

XenApp und XenDesktop 7.6

Nov 10, 2016

Neue Features

Long Term Service Release (LTSR)

Feature Pack 3: Installation und Upgrade

Feature Pack 3: Behobene Probleme

Bekannte Probleme

Nicht in diesem Release enthaltene Features

Systemanforderungen

Technische Übersicht

Konzepte und Komponenten

Active Directory

Fehlertoleranz

Bereitstellungsmethoden

Referenzarchitekturen

Designanleitungen

Implementierungsanleitungen

Neue Bereitstellungen

Vorbereiten der Installation

Vorbereiten der Virtualisierungsumgebung: VMware

Vorbereiten der Virtualisierungsumgebung: Microsoft System Center Virtual Machine Manager

Vorbereiten der Verwendung von Microsoft System Center Configuration Manager

Installieren über die grafische Oberfläche

Installieren an der Befehlszeile

Erstellen einer Site

Installieren oder Entfernen von Virtual Delivery Agents mit Skripts

Maschinenkataloge

Bereitstellungsgruppen

Mit XenApp veröffentlichte Anwendungen und Desktops

VM Hosted Apps

VDI-Desktops

Remote-PC-Zugriff

App-V

Lokaler App-Zugriff und URL-Umleitung

Server-VDI

Entfernen von Komponenten

Upgrades und Migration

Upgrade einer Bereitstellung

Migrieren von XenApp 6.x

Migrieren von XenDesktop 4

Sicherheit

Sicherheit mit Citrix XenApp und XenDesktop: Erste Schritte

Best Practices und Überlegungen zur Sicherheit

Delegierte Administration

Smartcards

SSL

Richtlinien

Arbeiten mit Richtlinien

Richtlinienvorlagen

Erstellen von Richtlinien

Vergleichen, Priorisieren, Modellieren und Problembehandlung für Richtlinien

Standardrichtlinieneinstellungen

Referenz für Richtlinieneinstellungen

Drucken

Beispiel einer Druckkonfiguration

Bewährte Methoden, Sicherheitsüberlegungen und Standardvorgänge

Druckrichtlinien und Einstellungen

Bereitstellen von Druckern

Pflege der Druckumgebung

Lizenzierung

Verbindungen und Ressourcen

Verbindungsleasing

Virtuelle IP-Adressen und virtuelles Loopback

Sekundärer Datenbankspeicherort

Delivery Controller-Umgebung

Hinzufügen, Entfernen, Verschieben von Controllern und Verschieben eines VDAs

Auf Organisationseinheiten von Active Directory-basierte Controller-Discovery

Sitzungsverwaltung

Durchführung einer Suche in Studio

Unterstützung für IPv4/IPv6

Clientordnerumleitung

Personal vDisks

Installation und Upgrade

Konfiguration und Verwaltung

Tools

Anzeigen, Meldungen und Problembehandlung

Benutzerprofile

HDX

Thinwire-Kompatibilitätsmodus

HDX 3D Pro

Flash-Umleitung

Host-zu-Client-Umleitung

GPU-Beschleunigung für Windows Desktopbetriebssysteme

GPU-Beschleunigung für Windows-Serverbetriebssystem

OpenGL Software Accelerator

Audiofeatures

Prioritäten für den Netzwerkdatenverkehr

[Überlegungen zu USB und Clientlaufwerk](#)

[Überwachen](#)

[Director](#)

[Sitzungsaufzeichnung](#)

[Personal vDisk](#)

[Konfigurationsprotokollierung](#)

[Überwachungsdienst-OData-API](#)

[SDK](#)

[Grundlegendes zum XenDesktop-Verwaltungsmodell](#)

[Erste Schritte mit dem SDK](#)

[Hilfe zu PowerShell-Cmdlets](#)

[FIPS-Beispielbereitstellungen](#)

[Hinweise zu Drittanbietern](#)

[Citrix SCOM Management Pack für XenApp und XenDesktop](#)

[Citrix SCOM Management Pack für den Lizenzserver](#)

Neue Funktionen

Nov 02, 2016

Dieser Abschnitt enthält folgende Themen:

[Long Term Service Release \(LTSR\)](#)

[Feature Pack 3](#)

[Feature Pack 2](#)

[Feature Pack 1](#)

[XenApp und XenDesktop 7.6](#)

Feature Pack 3

Das Feature Pack 3-Release enthält die folgenden neuen, verbesserten Features.

Anweisungen zur Installation und zum Upgrade finden Sie unter [Installieren und Aktualisieren von Feature Pack 3-Komponenten](#).

[Behobene Probleme](#)

Virtual Delivery Agent (VDA)

Das aktualisierte Paket für den eigenständigen VDA bietet Folgendes:

- **Unterstützung von Windows 10** Enterprise Edition im Standard-VDA für Windows-Desktopbetriebssysteme.

Universeller Druckserver

Das aktualisierte Paket für den universellen Druckserver bietet Folgendes:

- **Unterstützung von Windows Server 2012 R2**

HDX-Technologie

Die folgenden neuen und erweiterten Features der HDX-Technologie sind im Feature Pack 3 VDA enthalten und werden mit dem aktualisierten Paket für die Gruppenrichtlinienverwaltung verwaltet:

- **Verbesserungen am ThinWire-Kompatibilitätsmodus:** Der HDX Thinwire Plus-Grafikmodus verwendet einen aktualisierten Encoder, der die Anzeigeleistung einfacher grafischer Schnittstellen wie Microsoft Excel und Microsoft Word verbessert, ohne einen ressourcenintensiven Videocodec zu erfordern. Weitere Informationen finden Sie unter [ThinWire-Kompatibilitätsmodus](#).
- **Unterstützung von Signaturgeräten und Grafiktablets:** Administratoren können mit einer Richtlinie auf Geräten clientseitige Abfragen aktivieren und im Rahmen der Richtlinie Optimierungsmodi festlegen. Auf diese Weise können Benutzer effektiv in WANs mit Geräten wie dem Wacom Signatur-Tablet in XenApp- und XenDesktop-Sitzungen arbeiten. Weitere Informationen finden Sie unter [Einstellungen der Richtlinie "USB-Geräte"](#).

- **Richtlinie zum Verhindern von Videofallback:** Gewährt Administratoren mehr Kontrolle über das server- und clientseitige Abrufen und Wiedergeben von Multimediainhalten. Wenn z. B. ein Benutzergerät ein Video nicht lokal wiedergeben kann, kann der Administrator über eine neue Einstellung steuern, ob die automatische Wiedergabe des Videos auf dem Server zulässig ist, und dem Benutzer wird eine entsprechende Meldung angezeigt. Weitere Informationen finden Sie unter [Einstellungen der Richtlinie "Multimedia"](#) und [Einstellungen der Richtlinie "Flash-Umleitung"](#).
- **Angabe erforderlicher Mindestversionen für Flash-Umleitung:** Für Clientgeräte, die mit Receiver für Windows und Receiver für Linux auf VDAs zugreifen. Hiermit können Administratoren die Angabe erforderlicher Mindestversionen für die Flash-Umleitung besser steuern. Weitere Informationen zur Angabe erforderlicher Mindestversionen finden Sie unter [Flash-Umleitung](#).
- **Verbesserungen des virtuellen Kanals für Smartcards:** Umfasst Unterstützung für Smartcardleser-Plug-n-Play mit Receiver für Windows 4.3, virtuelle Smartcards, die auf Microsoft Windows Trusted Platform Module basieren, und Leistungsverbesserungen. Weitere Informationen zum Bereitstellen von Smartcards finden Sie unter [Smart cards](#) und [Virtual Smart Card Overview](#) im Microsoft Windows TechCenter.
- **Integration des virtuellen Framehawk-Anzeige Kanals mit dem VDA:** Der virtuelle Framehawk-Anzeige Kanal wurde in das eigenständige VDA-Paket integriert. Durch die Integration von Framehawk in die HDX-Technologie bietet Citrix eine reibungslose und intuitive Benutzererfahrung in Situationen, in denen Remotearbeitsbereichsbenutzer bisher Probleme bei der Interaktivität hatten, z. B. bei drahtlosen Verbindungen mit hohen Paketverlusten oder Engpässen. Weitere Informationen finden Sie in der [Administratordokumentation zum HDX Framehawk Virtual Channel](#).
- **Vollbildmodus für Apps für HDX 3D Pro VDI:** Der Virtual Delivery Agent (VDA) für HDX 3D Pro unterstützt Vollbild-Apps einschließlich 3D und Spiele-Apps in ICA-Sitzungen mit Einzelbildschirm.
- **Benutzerdefinierte Konfiguration von Tastenkombination für verlustfreie Komprimierung:** Mit HDX 3D Pro kann der Administrator eine Registrierungseinstellung konfigurieren und damit eine benutzerdefinierte Tastenkombination zum Überschreiben der Standardkombination (Alt + Umschalt + 1) festlegen. Benutzer können mit einer benutzerdefinierten Tastenkombination schnell zwischen verlustreicher und visuell verlustfreier Komprimierung oder zwischen verlustreicher und verlustfreier Komprimierung wechseln. Weitere Informationen finden Sie unter [GPU-Beschleunigung für Windows-Desktopbetriebssysteme](#).
- **Call Home:** In dieser Version von Call Home können Administratoren das PowerShell-Cmdlet **Start-TelemetryUpload** zum Extrahieren von VDA-Diagnoseinformationen ausführen. Mit dem Cmdlet können Sie eine lokale ZIP-Datei zum Versand per E-Mail an einen Supportmitarbeiter erstellen oder die Informationen unter Verwendung der Anmeldeinformationen und der Supportdienstleistungsnummer (SR-Nummer) automatisch an Citrix Insights Services hochladen. So können Administratoren und Supportmitarbeiter VDA-Probleme diagnostizieren und beheben. Das PowerShell-Cmdlet ist im Lieferumfang des VDA-Installationsprogramms in FP3 enthalten.
- **Verbesserte Unterstützung und Konfiguration von HDX-Drucker:** Umfasst immer aktive Protokollierung für den Druckserver und das Drucksubsystem auf dem VDA und die Option, dem Citrix Programm zur Verbesserung der Benutzerfreundlichkeit (CEIP) beizutreten. Weitere Informationen finden Sie unter [Installieren an der Befehlszeile](#) und [Bewährte Methoden, Sicherheitsüberlegungen und Standardvorgänge](#).
- **Aktualisierte Richtlinienvorlagen:** Dieses Feature Pack bietet neue Richtlinienvorlagen für hohe Serverskalierbarkeit und Optimierung für WAN und ersetzt die Vorlage "High Definition-Benutzererfahrung" durch die Vorlage "Besonders gute High Definition-Benutzererfahrung". Weitere Informationen finden Sie unter [Richtlinienvorlagen](#).
- **Unterstützung für 32-Bit-Cursor:** Ermöglicht eine verbesserte Benutzererfahrung für 3D-Anwendungen, in denen

benutzerdefinierte Cursor benötigt werden.

- **Desktopgestaltungsumleitung (DCR):** In der Standardeinstellung ist DCR deaktiviert.

StoreFront 3.0.1

Das aktualisierte StoreFront-Paket bietet Folgendes:

- **Unterstützung für TLS (Transport Layer Security) 1.0 und 1.1:** SSL (Secure Sockets Layer) 3.0 wird **nicht** unterstützt und Citrix rät dringend von einer Verwendung ab.

Weitere Informationen finden Sie in der [StoreFront](#)-Dokumentation.

Linux Virtual Desktop 1.1

Sie können virtuelle Linux-Desktops auf der Basis der SUSE- und Red Hat-Distributionen erstellen. Bereiten Sie die Linux-VMs vor, installieren Sie die neue [Linux-VDA-Software](#) darauf, konfigurieren Sie den Delivery Controller und stellen Sie die Desktops den Benutzern mit Studio zur Verfügung. Weitere Informationen finden Sie in den folgenden Dokumenten:

 [Installationsanleitung für SUSE](#)

 [Installationsanleitung für Red Hat](#)

Feature Pack 2

Das Feature Pack 2-Release enthält die folgenden neuen und erweiterten Features.

StoreFront 3.0

StoreFront enthält die folgenden neuen Features und Verbesserungen:

Aktualisierte StoreFront-Verwaltungskonsole

Unterstützung der einheitlichen Receiver-Benutzeroberfläche StoreFront 3.0 bietet eine zentralisierte Plattform für Anpassung und Branding der Benutzererfahrung bei der Auswahl von Anwendungen und Desktops in Receiver. Sie können die Benutzeroberfläche mit dem Logo, den Farben usw. Ihres Unternehmens anpassen. Beispiele neuer Receiver- und StoreFront-Features:

- **Vereinfachte Anwendungsorganisation:** gewohnte und einheitliche Benutzererfahrung bei der Anwendungsauswahl.
 - App-Gruppen mit Highlights: vom Administrator konfigurierte Anwendungen, die logisch zusammengefasst und Benutzern auf der Benutzeroberfläche zur Anwendungsauswahl angeboten werden
 - Ordneransicht von Anwendungen: Webinterface-Funktionalität in StoreFront
 - Favoriten: einfaches Hinzufügen (oder Entfernen) von Anwendungen zu Favoriten ermöglicht schnellen Zugriff
- **Über den Server verwaltete Benutzererfahrung:** Die Benutzererfahrung bei der Anwendungsauswahl ist serverkonfiguriert.
 - Einmaliges Ändern, globale Bereitstellung: Änderungen an der Benutzererfahrung bei der Anwendungsauswahl werden vom Administrator auf dem StoreFront-Server vorgenommen.
 - Einheitliche Benutzererfahrung bei der Anwendungsauswahl: Auf jedem Gerät, auf dem Receiver ausgeführt wird, wird die Benutzererfahrung bei der Anwendungsauswahl über HTML5 vom Server gesteuert und ist somit auf Smartphones, Tablets und Desktops einheitlich.

- Entkopplung der Benutzererfahrung bei der Anwendungsauswahl von der Receiver-Funktionalität: Clientupgrades ohne Erfordernis einer Endbenutzerschulung oder Auswirkungen auf die Benutzererfahrung; Unterstützung der Benutzererfahrung der Vorgängerversion ohne Kosten für Vorgängerclients.
- **Optionen für zweckgesteuerte Anpassung der Benutzererfahrung:** Sie wählen den passenden Grad der Anpassung.
 - Aktualisierte standardmäßige Benutzererfahrung: modernisiertes Erscheinungsbild bei der Anwendungsauswahl.
 - Serverkonfiguriertes Branding: Passen Sie das Branding mit Ihren Unternehmens-Logos sowie Farben und App-Gruppen mit Highlights an.
 - Umfassende Optionen für Anpassung und Branding: CSS und Skript-APIs sind für noch umfassendere Anpassung verfügbar.
- **Verwenden Sie die StoreFront-Verwaltungskonsole zur Ausführung folgender Aufgaben für Receiver:**
 - Festlegen von Receiver für Web als Standard für den Store
 - Erstellen einer Receiver für Web-Website
 - Ändern des Websitetyps
 - Anpassen der Websitedarstellung

Siehe Artikel zum [Verwalten einer Citrix Receiver für Web-Site](#).

- **App-Gruppen mit Highlights:** App-Gruppen sind Gruppen von Anwendungen, die zu einer bestimmten Kategorie gehören und in der Gruppe einfacher gefunden werden können. Mit Studio und der StoreFront-Verwaltungskonsole können Sie App-Gruppen unter Verwendung von Schlüsselwörtern, Anwendungsnamen oder Kategorien definieren.

Weitere Informationen finden Sie im Artikel [Erstellen und Verwalten empfohlener Apps](#).

Authentifizierung auf Basis des XML-Diensts

Wenn Sie alle konfigurieren, wird die Sitzung jeweils durch das zuerst auftretende Ereignis beendet. Verwenden Sie PowerShell-Cmdlets, um dieses Feature zu aktivieren oder zu deaktivieren. Weitere Informationen finden Sie unter [Authentifizierung auf Basis des XML-Diensts](#).

Unterstützung für HDX Real-Time Optimization Pack 1.8 für Microsoft Lync

HDX RealTime Optimization Pack bietet die folgenden neuen Features und Verbesserungen:

- Unterstützung für den Microsoft Skype for Business-Client im Lync-UI-Modus, den Microsoft Lync 2013-Client und den Microsoft Lync 2010-Client
- Unterstützung für Lync Server 2013 AutoErmittlungsdienst
- Anruf parken und Anruf annehmen im Lync 2010-Client
- Weiterleiten von Anrufen und gleichzeitiges Anrufen für den Lync 2010-Client
- Unterstützung für Mac
- Kerberos-Authentifizierung
- Pluszeichen (+) auf der Wähltastatur
- Unterstützung für Microsoft Windows 10 Technical Preview

Weitere Informationen finden Sie unter [HDX RealTime Optimization Pack 1.8 für Microsoft Lync](#).

Sitzungsaufzeichnung

Die Sitzungsaufzeichnung hat die folgenden neuen Features und Verbesserungen:

- Sie können die Anmeldeinformationen für die Verbindung mit der Datenbank bei der Installation der Datenbank für die Sitzungsaufzeichnung angeben.

- Sie können während der Installation der Datenbank für die Sitzungsaufzeichnung und des Sitzungsaufzeichnungsservers die Verbindung mit der Datenbank und während der Installation des Sitzungsaufzeichnungsagents die Verbindung mit dem Sitzungsaufzeichnungsserver testen.
- Microsoft Shared Management Objects ist für die Datenbank der Sitzungsaufzeichnung nicht mehr erforderlich.
- Das Citrix Programm zur Verbesserung der Benutzerfreundlichkeit (CEIP) ist in die Sitzungsaufzeichnung integriert. Weitere Informationen finden Sie unter [Citrix Programm zur Verbesserung der Benutzerfreundlichkeit \(CEIP\)](#).

Weitere Informationen finden Sie unter [Sitzungsaufzeichnung - XenApp 7.6 FP1, FP2 und LTSR](#).

Linux Virtual Desktop

Sie können jetzt virtuelle Linux-Desktops auf der Basis der SUSE- und Red Hat-Distributionen erstellen. Bereiten Sie die Linux-VMs vor, installieren Sie die neue [Linux-VDA-Software](#) darauf, konfigurieren Sie den Delivery Controller und stellen Sie die Desktops den Benutzern mit Studio zur Verfügung. Weitere Informationen finden Sie in den aktuellen Dokumenten im Abschnitt zum Feature Pack 3.

Framehawk Virtual Channel

Framehawk ist ein neuer virtueller ICA-Kanal, der Citrix HDX durch Funktionen erweitert und den Benutzern virtuelle Desktops und Anwendungen in HD bereitstellt.

Der virtuelle Framehawk-Kanal optimiert die Bereitstellung virtueller Desktops und Anwendungen für Benutzer über drahtlose Breitbandverbindungen. Er eignet sich ideal für Mobilgeräte bei verlustreichen Netzwerkverbindungen. Durch die Integration von Framehawk in die führende HDX-Technologie bietet Citrix eine reibungslose und intuitive Benutzererfahrung in Situationen, in denen Remotearbeitsbereichsbenutzer bisher Probleme bei der Interaktivität hatten, z. B. bei drahtlosen Verbindungen mit hohen Paketverlusten oder Engpässen.

Weitere Informationen finden Sie im folgenden Dokument, das Informationen über XenApp und XenDesktop 7.6 Feature Pack 3 und Feature Pack 2 enthält.

 [Framehawk-Administratordokumentation](#)

Director 7.6.300

Diese Version von Director unterstützt den virtuellen Framehawk-Kanal und enthält Fixes für folgende Probleme, die in Version 7.6.200 auftraten. Weitere Informationen zur Installation dieser Komponente finden Sie in der [Dokumentation für Director](#).

[Behobene Probleme](#)

Feature Pack 1

Das Feature Pack 1-Release enthält die folgenden neuen und erweiterten Features.

Sitzungsaufzeichnung

Die Sitzungsaufzeichnung ermöglicht das Aufzeichnen von Bildschirmaktivitäten über eine beliebige Verbindung von einem XenApp-Server (im Einklang mit den Unternehmensrichtlinien und gesetzlichen Vorschriften). Mit der Sitzungsaufzeichnung können Sie Sitzungen aufzeichnen, katalogisieren und archivieren und sie erneut aufrufen und wiedergeben. Weitere Informationen finden Sie unter [Sitzungsaufzeichnung - XenApp 7.6 FP1, FP2 und LTSR](#).

Citrix Director 7.6.200

Director 7.6.200 ermöglicht das Aktivieren und Deaktivieren der Sitzungsaufzeichnungsrichtlinie in den Ansichten "Benutzerdetails" und "Maschinendetails". Mit der Director-Konsole können Sie Sitzungsaufzeichnungsrichtlinien erstellen und aktivieren.

Weitere Informationen zur Sitzungsaufzeichnungs-/Director-Integration finden Sie unter <http://support.citrix.com/article/CTX142260>. Weitere Informationen zur Sitzungsaufzeichnung finden Sie unter [Sitzungsaufzeichnung - XenApp 7.6 FP1, FP2 und LTSR](#).

Citrix Lizenzierung 11.12.1

Das Programm zur Verbesserung der Benutzerfreundlichkeit (CEIP) für die Citrix Lizenzierung und Call Home sind zwei freiwillige Datenerfassungsprogramme, durch die Produkte von Citrix anonyme oder identifizierende Daten zu Konfiguration, Leistung, Fehlern und Nutzung Ihrer Bereitstellung erfassen und automatisch an Citrix senden. Weitere Informationen finden Sie unter [Programm zur Verbesserung der Benutzerfreundlichkeit \(CEIP\) für die Citrix Lizenzierung und Call Home](#). Nicht unterstützt von Lizenzserver VPX.

Durch das Citrix Programm zur Verbesserung der Benutzerfreundlichkeit (Customer Experience Improvement Program = CEIP) können Sie zum Design und zur Entwicklung von Citrix Produkten beitragen. Wenn Sie sich für das Programm registrieren, sammelt Citrix anonyme Informationen über Ihre Bereitstellung, die für die Verbesserung der Produktqualität, der Zuverlässigkeit und der Leistung verwendet werden.

Weitere Informationen finden Sie unter [Citrix Programm zur Verbesserung der Benutzerfreundlichkeit](#).

Unterstützung für HDX Real-Time Optimization Pack 1.7 für Microsoft Lync

Citrix HDX RealTime Optimization Pack 1.7 für Microsoft Lync bietet eine flexibel skalierbare Lösung für Echtzeit-Audio-/Videokonferenzen und Enterprise VoIP-Telefonie über Microsoft Lync in XenDesktop- und XenApp-Umgebungen für Benutzer von Linux- und Windows-Geräten. Das Optimization Pack nutzt die vorhandene Microsoft Lync-Infrastruktur und funktioniert mit anderen Microsoft Lync-Endpunkten, die nativ auf Geräten ausgeführt werden.

Weitere Informationen finden Sie unter [HDX RealTime Optimization Pack 1.7 für Microsoft Lync](#).

XenApp 7.6 und XenDesktop 7.6

Dieses Produktrelease enthält die folgenden neuen und erweiterten Features.

Sitzungsvorabstart und Sitzungsfortbestehen

Mit dem Sitzungsvorabstart und dem Fortbestehen von Sitzungen greifen Benutzern schnell auf Anwendungen zu, da Sitzungen gestartet werden, bevor sie angefordert werden (Sitzungsvorabstart) und Anwendungssitzungen aktiv bleiben, nachdem ein Benutzer alle Anwendungen geschlossen hat (Sitzungsfortbestehen). Diese Features werden nur für Serverbetriebssystemmaschinen unterstützt.

In der Standardeinstellung wird der Sitzungsvorabstart und das Sitzungsfortbestehen nicht verwendet. Eine Sitzung wird gestartet, wenn ein Benutzer eine Anwendung startet und sie bleibt aktiv, bis die letzte geöffnete Anwendung in der Sitzung geschlossen wird. Sie können die Features für alle Benutzer einer Bereitstellungsgruppe oder nur für bestimmte Benutzer aktivieren.

Wie lange eine nicht genutzte Sitzung aktiv bleibt, wenn der Benutzer keine Anwendung startet, kann über ein Timeout oder über Serverlast-Schwellenwerte angegeben werden. Sie können alle Parameter konfigurieren und die Sitzung wird jeweils durch das zuerst auftretende Ereignis beendet.

Weitere Informationen finden Sie unter [Konfigurieren des Vorabstarts und des Fortbestehens von Sitzungen](#).

Unterstützung für nicht authentifizierte (anonyme) Benutzer

Beim Erstellen oder Bearbeiten von Bereitstellungsgruppen mit Serverbetriebssystemmaschinen können Sie Benutzern Zugriff auf Anwendungen und Desktops gewähren, ohne dass die Benutzer Anmeldeinformationen in StoreFront oder Citrix Receiver eingeben müssen. Wenn Benutzer beispielsweise über Kioske auf Anwendungen zugreifen, werden ggf. Anmeldeinformationen von der Anwendung benötigt, nicht aber für das Citrix Zugriffsportal und die Tools.

Beim Konfigurieren der Bereitstellungsgruppe können Sie authentifzierten Benutzern oder nicht authentifzierten Benutzern oder beiden Gruppen von Benutzern Zugriff gewähren. Wenn Sie nicht authentifzierten (anonymen) Benutzern Zugriff gewähren, müssen Sie einen StoreFront-Store ohne Authentifizierung bereitstellen.

Weitere Informationen finden unter [Benutzer](#).

Verbindungsleasing

Um sicherzustellen, dass die Sitedatenbank immer verfügbar ist, empfiehlt Citrix, unter Befolgung der bewährten Methoden zur hohen Verfügbarkeit von Microsoft mit einer fehlertoleranten SQL Server-Bereitstellung zu beginnen. Delivery Controller können jedoch aufgrund von Netzwerkproblemen und Unterbrechungen möglicherweise nicht auf die Datenbank zugreifen, was dazu führt, dass Benutzer keine Verbindung mit ihren Anwendungen oder Desktops herstellen können.

Das Feature für Verbindungsleasing ergänzt die bewährten Methoden zur hohen Verfügbarkeit bei SQL Server, da es Benutzern die Wiederverbindung mit den zuletzt verwendeten Anwendungen und Desktops ermöglicht, selbst wenn die Sitedatenbank nicht verfügbar ist.

Benutzer verfügen zwar über eine große Anzahl veröffentlichter Ressourcen, verwenden jedoch oft nur wenige davon regelmäßig. Wenn Sie das Verbindungsleasing aktivieren, werden die Benutzerverbindungen mit den zuletzt verwendeten Anwendungen und Desktops im Normalbetrieb (wenn die Datenbank verfügbar ist) von jedem Controller zwischengespeichert. Wenn die Datenbank nicht mehr verfügbar ist, wechselt der Controller in den Leasingverbindungsmodus und beim Versuch der Herstellung oder des Neuaufbaus einer Verbindung mit einer kürzlich verwendeten Anwendung oder einem kürzlich verwendeten Desktop von StoreFront aus werden die zwischengespeicherten Vorgänge "wiedergegeben".

Weitere Informationen finden Sie unter [Verbindungsleasing](#).

Anwendungsordner

Sie können Anwendungen in Ordnern organisieren; dies vereinfacht die Verwaltung vieler Anwendungen in Studio.

Standardmäßig werden die Anwendungen einer Bereitstellungsgruppe in einem Ordner angezeigt. Im Bereitstellungsgruppenbereich in Studio können Sie zusätzliche Ordner erstellen und Anwendungen in sie verschieben. Mit Drag & Drop können Sie Ordner leicht verschieben, verschachteln und umbenennen. Sie können dazu auch die Elemente im Menü "Aktionen" verwenden. Außerdem können Sie Zielordner angeben und neue Ordner erstellen, wenn Sie Anwendungen zu einer Bereitstellungsgruppe hinzufügen.

Weitere Informationen finden Sie unter [Verwalten von Anwendungsordnern](#).

XenApp 6.5-Migration

Der XenApp 6.5-Migrationsprozess ermöglicht eine effiziente und schnelle Migration von einer XenApp 6.5-Farm zu einer Site mit XenApp 7.6 (oder einem neueren unterstützten Release). Dies ist nützlich bei Bereitstellungen mit vielen Anwendungen und Citrix Gruppenrichtlinien, da das Fehlerisiko beim manuellen Verschieben von Anwendungen und Citrix Gruppenrichtlinien in die neue XenApp-Site verringert wird.

Nach der Installation der XenApp 7.6-Kernkomponenten und dem Erstellen einer Site wird der Migrationsprozess in der folgenden Reihenfolge ausgeführt:

- Führen Sie das XenApp 7.6-Installationsprogramm auf jedem XenApp 6.5-Worker aus. Damit wird dieser automatisch auf einen neuen Virtual Delivery Agent für Windows-Serverbetriebssysteme zur Verwendung in der neuen Site aktualisiert.
- Führen Sie PowerShell-Export-Cmdlets auf einem XenApp 6.5-Controller aus, um die Anwendung und Citrix Richtlinieneinstellungen in XML-Dateien zu exportieren.
- Bearbeiten Sie, falls gewünscht, die XML-Dateien, um genau zu bestimmen, welche Elemente Sie in die neue Site importieren möchten. Durch Anpassen der Dateien können Sie Richtlinien- und Anwendungseinstellungen phasenweise (d. h. einige sofort, andere später) in die XenApp 7.6-Site importieren.
- Führen Sie PowerShell-Import-Cmdlets auf dem neuen XenApp 7.6-Controller aus, um die Einstellungen aus den XML-Dateien in die neue XenApp-Site zu importieren.
- Konfigurieren Sie die neue Site nach Bedarf um und testen Sie sie.

Weitere Informationen finden Sie unter [Migrieren von XenApp 6.x](#)

Citrix Programm zur Verbesserung der Benutzerfreundlichkeit

Durch das Citrix Programm zur Verbesserung der Benutzerfreundlichkeit (Customer Experience Improvement Program = CEIP) können Sie zum Design und zur Entwicklung von Citrix Produkten beitragen. Wenn Sie sich für das Programm registrieren, sammelt Citrix anonyme Informationen über Ihre Bereitstellung, die für die Verbesserung der Produktqualität, der Zuverlässigkeit und der Leistung verwendet werden.

Es ist ganz einfach, sich nach dem Erstellen oder Aktualisieren einer Site beim Programm zu registrieren. Sie können auch später jederzeit am Programm teilnehmen oder sich abmelden, indem Sie im Navigationsbereich von Studio Konfiguration wählen und den Anweisungen folgen.

Weitere Informationen finden Sie unter [Citrix Programm zur Verbesserung der Benutzerfreundlichkeit](#).

Verbesserte Drosselungseinstellungen für Verbindungen

Um die Leistung zu verbessern können Sie jetzt die maximale Anzahl der gleichzeitigen Aktionen sowie der gleichzeitigen Updates für Personal vDisk-Inventar und der Aktionen pro Minute für eine Hostverbindung festlegen.

Weitere Informationen finden Sie unter [Bearbeiten einer Verbindung](#).

Verbesserte Berichterstellung in Studio

In Studio werden detaillierte Status- und Fehlerberichte beim Aktualisieren von PvD-Images sowie ausführliche Lizenzierungsbenachrichtigungen angezeigt, wenn Sie im Knoten "Lizenzierung" sind.

SSL/TLS

Sie können Secure Sockets Layer-Verbindungen (SSL/TLS) zwischen Benutzern und VDAs aktivieren. Konfigurieren Sie hierfür SSL/TLS auf den Maschinen, auf denen die VDAs installiert sind, und in den Bereitstellungsgruppen, die VDAs enthalten.

Weitere Informationen finden Sie unter [SSL](#).

Virtuelle IP-Adressen und virtuelles Loopback

Für veröffentlichte Anwendungen können Sie das Feature "Virtuelle IP" von Microsoft auf Maschinen aktivieren und verwenden, die unter Windows Server 2008 R2 und Windows Server 2012 R2 ausgeführt werden. Außerdem können Sie neue Citrix Richtlinieneinstellungen zum Verwalten des virtuellen Loopbacks hinzufügen. Eine Option "Bevorzugtes Loopback" ist auch verfügbar.

Weitere Informationen finden Sie unter [Virtuelle IP-Adressen und virtuelles Loopback](#).

Schreiben in Client- und Sitzungszwischenablage

Sie können das Freigeben von Daten von der Hostzwischenablage zum Client oder von der Clientzwischenablage zur Benutzersitzung einschränken oder aktivieren. Wenn die Freigabe eingeschränkt ist, können Sie bestimmte Datenformate freigeben.

Weitere Informationen finden Sie unter [Einstellungen der Richtlinie "ICA"](#).

Remote-PC-Zugriff

Sie können jetzt verhindern, dass ein lokaler Benutzer eine Remotesitzung ohne Genehmigung des Remotebenutzers trennt.

Beim Trennen einer Remotesitzung wird beim Verschieben der Maus oder beim Druck einer Taste auf der Tastatur der lokale Monitor aktiviert. (In Vorversionen wurde beim doppelten Drücken von Strg+Alt+Entf der Anmeldebildschirm angezeigt.)

Die Position von Symbolen wird jetzt beibehalten, wenn Sie eine Verbindung von einem Gerät mit niedrigerer Auflösung herstellen und dann zu einem Gerät mit höherer Auflösung zurückkehren.

Generische USB-Umleitung

Dieses Release bietet Unterstützung für generische USB-Umleitung für spezielle USB-Geräte, für die es keinen optimierten virtuellen Kanal gibt. Diese Funktion leitet beliebige USB-Geräte von Clientmaschinen an virtuelle Desktops um. Durch dieses Feature können Endbenutzer mit vielen generischen USB-Geräten in der Desktopsitzung so interagieren, als wären die Geräte direkt angeschlossen.

Vorteile von generischer USB-Umleitung:

- Benutzer müssen keine Gerätetreiber auf den Benutzergeräten installieren.
- USB-Clienttreiber werden auf der VDA-Maschine installiert.

Dieses Feature erfordert Windows Server 2012 R2 und funktioniert mit vorhandenen Versionen von Windows Receiver für veröffentlichte Desktopsitzungen, die auf RDS-Hosts in Single-Hop-Szenarios gehostet werden. Mit diesem Feature werden USB-Clienttreiber auf dem Host installiert, daher müssen die Treiber mit RDSH für Windows 2012 R2-Plattformen kompatibel sein.

Citrix Director 7.6.100

Director 7.6.100 enthält die folgenden neuen und erweiterten Features. Informationen zu Download, Installation und Upgrade finden Sie unter <http://support.citrix.com/article/CTX200330>.

- **VM-Nutzung:** ermöglicht Administratoren einen Überblick über die Nutzung der VMs in Echtzeit, sodass sie den Kapazitätsbedarf einer Site schnell einschätzen können. VM-Nutzung wird in Nutzungsdaten für

Desktopbetriebssysteme und Nutzungsdaten für Serverbetriebssysteme unterteilt.

- **Verfügbarkeit der Desktopbetriebssysteme:** zeigt den aktuellen Zustand der Desktopbetriebssystemmaschinen (VDIs) nach Verfügbarkeit für die gesamte Site oder für bestimmte Bereitstellungsgruppen an.
- **Verfügbarkeit der Serverbetriebssysteme:** zeigt den aktuellen Zustand der Serverbetriebssystemmaschinen nach Verfügbarkeit für die gesamte Site oder für bestimmte Bereitstellungsgruppen an.
- **Exportverbesserungen:** bietet die Option zum Exportieren von Trendberichten in den Formaten CSV, PDF und Excel.
- **Drilldown-Verbesserungen:** Director 7.6 verfügt nun über weitere Drilldownfunktionen. Dieses erweiterte Feature ermöglicht Administratoren das Navigieren durch Trenddiagramme, indem sie bestimmte Zeiträume vergrößern (durch Klicken auf einen Datenpunkt im Diagramm) und die Detailinformationen zum Trend anzeigen. Administratoren können die genauen Auswirkungen der angezeigten Trends nun besser verstehen.

Neue Features in Director 7.6

Lizenzierungsbenachrichtigungen weisen Sie auf Probleme hin, die sich auf Benutzerverbindungen auswirken können. Darüber hinaus empfiehlt Director Aktionen, um Abhilfe zu schaffen. In Director werden u. a. folgende Zustände angezeigt:

- Alle Lizenzen sind abgelaufen.
- Lizenzen laufen bald ab.
- Citrix Lizenzkulanzzzeitraum ist abgelaufen.
- Der Zusatzkulanzzzeitraum ist aktiv und alle installierten Lizenzen werden zurzeit verwendet.

Anzeigen der Verwendung gehosteter Anwendungen: Sie können die Bereitstellungsgruppe und den Zeitraum auswählen, um eine Kurve der höchsten gleichzeitigen Nutzung sowie eine Tabelle mit der anwendungsbasierten Verwendung anzuzeigen. In der Tabelle "Anwendungsbasierte Verwendung" können Sie eine bestimmte Anwendung auswählen, um Details und eine Liste der Benutzer anzuzeigen, die die Anwendung verwenden oder verwendet haben.

Überwachen von Hotfixes: Zum Anzeigen der auf einem bestimmten Maschinen-VDA (physisch oder VM) installierten Hotfixes wählen Sie die Ansicht "Benutzerdetails" oder "Maschinendetails".

Die Filterfunktion wurde erweitert: Die Funktion zum Filtern von Daten kann durch Mausklick ausgewählt werden und führt zu Benutzerdetails, Maschinendetails, Endpunktdetails und Anonyme Sitzungen.

Director ist mit XenApp 6.5 kompatibel. Sie können mit Director Ihre XenApp 6.5-Bereitstellungen überwachen.

Weitere Informationen finden Sie unter [Director](#).

AppDNA 7.6

Citrix AppDNA beschleunigt die Migration und Transformation von Desktop- und Webanwendungen für neue Umgebungen über schnelle Analysis, automatische Anwendungslösung und -verpackung und tägliche Anwendungsverwaltung. AppDNA 7.6 enthält eine neue Build-Bewertungslösung, die testet, ob Anwendungen auf zusätzlichen Builds derselben Betriebssystemfamilie funktionieren. Erweiterungen der AppDNA 7.6-Analyse geben jetzt an, ob erforderliche Anwendungen, Anwendung-Frameworks und Dateien vorhanden sind, ob aktivierte Gruppenrichtlinienobjekte Probleme verursachen werden, und ob Webanwendungen mit Citrix WorxWeb kompatibel sind.

Weitere Informationen finden Sie unter [AppDNA 7.6](#).

Citrix StoreFront 2.6

Vereinfachte Storekonfiguration in der Verwaltungskonsolle: Die aktualisierte StoreFront-Konsole vereinfacht die Konfiguration der folgenden StoreFront-Features:

- Benutzerabonnements
- Festlegen des Sitzungstimeouts für Receiver für Web
- Anzeigen von Domänenlisten auf der Anmeldeseite

Ordneransicht "Eigene Apps" für Receiver für Web: In dieser neuen Ansicht werden die Anwendungen in einer Ordnerhierarchie angezeigt und sie enthält eine Breadcrumbspur für Stores ohne Authentifizierung und obligatorische Stores. Diese Ordneransicht kann Benutzern den Wechsel vom Webinterface nach Receiver für Web erleichtern.

Eingeschränkte Kerberos-Delegierung für XenApp 6.5: StoreFront mit eingeschränkter Kerberos-Delegierung ermöglicht Passthrough-Authentifizierung, wobei Client und Gerät nicht mehr Windows mit Receiver ausführen müssen.

Zugriff über einen vollqualifizierten Domännennamen (FQDN): Mit diesem Feature können Sie internen und externen Zugriff auf Ressourcen mit einem einzelnen vollqualifizierten Domännennamen bieten.

Smartcard-Authentifizierung für XenApp Services-Support: Der StoreFront-Server führt die Authentifizierung an XenApp Services-Supportsites mit Smartcards durch und benötigt keine bestimmten Versionen von Receiver und den Betriebssystemen.

Smartcard-Authentifizierung für Receiver für Android, iOS und Linux: Neue Versionen von Receiver unterstützen die lokale und remote Verwendung von Smartcards für den Zugriff auf Anwendungen und Desktops.

Erweiterbare Authentifizierung: Die Unterstützung für die erweiterbare Authentifizierung bietet einen Anpassungspunkt für die Erweiterung der formularbasierten Authentifizierung in StoreFront. WorxHome und Receiver für Web verwenden dies für die Authentifizierung an XenMobile und XenApp sowie XenDesktop bei internen (direkten) und externen (mit NetScaler Gateway) Zugriffsszenarios.

Weitere Informationen finden Sie unter [StoreFront 2.6](#).

Unterstützung für Citrix Connector 7.5

Citrix Connector 7.5 stellt eine Brücke zwischen Microsoft System Center Configuration Manager und XenApp oder XenDesktop bereit, sodass Sie die Verwendung von Configuration Manager auf Ihre Citrix Umgebungen ausweiten können. Die Unterstützung von Citrix Connector 7.5 ist jetzt in den Platinum Editionen von XenApp 7.6 und XenDesktop 7.6 enthalten.

Weitere Informationen finden Sie unter [Citrix Connector 7.5 für System Center Configuration Manager 2012](#).

Unterstützung für Receiver für Chrome 1.4 und Receiver für HTML5 1.4

Mit Receiver für Chrome für den Zugriff können Benutzer auf virtuelle Desktops und gehostete Anwendungen von Geräten mit dem Google Chrome-Betriebssystem zugreifen. Benutzer greifen auf diese Ressourcen über Receiver für Chrome zu und die Desktops und Anwendungen werden in einem Fenster angezeigt.

Receiver für HTML5 wird auf StoreFront-Servern ausgeführt und ermöglicht den Benutzerzugriff auf virtuelle Desktops und gehostete Anwendungen über einen Webbrowser. Benutzer greifen auf Desktops und Anwendungen in einem Webbrowser zu, ohne Citrix Receiver auf den Geräten zu installieren.

In diesem Release bieten beide Receiver Folgendes: Konvertieren von Dokumenten von gehosteten Anwendungen oder von auf virtuellen Desktops ausgeführten Anwendungen in das PDF-Format und Anzeige auf einem lokalen Gerät oder Ausgabe auf einem lokal angeschlossenen Drucker, verbesserte Zwischenablagenvorgänge, Metrik zur Endbenutzererfahrung und Verfolgen der Lizenznutzung für gehostete Anwendungen.

Weitere Informationen finden Sie unter [Receiver für Chrome 1.4](#) und [Receiver für HTML5 1.4](#).

Unterstützung für HDX Real-Time Optimization Pack 1.5 für Microsoft Lync

HDX Real-Time Optimization Pack 1.5 für Microsoft Lync unterstützt Lync-zertifizierte USB-Telefone, gemischte Lync 2010 Client- und Lync Server 2013-Konfiguration und asynchrone Upgrades.

Weitere Informationen finden Sie unter [HDX Real-Time Optimization Pack 1.5 für Microsoft Lync](#).

Long Term Service Release (LTSR)

Jan 05, 2017

Releasedatum: 11. Januar 2016

Installieren und Aktualisieren von LTSR-Komponenten

Für das XenApp und XenDesktop 7.6 Long Term Service Release (LTSR) müssen Sie die Komponenten von XenApp und XenDesktop 7.6, die zum LTSR und Ihrer Bereitstellung gehören, auf die LTSR-Versionen aktualisieren. Beispiel: Wenn Provisioning Services zur Bereitstellung gehört, aktualisieren Sie die Provisioning Services-Komponente auf die LTSR-Version. Wenn Provisioning Services nicht zu Ihrer Bereitstellung gehört, brauchen Sie es nicht zu installieren oder zu aktualisieren.

Das Upgrade auf die LTSR-Versionen ist verbindlich, damit Ihre Bereitstellung für die Vorteile von LTSR qualifiziert ist.

Darüber hinaus empfiehlt Citrix auch bestimmte Versionen von Citrix Receiver und anderen Komponenten. Obwohl es nicht für LTSR erforderlich ist, wird durch ein Upgrade auf die aktuellen Versionen dieser Komponenten sichergestellt, dass Ihre Bereitstellung wartungsfreundlich ist und die aktuellen Fixes vorhanden sind.

Hilfreiche Links:

- [Download LTSR \(XenApp\)](#)
- [Download LTSR \(XenDesktop\)](#)
- [XenApp and XenDesktop Servicing Options](#)
- [LTSR Frequently Asked Questions \(FAQs\)](#)
- [Product Lifecycle Dates](#)
- [LTSR Program for Receiver for Windows](#)

LTSR-Basiskomponenten und verbindliche Versionen

Hinweis

Die folgenden Informationen gelten spezifisch für das LTSR-Basisrelease. Die entsprechenden Informationen für [KU1](#) bzw. [KU2](#) finden Sie in der zugehörigen Dokumentation.

Obwohl es für LTSR nicht erforderlich ist, dass die folgenden Komponenten zu Ihrer Bereitstellung gehören, müssen Sie jede vorhandene Komponente in der Bereitstellung auf die unten angegebene Version aktualisieren.

LTSR-Basiskomponenten	Version	Hinweise
VDA für Desktopbetriebssystem	7.6.300	Für Windows 10 gelten besondere Regeln. Weitere Informationen

		finden Sie unter Kompatible Komponenten und Plattformen .
VDA für Serverbetriebssystem	7.6.300	
Delivery Controller	7.6 Update 3	
Citrix Studio	7.6 Update 3	
Citrix Director	7.6.300	
Gruppenrichtlinienverwaltung	7.6.300 (2.5)	
StoreFront	3.0.1	
Provisioning Services	7.6 Update 1	Für Windows 10 gelten besondere Regeln. Weitere Informationen finden Sie unter Kompatible Komponenten und Plattformen .
Universeller Druckserver	7.6.300	Nur Windows 2008 R2 SP1 Windows 2012 Windows 2012 R2 wird unterstützt
Sitzungsaufzeichnung	7.6.100	nur Platinum Edition

Kompatible Komponenten und Plattformen

Es empfiehlt sich, die folgenden Komponenten in 7.6 LTSR-Umgebungen zu verwenden. Für diese Komponenten können die LTSR-Vorteile (erweiterter Lebenszyklus und kumulative Updates mit Fixes) nicht beansprucht werden. Citrix fordert Sie u. U. auf, ein Upgrade auf eine neuere Version der folgenden Komponenten in Ihren 7.6 LTSR-Umgebungen durchzuführen.

Hinweis zu Windows 10: Regulärer Support für Windows 10 ist im Rahmen des aktuellen Releases verfügbar. Windows 10 erhält nicht die vollen mit 7.6 LTSR verbundenen Vorteile. Bei Bereitstellungen, die Windows 10-Maschinen enthalten, empfiehlt Citrix, Version 7.9 des VDA für Desktopbetriebssysteme und Provisioning Services zu verwenden.

Weitere Informationen finden Sie im [Blogbeitrag zum Hinzufügen von Windows 10-Kompatibilität zu XenApp und XenDesktop 7.6 LTSR](#) und in den [FAQ zu den XenApp- und XenDesktop-Wartungsoptionen \(LTSR\)](#).

Mit LTSR kompatible Komponenten und Plattformen	Version
Profilverwaltung	5.4

AppDNA	7.6.5
Lizenzserver	11.12.1
HDX RealTime Optimization Pack	2.0
Windows 10	VDA: Version 7.9 Provisioning Services: Version 7.9

Kompatible Versionen von Citrix Receiver

Damit Wartungsfreundlichkeit und optimale Leistung gewährleistet sind, empfiehlt Citrix, dass Sie ein Upgrade auf die aktuelle Version von Citrix Receiver ausführen, sobald sie verfügbar ist. Die aktuellen Versionen sind zum Download hier verfügbar: <https://www.citrix.com/downloads/citrix-receiver.html>. Wenn Sie den [Citrix Receiver-RSS-Feed](#) abonnieren, erhalten Sie eine Benachrichtigung, wenn eine neue Version von Citrix Receiver zur Verfügung steht.

Für Citrix Receiver können die XenApp und XenDesktop LTSR-Vorteile (erweiterter Lebenszyklus und kumulative Updates mit Fixes) nicht beansprucht werden. Citrix fordert Sie u. U. auf, in Ihren 7.6 LTSR-Umgebungen ein Upgrade auf eine neuere Version von Citrix Receiver durchzuführen. Citrix hat ein spezielles LTSR-Programm für Citrix Receiver für Windows angekündigt. Weitere Informationen zu diesem Programm sind auf der Seite [Lifecycle Milestones for Citrix Receiver](#) verfügbar.

LTSR unterstützt insbesondere die folgenden Versionen von Citrix Receiver und alle neueren Versionen:

Mit LTSR kompatible Citrix Receiver	Version
Citrix Receiver für Windows	4.4 oder höher
Citrix Receiver für Linux	13.2.1 oder höher
Citrix Receiver für Mac	12.1 oder höher
Citrix Receiver für Chrome	1.8 oder höher
Citrix Receiver für HTML5	1.8 oder höher
Citrix Receiver für iOS	6.1.1 oder höher
Citrix Receiver für Android	3.8 oder höher

Ausnahmen

Für die folgenden Features, Komponenten und Plattformen können die LTSR-Lebenszyklusmeilensteine und Vorteile nicht in Anspruch genommen werden. Insbesondere sind kumulative Updates und die mit dem erweiterten Lebenszyklus verbundenen Vorteile ausgeschlossen. Updates für ausgeschlossene Features und Komponenten sind durch regelmäßige aktuelle Releases verfügbar.

Ausgeschlossene Features
Lokaler App-Zugriff
Framehawk

Ausgeschlossene Komponenten
Linux VDA
Personal vDisk

Ausgeschlossene Windows Plattformen*
Windows 2008 32 Bit (für Universal Print Server)

* Citrix behält sich das Recht vor, die Plattformunterstützung basierend auf den Lebenszyklusmeilensteinen von Drittanbietern zu aktualisieren.

Upgrade auf XenApp und XenDesktop 7.6 LTSR

Sie können ein Upgrade auf LTSR direkt von XenApp und XenDesktop 7.6 und von jedem der drei 7.6 Feature Packs durchführen.

Download:

- [Download LTSR \(XenApp\)](#)
- [Download LTSR \(XenDesktop\)](#)

Die gute Nachricht ist, dass die LTSR-Versionen vieler Komponenten schon länger verfügbar sind, und zwar hauptsächlich als Teil von Feature Pack 3. Wenn Sie Ihre Bereitstellung mit Feature Pack 3 aktualisiert haben, sind viele Komponenten bereits LTSR-kompatibel. In diesen Fällen brauchen Sie nichts weiter zu tun. Unter "Erstveröffentlichung dieser Version" können Sie für die verschiedenen Komponenten prüfen, ob Sie ein Upgrade über Feature Pack 3 hinaus durchführen müssen.

Vorsicht beim Upgrade des Controllers: Beim Upgrade auf die LTSR-Version des Controllers wird mindestens eines der Datenbankschemas des Datenspeichers Ihrer Site geändert. Diese Änderungen sind permanent und können nicht rückgängig gemacht werden. Sie können diese Änderungen nicht automatisch rückgängig machen. Vor dem Upgrade des Controllers sollten Sie daher den Abschnitt zum Upgrade auf die LTSR-Version des Controllers sorgfältig durchlesen.

Virtual Delivery Agent (VDA) für Desktopbetriebssysteme 7.6.300

LTSR-Version: VDA für Desktopbetriebssysteme 7.6.300

Erstveröffentlichung dieser Version: 30. September 2015 als Teil von Feature Pack 3 (VDAWorkstationSetup_7.6.300.exe)

[Systemanforderungen](#)

[Behobene Probleme](#)

Installation und Upgrade

Laden Sie VDAWorkstationSetup_7.6.300.exe auf die Maschine herunter, auf der Sie den VDA installieren möchten, und führen Sie die Datei aus. Verwenden Sie die grafischen Benutzeroberfläche oder die Befehlszeile.

Weitere Informationen finden Sie unter [Installieren von VDAs mit dem eigenständigen Paket](#).

Virtual Delivery Agent (VDA) für Serverbetriebssystemmaschinen 7.6.300

LTSR-Version: VDA für Serverbetriebssystemmaschinen 7.6.300

Erstveröffentlichung dieser Version: 30. September 2015 als Teil von Feature Pack 3 (VDAServerSetup_7.6.300.exe)

[Systemanforderungen](#)

[Behobene Probleme](#)

Installation und Upgrade

Laden Sie VDAServerSetup_7.6.300.exe auf die Maschine herunter, auf der Sie den VDA installieren möchten, und führen Sie die Datei aus. Verwenden Sie die grafischen Benutzeroberfläche oder die Befehlszeile.

Der VDA für Windows-Serverbetriebssysteme stellt automatisch Microsoft Visual C++ 2013 Runtime (32 Bit und 64 Bit) sowie 2008 und 2010 Runtimes (32 Bit und 64 Bit) bereit. Microsoft Visual C++ 2005 wird nicht mehr bereitgestellt. Diese Voraussetzungen initiieren einen Serverneustart. Die VDA-Installation wird nach dem Neustart fortgesetzt.

Weitere Informationen finden Sie unter [Installieren von VDAs mit dem eigenständigen Paket](#).

Delivery Controller 7.6.3 (Controller Hotfixes Update 3)

LTSR-Version: Delivery Controller 7.6.3

Erstveröffentlichung dieser Version: 12. November 2015 als Delivery Controller 7.6.3 (Controller Hotfixes Update 3)

[Systemanforderungen](#)

[Behobene Probleme](#) (32 Bit)

Behobene Probleme (64 Bit)

Installation und Upgrade

Wenn Ihrer Controller Version 7.6.3 aufweist:

Version 7.6.3 (Controller Hotfixes Update 3) des Controllers ist die LTSR-Version. Wenn Sie bereits ein Upgrade auf Version 7.6.3 ausgeführt haben, ist der Controller LTSR-kompatibel und es sind keine weiteren Upgrades nötig. Fahren Sie mit dem Abschnitt zu Citrix Studio fort. Wichtig: Stellen Sie sicher, dass alle Komponenten der Version 7.6.3 installiert sind, sonst ist der Controller möglicherweise instabil.

Wenn Ihrer Controller Version 7.6, 7.6.1 oder 7.6.2 aufweist:

Damit der Controller LTSR-kompatibel ist, müssen Sie ein Upgrade auf die LTSR-Version durchführen. Laden Sie dazu die LTSR-Version auf den Controller herunter und folgen Sie den nachstehenden Upgradeanweisungen.

Achtung: Downgrades, auch Rollbacks genannt, von einzelnen Controllerkomponenten werden nicht unterstützt und können das System in einem instabilen Zustand zurücklassen. Die Controllerkomponenten werden nicht als Patch auf die vorhandenen Installationen angewendet, jede Komponente ersetzt die ursprüngliche Komponente mit einer neuen Installation. Folglich wird durch das Deinstallieren einer Komponente die gesamte Komponente vom Computer entfernt. Wenn Sie zu einer früheren Version des Controllers zurückkehren möchten, müssen Sie jede Komponente deinstallieren und dann die frühere Version neu installieren. Beim Wiederherstellen einer früheren Version einer Komponente gehen möglicherweise die Einstellungen verloren, die Sie bei der Installation dieses Upgrades konfiguriert haben.

Wichtig: Alle Komponenten der LTSR-Version müssen installiert sein, sonst ist der Controller möglicherweise instabil.

Wenn Sie ein Upgrade von der Basisversion (RTM) des 7.6-Controllers durchführen, installieren Sie alle Komponenten des LTSR-Controllers.

Wenn Sie ein Upgrade von Delivery Controller 7.6.1 (Controller-Hotfixes Update 1) oder Delivery Controller 7.6.2 (Controller-Hotfixes Update 2) durchführen, installieren Sie nur die Komponenten, die gegenüber dem früheren, von Ihnen bereits installierten Release in der LTSR-Version neu sind. Die einzelnen Komponenten brauchen nicht in einer bestimmten Reihenfolge installiert zu werden.

- Damit dieses Upgrade erfolgreich durchgeführt werden kann, dürfen auf den Servern keine Einschränkungen für die Bearbeitung der Registrierung gelten.
- Zusätzliche Informationen über die Installation von Updates für den XenDesktop/XenApp 7.x-Controller finden Sie unter [CTX201988](#).

Upgrade von Delivery Controller 7.6.2 (Controller Hotfixes Update 2)

Achtung: Standardmäßig ändert die Brokerdienstkomponente (BrokerSvc760WX64003.msi) das Broker-Datenbankschema des Datenspeichers Ihrer Site. Diese Änderungen sind permanent und können nicht rückgängig gemacht werden. Wenn Sie die Brokerdienstkomponente zu einem späteren Zeitpunkt deinstallieren, werden diese Änderungen nicht automatisch rückgängig gemacht. Citrix empfiehlt daher dringend, dass Sie ein Backup des Datenspeichers der Site erstellen, bevor Sie die Brokerdienstkomponente installieren. Auf diese Weise können Sie den Datenspeicher der Site mit der Backupversion manuell wiederherstellen. Änderungen, die Sie am Datenspeicher der Site zwischen Backup und Wiederherstellung vorgenommen haben, gehen jedoch verloren. Weitere Informationen zum Sichern und Wiederherstellen von Datenspeichern finden Sie unter [CTX135207](#).

Das Update des Datenbankschemas ist nur erfolgreich, wenn mindestens eine Site vorhanden ist. Wenn Sie noch keine Site

erstellt haben, erstellen Sie mindestens eine Site, bevor Sie das Update installieren. Andernfalls wird bei der Installation das vorhandene Datenbankschema nicht aktualisiert und Sie müssen XenDesktop neu erstellen.

Hinweis: Nach dem Upgrade auf dieses Release werden Sie zur Lizenzserver-Kompatibilitätsprüfung in Citrix Studio aufgefordert, damit sichergestellt ist, dass der Lizenzserver die erforderliche Version hat. Wenn Sie den mit XenDesktop 7.6 oder einer neueren Version veröffentlichten Lizenzserver verwenden, müssen Sie den Lizenzserver nicht aktualisieren. Klicken Sie auf "Weiter", um mit dem Upgrade des Datenbankschemas fortzufahren.

1. Stellen Sie sicher, dass Sie einen 7.6.2-Controller aktualisieren. Andernfalls finden Sie nachfolgend Informationen zum Aktualisieren von Delivery Controller 7.6.1 oder 7.6.
2. Stellen Sie sicher, dass mindestens eine Site vorhanden ist.
3. Erstellen Sie ein Backup des Datenspeichers der Site.
4. Kopieren Sie das Releasepaket in einen freigegebenen Ordner auf dem Netzwerk.
5. Speichern Sie die MSI-Datei(en) für die Komponente auf dem Delivery Controller, den Sie aktualisieren möchten.
6. Führen Sie die MSI-Dateien aus.
7. Starten Sie den Delivery Controller neu, auch wenn Sie nicht dazu aufgefordert werden.
8. Um ein Upgrade auf das aktuelle, von diesem Release installierte Datenbankschema auszuführen, gehen Sie zum Citrix Studio Dashboard und klicken Sie auf "Upgrade".

Upgrade von Delivery Controller 7.6.1 oder 7.6

Hinweis: Die Anweisungen in diesem Abschnitt gelten nicht, wenn Sie ein Upgrade von einem Controller der Version 7.6.2 durchführen.

Achtung: Standardmäßig ändern beide Komponenten, der Brokerdienst (BrokerSvc760WX64003.msi) und der Hostdienst (HostSvc760WX64003.msi), die Datenbankschemata von Broker und Host im Datenspeicher Ihrer Site. Diese Änderungen sind permanent und können nicht rückgängig gemacht werden. Wenn Sie die Brokerdienst- oder die Hostdienstkomponente zu einem späteren Zeitpunkt deinstallieren, werden diese Änderungen nicht automatisch rückgängig gemacht. Citrix empfiehlt daher dringend, dass Sie ein Backup des Datenspeichers der Site erstellen, bevor Sie die Brokerdienst- oder die Hostdienstkomponente installieren. Auf diese Weise können Sie den Datenspeicher der Site mit der Backupversion manuell wiederherstellen. Änderungen, die Sie am Datenspeicher der Site zwischen Backup und Wiederherstellung vorgenommen haben, gehen jedoch verloren. Weitere Informationen zum Sichern und Wiederherstellen von Datenspeichern finden Sie unter [CTX135207](#).

Das Update des Datenbankschemas ist nur erfolgreich, wenn mindestens eine Site vorhanden ist. Wenn Sie noch keine Site erstellt haben, erstellen Sie mindestens eine Site, bevor Sie das Update installieren. Andernfalls wird bei der Installation das vorhandene Datenbankschema nicht aktualisiert und Sie müssen XenDesktop neu erstellen.

Hinweis: Nach dem Upgrade auf dieses Release werden Sie zur Lizenzserver-Kompatibilitätsprüfung in Citrix Studio aufgefordert, damit sichergestellt ist, dass der Lizenzserver die erforderliche Version hat. Wenn Sie den mit XenDesktop 7.6 oder einer neueren Version veröffentlichten Lizenzserver verwenden, müssen Sie den Lizenzserver nicht aktualisieren. Klicken Sie auf "Weiter", um mit dem Upgrade des Datenbankschemas fortzufahren.

1. Stellen Sie sicher, dass mindestens eine Site vorhanden ist.
2. Erstellen Sie ein Backup des Datenspeichers der Site.
3. Kopieren Sie das Releasepaket in einen freigegebenen Ordner auf dem Netzwerk.
4. Speichern Sie die MSI-Datei(en) für die Komponente auf dem Delivery Controller, den Sie aktualisieren möchten.
5. Führen Sie die MSI-Dateien aus.
6. Starten Sie den Delivery Controller neu, auch wenn Sie nicht dazu aufgefordert werden.
7. Um ein Upgrade auf das aktuelle, von diesem Release installierte Datenbankschema auszuführen, gehen Sie zum Citrix

Studio Dashboard und klicken Sie auf "Upgrade".

Deinstallieren von Delivery Controller-Komponenten und Wiederherstellen vorherigen Versionen der Komponenten und des Datenspeichers der Site

1. Deinstallieren Sie die Komponenten über "Software" bzw. "Programme und Funktionen".
2. Stellen Sie die Datenspeicher wie in [CTX135207](#) beschrieben wieder her.
3. Installieren Sie die gewünschte Komponentenstufe (Basis oder ein höheres Release).
4. Starten Sie den Delivery Controller neu, auch wenn Sie nicht dazu aufgefordert werden.

Citrix Studio 7.6 Update 3

LTSR-Version: Citrix Studio 7.6 Update 3

Erstveröffentlichung dieser Version: 29. Oktober 2015 als Hotfix DStudio760WX64003; Hotfix DStudio760WX86003

Systemanforderungen

Behobene Probleme

- [64 Bit](#)
- [32 Bit](#)

Bekannte Probleme

If Citrix Studio is open during an upgrade, if you select the setting "Close the applications and attempt to restart them" on the Files in Use page of the installation wizard of this hotfix, the following message might appear:

"Setup was unable to automatically close all requested applications. Please ensure that the applications holding files in use are closed before continuing with the installation."

If the message appears, you can safely close the message and click OK to continue with the installation.

Installation und Upgrade

Laden Sie die LTSR-Version von Citrix Studio herunter und befolgen Sie die Installationsanweisungen in [CTX201572](#).

Citrix Director 7.6.300

LTSR-Version: Director 7.6.300

Erstveröffentlichung dieser Version: 30. September als Teil von Feature Pack 3 (Director_7.6.300.zip)

Systemanforderungen

Vergewissern Sie sich, dass Sie alle erforderlichen Features in IIS ausgewählt haben. Eine vollständige Liste finden Sie unter [CTX142260](#). Installieren Sie die Citrix Gruppenrichtlinienverwaltungskomponente, wenn sie noch nicht vorhanden ist.

Behobene Probleme

Installation und Upgrade

Laden Sie die LTSR-Version von Citrix Director auf den Server herunter, auf dem Director ausgeführt wird, und befolgen Sie die Anweisungen unter [Director](#).

Gruppenrichtlinienverwaltung 7.6.300 (2.5)

LTSR-Version: Gruppenrichtlinienverwaltung 7.6.300

Erstveröffentlichung dieser Version: 30. September als Teil von Feature Pack 3 (CitrixGroupPolicyManagement_7.6.300.zip)

Systemanforderungen:

Computer mit Windows 7, Windows 8, Windows 8.1, Server 2008 R2, Server 2012 oder Server 2012 R2

Die neuen und erweiterten Features der HDX-Technologie im VDA werden mit dem aktualisierten Paket für die Gruppenrichtlinienverwaltung verwaltet. Hinweis: Nach der Installation wird diese Komponente als Version 2.5.0.0 in "Programme und Funktionen" angezeigt.

Installation und Upgrade

Citrix Gruppenrichtlinienverwaltung muss auf einem System mit Director installiert werden, damit Richtlinien in der Benutzerdetailansicht angezeigt werden. Laden Sie die LTSR-Version von Citrix Gruppenrichtlinienverwaltung (Citrix Richtlinie) herunter und installieren Sie sie auf dem Server, auf dem Director ausgeführt wird. Starten Sie Studio oder GPMC. Die neuen und aktualisierten Richtlinien werden dann angezeigt.

Weitere Informationen zu den aktualisierten Richtlinien finden Sie unter folgenden Links: [Einstellungen der Richtlinie "Visuelle Anzeige"](#) für den verbesserten ThinWire-Kompatibilitätsmodus; [Einstellungen der Richtlinie "USB-Geräte"](#) für die Unterstützung von Signaturgeräten und Zeichentablets; [Flash-Umleitung](#) und [Richtlinieneinstellungen für Multimedia](#) zum Verhindern von Videofallback.

StoreFront 3.0.1

LTSR-Version: 3.0.1

Erstveröffentlichung dieser Version: 30. September 2015 als Teil von Feature Pack 3 (CitrixStoreFront-x64.exe)

[Systemanforderungen](#)

[Behobene Probleme](#)

[Bekannte Probleme](#)

Installation und Upgrade

Laden Sie die LTSR-Version von StoreFront auf den StoreFront-Server herunter und folgen Sie den [Upgradeanweisungen](#).

Provisioning Services 7.6 Update 1

LTSR-Version: Provisioning Services 7.6 Update 1 (Provisioning Services 7.6 Cumulative Update 1 für Server und Console); PVS760TargetDeviceWX64001.zip, PVS760TargetDeviceWX86001.zip

Erstveröffentlichung dieser Version: 15. September 2015 als PVS760ConsoleServerWX86001.zip; PVS760ConsoleServerWX64001.zip

Provisioning Services 7.6 Update 1 enthält Fixes für mehr als 40 Probleme in der 7.6-Basisversion.

[Systemanforderungen](#)

Behobene Probleme

- Konsole, Server ([64 Bit](#) | [32 Bit](#))
- Zielgeräte ([64 Bit](#) | [32 Bit](#))

Installation und Upgrade

Laden Sie die LTSR-Version von Provisioning Services herunter und folgen Sie den Installationsanweisungen unter [Installieren der Provisioning Services Console-Software](#) (Konsole), [Installieren der Serversoftware für die Provisioning Services](#) (Server) und in [CTX135746](#) (Zielgeräte).

Sitzungsaufzeichnung 7.6.100

LTSR-Version: 7.6.100

Erstveröffentlichung dieser Version: 30 Juni 2015 als Teil von Feature Pack 2 (SessionRecording7.6.100.zip)

Sitzungsaufzeichnung 7.6.100 wurde als Teil von Feature Pack 2 veröffentlicht und enthält die folgenden neuen Features und Verbesserungen.

- Sie können die Anmeldeinformationen für die Verbindung mit der Datenbank bei der Installation der Datenbank für die Sitzungsaufzeichnung angeben.
- Sie können während der Installation der Datenbank für die Sitzungsaufzeichnung und des Sitzungsaufzeichnungsservers die Verbindung mit der Datenbank und während der Installation des Sitzungsaufzeichnungsagents die Verbindung mit dem Sitzungsaufzeichnungsserver testen.
- Microsoft Shared Management Objects ist für die Datenbank der Sitzungsaufzeichnung nicht mehr erforderlich.
- Das Citrix Programm zur Verbesserung der Benutzerfreundlichkeit (CEIP) ist in die Sitzungsaufzeichnung integriert. Weitere Informationen finden Sie unter Citrix Programm zur Verbesserung der Benutzerfreundlichkeit. Während des Upgrades werden die vorhandenen Einstellungen beibehalten.

Systemanforderungen

Installation und Upgrade

Laden Sie die LTSR-Version herunter und folgen Sie den [Upgradeanweisungen](#).

Bekannte Probleme in Sitzungsaufzeichnung 7.6.100 bei Verwendung in einer LTSR-Bereitstellung:

- Aufzeichnungen von Microsoft Paint-Sitzungen werden im Sitzungsaufzeichnungsplayer nicht richtig wiedergeben. [#0604700]
- Bei der Wiedergabe einer Sitzung, die auf einem Benutzergerät mit mehreren Monitoren aufgezeichnet wurde, tritt ein Fehler auf. [#0605129]

Universeller Druckserver 7.6.300

LTSR-Version: 7.6.300

Erstveröffentlichung dieser Version: 30 September 2015 als Teil von Feature Pack 3 (UpsServer_7.6.300.zip)

Hinweis: Der universelle Druckserver umfasst Client- und Serverkomponenten. Die Clientkomponente wird als Teil der VDAs installiert, daher sind in der LTSR-Version keine Installationsdateien für den Client enthalten. Auf der Serverseite unterstützt LTSR den universellen Druckserver nicht auf 32-Bit-Windows-Betriebssystemen. Daher ist nur der Installer für 64-Bit-Server

enthalten.

Systemanforderungen

Behobene Probleme

Installation und Upgrade

Das Paket für den universellen Druckserver enthält aktualisierte Versionen der eigenständigen UPS-Serverkomponente (UpsServer_x64.msi) und die erforderlichen Dateien vcredist_x64.exe, vcredist_x86.exe und cdf_x64.msi.

1. Laden Sie die LTSR-Version auf einen Druckserver mit Windows 2008 R2 SP1, Windows Server 2012 oder Windows Server 2012 R2 herunter.
2. Installieren Sie die erforderlichen Dateien: vcredist_x64.exe, vcredist_x86.exe und cdf_x64.msi.
3. Installieren Sie die UPS-Komponente UpsServer_x64.msi.
4. Starten Sie den Server nach der Installation der UPS-Komponente neu.

Die UPClient-Komponente ist in der VDA-Installation enthalten. Daher müssen Sie die Clientkomponente nicht manuell installieren und sie ist nicht als eigenständige Komponente der LTSR-Version enthalten.

Weitere Informationen finden Sie unter [Bereitstellen von Druckern](#).

HDX Flash-Umleitung

Bei der HDX Flash-Umleitung wird die Verarbeitung der meisten Adobe Flash-Inhalte (einschließlich Animationen, Videos und Anwendungen) an über LAN- und WAN-angeschlossene Windows-Benutzergeräte übertragen, wodurch Server- und Netzwerklast verringert werden. Dies führt zu größerer Skalierbarkeit, während gleichzeitig eine High Definition-Benutzererfahrung sichergestellt wird.

Die clientseitigen Komponenten werden als Teil der Desktop- und Serverbetriebssystem-VDAs installiert. Beim Upgrade auf die LTSR-Versionen der VDAs wird die Bereitstellung daher auf die neueste Version von HDX Flash-Umleitung aktualisiert.

Es brauchen keine serverseitigen Komponenten installiert werden. Das Konfigurieren der Flash-Umleitung erfordert jedoch sowohl server- als auch clientseitige Einstellungen. Weitere Informationen zum Konfigurieren der Flash-Umleitung finden Sie unter [Flash-Umleitung](#). Aktuelle Updates für HDX Flash-Kompatibilität finden Sie unter [CTX136588](#).

Feature Pack 3: Installation und Upgrade

May 10, 2016

In diesem Artikel finden Sie Installationsanweisungen für die Komponenten in XenApp und XenDesktop 7.6 FP3. Detaillierte Anweisungen finden Sie unter [Systemanforderungen](#) und unter den Links zu Artikeln über Produkte.

Laden Sie die FP3-Komponenten unter [citrix.com](#) herunter. Die Verfügbarkeit hängt von Ihren aktuellen Berechtigungen ab.

Workstation OS VDA 7.6.300

Laden Sie **VDAWorkstationSetup.exe** auf die Maschine herunter, auf der Sie den VDA installieren möchten, und führen Sie die Datei aus. Verwenden Sie die grafischen Benutzeroberfläche oder die Befehlszeile.

Weitere Informationen finden Sie unter [Installieren von VDAs mit dem eigenständigen Paket](#).

Für ein Upgrade von VDAs auf Maschinen mit Windows 8.x oder Windows 7 bis Windows 10 spielen Sie ein neues Image von Windows 10 auf die Maschinen mit Windows 7 und 8.x auf und installieren Sie dann die VDAs für Windows 10 mit dem eigenständigen Installationspaket.

Weitere Informationen finden Sie unter [Upgrade einer Bereitstellung](#).

Server OS VDA 7.6.300

Laden Sie **VDAServerSetup.exe** auf die Maschine herunter, auf der Sie den VDA installieren möchten, und führen Sie die Datei aus. Verwenden Sie die grafische Benutzeroberfläche oder die Befehlszeile.

Der VDA für Windows-Serverbetriebssysteme stellt automatisch Microsoft Visual C++ 2013 Runtime (32 Bit und 64 Bit) sowie 2008 und 2010 Runtimes (32 Bit und 64 Bit) bereit. Microsoft Visual C++ 2005 wird nicht mehr bereitgestellt. Diese Voraussetzungen initiieren einen Serverneustart. Die VDA-Installation wird nach dem Neustart fortgesetzt.

Weitere Informationen finden Sie unter [Installieren von VDAs mit dem eigenständigen Paket](#).

Gruppenrichtlinienverwaltung 7.6.300

Die neuen und erweiterten Features der HDX-Technologie in diesem Release sind im VDA von Feature Pack 3 enthalten und werden mit dem aktualisierten Paket für die Gruppenrichtlinienverwaltung verwaltet:

Laden Sie die folgende Datei auf den Server herunter, auf dem Sie Citrix Richtlinien verwalten, und führen Sie sie aus: **CitrixGroupPolicyManagement_x64.msi** oder **CitrixGroupPolicyManagement_x86.msi**. Starten Sie Studio oder GPMC. Die neuen und aktualisierten Richtlinien werden dann angezeigt.

Weitere Informationen zu den aktualisierten Richtlinien finden Sie unter folgenden Links: [Einstellungen der Richtlinie "Visuelle Anzeige"](#) für den verbesserten ThinWire-Kompatibilitätsmodus; [Einstellungen der Richtlinie "USB-Geräte"](#) für die Unterstützung von Signaturgeräten und Zeichentablets; [Flash-Umleitung](#) und [Richtlinieneinstellungen für Multimedia](#) zum Verhindern von Videofallback; [Framehawk-Richtlinieneinstellungen](#) für den HDX Framehawk Virtual Channel.

Active Directory-Bereitstellungstools 7.6

Enthalten sind Skripts (**InstallVDA.bat** und **UninstallVDA.bat**) zum Installieren, Aktualisieren oder Entfernen von Virtual Delivery Agents (VDAs) für Gruppen von Maschinen in Active Directory. Die Skripts müssen vor der Verwendung modifiziert werden.

Weitere Informationen finden Sie unter [Installieren oder Entfernen von Virtual Delivery Agents mit Skripts](#).

Universeller Druckserver 7.6.300

Das UPS-Paket enthält aktualisierte Versionen der eigenständigen UPS-Serverkomponente (**UpsServer_x64.msi** und **UpsServer_x86.msi**) und Voraussetzungen (**vcredist_x64.exe**, **vcredist_x86.exe**, **cdf_x64.msi** und **cdf_x86.msi**).

Installieren Sie die UPSEServer-Komponente auf einem Druckerserver für Windows Server 2012 R2 oder Windows Server 2012, indem Sie die MSI der Komponente, UpsServer_x64.msi oder UpsServer_x86.msi, extrahieren und starten. Nach der Installation der UPSEServer-Komponente ist ein Neustart erforderlich.

Die UPClient-Komponente, die Sie auf XenApp- und XenDesktop-Hosts installieren, die Netzwerksitzungsdrucker bereitstellen, gehört zur VDA-Installation.

Zusätzliche Schritte sind in Umgebungen erforderlich, in denen Sie die UPClient-Komponente separat bereitstellen, z. B. mit XenApp 6.5.

Weitere Informationen zu den Konfigurationen in FP3 finden Sie unter [Bereitstellen von Druckern](#).

Zusätzliche Schritte sind zudem für die Bereitstellung von UpsServer_x86.msi auf der Windows 2008 32-Bit-Plattform erforderlich. Weitere Informationen finden Sie unter [Installieren an der Befehlszeile](#).

Director 7.6.300

Vergewissern Sie sich, dass Sie alle erforderlichen Features in IIS ausgewählt haben. Eine vollständige Liste finden Sie unter [CTX142260](#). Laden Sie **DesktopDirector.msi** herunter und führen Sie die Datei auf dem Server aus, auf dem Director ausgeführt wird. Installieren Sie die CitrixGroupPolicyManagement.msi, wenn sie nicht bereits vorhanden ist.

Verwenden Sie die im Paket enthaltenen WMIPProxy MSI-Dateien zum Installieren oder Aktualisieren von WMIPProxy auf dem VDA.

Verwenden Sie die im Paket enthaltene XDPoshSnapin MSI-Datei zum Aktualisieren von XDPoshSnapin_Hotfix auf dem Delivery Controller. Dies ist erforderlich, damit delegierte Administratoren und benutzerdefinierte Administratoren die Framehawk Virtual Channel-Informationen in Director anzeigen können.

Vollständige Installations- und Konfigurationsanweisungen finden Sie unter [Director](#).

AppDNA 7.6.5

AppDNA 7.6.5 enthält neue Windows 10-Berichte, mit denen Sie analysieren können, ob Ihre unter Windows XP oder höher ausgeführte Anwendung unter Windows 10 ausgeführt werden kann. Laden Sie den Installer **Citrix-AppDNA.msi** herunter und führen Sie ihn auf einer VM aus. Der Installer eignet sich für Erstinstallationen und Upgrades für alle Lizenzierungstypen. Bei Testversionen erstellt der Installer eine Datenbank, die mit Microsoft SQL Server Express kompatibel ist.

Weitere Informationen finden Sie unter [Install AppDNA](#).

StoreFront 3.0.1

Laden Sie für die Installation **CitrixStoreFront-x64.exe** herunter und führen Sie die Datei als Administrator auf dem StoreFront-Server aus. StoreFront 3.0.1 umfasst Citrix Receiver für HTML5 1.8.

Weitere Informationen finden Sie unter [Installieren, Einrichten und Deinstallieren](#).

Um das Upgrade durchzuführen, deaktivieren Sie den Zugriff auf die StoreFront-Bereitstellung und starten Sie den StoreFront-Server neu. Laden Sie **CitrixStoreFront-x64.exe** herunter und führen Sie die Datei als Administrator auf den StoreFront-Servern aus.

Weitere Informationen finden Sie unter [Upgrade](#).

Citrix Receiver für HTML5 1.8

Receiver für HTML5 wird auf StoreFront gehostet und eine Version von Receiver für HTML5 ist standardmäßig mit StoreFront verfügbar. StoreFront 3.0.1 umfasst Citrix Receiver für HTML5 1.8. Wenn Sie eine frühere Version von StoreFront ausführen und diese Version von Citrix Receiver für HTML5 installieren oder ein Upgrade darauf durchführen möchten, laden Sie Receiver für HTML5 1.8 herunter (**CitrixHTML5Client-x64.exe**) und führen Sie die Datei auf StoreFront 2.5 oder höher aus.

Weitere Informationen finden Sie unter [Konfigurieren von Citrix Receiver für HTML5](#).

Citrix Receiver für Windows 4.3.100

Receiver für Windows 4.3.100 ist in den FP3-Paketen mit VDA 7.6.300 eingeschlossen. Mit Citrix Receiver für Windows 4.3.100 (CitrixReceiver.exe) können Sie Receiver auf älteren VDAs aktualisieren oder auf Benutzergeräten installieren oder aktualisieren.

Wenn das Citrix Lync Optimization Pack auf dem Endpunktgerät installiert ist, muss es zuerst deinstalliert und nach dem Upgrade von Citrix Receiver für Windows neu installiert werden. Weitere Informationen finden Sie unter [CTX200340](#).

Weitere Informationen finden Sie unter [Installieren](#).

Linux Virtual Desktop 1.1

Bereiten Sie die Linux-VMs vor, installieren Sie die neue [Linux-VDA-Software](#) darauf, konfigurieren Sie den Delivery Controller und stellen Sie die Desktops den Benutzern mit Studio zur Verfügung.

Weitere Informationen finden Sie unter [Linux Virtual Desktop Installation Guide for SUSE](#) und unter [Linux Virtual Desktop Installation Guide for Red Hat](#).

XenServer 6.5 SP1 Hotfix für zertifizierte Windows 10-Treiber

Dies ist ein Hotfix für Kunden, die XenServer 6.5.0 Service Pack 1 ausführen. Dieser Hotfix bietet Windows 10-Zertifizierung für die Windows-Standardtreiber.

Weitere Informationen finden Sie unter [CTX201974](#).

Feature Pack 3: Behobene Probleme

Jul 13, 2016

Citrix Director 7.6.300; 7.6 LTSR

- Der Aktivitätsmanager in Director zeigt möglicherweise einigen Benutzern die geöffneten Anwendungen nicht an.
[#LC2325]
- Wenn mehrere Sites konfiguriert sind, werden die Director-Sitenamen u. U. abgeschnitten.
[#LC2967]

Citrix Universeller Druckserver 7.6.300; 7.6 LTSR

[Universeller Druckclient](#)

[Universeller Druckserver](#)

Probleme beim universellen Druckclient

- Bei der Verwendung von Netzwerkdruckern ist die Druckvorschau von Office-Dokumenten u. U. langsam.
[#LC1205]
- Bei Servern mit XenApp 6.5 Hotfix Rollup Pack 4 und dem universellen Druckclient 7.6 reagiert die Sitzung auf dem Benutzergerät möglicherweise nicht mehr.
[#LC1562]
- Diese Funktionserweiterung führt Leistungsindikatoren für den universellen Druckserver und den universellen Druckclient ein.
[#LC1820]
- Beim Drucken von Adobe Reader X1 dauert es u. U. 20 Sekunden, bis das Druckdialogfeld angezeigt wird.
[#LC2412]
- Gelegentlich stoppt der Citrix Druckmanagerdienst (cpsvc.exe) unerwartet, wenn sich Benutzer an einem VDA anmelden.
[#LC2716]
- Wenn der universelle Druckserver auf einem VDA installiert aber durch Citrix Richtlinien deaktiviert ist, schlagen PowerShell get-printer-Befehle auf dem VDA u. U. fehl.
[#LC2966]
- Gelegentlich tritt auf UpProv.dll des Druckspoolerdiensts eine schwerwiegende Ausnahme auf.
[#LC3055]

Probleme beim universellen Druckserver

- Dieser Fix behebt ein Speicherproblem in einer Hintergrundkomponente.

[#LC1381]

- Diese Funktionserweiterung führt Leistungsindikatoren für den universellen Druckerserver und den universellen Druckclient ein.

[#LC1820]

- Dieser Fix behebt ein Speicherproblem in einer Hintergrundkomponente.

[#LC2764]

VDA für Desktopbetriebssysteme 7.6.300, 7.6 LTSR, 7.7

Die Versionen der VDAs für Desktopbetriebssysteme, die in XenApp und XenDesktop 7.6 Feature Pack 3 (7.6.300) und 7.6 LTSR enthalten sind, sind identisch. Die in XenApp und XenDesktop 7.7 enthaltene Version enthält die gleichen Fixes wie Version 7.6.300 sowie Kompatibilitätsupdates für XenApp und XenDesktop 7.7.

Broker Agent

Remote PC

HDX 3d Pro

Seamlessfenster

HDX MediaStream Flash-Umleitung

Sitzung/Verbindung

HDX MediaStream Windows Media-Umleitung

Site-/Farmverwaltung

Installieren, Deinstallieren und Aktualisieren

Smart cards

Tastatur/Maus

Systemausnahmen

Lokaler App-Zugriff

Benutzererfahrung

Anmeldung und Authentifizierung

Benutzeroberfläche

Drucken

Sonstiges

Broker Agent

- VDAs können während des Registrierungsvorgangs bei der Initialisierung hängen bleiben. Das Problem tritt auf, wenn der Citrix Desktopdienst mehrere Tage ohne Neustart ausgeführt wird.

[#LC0570]

- Ist die Funktion "CName" aktiviert, kann die VDA-Registrierung übermäßig lange dauern. Durch diesen Fix wird beim Aktivieren des Werts "CNameSuppressOrgFqdnLookup" der Registrierungsprozess schneller.
Wichtig: Bei der Aktivierung von "CNameSuppressOrgFqdnLookup" müssen Sie auch "CName" aktivieren.

Zum Implementieren dieses Fixes erstellen Sie folgenden Registrierungsschlüssel:

- HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\VirtualDesktopAgent
Name: UseCnameLookup
Typ: REG_DWORD
Wert: Aktiviert = 1 (Deaktiviert = 0)
- HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\VirtualDesktopAgent
Name: CNameSuppressOrgFqdnLookup
Typ: REG_DWORD
Wert: Aktiviert = 1 (Deaktiviert = 0)

[#LC0791]

- Versuche, über eine Benutzersitzung die Verbindung mit einer VDA wiederherzustellen, schlagen fehl, wenn der VDI-in-a-Box-Modus aktiviert ist.

[#LC1522]

- Gehört ein Version 7.6 VDA zu einer XenDesktop 5.6-Site, schlagen Wiederverbindungsversuche fehl und der VDA ist nicht registriert, wenn Benutzer sich verbinden, die Verbindung trennen und dann wiederherstellen möchten.

[#LC1859]

HDX 3D Pro

- Laden Benutzer einen benutzerdefinierten Cursor in einem Desktop mit HDX 3D Pro, wird nicht die richtige Cursorfarbe angezeigt.
- Wird der Mauszeiger über die Toolboxleiste und den Zeichnungsbereich einer Drittanbieter-CAD-Anwendung bewegt, in der für die Desktops HDX 3D Pro aktiviert ist und Windows in vereinfachtem Chinesisch ausgeführt wird, verschwindet der Mauszeiger möglicherweise.

[#LC1321]

- Nach dem Trennen einer Citrix Receiver-Sitzung kann das Wiederverbinden mit einer gesperrten VDA-Sitzung auf einem Computer mit HDX 3D Pro und einer NVIDIA-Grafikkarte fehlschlagen.

[#LC1629]

- In HDX 3D Pro VDA-Sitzungen mit zwei Monitoren, die für 150% DPI kalibriert sind, können der Mauszeiger und andere Bildschirmeffekte außerhalb der VDA-Sitzung angezeigt werden und daher unzugänglich sein.

[#LC2502]

- Ist die Hardwarebeschleunigung aktiviert, schlägt AutoCAD 2011 in einer Receiver-Sitzung fehl, nachdem die Größe des CD Viewer-Fensters geändert wurde.

[#LC2531]

- Mit diesem Fix ist die Bandbreitenobergrenze für HDX-Netzwerkdatenverkehr 20 MBit/s.

[#LC2780]

- Grafiken können in VDA-Sitzungen in HDX 3D Pro-Umgebungen mit einem NVidia Grid K1-Grafikboard verschwommen sein, wenn der VDA mit einem WAN-Netzwerk mit hoher Latenz verbunden ist.

[#LC2877]

- Auf VDAs, auf denen HDX 3D Pro aktiviert ist, zeigt das Taskleistensymbol für "Verlustfrei" (für LLIndicator.exe) möglicherweise nicht den Status an und reagiert bei einem Klick mit der rechten Maustaste nicht.

[#LC2931]

- Der Mauszeiger verschwindet möglicherweise nachdem ein Benutzer eine Sitzung einige Mal sperrt und entsperrt auf einem HDX 3D Pro-Desktop.

[#LC3188]

- Wenn der Desktop Viewer vom Vollbildmodus in den Fenstermodus umgeschaltet wird, reagiert die HDX 3D Pro-Sitzung möglicherweise einige Minuten nicht.

[#LC3193]

- HDX 3D Pro startet u. U. unerwartet auf dem VDA, obwohl es nicht installiert wurde. Dies kann passieren, wenn Windows XP Display Driver Model (XPDM) statt Windows Display Driver Model (WDDM) auf dem VDA installiert ist.

[#LC3215]

HDX MediaStream Flash-Umleitung

- Mit Browserversionen Microsoft Internet Explorer 9 und höher werden nach dem Klicken auf einen Link zu einem umgeleiteten Flash-Video während bereits ein anderes umgeleitetes Flash-Video abgespielt wird, die Audiodaten der beiden Videos übereinander abgespielt.

Dieser Fix verhindert das Problem nicht, aber wenn Sie den folgenden Registrierungsschlüssel auf dem Server setzen, wird die Zeit verkürzt, für die das Problem auftritt:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\HdxMediaStreamForFlash\Server\PseudoServer
 Name: UrlListForTerminationFlashInst
 Typ: REG_MULTI_SZ
 Daten:

[#LC0453]

- Mit Microsoft Internet Explorer 9 oder höher kann die Flash-Umleitung für Videos fehlschlagen, die in Websites eingebettet sind. Ein schwarzes Rechteck wird statt des Videos angezeigt.

Zum Implementieren dieses Fixes legen Sie folgenden Registrierungsschlüssel fest:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\HdxMediaStreamForFlash\Server\PseudoServer
 Name: WorkWithAeroEnabled
 Typ: REG_DWORD
 Wert: 0

[#LC0477]

- Nachdem Sie das integrierte Suchfeld von YouTube in Microsoft Internet Explorer 8 und höher verwendet haben, wird die

HDX Flash-Umleitung auf die serverseitige Wiedergabe zurückgesetzt.

Um diesen Fix zu nutzen, müssen Sie den folgenden Registrierungsschlüssel auf dem Server setzen:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\HdxMediaStreamForFlash\Server\PseudoServer

Name: SupportedUrlHeads

Typ: REG_MULTI_SZ

Wert:

http://

https://

file://

Hinweis: SupportedUrlHeads wurde zuerst mit Fix LA4151 eingeführt. Er wird auch von diesem Fix verwendet. Wenn Sie der Registrierung SupportedUrlHeads hinzufügen, ignoriert die Flash-Umleitung Instanzierungsanfragen, es sei denn, die Ziel-URL beginnt mit einem der Header, die in SupportedUrlHeads angegeben wurden.

[#LC0505]

- Flash-Umleitung schlägt fehl für Videos, die in cnn.com und msn.com eingebettet sind. Ein schwarzes Rechteck wird statt des Videos angezeigt.

Um diesen Fix zu aktivieren, setzen Sie die folgenden Registrierungsschlüssel:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\HdxMediaStreamForFlash\Server\PseudoServer\ASJSCallbacks

Für cnn.com:

Name: cnn.com

Typ: REG_MULTI_SZ

Wert:	javaScriptLoaded
	setQueueAutoplay
	setAdVisibility
	setPlayerMode
	setAdSection
	setAdKeyValue
	playContent
	getContentEntry
	switchTrackingContext

Für msn.com:

Name: msn.com

Typ: REG_MULTI_SZ

Wert:	MsnVideoCallback
	MsnVideoPropertyCallback
	GetWidgetID

Hinweis: Dieser Fix funktioniert nur für Internet Explorer 9 und höher.

[#LC0665]

- Mit Microsoft Internet Explorer 10 kann die Flash-Umleitung für Videos fehlschlagen, die in cnn.com-Websites eingebettet sind. Dies passiert, wenn das erste Video auf einer Website automatisch abgespielt wird. Manche Inhalte der Seite für dieses Video, wie Titel, Status und Werbung fehlen möglicherweise.

[#LC0830]

- Flash-Umleitung schlägt fehl für Videos, die in m.mlb.com und msn.com eingebettet sind. Ein schwarzes Rechteck wird statt des Videos angezeigt.

Um diesen Fix zu aktivieren, setzen Sie die folgenden Registrierungsschlüssel:

Für mlb.com:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\HdxMediaStreamForFlash\Server\PseudoServer\ASJSCallbacks

Name: m.mlb.com

Typ: REG_MULTI_SZ

Wert:

setPlaylist

startPlaylist

Für msn.com:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\HdxMediaStreamForFlash\Server\PseudoServer\ASJSCallbacks

Name: msn.com

Typ: REG_MULTI_SZ

Wert:

MsnVideoCallback

MsnVideoPropertyCallback

GetWidgetID

Dieser Fix funktioniert nur für Internet Explorer 9 und höher. Damit dies funktioniert, stellen Sie für Internet Explorer den Emulationsmodus und für Internet Explorer Version 9 und 10 den Dokumentmodus ein. Weitere Informationen finden Sie unter [CTX136588](#) auf der Citrix Supportsite.

[#LC1043]

- Internet Explorer wird möglicherweise unerwartet geschlossen bei dem Versuch, ein Video von Websites abzuspielen, für die gilt:
 - Die Option "Flash-Beschleunigung deaktivieren" ist aktiviert oder die Site steht auf der Sperrliste.
 - Der Benutzer verwendet eine Receiver-Version, die die Flash-Umleitung nicht unterstützt, z. B. Receiver für Mac OS X, iPad, iPhone und Android.

[#LC2256]

HDX MediaStream Windows Media-Umleitung

- Dieses Release behebt das Problem, dass überflüssige Frames vom HDX-Grafikencoder generiert werden. So wird eine gleichmäßigere Framerate und bessere Interaktivität erreicht.

[#LC2177]

- Wenn eine Videodatei in einer VDA-Sitzung mit Receiver für Windows Version 13.3 oder 13.4 abgespielt wird, geschieht die Synchronisierung von Audio und Video in einer benutzerdefinierten Anwendung. Nach dem Upgrade auf eine neuere Version von Receiver schlägt die Windows Media-Umleitung fehl. Nachdem dies passiert können Benutzer nur Audio hören oder nur Video sehen.

Um diesen Fix zu aktivieren, müssen Sie Citrix Receiver für Windows 4.3 verwenden.

[#LC2516]

- Versuche, Windows Media Player zu schließen, während ein Video angehalten ist, kann dazu führen, dass Windows Media Player nicht mehr reagiert.

[#LC0728, #LC3410]

Installieren, Deinstallieren und Aktualisieren

- Frühere Updates für VDA für Desktopbetriebssysteme versuchten, ein WDDM-Treiber zu installieren. In Szenarios, in denen der Schalter /nocitrixwddm während der ursprünglichen Installation des VDAs verwendet wurde, um die Installation des WDDM-Treibers zu unterdrücken, gerät der Installationsprozess bei der Installation eines Updates in eine Schleife, in der der VDA immer wieder neu gestartet wird. Mit diesem Fix wird kein Versuch unternommen, den Treiber zu installieren, wenn Sie ein Update für ein VDA ausführen, für den die Treiberinstallation ursprünglich unterdrückt wurde.

[#LC0443]

- Installieren eines VDA-Hotfixes, der ICAWS750WX64012 veröffentlicht wurde, setzt den Status der Dienste "Citrix Remote Multi-Touch Device" und "Citrix Multi-Touch Redirection Service" auf die Standardeinstellungen zurück. Es wird nicht beachtet, ob Sie sie vorher manuell aktiviert oder deaktiviert haben.

[#LC0446]

- Installieren von Hotfixes für XenApp 7.5 und XenDesktop 7.1 und 7.5 VDA Core Services für Windows Desktop- und Serverbetriebssysteme, die vor September 2014 veröffentlicht wurden, führt dazu, dass der Leistungsindikatorzähler für ICA-Sitzungen entfernt wird. Dies wirkt sich negativ auf Tools und Prozesse aus, die auf diesem Zähler basieren. Dieser Fix stellt durch die Installation früherer Hotfix entfernter Zähler wieder her und verhindert, dass sie durch die Installation nachfolgender Hotfixes entfernt werden.

[#LC0771]

- Der VDA wird möglicherweise nach einem Downgrade der VDAs von Version 7.6 auf 5.6.400 nicht registriert.

[#LC1465]

- Nach dem Upgrade von VDAs treten folgende Probleme auf:

- Der Registrierungseintrag EnableReadImageFileExecOptionsExclusionList wird in "outlook.exe" geändert.
- Outlook funktioniert nicht mit einer Smartcard.

Dieser Fix entfernt "outlook.exe" aus dem Registrierungseintrag EnableReadImageFileExecOptionsExclusionList. Alle andere Zeichenfolgen in diesem Registrierungseintrag bleiben erhalten.

[#LC1655]

- Wenn Benutzer ein VDA auf Version 5.6.400 oder 5.6.500 aktualisieren und dann Dateien von einem zugeordneten Clientlaufwerk zum VDA kopieren oder die Dateien direkt auf dem Laufwerk öffnen, wird folgende Fehlermeldung angezeigt:

"Ungültiges Handle."

[#LC1910]

- Versuche, den Druckmanagerdienst über das Snap-In "Dienste" neu zu starten, schlagen fehl. Es wird eine Fehlermeldung angezeigt, dass der Dienst nicht angehalten werden kann, und der Status des Druckmanagerdiensts ist "Wird beendet".

[#LC2122]

- Nach dem VDA-Upgrade von XenDesktop 7.5 auf XenDesktop 7.6 wird Fehler 1030 angezeigt, wenn Benutzer eine Verbindung herstellen möchten. In dem Systemereignisprotokoll erscheint folgende Meldung: "The Citrix ICA Transport Driver received an invalid Transport packet on port 2598."

[#LC2501]

- Frühere Updates für VDA für Desktopbetriebssysteme versuchten, ein WDDM-Treiber zu installieren. In Szenarios, in denen der Schalter /nocitrixwddm während der ursprünglichen Installation des VDAs verwendet wurde, um die Installation des WDDM-Treibers zu unterdrücken, gerät der Installationsprozess bei der Installation eines Updates in eine Schleife, in der der VDA immer wieder neu gestartet wird. Mit diesem Fix wird kein Versuch unternommen, den Treiber zu installieren, wenn Sie ein Update für ein VDA ausführen, für den die Treiberinstallation ursprünglich unterdrückt wurde.

[#LC3052]

- In seltenen Fällen kann die Installation oder Neuinstallation des MSP-Hotfixpakets dazu führen, dass der Citrix Anzeigetreiber (Citrix Systems – WDDM) zur Löschung markiert wird. In diesem Fall schlägt die Installation fehl und die VDA-Anzeigeauflösung ist unbrauchbar.

[#LC3312]

Tastatur/Maus

- Die automatische Tastaturanzeige funktioniert möglicherweise nicht, wenn OpenOffice-Anwendungen auf einem iPad-Gerät verwendet werden.

[#LC2717]

Lokaler App-Zugriff

- In einer Desktopsitzung im Vollbildmodus, in der lokaler App-Zugriff aktiviert und die Einstellung "Fensterinhalt beim Verschieben anzeigen" deaktiviert ist, werden Bilder in hoher Auflösung verzerrt, wenn Anwendungssymbole über Bilder in Internet Explorer gezogen werden.

[#LC1722]

Anmeldung und Authentifizierung

- Die Fehlermeldung "Keine gültigen Zertifikate gefunden" wird angezeigt, wenn Benutzer versuchen, eine Anwendung über StoreFront mit Smartcard-Authentifizierung zu starten. Dadurch wird verhindert, dass Benutzer sich anmelden können.

[#LC1704]

Drucken

- Der Prozess für den Citrix Druckmanagerdienst (CpSvc.exe) wird möglicherweise unerwartet beendet.

[#LA5682, #LC1770]

- Sitzungsdruckerrichtlinien, die eingestellt werden ohne Netzwerkdrucker hinzuzufügen, können das Erstellen von Sitzungsdruckern blockieren, die in anderen Sitzungsdruckerrichtlinien angegeben werden.

[#LC0705]

- Nachdem der Standarddrucker in einem VDA geändert wurde, erscheint der vorherige Standarddrucker in dem VDA, wenn der VDA neu gestartet wird und Benutzer sich nicht von der ICA-Sitzung abgemeldet haben.

[#LC0898]

- Dieser Fix macht die Richtlinie "Legacydruckernamen" wieder verfügbar, die vorher nicht mehr in Version 7.x des Produkt enthalten war.

[#LC1002]

- Der Prozess für den Citrix Druckmanagerdienst (CpSvc.exe) wird möglicherweise unerwartet beendet, wenn der Wert für "pServerName" nicht richtig ist.

[#LC1241]

- Wenn die Richtlinieneinstellung "Generischen universellen Drucker automatisch erstellen" aktiviert ist, erstellt der Citrix Druckmanagerdienst (CpSvc.exe) den generischen Citrix Universellen Drucker nicht.

[#LC1255]

- Diese Funktionserweiterung führt Leistungsindikatoren für den universellen Druckerserver und den universellen Druckclient ein.

[#LC1820]

- Die erste Seite wird möglicherweise leer gedruckt, wenn der universelle Druckertreiber von Citrix verwendet wird.

[#LC2771]

- Die Richtlinien "Universelles Drucken - EMF-Verarbeitungsmodus" bleibt auf den Standardwert eingestellt, selbst wenn Sie sie auf "EMF-Datensätze für Drucker neu verarbeiten" eingestellt haben.

[#LC2521]

- Benutzer bemerken möglicherweise sporadisch Verzögerungen beim Drucken von Dokumenten oder Wählen eines anderen Druckers. Für bis zu eine Minute lang erscheint eine Meldung, dass eine Verbindung mit dem Drucker hergestellt wird.

[#LC2701]

- Nach dem Anwenden von Fix LC0898 und Konfigurieren von Sitzungsroaming schlägt das Erstellen von Clientdruckern und Sitzungsdruckern fehl.

[#LC3490]

Remote PC

- Wenn Benutzer eine Receiver-Sitzung starten und es bestehende Remotedesktop- und Konsolensitzungen zu dem gleichen RemotePC VDA gibt, schlägt die Receiver-Sitzung mit einer Wiederverbindenmeldung fehl. Diese Meldung wird angezeigt bis Benutzer manuell die Receiver-Sitzung schließen.

[#LC0627]

- Wenn Sie sich nach einem Neustart an einer physischen Remote PC-Konsole innerhalb von etwa 30 Sekunden nach der Anzeige des Anmeldebildschirms anmelden, können Sie keine Remote PC-ICA-Sitzung zu dem VDA herstellen, so lange die ursprüngliche Sitzung andauert.

[#LC3603]

Seamlessfenster

- Die Größe maximierter Seamlessanwendungen wird nicht richtig angepasst, wenn eine angedockte Anwendung verschoben wird, z. B. die Windows-Taskleiste.

[#LC1342]

- Das Seamlessfenster wird nicht angezeigt, obwohl der Anwendungsprozess ausgeführt wird und die Sitzung auf dem Server-VDA vorhanden ist.

[#LC1728]

Sitzung/Verbindung

- Beim Wiederverbinden mit einer VDA-Sitzung mit einer anderen Verbindungsmethode als für die anfängliche Sitzung (z. B. mit und ohne Access/NetScaler Gateway) werden Citrix Richtlinien nicht für die wiederverbundene Sitzung neu evaluiert.

Hinweis: Für diesen Fix müssen Sie alle Einträge an den folgenden Stellen auf dem Gerät mit Windows-Desktopbetriebssystem oder Windows-Serverbetriebssystem löschen. Wenden Sie diesen Fix nur an, wenn Sie wie beschrieben von dem Problem betroffen sind:

- Registrierung: HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Citrix
- Ordner: C:\ProgramData\Citrix\GroupPolicy*.*

[#LA5051]

- In Szenarios mit Endpunkten A und B und zwei Monitoren, die identische Bildschirmauflösung haben, werden alle Hauptfenster auf dem primären Monitor neu positioniert, wenn die Verbindung von Endpunkt A getrennt und von Endpunkt B wiederhergestellt wird.

[#LA5440]

- Wenn eine veröffentlichten Anwendung geöffnet und dann der Befehl explorer.exe ausgeführt wird, um Windows Explorer zu öffnen, kann es passieren, dass der Desktop statt Windows Explorer geöffnet wird.

Hinweis: Nachdem Sie diesen Fix installieren, wird der Prozess explorer.exe möglicherweise unerwartet beendet. Weitere Informationen finden Sie im Knowledge Center-Artikel [CTX128009](#).

[#LC0285]

- Beim Wiederverbinden mit einer Receiver-Sitzung kann die CPU-Auslastung hoch sein, wenn viele Windows Explorer-Instanzen auf einem VDA ausgeführt werden. Außerdem werden minimierte Fenster sporadisch maximiert, bevor sie wieder in die Windows-Taskleiste minimiert werden.

[#LC0315]

- Beim Anschließen eines USB-Lesegeräts für Magnetkarten wird das Gerät im virtuellen Desktop erkannt, aber es werden nicht die richtigen Treiber geladen.

[#LC0360]

- Drücken Sie die Tastenkombination Windows+Pfeil-nach-Links/Rechts, um eine Anwendung an den Bildschirmrand zu verschieben, und trennen dann die Sitzung, wird beim Wiederverbinden mit der Sitzung die Größe der Anwendung geändert. Das Ausmaß der Größenänderung ist je nach Anwendung unterschiedlich und der Effekt ist auf den nicht-primären Monitoren stärker.

[#LC0386]

- Gelegentlich reagieren Benutzersitzungen nicht mehr und können nicht abgemeldet werden. Dies passiert auch, wenn Benutzer sich manuell von der Konsole abmelden.

[#LC0472]

- Wenn die Einstellung "Geräteverbindungen vereinfachen" in einer USB-Umleitungsrichtlinie deaktiviert ist, funktionieren SafeNet UKey-USB-Geräte möglicherweise nicht.

[#LC0480]

- Wenn Benutzer eine veröffentlichte Version von Visual Basic Editor über die Taskleiste schließen, werden geöffnete Excel-Tabellen auch geschlossen.

[#LC0534]

- Wird eine SD-Karte in eine Benutzergerät eingelegt und dann eine VDA-Sitzung gestartet, erscheint die Karte in Windows-Explorer als Diskette, wenn sie durch die Clientlaufwerkzuordnung umgeleitet wurde.

Zum Implementieren dieses Fixes legen Sie folgende Registrierungsschlüssel fest:

- 32-Bit-Systeme:
HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\UNCLinks
Name: DisableFloppyIcon
Typ: REG_DWORD
Wert: 1
- 64-Bit-Systeme:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\UNCLinks

Name: DisableFloppyIcon

Typ: REG_DWORD

Wert: 1

[#LC0616]

- Versuche, das Kombinationsfeld-Steuerelement unter "Region und Sprache" > "Datums- und Uhrzeitformate" remote zu bedienen, schlagen möglicherweise fehl. Dies kann passieren, wenn das Kombinationsfeld-Steuerelement bereits einmal erfolgreich bedient wurde und dann ein zweiter Versuch gemacht wird.

[#LC0836]

- Umgeleitete COM-Portgeräte funktionieren möglicherweise nicht in einer Receiver-Sitzung.

[#LC0851]

- Ein Fenster "Verbindung aufgebaut. Funktionalität wird ausgehandelt" wird in Receiver angezeigt, wenn der VDA nicht mehr reagiert. Um andere Sitzungen erfolgreich zu starten, muss der VDA neu gestartet werden.

[#LC0909]

- Die Zeitzone ist nicht richtig, wenn sich Benutzer mit Receiver für Windows anmelden. Damit dieser Hotfix korrekt funktioniert, müssen Sie Folgendes installieren:
 - Den gleichen Microsoft-Zeitzoneupdatehotfix auf dem Benutzergerät und dem Server. Beispielsweise wenn Microsoft Hotfix [KB2998527](#) auf dem Benutzergerät installiert ist, installieren Sie dieses Hotfix auch auf dem Server.
 - Microsoft Hotfix [KB2870165](#) auf dem Server, wenn das Serverbetriebssystem Windows Server 2008 R2 Service Pack 1 ist.
 - Fix LC1392 ist auf dem Benutzergerät installiert.

[#LC1061]

- Wenn die USB-Umleitungsrichtlinie mehr als 1000 Zeichen hat, werden alle USB-Laufwerke umgeleitet, selbst wenn es für das Gerät eine Verweigerungsregel gibt.

[#LC1153]

- Ist ExcelHook aktiviert und sind mehrere Projekte in Microsoft Project 2013 geöffnet, erscheint das zuerst geöffnete Projekt möglicherweise nicht in der Taskleiste.

[#LC1158]

- Wird die Desktopgestaltungsumleitung (DCR) für WAN-Verbindungen verwendet, reagieren Webseiten oder Dokumente mit Text und Grafikinhalten möglicherweise nicht mehr und verbrauchen sehr viel Bandbreite.

[#LC1256]

- Die endgültige Bildqualität ist möglicherweise schlechter, wenn die Einstellungen "Zu verlustfrei verbessern" und "Visuell verlustfreie Komprimierung zulassen" auf dem Desktop Delivery Controller (DDC) aktiviert sind.

[#LC1271]

- Wenn sich Benutzer das erste Mal anmelden, wird der Bildschirm in einer Receiver-Sitzung möglicherweise beschädigt angezeigt. Das Problem tritt auf, wenn Sie die Bildqualität auf "Zu verlustfrei verbessern" ändern und die Einstellung "Visuell verlustfreie Komprimierung zulassen" aktivieren.

[#LC1272]

- In einer Umgebung mit zwei Monitoren für eine Server-VDA-Sitzung reagiert der primäre Monitor möglicherweise nicht mehr, wenn der sekundäre Monitor angeschlossen oder entfernt wird.

[#LC1359]

- Wenn Benutzer versuchen, sich mit einer getrennten Sitzung zu verbinden, kann es bis zu drei Minuten dauern.

[#LC1376]

- Benutzer können nicht von einer Remotedesktopsitzung (RDP) zu einer Receiver-Sitzung wechseln.

[#LC1383]

- Die Umleitung funktioniert möglicherweise nicht korrekt für das MailTo-Feature in XenDesktop- und XenApp-Sitzungen. Klickt der Benutzer auf den MailTo-Link im Webbrowser, wird nicht die veröffentlichte Instanz von Outlook geöffnet sondern ein Fehler angezeigt.

[#LC1400]

- Beim Starten von veröffentlichten Anwendungen kann Citrix Grafikprozess (CtxGfx.exe) dazu führen, dass der CPU-Verbrauch höher als normal ist.

[#LC1404]

- Ist Sitzungsfortbestehen aktiviert und eine Anwendung wird auf einem iOS- oder Android-Gerät geschlossen, erscheint ein schwarzes Fenster auf dem Gerätebildschirm und bleibt offen.

[#LC1407]

- Ist ein USB-Speichergerät auf einem Benutzergerät mit XenDesktop 5.6 verbunden, der Registrierungswert "UNCEnabled" auf "0" und der Registrierungswert für "InitialClientDriveLetter" auf "N" gesetzt, beginnen die Laufwerksbuchstaben der USB-Laufwerke in einer Sitzung mit dem Standardwert "V" statt "N".

Weitere Informationen finden Sie in den Knowledge Center-Artikeln [CTX127968](#) und [CTX122061](#).

[#LC1451]

- Melden sich Benutzer von einer Desktopsitzung ab, reagiert WFSHellWindow nicht mehr und die Meldung "Wfshell needs to close" wird drei Minuten lang angezeigt.

[#LC1591]

- Informationen bleiben auf dem Bildschirm sichtbar, nachdem der VDA in den Bildschirmschoner- oder Energiesparmodus wechselt. Erst wenn der Benutzer eine Eingabe macht (Maus oder Tastatur), wird die Sitzung mit einem leeren Bildschirm aktualisiert. Dies passiert, wenn die Bildschirmschoner- und Energiesparoption durch den DWORD-Wert HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\Graphics\SetDisplayRequiredMode = 0 aktiviert sind.

[#LC1650]

- Eine Drittanbieteranwendung reagiert möglicherweise in einer XenDesktop-Sitzung nicht mehr, wenn sie auf einen zugeordneten Dongle zugreift.

[#LC1657]

- Ist der Schlüssel UxPersistence in der Registrierung aktiviert, schlagen nach dem Trennen und Wiederverbinden einer Sitzung Versuche fehl, Anwendungen mit der Tastenkombination Windows+Pfeil-nach-rechts/links an den Bildschirmrand zu verschieben.

[#LC1684]

- Beim Wechseln von einer Receiver-Verbindung zu einer Remotedesktopsitzung (RDP) für VDI, schlägt die Wiederverbindung zu Receiver mit Status 1030 fehl, wenn Benutzer im Transferanmeldedialogfeld für die RDP-Verbindung mit "Nein" antworten.

[#LC1696]

- Melden sich Benutzer in XenCenter oder der vSphere-Konsole von XenDesktop ab, erscheint die Benutzerdesktopsitzung für mehrere Sekunden.

[#LC1782]

- Melden sich Benutzer mit einer Smartcard an und ab und anschließend mit einem Benutzernamen und Kennwort wieder an, wird die Verbindung gestartet und fast sofort wieder getrennt. Dies geschieht, bis der Benutzer sich wieder mit der Smartcard anmeldet.

[#LC1831]

- Ist das Tastaturlayout des Benutzergeräts anders als das Tastaturlayout des VDAs, reagiert die Anzeige für etwa 60 bis 70 Sekunden nicht, während eine virtuelle Desktopsitzung von Receiver für Windows gestartet wird.

[#LC1939]

- Der CPU-Verbrauch des WMI-Anbieterdienstes kann in Windows Server- und Desktop-VDAs höher sein und dazu führen, dass die VDAs nicht reagieren. Außerdem schlägt die Receiver-Verbindung fehl.

[#LC2041]

- Die Bildschirmauflösung für die Sitzung wird möglicherweise nicht angepasst, wenn Benutzer eine Sitzung mit Citrix Receiver für Windows trennen und sich dann mit der gleichen Sitzung von einem Gerät mit einer kleineren Bildschirmauflösung wieder verbinden und dafür Citrix Receiver für Linux oder Citrix Receiver für Windows CE verwenden. Dies kann passieren, wenn die Richtlinieneinstellung "Legacygrafikmodus" für die Sitzung aktiviert ist.

[#LC2079]

- Lotus Notes auf einem iPad mit Receiver für iOS wird möglicherweise unerwartet geschlossen.

[#LC2124]

- In einer XenDesktop-Sitzung wird ein Meldungsfeld angezeigt, dass ein Dienst eine Meldung anzeigen möchte. Klickt der Benutzer, um die Meldung anzuzeigen, zeigt die XenDesktop-Sitzung einen schwarzen Bildschirm. Das Problem tritt auf, wenn der XenDesktop VDA bereits Sitzung 0 verwendet.

[#LC2160]

- Sind Legacygrafiken aktiviert, wird der Schatteneffekt für den Mauszeiger nicht automatisch deaktiviert.

[#LC2222]

- Wenn Benutzer die Verbindung zu einer Sitzung trennen verursacht Citrix WinFrame Shell (WFShell.exe) eine hohe CPU-

Auslastung.

[#LC2235]

- Melden sich Benutzer von innerhalb einer VDA-Desktopsitzung am Webinterface an und versuchen dann, den gleichen Desktop zu starten, wird die Sitzung möglicherweise getrennt und der VDA reagiert nicht mehr. Benutzer können möglicherweise den Desktop nicht starten und ein Neustart des VDAs ist erforderlich.

[#LC2267]

- Ist die Standardunicodeingabesprache des Benutzergeräts anders als die Eingabesprache des VDAs, reagiert die Sitzung möglicherweise nicht mehr während der Sitzungsdesktop geladen wird.

[#LC2287]

- Der Zeitstempel wird nicht aktualisiert, wenn eine Datei von einem zugeordneten Clientlaufwerk in einer veröffentlichten Anwendung geöffnet und nach dem Bearbeiten gespeichert wird.

[#LC2305]

- Wfshell.exe verwendet weiter eine große Menge an CPU-Ressourcen nachdem Benutzer die Sitzung trennen.

[#LC2391]

- Ist die USB-Umleitung aktiviert, schlagen Versuche fehl, ein Wacom-Tablet und Stift zu verwenden.

[#LC2409]

- Ist das Tastaturlayout des Benutzergeräts anders als das Tastaturlayout des VDAs, reagiert die Sitzung möglicherweise nicht mehr während der Sitzungsdesktop geladen wird.

[#LC2466]

- Nach dem Trennen von Receiver schlagen Verbindungsversuche mit Remotedesktop (RDP) fehl, wenn RDP 8 aktiviert ist und die Authentifizierung auf Netzwerkebene (NLA) nicht verwendet wird.

[#LC2480]

- Verbinden sich Benutzer über NetScaler Gateway mit Single Sign-On an StoreFront, wobei ICA-Proxy aktiviert ist, und melden sich ab und wieder an, funktionieren VDA-Richtlinien nicht. Trennen die Benutzer die Verbindung und stellen sie dann wieder her, funktionieren die Richtlinien.

[#LC2577]

- Ist Excelhook aktiviert, geht beim Wechseln zwischen geöffneten Arbeitsmappen der Fokus der ausgewählten Arbeitsmappe verloren. Außerdem können Benutzer keine der geöffneten Arbeitsmappen wählen, der Fokus bleibt auf der zuletzt gewählten Datei.

#LC2589]

- Benutzer können nach dem Update der Registrierungsschlüssel für Verschlüsselungsverfahren keine Anwendungen starten. Um diesen Fix anzuwenden, müssen Sie eine neue Version des SSL SDKs installieren.

[#LC2729]

- Ist "User Experience Persistence" aktiviert, können Anwendungsfenster im falschen Zustand sein, z. B. maximiert statt wiederhergestellt, wenn Benutzer die Verbindung zu einem VDA wiederherstellen.

[#LC2769]

- Wird ein Point-of-Sales-Gerät an einen freigegebenen Desktop über den COM-Port umgeleitet, erkennt möglicherweise Drittanbietersoftware, die auf dem freigegebenen Desktop ausgeführt wird, das Gerät nicht.

[#LC2775]

- Wird eine veröffentlichte Version von Internet Explorer in Windows Phone 8.1 geöffnet und das Gerät horizontal gedreht, zeigt der halbe Bildschirm ein leeres Fenster. Außerdem können Benutzer das Internet Explorer-Fenster nicht maximieren. Drehen Benutzer das Gerät wieder vertikal, erscheint der Bildschirm normal.

[#LC2786]

- Microsoft Office-Anwendungen werden möglicherweise unerwartet geschlossen, während sie in einem VDA für Windows-basierte Desktopbetriebssysteme ausgeführt werden.

[#LC2878]

- Beim Verwenden eines externen Monitors mit einem Laptop, kann ein weißes Quadrat in einer Ecke des externen Monitors sichtbar sein. Das Problem tritt speziell auf, wenn der externe Monitor eine andere Bildschirmauflösung als der primäre Monitor hat, links neben dem primären Monitor steht und seine obere Kante über die obere Kante der primären Monitors hinausragt.

[#LC3186]

- Microsoft Office-Anwendungen werden gelegentlich unerwartet aufgrund von Interaktionen mit Citrix API-Hooks beendet.

[#LC3280]

- Bei deaktivierter USB-Umleitung kann das Trennen von einer veröffentlichten Anwendung dazu führen, dass fälschlicherweise das folgende Ereignis 261 im Ereignisprotokoll erscheint:

Der Citrix Geräteumleitungsdienst konnte einen E/A-Vorgang mit dem Umleitungsbus nicht beenden.

[#LC3439]

Site-/Farmverwaltung

- Benutzerdefinierte Administratoren, die alle Berechtigungen haben, können keinen Server aus der Farm entfernen, der offline ist, und erhalten eine Zugriff-Verweigert-Meldung, wenn sie den PowerShell-Befehl "Remove-XAServer" ausführen.

- [#LC0752]

- Die Richtlinienverarbeitung dauert möglicherweise eine lange Zeit, wenn Server nur auf einen schreibgeschützten Domänencontroller zugreifen können.

[#LC0828]

- Verwenden Sie RES-Software zum Verwalten der Bereitstellung und die Richtlinie "Instant LogOff" ist aktiviert, werden Sitzungsinformationen möglicherweise nicht im Citrix Brokerdienst auf dem XenDesktop-Controller aktualisiert, wenn Benutzer Sitzungen abmelden. Der Controller zeigt dann nicht den richtigen Sitzungsstatus an.

[#LC1173]

- Dieser Fix ermöglicht, permanente Registrierungseinstellungen für Benutzergeräte zu erstellen.

[#LC1827]

- Bei hoher Latenz kann die Citrix Gruppenrichtlinienengine (CitrixCseEngine.exe) Verzögerungen bei der Anmeldung verursachen.

Um den Fix zu aktivieren, erstellen Sie die folgenden Registrierungsschlüssel:

- HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\GroupPolicy
Name: GpoCacheEnabled
Typ: REG_DWORD
Wert: 1
- HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\GroupPolicy
Name: CacheGpoExpireInHours
Typ: REG_DWORD
Wert: 5 bis 24

[#LC1987]

Smartcards

- Nach dem Upgrade von VDAs fordert Microsoft Outlook nicht zum Eingeben einer PIN auf, wenn Benutzer sich mit einer Smartcard anmelden wollen. Die folgende Fehlermeldung wird angezeigt:

"Please insert a smart card."

[#LC0556]

- Nach dem Authentifizieren mit Citrix Receiver für Linux während eine Smartcard in das Lesegerät eingelegt ist, das an den Endpunkt angeschlossen ist, kann es passieren, dass die Sitzung nicht reagiert.

[#LC0982]

- Der Prozess ActiIdentity (accompks.exe) wird möglicherweise unerwartet beendet mit einer Ausnahme bei scardhook64.dll. Das Resultat ist, dass Websites für die eine Smartcard erforderlich ist, nicht funktionieren bis der VDA neu gestartet wird.

[#LC1056]

- Mit dieser Erweiterung führt das Entfernen eines Smartcardlesers von einem Endpunkt zu dem gleichen Verhalten wie das Entfernen einer Smartcard. Das Resultat ist, dass mit der VDA-Smartcardrichtlinieneinstellung "Arbeitsstation sperren" das Entfernen des Smartcardlesers die Sitzung trennt, wodurch effektiv der VDA gesperrt wird

Zum Implementieren dieses Fixes legen Sie folgende Registrierungsschlüssel fest:

- HKEY_LOCAL_MACHINE\Software\Citrix\SmartCard
Name: ReaderUnpluggedDetectEnable
Typ: REG_DWORD
Wert: Wert ungleich Null
- HKEY_LOCAL_MACHINE\Software\Citrix\SmartCard
Name: ReaderUnpluggedDetectPollingInterval
Typ: REG_DWORD

Wert: 5 bis 60 Sekunden (Standardabrufintervall ist 10 Sekunden)

[#LC1059]

- Dieser Fix erweitert die Funktionalität, die mit Fix LA1059 eingeführt wurde, sodass sie auch für Verbindungen mit Version 7.5 VDAs von Windows 8.1-Endpunkten gilt.

Erläuterung zu Fix LA1059:

Mit dieser Erweiterung führt das Entfernen eines Smartcardlesers von einem Endpunkt zu dem gleichen Verhalten wie das Entfernen einer Smartcard. Das Resultat ist, dass mit der VDA-Smartcardrichtlinieneinstellung "Arbeitsstation sperren" das Entfernen des Smartcardlesers die Sitzung trennt, wodurch effektiv der VDA gesperrt wird

Zum Implementieren dieses Fixes legen Sie folgende Registrierungsschlüssel fest:

- HKEY_LOCAL_MACHINE\Software\Citrix\SmartCard
Name: ReaderUnpluggedDetectEnable
Typ: REG_DWORD
Wert: Wert ungleich Null
- HKEY_LOCAL_MACHINE\Software\Citrix\SmartCard
Name: ReaderUnpluggedDetectPollingInterval
Typ: REG_DWORD
Wert: 5 bis 60 Sekunden (Standardabrufintervall ist 10 Sekunden)

[#LC1713]

- Auf VDA-Servern kommt es zu einem Fehler mit icardd.dll und ein Bluescreen wird angezeigt.

[#LC2277]

- Versuche, eine VDA-Sitzung mit einer PIN wieder zu verbinden, schlagen fehl und eine der folgenden Fehlermeldungen wird angezeigt:

- Insert a smart card
- Reading Card.

Wenn sich Benutzer mit Name und Kennwort anmelden, wird die Sitzung verbunden.

[#LC2596]

Systemausnahmen

- Im Betriebssystem kommt es zu einem Fehler mit picadm.sys und ein Bluescreen mit Bugcheckcode 0x22 wird angezeigt.

[#LC0227]

- Die Wiedergabe eines Videos in einem Medienplayer in einer Passthrough-Sitzung kann zum unerwarteten Beenden der Sitzung führen.

[#LC0553]

- Die veröffentlichte Version von Microsoft Excel 2010 schlägt fehl beim Wechseln zwischen geöffneten Tabellendateien.

[#LC0809]

- Im VDA kommt es zu einem Fehler mit ctxdvc.sys und ein Bluescreen mit Bugcheckcode E3 wird angezeigt.

[#LC0857]

- Der VDA reagiert möglicherweise nicht mehr, wenn eine VDA-Sitzung beim Verwenden von Grafikanwendungen maximiert wird.

[#LC0908]

- Windows Explorer kann in folgenden Fällen unerwartet geschlossen werden:
 - Wenn eine große Anzahl von Dateien ausgewählt wird, deren Namen mehr als 260 Zeichen enthält, und die Option "Senden an > Faxempfänger" ausgewählt wird
 - Wenn versucht wird, Anwendungen von Drittherstellern zu öffnen
 - Wenn versucht wird, Dateien mit Nitro PDF zu kombinieren

[#LC0938]

- Ein Seitenfehler im ICA WSK-Transporttreiber (tdwsk.sys) führt dazu, dass der VDA mit einem Fehler auf einem Bluescreen fehlschlägt.

[#LC1032]

- Nach dem Downgrade von VDAs von Version 7.6 auf Version 5.6.400 kommt es im Betriebssystem zu einem Fehler bei vd3dk.sys und ein Bluescreen mit Stoppcode 0x3B wird angezeigt.

[#LC1200]

- Im Betriebssystem kommt es zu einem Fehler mit picadm.sys und ein Bluescreen wird angezeigt. Das Problem tritt auf, wenn der Wert von "TrucationResource" Null wird.

[#LC1430]

- Das Abspielen einer Audiodatei in Receiver für HTML5, wenn SSL in der ICA-Sitzung aktiviert ist, kann zu einer schwerwiegenden Ausnahme beim VDA führen. Es wird ein Bluescreen angezeigt.

[#LC1752]

- Gelegentlich wird der VDA möglicherweise unerwartet heruntergefahren und dann neu gestartet.

{#LC1876}

- Im Betriebssystem kommt es zu einem Fehler mit ctxdvc.sys und ein Bluescreen mit Bugcheckcode 3B wird angezeigt.

[#LC2119]

- Im Betriebssystem kommt es zu einem Fehler mit tdica.sys und ein Bluescreen mit Bugcheckcode 44 wird angezeigt. [MULTIPLE_IRP_COMPLETE_REQUESTS].

[#LC2171]

- Es kommt mit Windows Server 2008 R2 und Windows Server 2012 R2 möglicherweise zu einer schwerwiegenden Ausnahme, wenn Benutzer die Verbindung mit Receiver für Windows herstellen.

[#LC2179]

- Nach dem Upgrade von Version 5.6 Rollup 4 auf Version 7.6 kann es passieren, dass VDAs beim Herunterfahren nicht mehr reagieren.

[#LC2245]

- Es kommt auf Servern möglicherweise zu einem Fehler bei icausbbsys und die Meldung PNP_DETECTED_FATAL_ERROR wird auf einem Bluescreen angezeigt.

[#LC2811]

- Der Prozess wfshell.exe wird möglicherweise unerwartet geschlossen und neue Verbindungen verhindert.

[#LC3024]

- Beim Wiederverbinden mit einer getrennten Sitzung wird die Gruppenrichtlinienengine (CitrixCseEngine.exe) u. U. unerwartet beendet.

[#LC3091]

- Der VDA reagiert möglicherweise nicht mehr, wenn eine VDA-Sitzung beim Verwenden von Grafikanwendungen maximiert wird.

[#LC3279]

- Es kommt auf VDAs möglicherweise zu einer schwerwiegenden Ausnahme bei vdtw30.dll, ein Bluescreen wird angezeigt.

[#LC3354]

- Es kommt auf VDAs möglicherweise zu einer schwerwiegenden Ausnahme bei vd3dk.sys, ein Bluescreen mit Stoppcode 0x50 wird angezeigt. Dies kann passieren, wenn Java-basierte Upgradetools für das Upgrade des VDAs beim Starten konfiguriert sind.

[#LC3372]

- Der Prozess explorer.exe kann beim Wiederherstellen der Verbindung mit einer Sitzung unerwartet beendet werden.

[#LC3411]

Benutzererfahrung

- Dieser Fix die folgenden Probleme mit der Darstellung von Anwendungen in der Receiver für Windows-Taskleiste:
 - Einzelne Anwendungen können mit dem Receiver-Standardsymbol statt dem anwendungsspezifischen Symbol angezeigt werden.
 - Werden mehrere Anwendungen in der gleichen Sitzung ausgeführt, kann es passieren, dass sie unter einem einzelnen Receiver-Standardsymbol erscheinen.

[#LC0094]

- Dieser Fix behandelt ein Benutzerfreundlichkeitsproblem der Desktopgestaltungsumleitung mit Receiver für Windows vor Version 4.200 bei LAN-Verbindungen.

[#LC1507]

Benutzeroberfläche

- Gelegentlich bleiben Fensterpositionen nicht erhalten, wenn Benutzer sich mit einem VDA wieder verbinden und mehrere Monitore verwenden.

[#LC2075]

- Die automatische Tastaturanzeige funktioniert möglicherweise nicht, wenn Benutzer eine veröffentlichte Anwendung mit Receiver für Windows oder den mobilen Receiver öffnen.

[#LC2747]

Sonstiges

- Dieser Fix führt die Option /Verbose für vdaredirector.exe ein. Es wird damit ein Meldungsfenster beim Ausführen von vdaredirector.exe angezeigt. Das Meldungsfenster wird nur angezeigt, wenn die Option /Verbose angegeben wurde.

[#LC0338]

- Nach dem Installieren einer Antivirensoftware von Drittanbietern auf einem Microsoft Windows 8 (32 Bit) VDA können Citrix Receiver-Sitzungen nicht gestartet werden.

[#LC1636]

- Benutzer können den Lync-Status in der Konsolensitzung nicht ändern, wenn einer der Hotfixes den Fix LA5736 enthält oder einer der folgenden Hotfixes auf dem VDA installiert wurden:

- ICAWS750WX64007
- ICAWS750WX64008
- ICAWS750WX64012
- XD710ICAWSWX64003

Der Lync-Status wird immer als abwesend angezeigt.

Wichtig: Ist NetScaler Gateway Teil der Bereitstellung, wird nachdem Sie diesen Hotfix angewendet haben, der Lync-Status immer noch als abwesend angezeigt, wenn Benutzer sich von der VDA-Sitzung trennen und sich dann mit einer Konsolensitzung verbinden.

[#LC1691]

- Benutzer können den Lync-Status in der Konsolensitzung nicht ändern, wenn einer der Hotfixes den Fix LA5736 enthält oder einer der folgenden Hotfixes auf dem VDA installiert wurden:

- ICAWS750WX64007
- ICAWS750WX64008
- ICAWS750WX64012
- ICAWS750WX64028
- XD710ICAWSWX64003

Der Lync-Status wird immer als abwesend angezeigt.

[#LC2479]

- In diesem Release wird Framehawk Virtual Channel, ein neuer virtueller ICA-Kanal unterstützt, der die HDX-Technologien von Citrix erweitert und die Benutzererfahrung für drahtlose Breitbandverbindungen verbessert, bei denen Packetverlust und Latenz ein häufiges Problem sind. Weitere Informationen finden Sie unter [Neue Features](#).

[#LC2782]

- Dies ist eine Funktionserweiterung zum Unterstützen eines bestimmten 5-in-1-COM-Portgeräts. Sie stellt sicher, dass die ID-Kartenleserfunktion des Geräts in VDA-Sitzungen korrekt funktioniert.

[#LC0325]

- Versuche, VM-gehostete Apps zu starten, schlagen möglicherweise fehl, wenn Benutzer indirekt zu einer geschachtelten Gruppe gehören, für eingeschränkte Sichtbarkeit eingestellt ist.

[#LC3589]

VDA für Serverbetriebssysteme 7.6.300, 7.6 LTSR, 7.7

Die Versionen der VDAs für Serverbetriebssysteme, die in XenApp und XenDesktop 7.6 Feature Pack 3 (7.6.300) und 7.6 LTSR enthalten sind, sind identisch. Die in XenApp und XenDesktop 7.7 enthaltene Version enthält die gleichen Fixes wie Version 7.6.300 sowie Kompatibilitätsupdates für XenApp und XenDesktop 7.7.

Broker Agent

HDX 3d Pro

Seamlessfenster

HDX MediaStream Flash-Umleitung

Sitzung/Verbindung

HDX MediaStream Windows Media-Umleitung

Site-/Farmverwaltung

Installieren, Deinstallieren und Aktualisieren

Smart cards

Tastatur/Maus

Systemausnahmen

Lokaler App-Zugriff

Benutzererfahrung

Anmeldung und Authentifizierung

Benutzeroberfläche

Drucken

Sonstiges

Broker Agent

- VDAs können während des Registrierungsvorgangs bei der Initialisierung hängen bleiben. Das Problem tritt auf, wenn der Citrix Desktopdienst mehrere Tage ohne Neustart ausgeführt wird.

[#LC0570]

- Ist die Funktion "CName" aktiviert, kann die VDA-Registrierung übermäßig lange dauern. Durch diesen Fix wird beim Aktivieren des Werts "CNameSuppressOrgFqdnLookup" der Registrierungsvorgang schneller.
Wichtig: Bei der Aktivierung von "CNameSuppressOrgFqdnLookup" müssen Sie auch "CName" aktivieren.

Zum Implementieren dieses Fixes erstellen Sie folgenden Registrierungsschlüssel:

- HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\VirtualDesktopAgent
Name: UseCNameLookup
Typ: REG_DWORD

Wert: Aktiviert = 1 (Deaktiviert = 0)

- HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\VirtualDesktopAgent

Name: CNameSuppressOrgFqdnLookup

Typ: REG_DWORD

Wert: Aktiviert = 1 (Deaktiviert = 0)

[#LC0791]

- Versuche, eine veröffentlichte Anwendung in einem mehrfach vernetzten Server zu starten, schlagen möglicherweise mit der folgenden Meldung fehl:

"An dieser Adresse ist kein Citrix XenApp-Server konfiguriert."

[#LC1686]

- Gehört ein Version 7.6 VDA zu einer XenDesktop 5.6-Site, schlagen Wiederverbindungsversuche fehl und der VDA ist nicht registriert, wenn Benutzer sich verbinden, die Verbindung trennen und dann wiederherstellen möchten.

[#LC1859]

HDX 3D Pro

- Mit diesem Fix ist die Bandbreitenobergrenze für HDX-Netzwerkdatenverkehr 20 MBit/s.

[#LC2780]

HDX MediaStream Flash-Umleitung

- Mit Browserversionen Microsoft Internet Explorer 9 und höher werden nach dem Klicken auf einen Link zu einem umgeleiteten Flash-Video während bereits ein anderes umgeleitetes Flash-Video abgespielt wird, die Audiodaten der beiden Videos übereinander abgespielt.

Dieser Fix verhindert das Problem nicht, aber wenn Sie den folgenden Registrierungsschlüssel auf dem Server setzen, wird die Zeit verkürzt, für die das Problem auftritt:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\HdxMediaStreamForFlash\Server\PseudoServer

Name: UrlListForTerminationFlashInst

Typ: REG_MULTI_SZ

Daten:

[#LC0453]

- Nachdem Sie das integrierte Suchfeld von YouTube in Microsoft Internet Explorer 8 und höher verwendet haben, wird die HDX Flash-Umleitung auf die serverseitige Wiedergabe zurückgesetzt.

Um diesen Fix zu nutzen, müssen Sie den folgenden Registrierungsschlüssel auf dem Server setzen:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\HdxMediaStreamForFlash\Server\PseudoServer

Name: SupportedUrlHeads

Typ: REG_MULTI_SZ

Wert:

http://

https://

file://

Hinweis: SupportedUrlHeads wurde zuerst mit Fix LA4151 eingeführt. Er wird auch von diesem Fix verwendet. Wenn Sie der Registrierung SupportedUrlHeads hinzufügen, ignoriert die Flash-Umleitung Instanzierungsanfragen, es sei denn, die Ziel-URL beginnt mit einem der Header, die in SupportedUrlHeads angegeben wurden.

[#LC0505]

- Mit Microsoft Internet Explorer 10 kann die Flash-Umleitung für Videos fehlschlagen, die in cnn.com-Websites eingebettet sind. Dies passiert, wenn das erste Video auf einer Website automatisch abgespielt wird. Manche Inhalte der Seite für dieses Video, wie Titel, Status und Werbung fehlen möglicherweise.

[#LC0830]

- Internet Explorer wird möglicherweise unerwartet geschlossen bei dem Versuch, ein Video von Websites abzuspielen, für die gilt:
 - Die Option "Flash-Beschleunigung deaktivieren" ist aktiviert oder die Site steht auf der Sperrliste.
 - Der Benutzer verwendet eine Receiver-Version, die die Flash-Umleitung nicht unterstützt, z. B. Receiver für Mac OS X, iPad, iPhone und Android.

[#LC2256]

HDX MediaStream Windows Media-Umleitung

- Nach der Installation von Version 4.1 oder 4.1.2 von Receiver für Windows, wenn die HDX MediaStream-Multimediabeschleunigung aktiviert ist, lädt Windows Media Player das Video, aber es wird nicht mehr als der erste Frame abgespielt.

[#LC1460]

- Dieses Release behebt das Problem, dass überflüssige Frames vom HDX-Grafikencoder generiert werden. So wird eine gleichmäßigere Framerate und bessere Interaktivität erreicht.

[#LC2177]

- Wenn eine Videodatei in einer VDA-Sitzung mit Receiver für Windows Version 13.3 oder 13.4 abgespielt wird, geschieht die Synchronisierung von Audio und Video in einer benutzerdefinierten Anwendung. Nach dem Upgrade auf eine neuere Version von Receiver schlägt die Windows Media-Umleitung fehl. Nachdem dies passiert können Benutzer nur Audio hören oder nur Video sehen.

Um diesen Fix zu aktivieren, müssen Sie Citrix Receiver für Windows 4.3 verwenden.

[#LC2516]

- Versuche, Windows Media Player zu schließen, während ein Video angehalten ist, kann dazu führen, dass Windows Media Player nicht mehr reagiert.

[#LC3410]

Installieren, Deinstallieren und Aktualisieren

- Wenn Sie eines der Hotfix Rollup Packs 1 bis 4 nach der Installation des universellen Druckserverclients installieren, kann die Enumeration der Sitzungsdrucker sehr lange dauern.

[#LA5977]

- Installieren von Hotfixes für XenApp 7.5 und XenDesktop 7.1 und 7.5 VDA Core Services für Windows Desktop- und Serverbetriebssysteme, die vor September 2014 veröffentlicht wurden, führt dazu, dass der Leistungsindikatorzähler für ICA-Sitzungen entfernt wird. Dies wirkt sich negativ auf Tools und Prozesse aus, die auf diesem Zähler basieren. Dieser Fix stellt durch die Installation früherer Hotfix entfernte Zähler wieder her und verhindert, dass sie durch die Installation nachfolgender Hotfixes entfernt werden.

[#LC0771]

- Nach der Installation von Hotfix XA650R04W2K8R2X64008 werden bestehende Symbole, die mit früheren Versionen von XenApp korrekt funktionierten, möglicherweise als beschädigt gemeldet.

[#LC1081]

- Nach der Installation eines Hotfixes oder Hotfix Rollup Packs mit Fix LA5872 wird der Windows-Anwendungssicherheitshinweis beim Anmelden mit Receiver nicht ausgeblendet, bis die veröffentlichten Anwendungen des Benutzers gestartet werden.

[#LC1215]

- Nach dem Installieren von XA650W2K8R2X64R04 schlägt gelegentlich die Discovery fehl und mehrere Versuche sind erforderlich, bis die Discovery erfolgreich durchgeführt wird.

[#LC1369]

- Nach dem Installieren von Hotfixes XA650W2K8R2X64R03, XA650W2K8R2X64R04 oder XA650W2K8R2X64R05 können folgende Probleme auftreten:

- Ereignisanzeigefehler, wenn die folgenden Registrierungseinstellungen deutsche Binärdateien verwenden:
 - HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\WINEVT\Publishers\{c6d0a271-2e59-40cd-b451-9000a9f104f9}
 - MessageFileName: C:\Programme (x86)\Citrix\System32\resource\de\RaveEventMessages.Dll
 - ResourceFileName: C:\Programme (x86)\Citrix\System32\resource\de\RaveEventMessages.Dll
- Wenn die Sprache des Betriebssystems Englisch ist und das Standardbenutzerkonto auf Deutsch eingestellt ist, verwendet die RAVE-Ereignisanzeige Deutsch statt Englisch.

[#LC2130]

- Nach der Installation von Hotfix Rollup Pack 5 schlägt der XTE-Dienst fehl und Benutzersitzungen werden getrennt.

{#LC2180

- Nach dem VDA-Upgrade von XenDesktop 7.5 auf XenDesktop 7.6 wird Fehler 1030 angezeigt, wenn Benutzer eine Verbindung herstellen möchten. In dem Systemereignisprotokoll erscheint folgende Meldung: "The Citrix ICA Transport Driver received an invalid Transport packet on port 2598."

[#LC2501]

- Nach dem Installieren von XA650W2K8R2X64R05 auf XML-Brokern kann es länger dauern bis der XML-Dienst Benutzer authentifiziert. In Umgebungen mit einer hohen Last kann diese Verzögerung lange genug sein, dass es zu einem Timeout der Benutzersitzung im Webinterface kommt.

[#LC2536]

Tastatur/Maus

- Bestimmte Tasten funktionieren nicht korrekt in einer Receiver für Linux-Sitzung mit einem tschechischen Tastaturlayout. Falsche Zuordnungen können für die folgenden Zeichen auftreten:

- d' wird ě zugeordnet
- ň wird ě zugeordnet
- ě wird ě zugeordnet

Hinweis: Nachdem Sie diesen Fix installieren, kann es sporadisch zu Problemen mit der Umschalttaste kommen, wenn Sie eine RDP-Sitzung innerhalb einer Receiver für Linux-Sitzung starten. Um dieses Problem zu vermeiden, empfiehlt Citrix, dass Sie den Fix aus Microsoft-Artikel [KB2592687](#) auf dem Windows-Server installieren. Alternativ finden Sie ein Workaround im Knowledge Center-Artikel [CTX110281](#).

[#LA3968]

- Wenn Sie mit der Maus zwischen lokalen und veröffentlichten Anwendungen wechseln, funktioniert die Tastatureingabe möglicherweise nicht.

[#LA5849]

- Der Mauszeiger bleibt möglicherweise nicht innerhalb der Desktopgrenzen.

[#LC0383]

- Mauseingaben gehen gelegentlich in der Sitzung verloren.

[#LC0632]

- Ist die automatische Tastaturanzeige aktiviert, wird die Tastatur auf dem Benutzergerät nicht automatisch in der veröffentlichten Anwendung angezeigt.

Hinweis: Um dieses Problem vollständig zu lösen, installieren Sie Fix LC3158 auf dem Benutzergerät.

[#LC2432]

- Die automatische Tastaturanzeige funktioniert möglicherweise nicht, wenn OpenOffice-Anwendungen auf einem iPad-Gerät verwendet werden.

[#LC2717]

Lokaler App-Zugriff

- In einer Desktopsitzung im Vollbildmodus, in der lokaler App-Zugriff aktiviert und die Einstellung "Fensterinhalt beim Verschieben anzeigen" deaktiviert ist, werden Bilder in hoher Auflösung verzerrt, wenn Anwendungssymbole über Bilder in Internet Explorer gezogen werden.

[#LC1722]

Anmeldung und Authentifizierung

- Die Fehlermeldung "Keine gültigen Zertifikate gefunden" wird angezeigt, wenn Benutzer versuchen, eine Anwendung über StoreFront mit Smartcard-Authentifizierung zu starten. Dadurch wird verhindert, dass Benutzer sich anmelden können.

[#LC1704]

Drucken

- Der Prozess für den Citrix Druckmanagerdienst (CpSvc.exe) wird möglicherweise unerwartet beendet.

[#LA4967, #LA5682, #LC0344, #LC0427, #LC0675, #LC0781, #LC1090, #LC1770]

- Beim Roaming zu einem anderen Benutzergerät wird möglicherweise ein falscher Drucker als Standarddrucker für die Sitzung eingestellt.

[#LC0225]

- Dieser Fix behebt ein Problem, dass das Zuordnen der folgenden Drucker in Benutzersitzungen mit Receiver für Linux zugeordnet verhindert:

- Epson-LQ-1050
- Epson PLQ-20

Zum Implementieren dieses Fixes legen Sie folgenden Registrierungsschlüssel fest:

HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\ICA\PrintingSettings

Name: AllowPrinterMappingByName

Typ: REG_DWORD

Wert: 1

[#LC0389]

- Sitzungsdrucker werden möglicherweise bei Wiederverbinden mit einer getrennten, fortbestehenden Sitzung nicht zugeordnet, wenn die veröffentlichte Anwendung eine festgelegte Fenstergröße hat.

[#LC0615]

- Der Citrix Druckmanagerdienst (CpSvc.exe) schlägt aufgrund einer Zugriffsverletzung fehl.

[#LC0660]

- Ist Sitzungsfortbestehen aktiviert, werden umgeleitete Drucker auf Benutzergeräten möglicherweise beim Wiederverbinden mit einer Sitzung nicht zugeordnet.

[#LC0816]

- Dieser Fix macht die Richtlinie "Legacydruckernamen" wieder verfügbar, die vorher nicht mehr in Version 7.x des Produkt enthalten war.

[#LC1002]

- Wenn sich Benutzer an einer neuen Sitzung anmelden, wird die Druckerliste nicht angezeigt. Die Druckerliste wird nicht angezeigt, wenn Benutzer sich mit einer getrennten Sitzung wieder verbinden.

[#LC1226]

- Wenn die Richtlinieneinstellung "Generischen universellen Drucker automatisch erstellen" aktiviert ist, erstellt der Citrix Druckmanagerdienst (CpSvc.exe) den generischen Citrix Universellen Drucker nicht.

[#LC1255]

- Diese Funktionserweiterung führt Leistungsindikatoren für den universellen Druckerserver und den universellen Druckclient ein.

[#LC1820]

- Versuche, den Druckmanagerdienst über das Snap-In "Dienste" neu zu starten, schlagen fehl. Es wird eine Fehlermeldung angezeigt, dass der Dienst nicht angehalten werden kann, und der Status des Druckmanagerdiensts ist "Wird beendet".

[#LC2122]

- Die Richtlinien "Universelles Drucken - EMF-Verarbeitungsmodus" bleibt auf den Standardwert eingestellt, selbst wenn Sie sie auf "EMF-Datensätze für Drucker neu verarbeiten" eingestellt haben.

[#LC2521]

- Der Citrix Druckmanagerdienst (CpSvc.exe) schlägt unerwartet aufgrund einer Zugriffsverletzung fehl.

[#LC2624]

- Benutzer bemerken möglicherweise sporadisch Verzögerungen beim Drucken von Dokumenten oder Wählen eines anderen Druckers. Für bis zu eine Minute lang erscheint eine Meldung, dass eine Verbindung mit dem Drucker hergestellt wird.

[#LC2701]

- Die erste Seite wird möglicherweise leer gedruckt, wenn der universelle Druckertreiber von Citrix verwendet wird.

[#LC2771]

- Benutzer können nicht mit einem Diebold-Drucker (Model P361) drucken, wenn sie mit Receiver angemeldet sind.

[#LC2908]

- Diese Funktionserweiterung erstellt einen automatisch wiederhergestellten Drucker pro Portnummer, wenn Benutzer sich anmelden.

[#LC2943]

Seamlessfenster

- Bei Instanzen von Microsoft Excel, die im Seamlessmodus veröffentlicht wurden, kommt es möglicherweise zu einem Bildschirmflimmern. Der Tastaturfokus kann außerdem an lokal ausgeführte Anwendungen verloren gehen.

Nach der Installation dieses Fixes müssen Sie den folgenden Registrierungsschlüssel erstellen und dann die folgenden beiden Werte setzen:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\wfshell\TWI\XLMAIN

Name: ClassName

Typ: REG_SZ

Daten: XLMAIN

Name: Type

Typ: REG_DWORD

Daten: 0x00000808

[#LC0401]

- Die Größe maximierter Seamlessanwendungen wird nicht richtig angepasst, wenn eine angedockte Anwendung verschoben

wird, z. B. die Windows-Taskleiste.

[#LC1342]

- Anwendungen mit erhöhten Rechten, die in Citrix Studio veröffentlicht sind, können nicht im Seamlessmodus gestartet werden.

[#LC1537]

- Versuchen Sie in einer Seamlessitzung, eine Seamlessymbolleiste im zweiten Monitor anzudocken, wird sie immer im ersten Monitor andockt, wenn das Seamlessflag für veröffentlichte Symbolleisten auf ON gesetzt ist. Weitere Informationen über das Seamlessflag finden Sie im Knowledge Center-Artikel [CTX101644](#).

[#LC1599]

- Das Seamlessfenster wird nicht angezeigt, obwohl der Anwendungsprozess ausgeführt wird und die Sitzung auf dem Server-VDA vorhanden ist.

[#LC1728]

- In einer Multimonitumgebung wird das Seamlessfenster möglicherweise auf den primären Monitor verschoben, wenn der Benutzer das Seamlessfenster auf einen sekundären Monitor verschiebt und dann die Auflösung des primären oder sekundären Monitors ändert.

[#LC1819]

Sitzung/Verbindung

- Wenn Benutzer versuchen, eine Datei auf ein Benutzergerät über die Clientlaufwerkzuordnung zu speichern, enthält die auf dem zugeordneten Laufwerk gespeicherte Datei möglicherweise keine Daten.

[#LA3566]

- Nach dem Kopieren einer Datei auf ein zugeordnetes Clientlaufwerk kann der verbleibende Speicherplatz auf dem zugeordneten Laufwerk falsch berechnet und angezeigt werden.

[#LA4799]

- Bei Versuchen, die Verbindung zu einer Sitzung wiederherzustellen, kann es zu einer Verzögerung von bis zu 60 Sekunden beim Anmelden kommen, wenn mehrere WMI-Filter durch GPOs konfiguriert sind.

Um die Richtlinienberechnung während der Wiederverbindung zu deaktivieren, müssen Sie folgenden Registrierungsschlüssel einstellen:

HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\Reconnect

Name: DisableGPCalculation

Typ: REG_DWORD

Wert: 1

[#LA4932]

- ConnectTime wird möglicherweise während der Wiederverbindung nicht neu berechnet. Das Resultat ist, dass der SCSD-Wert (Serverdauer für Sitzungserstellung) und der SSSD-Wert (Sitzungsstartdauer) für bestimmte Sitzungen in EdgeSight-Berichten unerwartet hoch sein können.

Um den Fix zu aktivieren, setzen Sie den folgenden Registrierungsschlüssel:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\wfshell

Name: DisableServerStartupEventForReconnect

Typ: REG_DWORD

Wert: 1

[#LA5141]

- Versuche, eine Audiodatei in Microsoft Outlook abzuspielen, schlagen möglicherweise fehl, wenn Benutzer mit Receiver für Mac verbunden sind.

[#LA5657]

- Ist Excelhook aktiviert, können Versuche fehlschlagen, zwischen Arbeitsmappen zu wechseln und Excel scheint nicht zu reagieren.

[#LA5708]

- Nach dem Neustart des XenApp-Servers schlagen Verbindungsversuche mit dem Lizenzserver möglicherweise fehl.

Zum Implementieren dieses Fixes erstellen Sie folgenden Registrierungsschlüssel:

- Unter 32-Bit-Windows:

HKEY_LOCAL_MACHINE\Software\Citrix\Licensing

Name: RA_retries

Typ: REG_DWORD

Wert: 1 (ist der Wert nicht vorhanden oder geringer als 5, wird der Standardwert als 5 angenommen)

- Unter 64-Bit-Windows:

HKEY_LOCAL_MACHINE\Software\Wow6432Node\Citrix\Licensing

Name: RA_retries

Typ: REG_DWORD

Wert: 1 (ist der Wert nicht vorhanden oder geringer als 5, wird der Standardwert als 5 angenommen)

[#LA5773]

- Versuche, die Verbindung mit einer gesperrten veröffentlichten Desktopsitzung wiederherzustellen, können mit der folgenden Fehlermeldung fehlschlagen:

"Verbindungsfehler: Auf dem Citrix Server wurde das maximale Limit für gleichzeitige Anwendungen erreicht. Wenden Sie sich an den Systemadministrator."

Wenn dies passiert, wird zusätzlich zu der bestehenden Sitzung eine neue Sitzung gestartet.

[#LC0158]

- Versuche, die Verbindung zu einer getrennten Sitzung wiederherzustellen, können fehlschlagen und es wird stattdessen eine neue Sitzung gestartet. Das Problem tritt auf, wenn die Einstellung "Remotedesktopdienste-Benutzer auf eine Remotedesktopdienste-Sitzung beschränken" aktiviert ist.

[#LC0201]

- Wird eine Remotedesktopverbindung (RDP) zu einem VDA hergestellt und dann die RDP-Sitzung getrennt, erscheint die folgende Fehlermeldung (Ereignis-ID 1048) im Ereignisprotokoll:

Der Citrix Desktopdienst wird wieder beim Desktop Delivery Controller registriert:

NotificationManager:NotificationServiceThread: WCF-Fehler oder Ablehnung durch Broker (DDC: bruin.newer.In)""

[#LC0251]

- Ist das Anwendungsinstanzlimit auf "1" gesetzt, können Versuche, eine Anwendung von einer getrennten Sitzung neu zu starten, fehlschlagen und es wird eine neue Sitzung gestartet. Dies wird durch eine versuchte automatische Wiederverbindung des Clients verursacht, die den falschen Wert für den internen Sitzungsstatus für Wiederverbindungen setzt.

[#LC0259]

- Wenn eine veröffentlichte Anwendung geöffnet und dann der Befehl explorer.exe ausgeführt wird, um Windows Explorer zu öffnen, kann es passieren, dass der Desktop statt Windows Explorer geöffnet wird.

Hinweis: Nachdem Sie diesen Fix installieren, wird der Prozess explorer.exe möglicherweise unerwartet beendet. Weitere Informationen finden Sie im Knowledge Center-Artikel [CTX128009](#).

[#LC0285]

- Dateien können beschädigt werden, wenn Dateien zwischen den UNC-Pfaden von zwei zugeordneten Laufwerken kopiert werden, wenn das zweite Laufwerk mit dem Windows-Befehl SUBST vom ersten UNC-Laufwerk erstellt wurde.

[#LC0288]

- Ist die Multistreaming per Richtlinie aktiviert, dauert die Datenübertragung der Zwischenablage vom Server zum Benutzergerät möglicherweise sehr lange.

Sie können den folgenden Registrierungsschlüssel erstellen, um die Standardausgabebandbreite des Servers auf einen höheren Wert einzustellen.

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server\Wds\icawd

Name: InitialOutputSpeed

Typ: REG_DWORD

Wert: 10000 – 1000000 (10KB/s - 1000KB/s; empfohlener Wert: 300000)

[#LC0390]

- Nach dem Konfigurieren von Excelhook, wie in Knowledge Center-Artikel [CTX133198](#) beschrieben, können in Sitzungen, in denen sowohl Microsoft Excel als auch Microsoft Visual Basic for Applications ausgeführt wird, die folgenden Probleme auftreten:
 - Das Hauptfenster von Excel bleibt im Vordergrund, selbst wenn das Fenster von Visual Basic for Applications den Fokus hat.
 - Schließen des Fensters von Visual Basic for Applications schließt auf das Hauptfenster von Excel.

[#LC0398]

- Benutzersitzungen bleiben beim Abmelden möglicherweise in einem getrennten Zustand hängen. Die Sitzungen können dann nicht geschlossen werden und sie können den Server nicht neu starten.

[#LC0399]

- Versuche, Mozilla Firefox 28 und spätere Versionen in XenApp zu veröffentlichen, können mit der folgenden Meldung fehlschlagen:

"System.NotSupportedException: Data is not in ICO format"

[#LC0402]

- Citrix XenApp 7.6 und XenDesktop 7.6 VDA-Kerndienste, die unter Windows Server 2008 R2 (Server OS) ausgeführt werden, reagieren beim Willkommenbildschirm möglicherweise nicht mehr. Wenn dies passiert, schlagen neue Receiver und RDP-Verbindungen zum Server fehl.

[#LC0405]

- Gelegentlich reagieren Benutzersitzungen nicht mehr und können nicht abgemeldet werden. Dies passiert auch, wenn Benutzer sich manuell von der Konsole abmelden.

[#SLC0472]

- In veröffentlichten Desktops wird die Statusanzeige nicht vervollständigt und bestimmte Anmeldestatusmeldungen werden nicht angezeigt, wenn Benutzer sich anmelden oder Anwendungen starten.

[#LC0482]

- Die automatische Tastaturanzeige auf Mobilgeräten (wie iOS und Android) wird unerwartet beim Bildübergang in der Microsoft Silverlight-Webanwendung angezeigt.

[#LC0499]

- Wenn Benutzer eine veröffentlichte Version von Visual Basic Editor über die Taskleiste schließen, werden geöffnete Excel-Tabellen auch geschlossen.

[#LC0534]

- Beim Starten einer veröffentlichten Anwendung können Benutzer langsame Anmeldungen bemerken.

[#LC0596]

- Ist die Richtlinie "Mobilerfahrung" in XenApp 6.5 aktiviert und Benutzer stellen die Verbindung mit Receiver für iOS her, wird die folgende Fehlermeldung beim Abmelden der Benutzer angezeigt:

"There was a problem logging off. Please contact the help desk for assistance."

[#LC0628]

- Verbindungsversuche mit XenDesktop 7.x auf Windows Server 2008- oder Windows Server 2012-VDAs schlagen möglicherweise fehl, wenn das Benutzerkennwort mehr als 45 Zeichen enthält.

[#LC0689]

- Ist die Option "Zeitzonenumleitung zulassen" aktiviert, funktioniert die Zeitzonensynchronisierung möglicherweise nicht für veröffentlichte Java-Anwendungen in Sitzungen, die auf nicht-englischen Windows 7-Benutzergeräten ausgeführt werden. Das Problem tritt auf, weil die GetDynamicTimeZoneInformation-API die Zeitzoneneinformationen nicht an die Receiver-Sitzung zurückgeben kann.

Stellen Sie auf Benutzergeräten das Gebietsschema des Systems auf die gleiche Sprache ein wie das Language Pack.

Hinweis: Dieser Fix behebt nicht das Problem mit russischen Zeitzonen.

[#LC0807]

- Ist die Clientlaufwerkzuordnung aktiviert, werden gelegentlich unerwartete Zeichen an das Ende von Dateien angehängt, die von einer veröffentlichten VB-Anwendung mit der Funktion FileStream() geschrieben werden.

[#LC0815]

- Versuche, das Kombinationsfeld-Steuerelement unter "Region und Sprache" > "Datums- und Uhrzeitformate" remote zu bedienen, schlagen möglicherweise fehl. Dies kann passieren, wenn das Kombinationsfeld-Steuerelement bereits einmal erfolgreich bedient wurde und dann ein zweiter Versuch gemacht wird.

[#LC0836]

- Wenn Benutzer versuchen, sich mit einer getrennten Sitzung zu verbinden, kann es bis zu drei Minuten dauern.

[#LC0850]

- Beim Starten einer veröffentlichten Anwendung kann die folgende Fehlermeldung auf dem Benutzergerät angezeigt werden:

"Der Citrix Server kann Ihre Anfrage zum Starten dieser veröffentlichten Anwendung momentan nicht bearbeiten. Bitte versuchen Sie es später noch einmal. Wenn das Problem weiterhin besteht, wenden Sie sich an den Administrator."

Hintergrund: Es kann eine Weile dauern, bis der Prozess wfshell das Überprüfen der Anwendungseigenschaften beim Start der Anwendung abgeschlossen hat. Abhängig von den Netzwerkbedingungen kann es zu einem Timeout für die Anforderung kommen, bevor die Überprüfung abgeschlossen ist, und die Fehlermeldung wird angezeigt.

Dieser Fix führt die Unterstützung für den folgenden Registrierungsschlüssel ein, mit dem Sie das Timeout beim Überprüfen der Anwendungseigenschaften um bis zu vier Minuten verzögern können:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\ICAST

Name: WaitTimeOutValue

Typ: REG_DWORD

Wert:

[#LC0860]

- Wenn sich Benutzer mit einer der Receiver-Versionen für mobile Geräte anmelden und dann eine veröffentlichte Version von Lotus Notes öffnen, für die die automatische Tastaturanzeige aktiviert ist, wird die Tastatur nicht angezeigt, um E-Mail-Adressen in die Adressfelder einzugeben.

[#LC0899]

- Ein Fenster "Verbindung aufgebaut. Funktionalität wird ausgehandelt" wird in Receiver angezeigt, wenn der VDA nicht mehr reagiert. Um andere Sitzungen erfolgreich zu starten, muss der VDA neu gestartet werden.

[#LC0909]

- Die Zeitzone ist nicht richtig, wenn sich Benutzer mit Receiver für Windows anmelden. Damit dieser Hotfix korrekt funktioniert, müssen Sie Folgendes installieren:
 - Den gleichen Microsoft-Zeitzoneupdatehotfix auf dem Benutzergerät und dem Server. Beispielsweise wenn Microsoft Hotfix [KB2998527](#) auf dem Benutzergerät installiert ist, installieren Sie dieses Hotfix auch auf dem Server.
 - Microsoft Hotfix [KB2870165](#) auf dem Server, wenn das Serverbetriebssystem Windows Server 2008 R2 Service Pack 1 ist.

- Fix LC1392 ist auf dem Benutzergerät installiert.

[#LC1061]

- Versuche, sich mit einer aktiven Sitzung auf dem Server wieder zu verbinden, schlagen möglicherweise fehl. Ist das Anwendungsinstanzlimit auf 1 gesetzt, können Versuche fehlschlagen, eine andere Sitzung zu öffnen. Benutzer müssen sich entweder abmelden oder die bestehende Sitzung trennen, um eine neue Sitzung starten zu können.

[#LC1084]

- Versuche, Anwendungen zu enumerieren, können fehlschlagen, wenn die Vertrauensstellung zwischen Domänen sich ändert, während Server in der Farm heruntergefahren sind. Anmeldeversuche mit einem UPN gelingen, aber Versuche, einen untergeordneten Anmeldenamen zu verwenden, schlagen fehl.

[#LC1136]

- Wenn die USB-Umleitungsrichtlinie mehr als 1000 Zeichen hat, werden alle USB-Laufwerke umgeleitet, selbst wenn es für das Gerät eine Verweigerungsregel gibt.

[#LC1153]

- Ist ExcelHook aktiviert und sind mehrere Projekte in Microsoft Project 2013 geöffnet, erscheint das zuerst geöffnete Projekt möglicherweise nicht in der Taskleiste.

[#LC1158]

- Gelegentlich werden visuelle Einstellungen von Seamlessanwendungen, wie Beschriftungsschriftart, Scrollleistengröße, Menühöhe, möglicherweise nicht korrekt mit den Benutzergeräteinstellungen synchronisiert.

[#LC1169]

- Die Sitzungsfreigabe schlägt sporadisch fehl.

[#LC1184]

- Wenn ein anderer Prozess die gleiche Sperre wie picadm.sys hat, können sich Benutzer nicht von der Sitzung abmelden und die Sitzung bleibt im getrennten Zustand.

[#LC1243]

- Die endgültige Bildqualität ist möglicherweise schlechter, wenn die Einstellungen "Zu verlustfrei verbessern" und "Visuell verlustfreie Komprimierung zulassen" auf dem Desktop Delivery Controller (DDC) aktiviert sind.

[#LC1271]

- Wenn sich Benutzer das erste Mal anmelden, wird der Bildschirm in einer Receiver-Sitzung möglicherweise beschädigt angezeigt. Das Problem tritt auf, wenn Sie die Bildqualität auf "Zu verlustfrei verbessern" ändern und die Einstellung "Visuell verlustfreie Komprimierung zulassen" aktivieren.

[#LC1272]

- Versuche, Informationen zu kopieren, schlagen in einer veröffentlichten Version von Microsoft Access 2007 fehl. Die Informationen erscheinen nicht in der Office-Zwischenablage.

[#LC1290]

- Die veröffentlichte Version von Microsoft Excel 2010 schlägt fehl beim Wechseln zwischen geöffneten Tabellendateien.

[#LC1356]

- Die Umleitung funktioniert möglicherweise nicht korrekt für das MailTo-Feature in XenDesktop- und XenApp-Sitzungen. Klickt der Benutzer auf den MailTo-Link im Webbrowser, wird nicht die veröffentlichte Instanz von Outlook geöffnet sondern ein Fehler angezeigt.

[#LC1400]

- Ist Sitzungsfortbestehen aktiviert und eine Anwendung wird auf einem iOS- oder Android-Gerät geschlossen, erscheint ein schwarzes Fenster auf dem Gerätebildschirm und bleibt offen.

[#LC14074]

- Wenn ein Benutzer sich mit der automatische Wiederverbindung von Clients mit einer Sitzung wieder verbindet, wird die clientseitige Zeitzone möglicherweise nicht zum Benutzergerät umgeleitet und die serverseitige Zeitzone wird stattdessen auf dem Benutzergerät angezeigt.

[#LC1426]

- Wenn Benutzer sich mit einer Desktopsitzung wieder verbinden, werden die Werte für "Serverdauer für Sitzungserstellung" (SCSD) und "Serverstartdauer Sitzung" (SSSD) nicht zurückgesetzt.

[#LC1464]

- Auf einem deutschen Windows-basierten Gerät kann der deutschsprachige Dienst "Multimedia Rave" (RaveEventMessages.dll) dazu führen, dass der Prozess svchost.exe fehlschlägt, der den Ereignisprotokolldienst enthält. Wenn dies der Fall ist, können sich Benutzer nicht anmelden.

[#LC1543]

- Wenn Benutzer eine veröffentlichte Anwendung starten, reagiert der Citrix Stack Control Service (SCService64.exe) nicht mehr.

[#LC1545]

- Melden sich Benutzer von einer Desktopsitzung ab, reagiert WFSHELLWINDOW nicht mehr und die Meldung "Wfshell needs to close" wird drei Minuten lang angezeigt.

[#LC1591]

- In einer Multimonitumgebung wird die Anwendung möglicherweise hinter der veröffentlichte Symbolleistanwendung im Seamlessmodus angezeigt.

Um den Fix zu aktivieren, ist möglicherweise die Installation von Fix LC1342 und LC0491 erforderlich.

[#LC1673]

- Wenn Sie sich an einem Windows-basierten Gerät anmelden und dann mit XenApp-VMs verbinden, erhöht das TWI-Modul (TWI3.dll) die CPU-Nutzung.

[#LC1810]

- In Umgebungen mit zwei Monitoren am Benutzergerät werden manche Anwendungen als schwarze Fenster angezeigt,

wenn mehrere Anwendungen mit HDX-Features gestartet wurden.

[#LC1818]

- Bei der Aufnahme einer XenApp-Sitzung, in der der Legacygrafikmodus deaktiviert ist, können die aufgezeichneten Dateien nicht gerollt werden.

[#LC1850]

- Ein Timeout kann bei einer fortbestehenden Sitzung auftreten. Die Sitzung wird unerwartet beendet und ein schwarzes Fenster angezeigt.

[#LC1903]

- Wenn Benutzer versuchen, sich mit einer getrennten Sitzung zu verbinden, kann es bis zu drei Minuten dauern.

[#LC1908]

- Sind Active Directory-Server als Server mit Lesezugriff eingestellt, können Benutzer keine Anwendungen starten und die Meldung "Verbindung aufgebaut. Funktionalität wird ausgehandelt" wird unendlich lange angezeigt.

[#LC1964]

- Ist die Richtlinie "Automatische Wiederverbindung von Clients" in einer Sitzung aktiviert, wird die Authentifizierungsaufforderung möglicherweise nicht angezeigt.

Zum Implementieren dieses Fixes legen Sie folgende Registrierungsschlüssel fest:

- HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\WFShell
Name: SessionReconnectMinTimeInMilliseconds
Typ: REG_DWORD
Wert: maximale Sitzungsdauer
- HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\WFShell
Name: LockOnReconnect
Typ: REG_DWORD
Wert: 1

Nach dem Konfigurieren des Registrierungsschlüssels starten Sie den Server neu oder führen Sie den Befehl "gpupdate /force" aus.

[#LC1990]

- Beim Starten von veröffentlichten Anwendungen kann Citrix Grafikprozess (CtxGfx.exe) dazu führen, dass der CPU-Verbrauch höher als normal ist.

[#LC2007]

- Die Bildschirmauflösung für die Sitzung wird möglicherweise nicht angepasst, wenn Benutzer eine Sitzung mit Citrix Receiver für Windows trennen und sich dann mit der gleichen Sitzung von einem Gerät mit einer kleineren Bildschirmauflösung wieder verbinden und dafür Citrix Receiver für Linux oder Citrix Receiver für Windows CE verwenden. Dies kann passieren, wenn die Richtlinieneinstellung "Legacygrafikmodus" für die Sitzung aktiviert ist.

[#LC2079]

- Wenn Benutzer den Inhalt einer eingebetteten Word-Datei kopieren, die von SAP aus geöffnet wurde, dann schlägt das

Kopieren in eine neue eingebettete Word-Datei fehl.

[#LC2092]

- Wenn Benutzer einen veröffentlichten Desktop vom Webinterface starten, reagiert die Sitzung nicht mehr beim Einrichten der personalisierten Einstellungen für Microsoft Windows Media Player und Audio funktioniert nicht für den Desktop.

[#LC2103]

- Lotus Notes auf einem iPad mit Receiver für iOS wird möglicherweise unerwartet geschlossen.

[#LC2124]

- Versuche, die Verbindung zu einer getrennten Sitzung wiederherzustellen, schlagen gelegentlich fehl, weil das Sitzungsattribut "AllowReconnect" auf "False" gesetzt ist.

[#LC2174]

- Wenn Benutzer die Verbindung zu einer Sitzung trennen verursacht Citrix WinFrame Shell (WFShell.exe) eine hohe CPU-Auslastung.

[#LC2235]

- Wird Internet Explorer als veröffentlichte Anwendung mit dem Argument /appve gestartet, schlägt das Laden des zugeordneten Pakets fehl.

[#LC2292]

- Wfshell.exe verwendet weiter eine große Menge an CPU-Ressourcen nachdem Benutzer die Sitzung trennen.

[#LC2391]

- Wenn Sie AutoCAD und AutoCAD Civil 3d verwenden, wird der Mauszeiger dauerhaft als Sanduhr angezeigt.

[#LC2438]

- Der Zeitstempel wird nicht aktualisiert, wenn eine Datei von einem zugeordneten Clientlaufwerk in einer veröffentlichten Anwendung geöffnet und nach dem Bearbeiten gespeichert wird.

[#LC2305]

- Versuche, an einer Datei zu arbeiten, die auf einem zugeordneten Clientlaufwerk (CDM) ist, führt möglicherweise zu einem Deadlock bei picadmsys und der veröffentlichte Desktop reagiert möglicherweise nicht mehr.

[#LC2312]

- Wenn eine Excel CSV-Datei eine große Anzahl von Einträgen enthält, kann das Kopieren der Datei vom Benutzergerät zum Server lange dauern.

[#LC2366]

- Beim Maximieren einer veröffentlichten Version von Excel oder Internet Explorer wird der obere Bereich des Namensfelds, der Bearbeitungsleiste oder der Menüleiste weiß angezeigt.

Um diesen Fix für Excel und Internet Explorer zu aktivieren, erstellen Sie die folgenden Registrierungsschlüssel:

Excel

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\wfshell\TWI\XLMAIN

Name: ClassName

Typ: REG_SZ

Wert: XLMAIN

Name: Type

Typ: REG_DWORD

Wert: 0x00020000

Internet Explorer

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\wfshell\TWI\IEFrame

Name: ClassName

Typ: REG_SZ

Wert: IEFrame

Name: Type

Typ: REG_DWORD

Wert: 0x00020000

Um diesen Fix auf mehrere Server anzuwenden, erstellen Sie ein Registrierungsflag. Weitere Informationen finden unter "Seamless Per Application Window Registry Flags" in [Seamless_Configuration.pdf](#) auf der Citrix Supportsite.

[#LC2504]

- Ist Excelhook aktiviert, geht beim Wechseln zwischen geöffneten Arbeitsmappen der Fokus der ausgewählten Arbeitsmappe verloren. Außerdem können Benutzer keine der geöffneten Arbeitsmappen wählen, der Fokus bleibt auf der zuletzt gewählten Datei.

#LC2589]

- Wenn die Einstellung "Nur eine Anwendungsinstanz pro Benutzer zulassen" oder die Richtlinie "Automatische Wiederverbindung von Clients" aktiviert sind, erhalten Benutzer möglicherweise den Fehler "Sie haben bereits eine Instanz dieser Anwendung geöffnet und sind nicht zur Ausführung von mehreren Instanzen berechtigt. Wenden Sie sich an den Systemadministrator." Dies passiert, weil die Sitzungs freigabe zwischen zwei Gesamtstrukturen (forests) fehlschlägt.

[#LC2592]

- Videos sind leer in der Anwendung "Unreal Streaming Media Player", wenn sie in einer veröffentlichten Version von Internet Explorer 8 auf einem Windows 7 64-Bit-Computer abgespielt werden.

[#LC2650]

- Kopieren von mehr als 1000 Dateien mit Clientlaufwerkzuordnung kann zu Leistungsproblemen führen.

[#LC2702]

- Benutzer können nach dem Update der Registrierungsschlüssel für Verschlüsselungsverfahren keine Anwendungen starten. Um diesen Fix anzuwenden, müssen Sie eine neue Version des SSL SDKs installieren.

[#LC2729]

- Sitzungen werden möglicherweise nicht gestartet und die folgende Fehlermeldung angezeigt:

"A device attached to the system is not functioning."

Dies passiert in Kerberos-gesicherten Umgebungen mit bidirektionalen Vertrauensstellungen für Domänen, wenn Benutzer und Server in unterschiedlichen Domänen sind.

[#LC2762]

- Wird eine veröffentlichte Version von Internet Explorer in Windows Phone 8.1 geöffnet und das Gerät horizontal gedreht, zeigt der halbe Bildschirm ein leeres Fenster. Außerdem können Benutzer das Internet Explorer-Fenster nicht maximieren. Drehen Benutzer das Gerät wieder vertikal, erscheint der Bildschirm normal.

[#LC2786]

- Wenn Sitzungs freigabe und Sitzungsfortbestehen aktiviert sind, werden getrennte Sitzungen nicht mit dem gleichen VDA wieder verbunden sondern mit einem anderen VDA. Wenn dies passiert werden auf dem VDA Ereignis ID 1048, 1050 und 7 protokolliert.

[#LC2860]

- Wenn eine veröffentlichte Anwendung nicht mehr funktioniert, lässt der Desktopfenster-Manager möglicherweise nicht zu, dass der Benutzer die Anwendung schließt.

[#LC2922]

- Wenn sich ein Benutzer mit einer Sitzung auf einem Server verbindet, kommt die Sitzung möglicherweise nicht über den Willkommensbildschirm hinweg. Dies kann für alle Sitzungen auf dem Server passieren, bis der Server neu gestartet wird.

[#LC2936]

- Schlüssel und Wert in dem Registrierungsspeicherort HKEY_CURRENT_USER\Control Panel\Desktop\UserPreferencesMask auf dem VDA werden möglicherweise jedes Mal, wenn sich ein Benutzer an dem VDA anmeldet, von dem Prozess wfshell.exe überschrieben. Sie verhindern dies, indem Sie den folgenden Registrierungsschlüssel auf dem VDA erstellen und den Wert auf 1 setzen:

HKLM\SYSTEM\CurrentControlSet\Control\Citrix

DWORD-Wert:

EnableVisualEffect

[#LC2965]

- Verknüpfungen für Desktops und Downloads im Favoritenmenü von Windows Explorer unter %userprofile%\Links werden nach dem Löschen wiederhergestellt, wenn sich der Benutzer wieder beim VDA anmeldet.

[#LC2998]

- Wenn sich Benutzer wiederholt an- und abmelden, reagiert Citrix WinFrame Shell (Wfshell.exe) nicht mehr.

[#LC3003]

- Wenn sich Benutzer an Receiver 4.2 anmelden, stimmt die Uhrzeit, die in der Sitzung angezeigt wird, nicht mit der Uhrzeit auf dem Benutzergerät überein.

[#LC3067]

- Versuche von Administratoren, Anwendungen zu starten, die aus dem System32-Ordner veröffentlicht wurden, können fehlschlagen.

[#LC3148]

- Beim Verwenden eines externen Monitors mit einem Laptop, kann ein weißes Quadrat in einer Ecke des externen Monitors sichtbar sein. Das Problem tritt speziell auf, wenn der externe Monitor eine andere Bildschirmauflösung als der primäre Monitor hat, links neben dem primären Monitor steht und seine obere Kante über die obere Kante der primären Monitors hinausragt.

[#LC3186]

- Versuche schlagen fehl, eine Verbindung zu einer getrennten Sitzung herzustellen, nachdem Server mit Volllast über das Lastauswertungsprogramm in den Offlinezustand versetzt wurden.

[#LC3226]

- Wenn Benutzer eine Verbindung zu einem veröffentlichten Desktop mit Receiver 4.2 herstellen, schlägt Citrix WinFrame Shell (wfshell.exe) fehl.

[#LC3227]

- Wenn veröffentlichte Anwendungen oder Desktops auf einem XenApp-Server gestartet werden, funktioniert Single Sign-On möglicherweise nicht.

[#LC3301]

- Bei deaktivierter USB-Umleitung kann das Trennen von einer veröffentlichten Anwendung dazu führen, dass fälschlicherweise das folgende Ereignis 261 im Ereignisprotokoll erscheint:

Der Citrix Geräteumleitungsdienst konnte einen E/A-Vorgang mit dem Umleitungsbus nicht beenden.

[#LC3439]

- Beim Beenden einer 64 Bit ThinAPP-verpackten Anwendung (BMC Control-M) kann für die Anwendung zu einer unerwarteten Ausnahme bei sfrhook64.dll kommen.

[#LC3484]

- Gelegentlich schlägt die Zwischenablagezuordnung fehl, nachdem Sie die Verbindung zu einer getrennten Sitzung wiederherstellen. Das Problem tritt auf, wenn der virtuelle Kanal für die Zwischenablage unerwartet bei der Wiederverbindung geschlossen wird.

[#LC3511]

- SSL-Verbindungen (HTTPS) funktionieren möglicherweise nicht richtig in einem VDA.

[#LC3535]

Site-/Farmverwaltung

- Ist der Registrierungswert HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\Logon\DisableStatus aktiviert (Wert=1), fehlen für End User Experience Monitoring (EUEM) die Werte "Serverdauer für Laden des Profils" und "Serverdauer

für Ausführung des Anmeldeskripts" möglicherweise.

[#LA5764]

- Versuche, Einstellungen von Lastauswertungsprogrammen zu überprüfen, können fehlschlagen und die folgende Fehlermeldung tritt auf:

System.Runtime.InteropServices.COMException (0x80004005): Error HRESULT E_FAIL has been returned from a call to a COM component.

[#LC0359]

- Mehrere XenApp-Server werden zur gleichen Zeit neu gestartet, statt zu unterschiedlichen Zeiten, wie in der Richtlinie "Willkürliches Neustartintervall" festgelegt.

[#LC1402]

- Nicht-Administrator-Benutzer können keine RDP-Sitzung zu einem XenApp 7.6-Server herstellen, wenn ein Remotedesktopgateway angegeben wurde.

[#LC2371]

- Es werden möglicherweise fehlerhafte Updates am lokalen Hostcache des Servers vorgenommen, nachdem die Microsoft-API-Funktion DsBind() API fehlschlägt. Symptome für dieses Problem sind u.a. schnelles Anwachsen der Größe des lokalen Hostcache auf Controllern und Änderungen am Inhalt von Workergruppen, die auf Active Directory-Organisationseinheiten verweisen.

[#LC2631]

- Der Befehl "dscheck /full workergroup" berichtet möglicherweise Fehler nicht richtig beim Überprüfen von Workergruppen, die keine Farmserverobjekt enthalten. Folgende Fehlermeldung wird angezeigt:

"Error:: Failed to retrieve Server UIDs for the WorkGroupUID."

Mit diesem Fix wird die angezeigte Meldung in diesem Fall in Folgendes geändert:

"This workergroup is configured for Farm Server and has no Server UID, verification OK."

[#LC3383]

- Das Willkürlichkeitsintervall für den Neustartzeitplan wird möglicherweise nicht für Server umgesetzt, die mit Provisioning Services bereitgestellt wurden.

[#LC3430]

- Wurde eine Anwendung in einer Workergruppe veröffentlicht, gibt der Befehl dscheck /full apps /servercheck fälschlicherweise das Workergruppenobjekt als unbekannten Server zurück und bietet die Option, ihn zu löschen.

[#LC3556]

Smartcards

- Der Citrix Smartcarddienst verursacht möglicherweise eine sehr hohe CPU-Auslastung. Um dieses Problem zu beheben, starten Sie den Citrix Smartcarddienst neu.

[#LA5919]

- Nach dem Upgrade von VDAs fordert Microsoft Outlook nicht zum Eingeben einer PIN auf, wenn Benutzer sich mit einer Smartcard anmelden wollen. Die folgende Fehlermeldung wird angezeigt:

"Please insert a smart card."

[#LC0556]

- Nach dem Anmelden an einem Server mit einer Smartcard, kann es passieren, dass der Server auf der Willkommenseite nicht mehr reagiert und keine neuen Sitzungen annimmt. Neustarten des Smartcarddienstes löst das Problem und der Server nimmt weiter neue Sitzungen an.

Konfigurieren der folgenden Registrierungsschlüssel ist möglicherweise nicht notwendig, wenn Fix LA0983 bereits installiert ist und Sie die Registrierungsschlüssel bei der vorangegangenen Hotfixinstallation konfiguriert haben.

HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\SmartCard

Name: TransactionTimeoutEnable

Typ: REG_DWORD

Wert: 1 (aktiviert)

HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\SmartCard

Name: TransactionTimeoutValue

Typ: REG_DWORD

Wert:

HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\SmartCard

Name: SendRecvTimeout

Typ: REG_DWORD

Wert: Mindesttimeoutwert in Sekunden, sollte 30 Sekunden oder mehr sein. Niedrigere Werte werden automatisch in 30 Sekunden geändert. Dieser Wert sollte mindestens 10 Sekunden mehr als der Wert für "TransactionTimeoutValue" sein.

[#LC0910]

- Nach dem Authentifizieren mit Citrix Receiver für Linux während eine Smartcard in das Lesegerät eingelegt ist, das an den Endpunkt angeschlossen ist, kann es passieren, dass die Sitzung nicht reagiert.

[#LC0982]

- Der Prozess ActivIdentity (accompkcs.exe) wird möglicherweise unerwartet beendet mit einer Ausnahme bei scardhook64.dll. Das Resultat ist, dass Websites für die eine Smartcard erforderlich ist, nicht funktionieren bis der VDA neu gestartet wird.

[#LC1056]

- Der Citrix Smartcarddienst reagiert möglicherweise nicht mehr und Benutzer können sich nicht am Server anmelden bis der Dienst neu gestartet wird.

[#LC1276]

- Wird eine Smartcard während der Sitzungsanmeldung aus dem Lesegerät entfernt, kann es verhindern, dass die Sitzung auf dem Server richtig beendet wird. Dies führt dazu, dass der Server in einem Zustand ist, in dem Smartcardanmeldungen für zukünftige Sitzungen verhindert werden.

[#LC3020]

- Beim Verbinden mit einem veröffentlichten Desktop über eine Smartcard reagiert der Citrix Smartcarddienst möglicherweise nicht mehr.

[#LC3421]

Systemausnahmen

- Im Betriebssystem kommt es zu einem Fehler mit picaser.sys und ein Bluescreen mit Bugcheckcode 0x0000000A wird angezeigt.

[#LC0222]

- Im Betriebssystem kommt es zu einem Fehler mit picadm.sys und ein Bluescreen mit Bugcheckcode 0x22 wird angezeigt.

[#LC0227]

- Auf XenApp 6.5-Servern mit Hotfix Rollup Pack 3 oder 4, die als XML-Broker fungieren, können die Remotedesktopdienste unerwartet beendet werden.

[#LC0440]

- Es kommt möglicherweise auf Servern zu einem schwerwiegenden Fehler mit Vdtw30.dll, ein Bluescreen wird angezeigt.

[#LC0471]

- Im Betriebssystem kommt es zu einem Fehler mit picadm.sys und ein Bluescreen mit Bugcheckcode 0x00000022 wird angezeigt.

[#LC0483]

- In Sitzungen auf Systemen mit Fix LA5282 (zuerst veröffentlicht in XA650W2K8R2X64R04 - XenApp 6.5 Hotfix Rollup Pack 4) kann das Klicken auf URLs in einer veröffentlichten Anwendung dazu führen, dass der Prozess winlogon.exe unerwartet beendet wird. Die veröffentlichte Anwendung wird dann getrennt.

[#LC0523]

- Der Citrix XML-Dienst kann unerwartet fehlschlagen.

[#LC0548]

- Die Wiedergabe eines Videos in einem Medienplayer in einer Passthrough-Sitzung kann zum unerwarteten Beenden der Sitzung führen.

[#LC0553]

- Es kommt auf Servern möglicherweise zu einem schwerwiegenden Fehler bei vdtw30.dll mit Bugcheckcode 0x0000007E auf einem Bluescreen.

[#LC0599]

- Im Betriebssystem kommt es zu einem Fehler mit picadm.sys und ein Bluescreen wird angezeigt.

[#LC0623]

- Auf Servern kommt es möglicherweise zu einem Fehler mit picadm.sys und ein Bluescreen mit Bugcheckcode 0x0000000A wird angezeigt.

[#LC0649]

- Beim Anmelden von einer Remotedesktop (RDP)-Sitzung wird der Prozess winlogon.exe möglicherweise unerwartet bei twi3.dll beendet.

[#LC0793]

- Auf VDAs kann es zu einem Fehler bei picadd.sys kommen und ein Bluescreen wird angezeigt.

[#LC0798, #LC1096]

- Im VDA kommt es zu einem Fehler mit ctxdvc.sys und ein Bluescreen mit Bugcheckcode E3 wird angezeigt.

[#LC0857]

- Im Betriebssystem kommt es zu einem Fehler mit picadm.sys und ein Bluescreen mit Stopppcode 0x0000003B wird angezeigt.

[#LC0896]

- Windows Explorer kann in folgenden Fällen unerwartet geschlossen werden:
 - Wenn eine große Anzahl von Dateien ausgewählt wird, deren Namen mehr als 260 Zeichen enthält, und die Option "Senden an > Faxempfänger" ausgewählt wird
 - Wenn versucht wird, Anwendungen von Drittherstellern zu öffnen
 - Wenn versucht wird, Dateien mit Nitro PDF zu kombinieren

[#LC0938]

- Ein Seitenfehler im ICA WSK-Transporttreiber (tdwsk.sys) führt dazu, dass der VDA mit einem Fehler auf einem Bluescreen fehlschlägt.

[#LC1032]

- Beim Starten einer veröffentlichten Anwendung oder eines öffentlichen Desktops wird die Windows-Anmeldebenutzeroberfläche (LogonUI.exe) oder der Prozess explorer.exe möglicherweise unerwartet bei icaendpoint.dll beendet, wenn der Start des Citrix Audioumleitungsdiensts fehlschlägt.

[#LC1167]

- Im Betriebssystem kommt es zu einem Fehler mit picadm.sys und ein Bluescreen wird angezeigt.

[#LC1239]

- Server werden unerwartet beendet bei vdtw30.dll.

[#LC1328]

- Es kommt möglicherweise auf Servern zu einem schwerwiegenden Fehler mit Vdtw30.dll, ein Bluescreen wird angezeigt.

[#LC1414]

- Ein Fehler bei twi3.dll kann dazu führen, dass Server nicht mehr reagieren.

[#LC1417]

- Der Prozess svchost.exe für die Terminaldienste wird möglicherweise unerwartet mit Anwendungsereignis-ID 1000 beendet und dies kann dazu führen, dass Remotedesktop- und Receiver-Sitzungen zum Server fehlschlagen.

[#LC1419]

- Im Betriebssystem kommt es zu einem Fehler mit picadm.sys und ein Bluescreen wird angezeigt. Das Problem tritt auf, wenn der Wert von "TrucationResource" Null wird.

[#LC1430]

- Ein Fehler in vdtw30.dll kann einen Bluescreen mit Bugcheckfehler 0x0000003B, {c0000005, fffff96000863133, fffff880099aaf70, 0} verursachen.

[#LC1567]

- Es kann auf dem Server zu einem Deadlock bei picadm.sys kommen, der verhindert, dass der Server neue Benutzerverbindungen annimmt.

[#LC1685]

- Der Citrix Stack Control-Dienst wird unerwartet beendet, wenn ein ungültiger Sitzungsschlüssel vorliegt.

[#LC1717]

- Das Abspielen einer Audiodatei in Receiver für HTML5, wenn SSL in der ICA-Sitzung aktiviert ist, kann zu einer schwerwiegenden Ausnahme beim VDA führen. Es wird ein Bluescreen angezeigt.

[#LC1752]

- Der Prozess svchost.exe wird möglicherweise unerwartet angehalten durch die ICA-Audioendpunkt-DLL (icaendpoint.dll).

[#LC1775]

- Gelegentlich wird der VDA möglicherweise unerwartet heruntergefahren und dann neu gestartet.

{#LC1876}

- Terminaldienste werden möglicherweise unerwartet durch RPM.dll geschlossen.

[#LC2031]

- Es kommt auf Servern möglicherweise zu einem schwerwiegenden Fehler bei vdtw30.dll mit Bugcheckcode 0x0000003B auf einem Bluescreen.

[#LC2037]

- Im Betriebssystem kommt es zu einem Fehler mit ctxdvc.sys und ein Bluescreen mit Bugcheckcode 3B wird angezeigt.

[#LC2119]

- Im Betriebssystem kommt es zu einem Fehler mit tdica.sys und ein Bluescreen mit Bugcheckcode 44 wird angezeigt. [MULTIPLE_IRP_COMPLETE_REQUESTS].

[#LC2171]

- Es kommt mit Windows Server 2008 R2 und Windows Server 2012 R2 möglicherweise zu einer schwerwiegenden Ausnahme, wenn Benutzer die Verbindung mit Receiver für Windows herstellen.

[#LC2179]

- Ein Fehler bei picadm.sys mit Bugcheckcode 0x00000022 wird auf einem Bluescreen angezeigt.
[#LC2236]
- Der Prozess winlogon.exe wird möglicherweise unerwartet bei twi3.dll geschlossen.
[#LC2244]
- Im Betriebssystem kommt es zu einem Fehler mit picadm.sys und ein Bluescreen mit Bugcheckcode 0x00000022 wird angezeigt.
[#LC2265]
- Auf VDA-Servern kommt es zu einem Fehler mit icaridd.dll und ein Bluescreen wird angezeigt.
[#LC2277]
- Auf XenApp 6.5 Servern mit Hotfix Rollup Pack 5, die als XML-Broker fungieren, können die Remotedesktopdienste unerwartet beendet werden.
[#LC2376]
- Es kommt auf Servern möglicherweise zu einem schwerwiegenden Fehler bei ctxdvcs.sys mit Bugcheckcode 0x0000003B auf einem Bluescreen.
[#LC2444]
- Ein Windows Explorer-Prozess wird möglicherweise während einer veröffentlichten Desktopsitzung unerwartet beendet und die Fehlermeldung "Windows Explorer funktioniert nicht mehr" angezeigt.
[#LC2733]
- Wird ein Point-of-Sales-Gerät an einen freigegebenen Desktop über den COM-Port umgeleitet, erkennt möglicherweise Drittanbietersoftware, die auf dem freigegebenen Desktop ausgeführt wird, das Gerät nicht.
[#LC2775]
- Es kommt auf Servern möglicherweise zu einem Fehler bei icausbbsys und die Meldung PNP_DETECTED_FATAL_ERROR wird auf einem Bluescreen angezeigt.
[#LC2811]
- Der Prozess wfshell.exe wird möglicherweise unerwartet geschlossen und neue Verbindungen verhindert.
[#LC3024]
- Es kommt auf Servern möglicherweise zu einer schwerwiegenden Ausnahme bei vdtw30.dll und win32k.sys mit einem Fehler auf einem Bluescreen.
[#LC3103]
- Nachdem Sie die Richtlinie für die Clientzwischenablagenumleitung aktivieren, wird der Prozess wfshell.exe möglicherweise unerwartet geschlossen.
[#LC3249]

- Es kommt auf VDAs möglicherweise zu einer schwerwiegenden Ausnahme bei vdtw30.dll, ein Bluescreen wird angezeigt.
[#LC3354]

- Es kann mit dem Prozess csrss.exe zu einer schwerwiegenden Ausnahme bei vdtw30.dll mit Stopppcode 0x7e kommen.
[#LC3482]

Benutzererfahrung

- Dieser Fix behandelt ein Benutzerfreundlichkeitsproblem der Desktopgestaltungsumleitung mit Receiver für Windows vor Version 4.200 bei LAN-Verbindungen.

[#LC1507]

- Dieses Release enthält eine Funktionserweiterung, sodass Benutzer, die sich mit Receiver für HTML5 oder Receiver für Chrome anmelden, Dateien über Remotesitzungen und lokale Geräte übertragen können.

[#LC3014]

Benutzeroberfläche

- Dieser Fix die folgenden Probleme mit der Darstellung von Anwendungen in der Receiver für Windows-Taskleiste:
 - Einzelne Anwendungen können mit dem Receiver-Standardsymbol statt dem anwendungsspezifischen Symbol angezeigt werden.
 - Werden mehrere Anwendungen in der gleichen Sitzung ausgeführt, kann es passieren, dass sie unter einem einzelnen Receiver-Standardsymbol erscheinen.

[#LC0094, #LC1181]

- Ist die Richtlinie "Kombinationsfeld remoten" aktiviert, wird die Blase für die remoten Kombinationsfelder nicht richtig angezeigt, wenn Benutzer Internet Explorer in Receiver für iOS mit einem iPad öffnen. Beispiel: Wenn Sie auf Kombinationsfeld B klicken, scheint die Blase von Kombinationsfeld A zu kommen.

[#LC1800]

- Bestimmte Rahmen der Seamlessanwendungsfenster sind möglicherweise verborgen. Das Problem tritt auf, wenn die Funktion "GetWindowRect" nicht die richtige Fenstergröße abrufen kann und führt dazu, dass die Fensterkoordinaten ohne Rahmen erscheinen.

[#LC1943]

- Die automatische Tastaturanzeige funktioniert möglicherweise nicht, wenn Benutzer eine veröffentlichte Anwendung mit Receiver für Windows oder den mobilen Receiver öffnen.

[#LC2747]

Sonstiges

- Dieser Fix bietet allgemeine Kompatibilitätsverbesserungen mit Microsoft Updates.

[#LA5638]

- Dieser Fix behebt ein Speicherproblem in einer Hintergrundkomponente.

[#LA5765]

- Bei dem Versuch, die Lizenzserveradresse abzurufen, erscheinen falsche Zeichen im Lizenzservernamen.

[#LC0363]

- Dieser Fix behebt Handlecks im WMI-Prozess.

[#LC0662]

- Dieser Fix behebt ein Stabilitätsproblem mit der Umleitung spezieller Ordner und Dateinamen, die mehr als 205 Zeichen haben.

[#LC0719]

- Dieser Fix behebt ein Speicherproblem in einer Hintergrundkomponente.

[#LC0732, #LC0800, #LC0953, #LC2764, #LC2888]

- Wenn Benutzer eine veröffentlichte Version von Excel starten, eine Datei mit "Per E-Mail senden" verschicken, und dann in Outlook Text eingeben und die Rücktaste einige Mal drücken, werden chinesische Zeichen statt der Rücktaste eingefügt.

[#LC1155]

- COM-Port-Unterschriftstabletts mit USB-Adaptern funktionieren nicht korrekt, wenn Benutzer sich mit Receiver für Mac anmelden. Die Unterschriftstabletts funktionieren korrekt, wenn Benutzer sich mit Receiver für Windows anmelden.

[#LC1396]

- Ein Handleck in CTXCDF führt dazu, dass der WMI-Anbieterdienst (WMIPRVSE.exe) anhält. Ereignis-ID 5612 wird im Ereignisprotokoll angezeigt.

[#LC1764]

- Dieser Fix behebt Speicherlecks im Citrix Audioumleitungsdienst (CtxAudioSvc).

[#LC2054]

- Der Powershell-Befehl "Get-BrokerSession | Format-List -Property SessionState" zeigt nicht den richtigen Sitzungszustand an. Beispielsweise werden getrennte Sitzungen als aktive Sitzungen angezeigt.

[#LC2080]

- In diesem Release wird Framehawk Virtual Channel, ein neuer virtueller ICA-Kanal unterstützt, der die HDX-Technologien von Citrix erweitert und die Benutzererfahrung für drahtlose Breitbandverbindungen verbessert, bei denen Paketverlust und Latenz ein häufiges Problem sind. Weitere Informationen finden Sie unter [Neue Features](#).

[#LC2782]

- Diese Funktionserweiterung verbessert die Meldung für Ereignis-ID 1480 und enthält die folgenden Informationen:
 - Name der veröffentlichten Anwendung
 - Befehlszeile
 - Benutzer-ID

- Benutzerdomäne
- Clientname
- Sitzungs-ID

[#LC2830]

- Erstellen der Datei sessionchange.log im Stamm von des Laufwerks C: auf dem VDA kann nicht deaktiviert oder an einen anderen Ort verschoben werden.

[#LC3621]

Bekannte Probleme

Nov 02, 2016

Feature Pack 3

- In neuen Firefox-Versionen sind die Browser-Add-Ons für die URL-Umleitung standardmäßig deaktiviert, daher müssen Sie sie aktivieren. Dies ist ein Drittanbieterproblem [#283844]
- Benutzer können einige Dateien aus umgeleiteten Ordnern (z. B. "Downloads" und "Desktop") nicht öffnen, wenn sie die Dateien mit dem Edge-Browser auf einem Windows 10-VDA heruntergeladen haben. Zur Problemvermeidung verwenden Sie Internet Explorer für den Download von Dateien. [#591635]
- Die folgenden Registrierungsschlüssel sind nach dem Upgrade eines brokerlosen VDA auf FP3 nicht vorhanden und müssen manuell hinzugefügt werden [#591030]:
 - hklm\software\citrix\VirtualDesktopAgent\HaModeComputerName
 - hklm\software\citrix\VirtualDesktopAgent\HaModeTimeEnd
- Bei Verwendung von Receiver für Windows mit einem Wacom-Stiftgerät und einem Windows 10-VDA wird bei einer Unterbrechung der Verbindung mit dem Netzwerk der Cursor möglicherweise nicht angezeigt, nachdem die Sitzung über die automatische Wiederverbindung von Clients (Automatic Client Reconnection, ACR) wiederhergestellt wurde. Sie können dieses Problem beheben, indem Sie die Sitzungsverbindung schließen und anschließend wiederherstellen. [#593516]
- Die persistente CDF-Ablaufverfolgung ist nicht verfügbar, wenn der Citrix Telemetriedienst ausgeführt wird. Um dieses Problem zu umgehen, deaktivieren Sie vorübergehend den Citrix Telemetriedienst. [#593347]
- Es kommt zu einem Fehler in XenApp und XenDesktop 7.6 FP3, wenn ein Windows 10-Masterimage auf XenServer zur Verwendung von XenServer-Hotfix [XS65ESP1010](#) mit XenTools aktualisiert wird. [#593985]
- Bei Anwendungen, die von einem Windows 10-VDA veröffentlicht werden, werden in der maximierten Anzeige die Ränder nicht vollständig angezeigt [#593596]. Um dieses Problem zu umgehen, können Registrierungsschlüssel für jede Anwendungsklasse festgelegt werden. Zum Beheben des Problems bei Paint erstellen Sie beispielsweise den folgenden Schlüssel:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\wfshe\TWI\MSPaintApp]
```

```
"ClassName"="MSPaintApp"
```

```
"Type"=dword:00020000
```

Achten Sie darauf, dass Sie den Klassennamen und nicht den Namen der Anwendung hinzufügen. Fügen Sie beispielsweise für Paint **MSPaintApp** hinzu. Sie können die Klassennamen mit einem Tool wie Spy++ in Erfahrung bringen. Weitere Informationen zu Seamless-Konfigurationseinstellungen finden Sie unter [CTX101644](#).

- Einige "Weitere Informationen"-Links führen möglicherweise nicht zur aktuellen Produktdokumentationswebsite (docs.citrix.com). Um dieses Problem zu umgehen, greifen Sie direkt auf die Produktdokumentationswebsite zu. Die gesamte Dokumentation für 7.6 FP3 und ältere 7.6-Feature Packs steht auf der neuen Produktdokumentationswebsite unter [XenApp und XenDesktop 7.6](#) zur Verfügung.
- Temporäre Dateien des eigenständigen VDA-Installationspakets verbleiben nach Abschluss der Installation im Temp-Ordner des VDA-Betriebssystems. Um dieses Problem zu beheben, löschen Sie die temporären Dateien manuell. [#594829]

Feature Pack 2

Linux Virtual Desktop

Rich-Text-Formatierung geht verloren, wenn Text aus einer Anwendung in der Sitzung kopiert und in einer Anwendung außerhalb der Sitzung eingefügt wird, und wenn Text außerhalb der Sitzung kopiert und innerhalb der Sitzung eingefügt wird. [#0538497]

Hostnamen (Hostteil des vollqualifizierten Domännennamens) mit mehr als 15 Zeichen werden abgeschnitten, wodurch keine VDA-Registrierung durchgeführt werden kann. [#0544120]

Der Linux-VDA-Registrierungsstatus wird auf Endpunkten bei Wiederaufnahme des Betriebs nach dem Ruhezustand nicht geändert. Bei einem Energiesparmodus-Ereignis erhält der Endpunkt keine IP-Adresse, wodurch verhindert wird, dass er registriert wird. [#0569303]

Der Linux-VDA unterstützt derzeit keine DNS-Suche für Bereichszuordnungen. [#0557555]

KDE-basierte Anwendungen erscheinen in Linux-VDA-Sitzungen als nicht verfügbar. [#0568006]

Linux-VDA-s mit RedHat Enterprise können keine Verbindung herstellen, wenn der Quest-Authentifizierungsdienst aktiviert ist und SELinux im Erzwingungsmodus ist. [#0551761]

Framehawk Virtual Channel

Citrix Receiver für Windows (Version 4.3) und der klassische Citrix Receiver für iOS (Version 6.0) sind derzeit die einzigen unterstützten Receiver-Plattformen.

Sitzungsaufzeichnung bzw. Smart Auditor werden für Framehawk nicht unterstützt.

Dieses Framehawk-Release ist für Standardarbeitslasten in XenApp und XenDesktop konzipiert, z. B. für Apps für Wissensarbeiter und Büro-Apps. Es wurde nicht für grafikintensive High-End-Umgebungen getestet.

Clients mit mehreren Monitoren werden nicht unterstützt.

Die Vd3dn.dll kann auf dem Client unter Testbedingungen, z. B. beim Ändern der Bandbreite und Latenz auf einem WAN-Emulator während einer Sitzung, abstürzen. Trennen Sie die Sitzung, bevor Sie Änderungen an den WAN-Emulatoreinstellungen vornehmen, und stellen Sie dann die Verbindung wieder her. Framehawk wird während des Sitzungshandshakes neu kalibriert.

Wenn sowohl die Framehawk-Richtlinie als auch die Legacy-Richtlinie aktiviert sind, hat die Framehawk-Richtlinie auf Arbeitsstationsbetriebssystemen (wie Windows 7) Vorrang. In RDS-Umgebungen (z. B. Windows 2012 R2) wird die Legacy-Richtlinie angewendet.

In Windows 7-Umgebungen bietet das anfängliche XenDesktop 7.6 FP2 Release von Framehawk keine Unterstützung für Toucheingabe (z. B. die Gesten zum Vergrößern und Verkleinern), obwohl diese Funktionalität in einem Windows 7-Betriebssystem vorhanden ist. Auch Windows 2008 R2 unterstützt keine Toucheingabe.

StoreFront 3.0

Eine Liste der bekannten und der behobenen Probleme in StoreFront 3.0 finden Sie in der Dokumentation zu StoreFront 3.0 unter [Bekannte Probleme](#) und [Behobene Probleme](#).

HDX RealTime Optimization Pack 1.8

Eine Liste der bekannten und behobenen Probleme in HDX RealTime Optimization Pack 1.8 finden Sie in der Dokumentation zu HDX RealTime Optimization Pack 1.8 unter [Bekannte und behobene Probleme](#).

Feature Pack 1

Sitzungsaufzeichnung

- Wenn von Maschinenerstellungsdienste (MCS) oder Provisioning Services ein VDA mit einem konfigurierten Masterimage und installiertem Microsoft Message Queuing (MSMQ) erstellt wird, erhält der VDA die gleiche QMID wie MSMQ. Dies kann u. a. zu folgenden Problemen führen:
 - Sitzungen werden nicht aufgezeichnet, selbst wenn eine Aufzeichnungsvereinbarung akzeptiert wurde.
 - Das Sitzungsabmeldungssignal wird möglicherweise vom Sitzungsaufzeichnungsserver nicht empfangen, sodass die Sitzung permanent den Status "Live" hat. [#528678]
- Um zur Problemlösung eine eindeutige und bleibende QMID für jeden VDA zu erstellen, verwenden Sie ein Skript. Führen Sie zum Verwenden des Skripts folgende Schritte aus:

1. Stellen Sie sicher, dass die Ausführungsrichtlinie in PowerShell auf "RemoteSigned" oder "Unrestricted" festgelegt ist.
Set-ExecutionPolicy RemoteSigned
2. Erstellen Sie einen geplanten Task und legen Sie als Auslöser "Bei Systemstart" fest und für die Ausführung auf dem Computer mit dem Provisioning Services- oder MCS-Masterimage das Konto SYSTEM.
3. Fügen Sie den Befehl als Starttask hinzu.
powershell.exe -file C:\GenQMID.ps1

Warnung

Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

Zusammenfassung des Skripts:

1. Erstellen der QMID basierend auf dem Hashwert des Computer-FQDNs
2. Anhalten zugehöriger Dienste, einschließlich CitrixSmAudAgent und MSMQ
3. Festlegen der QMID in der Registrierung
4. Starten der zuvor angehaltenen Dienste zum Anwenden der geänderten QMID

Das folgende Skript dient als Referenz. KOPIEREN

```
function ConvertHexStringToByte($theString)
{
    $bytes = New-Object Byte[] ($theString.Length / 2)
    for ($i = 0; $i -lt $theString.Length; $i += 2) {
        $bytes[$i / 2] = [System.Convert]::ToByte($theString.Substring($i, 2), 16)
    }
    return $bytes
}

Try {
    # Get UUID of machine

    $strUUID = (Get-WmiObject -Class Win32_ComputerSystemProduct | Select-Object -Property UUID).UUID

    # Remove "."
    $strUUID = $strUUID.ToString().Replace(".", "")

    # Convert string to bytes
    $UUID = ConvertHexStringToByte($strUUID)

    # Set UUID as QMID
    $new_QMID = $UUID

} Catch { }
```

```

# IF exception occurred, just use MD5 digest of FQDN as QMID

# Get FQDN

$fqdn = [System.Net.Dns]::GetHostByName(($env:computerName)).HostName

# Calculate MD5 hash of FQDN

$md5 = new-object -TypeName System.Security.Cryptography.MD5CryptoServiceProvider

# Set md5 digest as QMID

$utf8 = new-object -TypeName System.Text.UTF8Encoding

$new_QMID = $md5.ComputeHash($utf8.GetBytes($fqdn))

}

# Write new_QMID into registry

Set-ItemProperty -Path HKLM:\Software\Microsoft\MSMQ\Parameters\MachineCache -Name "QMID" -Value $new_QMID

# Restart MSMQ to adopt new QMID

# Get dependent services

$depServices = Get-Service -name MSMQ -dependentservices | Select -Property Name

Restart-Service -force MSMQ

# Start dependent services

if ($depServices -ne $null) {

foreach ($depService in $depServices) {

$startMode = Get-WmiObject win32_service -filter "NAME = '$($depService.Name)'" | Select -Property StartMode

if ($startMode.StartMode -eq "Auto") {

Start-Service $depService.Name

}

}

}

```

- Beim Aufzeichnen einer Sitzung mit einer Auflösung über 4096 x 4096 ist die Aufzeichnungsanzeige u. U. fragmentiert. [#524973]
- Wenn Sie den Lizenztyp für XenApp oder XenDesktop ändern, wird die Änderung nicht sofort für die Sitzungsaufzeichnung wirksam. Workaround: Starten Sie den VDA-Computer neu. [#532393]
- In den folgenden beiden Fällen wird möglicherweise ein Fehler bei der Installation gemeldet. Sie können diese Meldung ignorieren. Um ihre Anzeige ganz zu vermeiden, starten Sie den Computer neu, bevor Sie die Komponenten der Sitzungsaufzeichnung neu installieren. [#544579]
 - Die Komponenten der Sitzungsaufzeichnung werden deinstalliert und dann ohne Neustart des Computers neu installiert.
 - Bei der Installation tritt ein Fehler auf und es erfolgt ein Rollback. Sie versuchen dann, die Komponenten der Sitzungsaufzeichnung neu zu installieren, ohne den Computer neu zu starten.
- Einschränkung für die Unterstützung vorab gestarteter Anwendungssitzungen durch die Sitzungsaufzeichnung [BUG0561109]
 - Problem:
 - Wenn die aktive Richtlinie versucht, den Anwendungsnamen zuzuordnen, wird die in der vorab gestarteten Sitzung gestartete Anwendung nicht zugeordnet, sodass die Sitzung nicht aufgezeichnet wird.
 - Wenn die aktive Richtlinie jede Anwendung aufzeichnet und der Benutzer sich bei Windows Receiver anmeldet (zur gleichen Zeit, zu der die vorab gestartete Sitzung eingerichtet wird), erscheint eine Aufzeichnungsbenachrichtigung und die leere Sitzung sowie alle darin anschließend gestarteten Anwendungen werden aufgezeichnet.
 - Problemumgehung:
 - Veröffentlichen Sie die Anwendungen gemäß ihrer Aufzeichnungsrichtlinie in separaten Bereitstellungsgruppen. Verwenden Sie nicht den Anwendungsnamen als Aufzeichnungsbedingung.

Damit wird sichergestellt, dass vorab gestartete Sitzungen aufgezeichnet werden. Benachrichtigungen werden jedoch weiterhin angezeigt.

- Beim Aufzeichnen einer Sitzung mit einer Auflösung über 4096 x 4096 ist die Aufzeichnungsanzeige u. U. fragmentiert. [#524973]
- Wenn Sie den Lizenztyp für XenApp oder XenDesktop ändern, wird die Änderung nicht sofort für die Sitzungsaufzeichnung wirksam. Workaround: Starten Sie den VDA-Computer neu. [#532393]
- In den folgenden beiden Fällen wird möglicherweise ein Fehler bei der Installation gemeldet. Sie können diese Meldung ignorieren. Um ihre Anzeige ganz zu vermeiden, starten Sie den Computer neu.
 - Die Komponenten der Sitzungsaufzeichnung werden deinstalliert und dann ohne Neustart des Computers neu installiert.
 - Bei der Installation tritt ein Fehler auf und es erfolgt ein Rollback. Sie versuchen dann, die Komponenten der Sitzungsaufzeichnung neu zu installieren, ohne den Computer neu zu starten.
- Beim Aufzeichnen einer Sitzung mit einer Auflösung über 4096 x 4096 ist die Aufzeichnungsanzeige u. U. fragmentiert. [#524973]
- Wenn Sie den Lizenztyp für XenApp oder XenDesktop ändern, wird die Änderung nicht sofort für die Sitzungsaufzeichnung wirksam. Workaround: Starten Sie den VDA-Computer neu. [#532393]
- In den folgenden beiden Fällen wird möglicherweise ein Fehler bei der Installation gemeldet. Sie können diese Meldung ignorieren. Um ihre Anzeige ganz zu vermeiden, starten Sie den Computer neu.
 - Die Komponenten der Sitzungsaufzeichnung werden deinstalliert und dann ohne Neustart des Computers neu installiert.
 - Bei der Installation tritt ein Fehler auf und es erfolgt ein Rollback. Sie versuchen dann, die Komponenten der Sitzungsaufzeichnung neu zu installieren, ohne den Computer neu zu starten.
- Beim Aufzeichnen einer Sitzung mit einer Auflösung über 4096 x 4096 ist die Aufzeichnungsanzeige u. U. fragmentiert. [#524973]
- Wenn Sie den Lizenztyp für XenApp oder XenDesktop ändern, wird die Änderung nicht sofort für die Sitzungsaufzeichnung wirksam. Workaround: Starten Sie den VDA-Computer neu. [#532393]
- In den folgenden beiden Fällen wird möglicherweise ein Fehler bei der Installation gemeldet. Sie können diese Meldung ignorieren. Um ihre Anzeige ganz zu vermeiden, starten Sie den Computer neu.
 - Die Komponenten der Sitzungsaufzeichnung werden deinstalliert und dann ohne Neustart des Computers neu installiert.
 - Bei der Installation tritt ein Fehler auf und es erfolgt ein Rollback. Sie versuchen dann, die Komponenten der Sitzungsaufzeichnung neu zu installieren, ohne den Computer neu zu starten.

Citrix Lizenzierung 11.12.1

Eine Liste der bekannten und behobenen Probleme in der Citrix Lizenzierung 11.12.1 finden Sie in der zugehörigen Dokumentation unter [Bekannte und behobene Probleme](#).

HDX RealTime Optimization Pack 1.7

Eine Liste der bekannten und behobenen Probleme in HDX RealTime Optimization Pack 1.7 finden Sie in der Dokumentation zu HDX RealTime Optimization Pack 1.7 unter [Bekannte und behobene Probleme](#).

XenApp 7.6 und XenDesktop 7.6

Allgemeine Probleme

Die folgende Notiz gilt für alle Workarounds, bei denen ein Registrierungseintrag geändert werden muss:

Achtung: Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

- Das Tool "CreateAnonymousUsersApp" löscht möglicherweise nicht alle anonymen Benutzerkonten von der lokalen Maschine. Alle anonymen Benutzer sind lokale Benutzer und nicht Active Directory-Benutzer. Da das Tool Kennwörter und Profile löscht, sind die Namen der anonymen Benutzerkonten unbrauchbar, die nicht gelöscht wurden. Löschen Sie unerwünschte Namen anonymer Benutzerkonten lokal vom Server mit "Benutzerkonten verwalten" oder "Computer Manager". [#4999679]
- Zum Konfigurieren eines nicht standardmäßigen HTTP/SOAP-Ports für den universellen Druckserverwebdienst müssen Sie mit PowerShell-Cmdlets die Sitzungsdruckerrichtlinie ändern. Informationen zum Konfigurieren von Gruppenrichtlinien finden Sie im Abschnitt zur Verwendung des Group Policy SDKs in den

— Informationen zum SDK

. So stellen Sie den Richtlinienwert ein:

```
Set-ItemProperty LocalGpo:\Computer\Unfiltered\Settings\ICA\Printing\UniversalPrintServer\UpsHttpPort -name Wert -Value <Portnummer>. [#268593]
```

- Nachdem das Anhalten eines Hyper-V-Hosts aufgehoben wird, aktualisiert Microsoft System Center Virtual Machine Manager (VMM) möglicherweise nicht sofort den Gesamtzustand des Hosts. Dies kann sich auf die Verwendung der Maschinenerstellungsdienste (MCS) auswirken. Wenn ein Hyper-V-Host einen angehaltenen Zustand meldet, wird dieser Host nicht für das Bereitstellen von virtuellen Maschinen verwendet. Wenn alle Hosts angehalten sind, schlägt die Katalogerstellung fehl. Als Workaround können Sie den Stammclusterknoten oder den Hyper-V-Host-Knoten manuell aktualisieren. Auf dem Host ausgeführte Umgebungstests melden auch Hosts mit einem Zustand "angehalten". [#285696]

- Das Brokering gehosteter Anwendungen auf Desktopbetriebssystemmaschinen wird nicht mit dem Remotedesktopprotokoll (RDP) unterstützt. [#377108]
- Dieses Release unterstützt nicht das Bereitstellen einer ISO-Datei, wenn die Clientlaufwerkzuordnung für Sitzungen unter Windows 8 konfiguriert ist. [#333111]
- Wenn der universelle Druckserver von Citrix aufgrund falscher Treiber fehlschlägt, versuchen Sie, diese Treiber unter der Druckertreiberisolation auszuführen. Weitere Informationen zur Konfiguration der Druckertreiberisolation finden Sie unter MSDN: [http://msdn.microsoft.com/en-us/library/windows/hardware/ff560836\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/hardware/ff560836(v=vs.85).aspx). [#381460]
- Die Richtlinieneinstellung für die Enhanced Desktop Experience hat keine Auswirkungen auf vorhandene Benutzer oder Administratorprofile. Löschen Sie als Workaround alle bereits vorhandenen Profile, bevor Sie die Einstellung aktivieren bzw. deaktivieren; löschen Sie auch die Profile, die für die VDA-Installation verwendet werden (nach der VDA-Installation). Wenn das integrierte Administratorkonto für die Installation von Virtual Delivery Agent (VDA) verwendet wurde, können Sie das Profil nicht löschen. In diesem Fall kann der Benutzer das Citrix Enhanced Desktop-Design nach der Anmeldung auswählen. [#363736]
- Wenn die Bildschirmauflösung betroffen ist, da die Auflösung der VDA-Desktopsitzung die Clientbildschirmauflösung übersteigt, ändern Sie die Auflösungseinstellung auf dem VDA-Desktop auf den unterstützten Höchstwert für den angeschlossenen Monitor. Sie können auch die Monitorkabel von der VDA-Grafikkarte abziehen. [#365877]
- Bildschirmschoner und der Energiesparmodus sind in Sitzungen deaktiviert. Bearbeiten Sie die Registrierung und erstellen Sie folgenden DWORD-Wert: HKLM\Software\Citrix\Graphics\SetDisplayRequiredMode = 0
Diese Änderung verhindert nicht, dass der Bildschirmschoner oder der Energiesparmodus auf dem Remotecomputer aktiviert wird. Wenn der Energiesparmodus aktiviert wird, wird die Remotesitzung erst aktualisiert, wenn der Benutzer eine Eingabe (Maus/Tastatur) macht, aber die Bildschirmangaben werden nicht entfernt. [#380550]
- Bei der versuchten Verwendung eines PowerShell SDK New-ProvScheme-Befehls oder eines MCS-Befehls von einem Remotecomputer, bevor die Hostadminadresse eingestellt ist, kann ein Fehler auftreten. Stellen Sie die Adminadresse mit dem Befehl Set-HypAdminConnection ein, bevor Sie den New-ProvScheme-Befehl ausführen. [#336902]
- In diesem Release wird die Verwendung dieses Produkts mit dem Microsoft RemoteFX vGPU-Feature in einem Hyper-V-Host nicht unterstützt. Greifen Sie stattdessen mit RDP, z. B. Hyper-V vGPU, auf die RemoteFX-Funktionalität zu. [#375577]
- Director meldet die Anzahl an GPU-Maschinen, die nicht gestartet werden konnten, auf der Dashboardseite im Bereich für Maschinenfehler. Diese Informationen werden jedoch nicht angezeigt, wenn Sie einen Drilldown ausführen, um Detailinformationen auf der Seite "Filter" anzeigen.

Sie können diese Informationen auch im historischen Trenddiagramm anzeigen. [#043722]

- Wenn mehr GPU-Maschinen eines bestimmten GPU-Typs bereitgestellt oder zugewiesen werden, als der XenServer-Ressourcenpool unterstützen kann, starten die GPU-Maschinen u. U. nicht. Benutzer, die einer Maschine zugewiesen wurden (einschließlich der, die Personal vDisk verwenden), die nicht gestartet werden konnte, können möglicherweise nicht über StoreFront auf ihre virtuellen Desktops zugreifen. [#0434505]

- Wenn Sie eine GPU-Maschine einschalten und sie startet nicht, wird möglicherweise die folgende Fehlermeldung angezeigt:

Exception Failure in PowerOn, PGPU_INSUFFICIENT_CAPACITY_FOR_VGPU

Dies bedeutet, dass unzureichende GPU-Ressourcen vorhanden sind, um eine weitere Maschine, die den GPU verwendet, zu starten. [#434509]

- Wenn sich mehrere Benutzer mit dem RDP- und ICA-Protokoll an einer Desktopbetriebssystemmaschine anmelden, und ein Benutzer die RDP-Sitzung sperrt, können sich Benutzer, die eine Verbindung mit ICA herstellen, nicht an der Sitzung anmelden. Benutzer sollten Benutzer die Verbindung mit RDP-Sitzungen trennen oder sich abmelden, um dieses Problem zu vermeiden. [#392311]
- Wenn Sie eine Anwendung mit einem VDA-Host unter Windows 8 veröffentlichen und die Anwendung Nicht-ASCII-Zeichen im Infobereich anzeigt, werden andere nicht in Verbindung stehende Zeichen nach der Wiederverbindung auch im Infobereich angezeigt. Melden Sie sich von der Sitzung ab und starten Sie die Anwendung neu, um dieses Problem zu vermeiden. [#387963]
- Das Deaktivieren der Sitzungs freigabe über einen Registrierungseintrag wird in diesem Release nicht unterstützt. Daher ist die Sitzungs freigabe immer aktiviert. Wenn der Registrierungsschlüssel auf das Deaktivieren der Sitzungs freigabe eingestellt ist, wird die erste Anwendung gestartet, aber nachfolgende Anwendungen werden nicht gestartet. Für dieses Problem gibt es keinen Workaround. [#383718]
- Wenn ein Benutzer sich mit einer getrennten Sitzung auf Windows Server 2012 wieder verbindet, besteht ein Arbeitsspeicherverlust im Desktop-Fenstermanager. Wenn sich Benutzer häufig mit einer längeren Sitzung verbinden und die Verbindung trennen, beispielsweise verwendet ein Benutzer dieselbe Sitzung im Büro und von zu Hause, können die Speicherressourcen auf dem Server verringert werden; dies führt zu langsamen Reaktionszeiten und einem möglichen Serverausfall. Dies ist ein Drittanbieterproblem im Microsoft Code. Weitere Informationen finden Sie unter <http://support.microsoft.com/KB/2855336> (Veröffentlichung im Juli 2013). [#374261]
- Beim Arbeiten über Windows PowerShell 3.0 funktioniert die Hilfeinfo nicht. Dies ist ein Drittanbieterproblem von Microsoft. Andere Helpthemen sind nicht betroffen. [#408866]
- Wenn Sie ein Upgrade von XenDesktop 5 auf dieses Release (oder auf XenDesktop 7.1 oder XenDesktop 7.0) durchführen, führt eine gehostete Anwendung, die sowohl einer freigegebenen als auch einer privaten Bereitstellungsgruppe zugewiesen ist, zu einem Bereitstellungsgruppenkompatibilitätsfehler während des Upgrades. Sie vermeiden diesen Fehler, indem Sie gehostete Anwendungen entweder einer freigegebenen oder einer privaten Bereitstellungsgruppe (aber nicht beiden) zuweisen. [#419424]
- Wenn die Profilverwaltung aktiviert ist, werden Anmeldeskripts für Sitzungen, in denen Windows Server 2012 R2 oder Windows 8.1 ausgeführt wird, standardmäßig um fünf Minuten verzögert. Wenn die Sitzung verfügbar ist, ist der Schritt mit der Anmeldeskriptdauer in Director nicht verfügbar. Die Verzögerung wird von der Richtlinie zum Konfigurieren der Anmeldeskriptverzögerung gesteuert (Aktiviert = 0). [#407978]
- Desktops können u. U. nicht auf Computern starten, auf denen Windows 8.1 ausgeführt wird und Microsoft Softwareupdateverwaltung installiert ist. Sie vermeiden dieses Problem, indem Sie in Windows 8.1 im Eigenschaftendialogfeld für Taskleiste und Navigation auf der Registerkarte **Navigation** alle Optionen im Bereich der **Startseite** auswählen. [#408439]
- Wenn Sie bei dem Versuch, über das Remotedesktopprotokoll eine Verbindung mit einer vermittelten Sitzung herzustellen, die Fehlermeldung "Sie können auf diese Sitzung nicht zugreifen, da keine Lizenzen vorhanden sind" erhalten, deaktivieren Sie die folgenden Einstellungen in C:\inetpub\wwwroot\Citrix\Store\App_Data\default.ica: [#422212 und #403855]:
 - RDPConnection=false
 - RDP-RedirectDrives=false
 - RDP-RedirectDynamicDrives=false
- Der Wert für die Richtlinie "Schwellenwert für persistenten Cache" ist in Studio fälschlicherweise mit KBit/s beschriftet. Der richtige Wert ist "Bits pro Sekunde" (Bit/s). [#429478]
- Bei der Auswahl eines benutzerdefinierten Datumsbereichs für einen Konfigurationsprotokollierungsbericht wird möglicherweise ein falscher Kalender angezeigt. [#452399]
- Unter Amazon Web Services oder CloudPlatform bereitgestellte Maschinen werden nicht angehalten, wenn die Energieverwaltungseinstellung ihrer Bereitstellungsgruppe für die Trennung der Verbindung auf "Anhalten" festgelegt ist. [#453780]
- Verbindungszeiten (Timer für längste Verbindung, Verbindungstimer und Trennungstimer) funktionieren möglicherweise auf Windows Server 2012-Maschinen mit VDAs für Windows-Serverbetriebssysteme nicht und verursachen unerwartetes Sitzungstimeoutverhalten. [#471698]
- Wenn die Richtlinieneinstellung Generischen universellen Drucker automatisch erstellen aktiviert ist, erstellt der Citrix Druckmanagerdienst (CpSvc.exe) den generischen Citrix Universellen Drucker nicht. Hotfixes für dieses Problem sind in den Knowledge Center-Artikeln CTX141565 und CTX141566 verfügbar.
- Benutzerverbindungen mit VDAs unter Windows Server 2012 R2 können fehlschlagen, wenn die Citrix Richtlinie **Maximal zugelassene Farbtiefe** aktiviert ist. Die Richtlinie **Maximal zugelassene Farbtiefe** gilt nicht für VDAs mit einem Windows Display Driver Model-Treiber für die primäre Anzeige, z. B. VDAs, auf denen Windows Server 2012 R2 ausgeführt wird.

Installationsprobleme

- Fehler 1904: Modul C:\Programme (x86)\Citrix\System32\vpdm.dll konnte nicht registriert werden. HRESULT -2147010895. Wenden Sie sich an den Support. Tritt auf, wenn der Virtual Delivery Agent (VDA) mit dem Citrix MetalInstaller installiert oder aktualisiert wird. Die Installation der Drittanbieterkomponente Microsoft Visual C++ 2005 Runtime schlägt fehl, ein Rollback findet statt und dann wird der MetalInstaller fälschlicherweise davon unterrichtet, dass die Installation erfolgreich abgeschlossen wurde. Daher fährt MetalInstaller mit der Installation der restlichen VDA-Komponenten bis zur Initialisierung der Komponenten nach der Installation fort. An dieser Stelle kann ohne Microsoft Visual C++ 2005 Runtime nicht fortgefahren werden. Um dieses Problem zu lösen, führen Sie die VDA-Installation mit dem MetalInstaller erneut aus, sodass auch die Installation von Microsoft Visual C++ 2005 Runtime erneut ausgeführt wird. Als Alternative lesen Sie <http://support.microsoft.com/?kbid=947821> und folgen Sie der dokumentierten Lösung von Microsoft. [#489633]
- Beim Installieren von Virtual Delivery Agent für Serverbetriebssysteme tritt möglicherweise Fehler 1935 auf. Grund: Abwärtskompatibilitätsfehler in Microsoft Visual C++ 2005 Redistributable. Weitere Informationen finden Sie auf der Microsoft Website oder versuchen Sie das Problem mit Windows Update zu beheben. [354833]
- Nach einer erfolgreichen Installation von VDA für Windows-Serverbetriebssysteme (aber vor dem Neustart der Maschine) enthält das Windows-Ereignisprotokoll möglicherweise mehrere Fehlermeldungen (z. B. TermService 1035 oder 1036, die angeben, dass der Terminalserver-Listenerstack ausgefallen war oder das Erstellen der Sitzung fehlgeschlagen ist. Wenn keine anderen Installationsfehler enthalten sind, können Sie diese Meldungen im Ereignisprotokoll ignorieren. [#374134]
- Wenn Sie ein Upgrade von XenDesktop 5.x durchführen, stellen Sie sicher, dass Desktop Studio von XenDesktop 5.x geschlossen ist, bevor Sie das Upgrade durchführen. Sonst wird Studio beim Upgrade ggf. unerwartet geschlossen. [#389374]
- Beim Upgrade wird ein Administrator, der in XenDesktop 5.6 deaktiviert wurde, möglicherweise ohne Rolle oder Geltungsbereich in die neuere Version verschoben. Überprüfen Sie die angezeigten Administratoren in Studio und bearbeiten Sie sie ggf. [#394765]
- Wenn Sie die Profilverwaltung aktivieren und Benutzer feststellen, dass die Windows 8-Standardanwendungen (z. B. Wetter, Nachrichten und Bing) bei der ersten Anmeldung aber nicht bei nachfolgenden Anmeldungen starten, müssen Sie dieses Feature neu konfigurieren. Dieses Problem wurde in Umgebungen beobachtet, in denen das Benutzerprofil nicht permanent ist und in denen der Ordner AppData\Local nicht ausgeschlossen wurde (Standardeinstellung). Fügen Sie als Workaround die Ordner AppData-Ordner\Local\Packages und AppData\Local\Microsoft\Application Shortcuts als Ausschlüsse hinzu. [#394802]
- Bei der Installation von Virtual Desktop Agent oder bei einem Upgrade auf Windows 7 werden Sie möglicherweise in einem Dialogfeld aufgefordert, den Computer neu starten, um die Änderungen zu übernehmen. Klicken Sie auf "Später neu starten", um das Upgrade fortzusetzen; wählen Sie nicht "Jetzt neu starten". [#396553]
- Das Installieren von VDAs über eine Active Directory-Gruppenrichtlinie mit einzelnen MSIs wird nicht empfohlen und kann fehlschlagen. Citrix empfiehlt die Verwendung der Startskripts, die auf dem Produktinstallationsmedium bereitgestellt werden (siehe [Installieren oder Entfernen von Virtual Delivery Agents in Active Directory mit Skripten](#)). [#383432, #372136]
- Der Abschluss der Optimierungsphase während der VDA-Installation kann lange dauern. In einigen Tests hat dies ungefähr 30 Minuten gedauert und kann länger dauern. Dies tritt auf, wenn der VDA auf einem Image mit dem Windows-Betriebssystem installiert wird, oder wenn im Installationsassistenten auf der Seite Features die Option Leistung optimieren ausgewählt wurde. Dies führt dazu, dass der Microsoft Native Image Generator (Ngen) ausgeführt wird. Wenn dieses Problem auftritt, warten Sie auf den Abschluss der Installation. Citrix empfiehlt, dass Sie Ngen auf dem VDA-Basisimage vor der Bereitstellung virtueller Desktops ausführen, um Verzögerungen zu vermeiden, die bei der Hintergrundausführung des Programms auf bereitgestellten virtuellen Desktops auftreten. [#381437]
- Wenn während eines Upgrades eine Fehlermeldung zu einem nichtgefundenen PICAIsPorticaV2-Einstiegspunkt angezeigt wird, können Sie sie bedenkenlos ignorieren. Schließen Sie das Upgrade ab und starten Sie den Computer neu, wenn Sie dazu aufgefordert werden. [#423947]
- Der VDA für Windows-Desktopbetriebssysteme wird möglicherweise unter Evaluierungsversionen von Windows 8 nicht installiert. Auf dem Bildschirm mit den Installationsoptionen wird die Meldung "Kann auf diesem Betriebssystem nicht installiert werden" angezeigt. Dieses Problem tritt auf, weil das Installationsprogramm die Windows 8-Evaluierungsversion fälschlicherweise als nicht

unterstütztes Betriebssystem identifiziert. Ein Hotfix für dieses Problem ist im Knowledge Center-Artikel CTX139660 verfügbar. Mit dem Hotfix installieren Sie einen Patch für das Installationsprogramm, damit Sie es auf Windows-Evaluierungsversionen ausführen können.

- Während der Produktinstallation können bei von Provisioning Services erstellten Maschinen Fehler auftreten, wenn .NET Framework 3.5 nicht vor der Installation vorhanden war. Zum Umgehen dieses Problems stellen Sie sicher, dass die .NET Framework-Versionen 2.0, 3.0, 3.5, 4.0, 4.5 und 4.5.1 installiert sind. [#442639, #447851]

Server- und Delivery Controller-Probleme

- XenDesktop 7.6 enthält eine neue Version des Volumeworkerpakets für CloudPlatform. Citrix empfiehlt, die Volumeworkervorlage in der Bereitstellung mit dieser neuen Version zu aktualisieren. Beachten Sie beim Aktualisieren, dass direkte Upgrades für dieses Paket nicht unterstützt werden. Daher müssen Sie zuerst das vorherige Paket vollständig deinstallieren oder eine neue Volumeworkervorlage erstellen. [493211]

Probleme bei Remote-PC-Zugriff

- Nach dem Upgrade von XenDesktop 5.6 FP1 wird der Name des Remote-PC-Zugriff-Dienstadministrators möglicherweise nicht richtig angezeigt. Dies wirkt sich nicht auf den Betrieb aus. [#437948]
- Wenn eine Büromaschine in den Ruhezustand versetzt wurde, kann anschließend der Start einer Remote-PC-Zugriff-Sitzung fehlschlagen. Als Lösungsansatz bei Desktopmaschinen, auf denen eine Fehlermeldung angezeigt wird, versuchen Sie noch einmal, die Sitzung zu starten. Starten Sie Laptops, auf denen dauerhaft ein grauer Neuverbindungsbildschirm angezeigt wird, erneut (entweder remote über die Administratorkonsole durch Erzwingen des Herunterfahrens und eines Neustarts oder lokal mit dem Netzschalter). Dies kann zu Datenverlust führen. [#441154]
- Bei Einsatz des Aktivierungsproxys anstelle von Intel Active Management Technology (AMT) oder Wake-On-LAN-Paketen wird eine Maschine möglicherweise nicht reaktiviert. Dies ist ein von Microsoft System Center Configuration Manager verursachtes Problem. [#441412]
- Das Starten von VDA von einem Benutzergerät mit einem Smartcardleser kann fehlschlagen, wenn der Benutzer den VDA vorher von diesem Gerät gestartet und in Desktop Viewer "Trennen" ausgewählt hat. In diesem Fall wird dem Benutzer der Bildschirm "Smartcardanmeldeinformationen" oder die Meldung "Smartcard wird gelesen" angezeigt. Wählen Sie als Workaround eine der folgenden Optionen:
 - Nehmen Sie die Smartcard aus dem Leser heraus und stecken sie wieder ein.
 - Klicken Sie auf "Abbrechen". Drücken Sie dann im VDA Strg+Alt+Entf. [#322301]

Probleme mit Studio

- Der Versuch, StoreFront und Studio zu starten, führt dazu, dass die Citrix Konsole unerwartet beendet wird, nachdem die XenApp- und XenDesktop-Software auf einer einzelnen Maschine mit Windows 2008 R2 SP1 installiert wurde. Dies tritt auf, wenn StoreFront und Studio am Ende der Installation über das Kontextmenü gestartet werden oder wenn Sie zuerst StoreFront und dann Studio in der Konsole öffnen. Sie umgehen das Problem, indem Sie die Aktualisierung der .NET-eigenen Images mit dem Native Image Generator (Ngen) erzwingen. Öffnen Sie dazu eine Eingabeaufforderung und navigieren Sie dann zu c:\windows\microsoft.net\framework64\\. Führen Sie ngen update /force aus. Dieser Vorgang kann einige Minuten dauern. Hinweis: Die framework64-Versionsnummer kann etwas abweichen, aber sie sollte immer mit 2.0 beginnen. [#490819]
 - Director meldet Lizenzierungsfehler nicht richtig und Studio kann nicht mit dem Lizenzserver kommunizieren. Dieses Problem kann auftreten, wenn sich die Lizenzserveradresse für eine Bereitstellungsstelle ändert, während mehrere Instanzen von Studio geöffnet sind und dieselbe XenDesktop- oder XenApp-Site verwalten. Sie umgehen dieses Problem, indem Sie die Lizenzierungsdaten in der Site aktualisieren. Schließen Sie dazu alle offenen Instanzen von Studio, öffnen Sie eine Instanz von Studio und navigieren Sie zum Lizenzierungsknoten. [#492971]
 - Wenn Sie mit Maschinenerstellungsdiensten (MCS) Maschinen auf VMware vSphere bereitstellen, spiegelt die Kombination der CPU-Sockets und Kernspeicher auf den bereitgestellten Maschinen die Kombination der CPU-Sockets und Kernspeicher auf dem Basisimage wider, das für die Maschinenkatalogerstellung verwendet wird. Wenn während der MCS-Katalogerstellung die Anzahl der ausgewählten virtuellen CPUs höher ist als die maximal mögliche auf dem Host, werden die bereitgestellten Maschinen mit der maximal möglichen Anzahl Sockets und Kernspeicher für diesen Host erstellt, ohne dass der Benutzer darauf hingewiesen wird. [#331269]
 - Wenn Sie System Center Virtual Machine Manager in einer reinen IPv6-Umgebung verwenden und Maschinenkataloge mit den Maschinenerstellungsdiensten (MCS) erstellen, haben alle VMs IPv4 und IPv6, selbst wenn die Master-VM ohne IPv4 im TCP/IP-Stapel konfiguriert ist. Dies ist ein Problem mit Microsoft und es gibt kein Workaround. [#371712]
 - Wenn Sie mit VMware vSphere in einer IPv4- und IPv6-Umgebung mit VMware ESX Hypervisoren verwenden, die mit VMXNET3-Netzwerkadaptern konfiguriert sind, haben alle VMs IPv4 und IPv6, selbst wenn die Master-VM im TCP/IP-Stapel ohne IPv4 konfiguriert ist. Dies ist ein Problem mit VMware und es gibt kein Workaround. [#371712]
 - Lange Maschinenkatalognamen und lange Speicherpfadnamen führen möglicherweise zu einem Datenträgeranbindungsfehler im VMM-Auftragsfenster. Microsoft hat eine Beschränkung von maximal 255 Zeichen für die Länge der Dateipfad für VM-Ressourcen festgelegt. Dieses Problem tritt bei lokalen Speichern auf einem eigenständigen Hyper-V-Host auf, wegen der langen Dateipfade zum Speichern von VMs, ist jedoch nicht auf eigenständige Hyper-V-Hosts beschränkt. [#359673]
- Lösungsansatz:
- Erstellen Sie VMM-MCS-Kataloge mit kurzen Namen, insbesondere dann, wenn über einen langen Pfad auf den Speicher zugegriffen wird.
 - Kürzen Sie den Pfad zum Speicher, in dem die VMs gespeichert sind.
- Um zu einer zuvor genutzten Datenbank zu wechseln, müssen Sie das SDK verwenden; dies kann nicht mit Studio erfolgen. [#355993]
- Wechseln zur Datenbank für die Konfigurationsprotokollierung
- Set-LogDbConnection -DataStore 'Logging' -DBConnection \$null
 - Set-LogDbConnection -DataStore 'Logging' -DBConnection '<neue Datenbankverbindungszeichenfolge>'
- Wechseln zur sekundären Überwachungsdatenbank
- Set-MonitorDbConnection -DataStore 'Monitor' -DBConnection \$null
 - Set-MonitorDbConnection -DataStore 'Monitor' -DBConnection '<neue Datenbankverbindungszeichenfolge>'
- Beispielsweise kann die neue Datenbankverbindungszeichenfolge sein:
- ```
'Server=dbserver;Initial Catalog = dbname; Integrated Security = True'
```
- Wenn Sie Maschinen auf der Hypervisorplattform mit Maschinenerstellungsdiensten bereitstellen, werden CPUs aber keine Kernspeicher hinzugefügt. Wenn Sie den CPU-Wert bei der Katalogerstellung ändern, werden für die Produkte, die pro CPU lizenziert sind, möglicherweise weitere Lizenzen benötigt. Beispiel: Das Masterimage hat eine CPU und vier Kernspeicher und Sie ändern den CPU-Wert während der Katalogerstellung. Wenn Sie bei der MCS-Katalogerstellung mehr CPUs wählen als maximal auf dem Host zugelassen, ist die Höchstanzahl der für diesen Host unterstützten bereitgestellten Maschinen erreicht und Sie werden nicht über die Differenz benachrichtigt. Beispielsweise kann ein Windows-Desktopbetriebssystem nur zwei physische CPUs nutzen; somit werden nur zwei angezeigt, selbst wenn mehr zugewiesen sind.
- Stellen Sie als Workaround sicher, dass die Masterimage-VM die gleiche virtuelle CPU-Konfiguration hat, die Sie für den Katalog bereitstellen möchten. [#331274]
- Wenn der Lync 2013-Client von einer Desktopbetriebssystemmaschine oder einer Serverbetriebssystemmaschine bereitgestellt wird, funktioniert das Videochat-Feature nicht. Informationen zur Verwendung von Lync 2013 finden Sie im Artikel zur Unterstützung für das Microsoft Lync 2013 VDI Plug-In durch XenDesktop 7.x, XenApp 6.x und Citrix Receiver 4.x (<http://support.citrix.com/article/CTX138408>). [#371818]
  - Beim Starten einer Seamlessanwendung auf einem Windows-Server hat das Fenster ggf. nicht das Aero-Design, selbst wenn Enhanced Desktop Experience aktiviert ist.
    - Benutzer, die nur Seamlessanwendungen starten, erhalten nie das Aero-Design.

- Benutzer mit Seamlessanwendungen und Desktops erhalten das Aero-Design nach dem Herstellen der ersten Desktopsitzung; Seamlessanwendungen haben danach das Aero-Design. Stellen Sie als Workaround das Citrix Enhanced Desktop-Design im Standardbenutzerprofil ein, sodass alle Benutzer auf diesem VDA Enhanced Desktop Experience für die Seamlessanwendungen erhalten. Das Design ist Teil der VDA-Installation und muss für alle VDAs eingestellt werden. [#348812]
- Von Studio an eine Windows 8-Maschine gesendete Nachrichten werden auf dem Windows 8-Desktop statt des Windows-Standarddisplays (früher Metro genannt) angezeigt. Der Benutzer muss zum Desktopmodus wechseln, um die Nachricht anzuzeigen. Dies ist ein Drittanbieterproblem und es gibt kein Workaround. [#387356]
- Wenn eine Site mehr als ein Hostinginfrastruktur-Objekt mit dem gleichen Namen enthält, wird Studio möglicherweise nicht richtig angezeigt. Beim Erstellen von Hostinginfrastruktur Objekte (z. B. Netzwerke und Speicher), sollten Sie einen eindeutigen Namen angeben. [#384959]
- Wenn Sie einen Administrator mit einem neuen Geltungsbereich mit der delegierten Administration erstellt haben, müssen Sie das Studio-Display aktualisieren. Sonst erhalten Sie u. U. einen Berechtigungsfehler, wenn Sie sich als der neue Administrator anmelden und versuchen, eine neue Verbindung oder Ressource zu erstellen. [#386634]
- Auf Desktopbetriebssystemmaschinen gehostete Anwendungen mit zufälliger Zuweisung werden möglicherweise nicht geöffnet, nachdem das Dialogfeld "Laden" ausgeblendet wird. Das Problem tritt auf, wenn der Start der Anwendung länger dauert als das standardmäßige Timeout von einer Minute und die Sitzung automatisch beendet wird. [#389025]  
Ändern Sie als Administrator das Basisimage und stellen es mit einem geänderten Timeoutwert wie folgt neu bereit:

Suchen Sie die Registrierungsschlüssel:HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\wfshell\TWI (erstellen Sie den Schlüssel, wenn er nicht verfügbar ist.)

Name: ApplicationLaunchWaitTimeoutMS

Typ: REG\_DWORD

Wert: Zusätzlich benötigter Timeout in Millisekunden

Hinweis: Wenn Sie einen Wert angeben, der kleiner als 10.000 ist, wird 10.000 verwendet, da 10 Sekunden der Mindestwert für die Überschreibung ist.

- Zum Durchführen der Discovery und zum Einrichten von Anwendungen in Bereitstellungsgruppen sollten Sie sicherstellen, dass einige Maschinen registriert sind und eingeschaltet werden können. Für die Discovery einer App-V-Anwendung müssen Sie die App-V-Veröffentlichung in Studio konfigurieren. [#393676]
- Wenn ein Benutzer die erste App-V-Anwendung startet, schlägt ein zu schnelles Starten einer zweiten App-V-Anwendung u. U. fehl. Wenn dieses Problem auftritt, sollte der Benutzer ein paar Minuten warten, damit die anfängliche Synchronisierung abgeschlossen werden kann, und dann die zweite Anwendung erneut starten. Weitere Informationen zu diesem Problem finden Sie unter [CTX138056](#). [#397521]
- Wenn das Antivirenprogramm BitDefender auf einem VDA installiert ist, können Sie möglicherweise keine Maschinenkataloge erstellen. Für dieses Problem gibt es keinen Workaround. [#392705]
- Nach dem Upgrade auf XenDesktop 7.5 können Sie keine Verbindung mit CloudPlatform oder Amazon Web Services (AWS) erstellen, wenn für die Site eine Verbindung mit einem lokalen Hypervisor konfiguriert ist. Sie müssen eine neue 7.5-Site erstellen. Bei einer Hybridbereitstellung, die Cloudsites und nicht cloudbasierte Sites umfasst, müssen Sie separate Sites erstellen, die dieselbe StoreFront-Site verwenden. [#454114]
- Das Erstellen eines Maschinenkatalogs schlägt bei Verwendung eines ODX-aktivierten Storage Area Network in einer Umgebung fehl, in der VMM 2012 SP1 und Hyper-V 2012 ausgeführt werden. Folgen Sie den Anweisungen in <http://support.citrix.com/article/CTX139333>, um dieses Problem zu beheben. [#424040]
- Die Citrix Richtlinieneinstellung **ICA-Listener - Verbindungstimeout** gilt nur für Virtual Delivery Agent 5.0, 5.5 und 5.6 Feature Pack 1, selbst wenn im Studio-Fenster zur Bearbeitung von Richtlinien angegeben wird, dass sie auch für 7.0 Desktopbetriebssystem, 7.1 Desktopbetriebssystem, 7.5 Desktopbetriebssystem und 7.6 Desktopbetriebssystem gilt.
- In Studio wird in der Beschreibung der Citrix Richtlinieneinstellung **Starten nicht-veröffentlichter Programme bei Clientverbindung** fälschlicherweise behauptet, dass diese Einstellung festlegt, ob Startanwendungen oder veröffentlichte Anwendungen über ICA oder RDP auf dem Server gestartet werden. Mit dieser Einstellung wird nur festgelegt, ob Startanwendungen oder veröffentlichte Anwendungen über RDP auf dem Server gestartet werden.
- Citrix Richtlinien werden möglicherweise nicht richtig angewendet, wenn sie nach Tags gefiltert werden. Um dies zu verhindern, wenden Sie den unter [CTX142439](#) verfügbaren Hotfix an. [#529165]

## Probleme mit Director

- Die benutzerbasierten Anwendungsnutzungsdaten in den Nutzungsberichten für gehostete Anwendungen sind nicht korrekt für Zeiträume, die über die letzten sieben Tage hinausgehen. Dieses Problem tritt auf, weil die Anwendungsdaten nach sieben Tagen gelöscht werden. Um zu verhindern, dass falsche benutzerbasierte Anwendungsnutzungsdaten verwendet werden, verwenden Sie nur die Daten der letzten sieben Tage. [504642]
- Director zeigt einen unerwarteten Fehler an. Überprüfen Sie die Netzwerkverbindung oder suchen Sie im Serverereignisprotokoll nach weiteren Informationen. Eine Popupmeldung zu einem Fehler auf den Seiten "Trends" und "Filter" wird angezeigt, weil der Name der XenDesktop-Datenbank Leerzeichen enthält. Die API-Aufrufe von Director unterstützen keine Verbindung zu einer Datenbank deren Name Leerzeichen enthält. Halten Sie sich beim Umbenennen einer Datenbank an die Namenskonventionen für Datenbanken. [494339]
- Nach einem Upgrade sind die Administratoreinstellungen für UI.GlobalSearchResults nicht mehr erhalten. Sie umgehen das Problem, indem Sie nach dem Upgrade die Einstellung UI.GlobalSearchResults in der Datei web.config manuell bearbeiten. [484066]
- Die Aktualisierung des Hotfixbestands kann mehrere Stunden dauern, wenn eine große Zahl Hotfixes gleichzeitig auf Umgebungen mit mehr als 20.000 Desktops angewendet wird. [#489604]
- Beim Löschen und erneuten Erstellen einer Bereitstellungsgruppe im selben Sitzungsmaschinenkatalog bleiben historische Informationen erhalten. Die historischen Daten werden dann fälschlicherweise im Trenddiagramm und der zugehörigen Tabelle angezeigt. Sie erstellen Bereitstellungsgruppen, ohne dass historische Trenddaten erhalten bleiben, indem Sie erst den Maschinenkatalog löschen und einen neuen Katalog erstellen, bevor Sie die Bereitstellungsgruppe erneut erstellen. [#480010]
- Beim Navigieren von der Seite "Filter" zur Seite "Benutzergerät" werden Benutzernamen möglicherweise nicht angezeigt. Wenn Delivery Controller aufgrund von Active Directory-Beschränkungen keinen Zugriff auf eine Domäne mit Benutzerkonten haben, werden die vollständigen Benutzernamen möglicherweise nicht auf der Seite "Benutzergerät" angezeigt. Um dieses Problem zu umgehen, stellen Sie sicher, dass die Domäne, in der XenDesktop installiert ist, eine vertrauenswürdige Beziehung zu der Domäne hat, zu der der Endbenutzer gehört. [#479517]
- Wenn nicht allen Sites in einer vorhandenen Sitegruppe delegierte Administratoren zugewiesen sind, schlägt die Suche nach Maschinen und Endpunktgeräten innerhalb der Sitegruppe fehl. Damit ein Administrator eine Suche über alle Sites in einer Sitegruppe durchführen kann, fügen Sie den Administrator der Gruppe "Delegierte Administratoren" hinzu, die Berechtigungen für die Sites hat. [#491740]
- Wenn Sie virtuelle Windows XP-Desktops überwachen, auf denen WinRM 2.0 für Benutzer mit VDAs vor 7 ausgeführt werden, müssen Sie die WinRM-Port-Abhörreihenfolge ändern. Weitere Informationen zum Ändern der Einstellung auf 5985.80 finden Sie unter [Erweiterte Konfiguration](#). [#273609]
- Sie können eine vorhandene Sitegruppe nicht mit dem Director-Konfigurationstool bearbeiten. Um dieses Problem zu umgehen, öffnen Sie die XML-Konfigurationsdatei im Konfigurationsordner von Director und bearbeiten Sie die Sitegruppeneinstellung. [#491681]
- Die Konfiguration von Benutzern mit Anmeldeskripts über Active Directory-Benutzer und -Computer schlägt fehl und Daten werden nicht im Bereich "Anmeldedauer" angezeigt. Verwenden Sie zum Konfigurieren von Benutzern mit Anmeldeskripts eine Gruppenrichtlinie. [#393259]
- Die Daten zur Anmeldedauer von einer Erstanmeldung an Personal vDisk-VDA werden in Director u. U. nicht gesammelt oder angezeigt. Für weitere Anmeldungen werden die Daten normal angezeigt. [#383941]
- Wenn Sie eine Konsolensitzung von einem Windows Server 2012-Desktop aus ausführen, kann das Navigieren zur Benutzerdetailseite in Director zu einem Fehler führen, sodass keine Daten angezeigt werden.

Um dieses Problem zu umgehen, registrieren Sie den VDA, melden Sie sich bei der aktuellen Sitzung ab und melden Sie sich dann erneut beim VDA an. [#388513]

- Im Bereich "Infrastruktur" der Seite "Dashboard" von Director wird für Citrix CloudPlatform-basierte Hostverbindungen, Amazon Web Services, Hyper-V und Microsoft System Center Configuration Manager "Nicht verfügbar" ohne Statusinformationen angezeigt. [#449806, #446397]
- Die Anmeldedauer wird im Dashboard von Director nicht aktualisiert, wenn Benutzer eine gehostete Sitzung starten. [#386860]

## Probleme mit HDX

**Hinweis:** Aktuelle Updates für HDX Flash-Kompatibilität finden Sie unter [CTX136588](#).

- Das Konfigurieren von HDX MediaStream-Umleitung auf einer neuen Maschine schlägt fehl. Um dieses Problem zu umgehen, starten Sie die Endpunktmaschine nach der Installation von Receiver neu. [#494741]
  - Sitzungen schlagen fehl, wenn Sie ein Zertifikat mit dem Algorithmus SHA512 für eine SSL/TLS-Verbindung verwenden. Dies tritt auf, wenn Sie mit Receiver für HTML5 Sitzungen mit Serverbetriebssystemmaschinen starten, die unter Windows Server 2012 R2 ausgeführt werden. Um dieses Problem zu vermeiden, verwenden Sie bei diesem Konfigurationstyp nicht den Algorithmus SHA512 für ein SSL/TLS-Zertifikat. [#487284]
  - Bei GPU-Passthrough und NVIDIA Kepler-Karten schlägt der erste Verbindungsversuch möglicherweise fehl, wenn Sie ein HDX 3D Pro-Benutzergerät mit drei oder vier Bildschirmen verwenden. Versuchen Sie in diesem Fall erneut, eine Verbindung herzustellen. [#422049]
  - Wenn ein Benutzer eine Verbindung herstellt, die Bildschirmauflösung oder die Größe des Sitzungsbildschirms ändert, flackert der Bildschirm möglicherweise oder er wird schwarz. Das Problem kann auftreten, wenn eine größere Auflösung ausgewählt ist und der Videotreiber beim Start nicht genügend Speicher zur Unterstützung der Auflösung zuweisen kann. Sie zeigen die aktuelle Auflösungseinstellung im Sitzungsfenster an, indem Sie "Erweiterte Einstellungen" im Dialogfeld "Bildschirmauflösung" auswählen. Erhöhen Sie die Speichermenge für den VDA, damit der Treiber genügend Speicher für höhere Auflösungen zuweisen kann. Verwenden Sie Folgendes als Richtlinien: Windows 7: Summe aller Bildschirme (Höhe x Breite x 4 Byte pro Pixel x 2 Hintergrundpuffer), Windows 8: Summe aller Bildschirme (Höhe x Breite x 4 Byte pro Pixel x 3 Hintergrundpuffer). [#494671]
  - Bei aktivierter USB-Umleitungsrichtlinie funktionieren USB-Speicher und Audiogeräte nicht, wenn Benutzer ihre Geräte umleiten. Dieses Problem tritt nur bei Serverbetriebssystemmaschinen mit Windows 2012 und höher auf. Schließen Sie Audio- und Speichergeräte mit der Richtlinie "Regeln für die Client-USB-Geräteumleitung" aus der Liste der Geräte aus. Weitere Informationen zu USB-Umleitungsregeln und -konfigurationen finden Sie unter <http://support.citrix.com/article/CTX137939>. [#479578]
  - Nachdem der Administrator die Richtlinieneinstellung für "Sitzungszuverlässigkeit - Verbindungen" von "Deaktiviert" in "Aktiviert" geändert hat, zeigt Receiver bei dem Versuch des Benutzers, eine Sitzung von einer Serverbetriebssystemmaschine zu starten, auf dem Benutzergerät die Fehlermeldung "The connection to failed with status (1030)" an. Um dieses Problem zu lösen, muss der Administrator den VDA neu starten, damit die Richtlinie wirksam wird. [#486073]
  - Bei der Verwendung von Remote PC zum Ausführen einer Lync 2013-Remotesitzung werden an Clientgeräte angeschlossene Webkameras in der Benutzersitzung nicht in Lync 2013 aufgeführt. Um dieses Problem zu umgehen, kann der Benutzer die Webkameras auf den Remotemaschinen mit Device Manager deaktivieren. [#482807]
  - Obwohl Webcams möglicherweise die H.264-Komprimierung unterstützen, wird die Hardwarekomprimierung in diesem Release nicht unterstützt. Sie müssen daher die Softwarekomprimierung für diese Webcams verwenden. Fügen Sie hierzu auf Benutzergeräten einen Registrierungseintrag unter HKEY\_CURRENT\_USER\Software\Citrix\HdxRealTime hinzu; fügen Sie den DWORD-Registrierungsnamen DeepCompress\_ForceSWEncode hinzu. Bei der Einstellung "1" wird die Softwarekomprimierung verwendet. Standardmäßig ist diese Einstellung deaktiviert und die Hardwarekomprimierung wird verwendet. [#357356]
  - Das HDX RealTime-Webcamvideo ist verzögert, wenn die Videoauflösung höher als 720 p (1280 x 720) ist. [#350187]
  - Bei der ständigen Verwendung der HDX Flash-Umleitung reagiert die Sitzung u. U. nicht mehr. [#350085, 361926]
  - HDX-Webcam unterstützt die meisten unformatierten Webcamformate, die eine Webcam unterstützt; in seltenen Fällen, wenn die Webcam ein nicht unterstütztes Format hat, funktioniert sie möglicherweise nicht wie die Citrix HDX-Webcam. [#338318]
  - Beim Verwenden von Citrix HDX-Webcam mit gewissen Webcammodellen werden manchmal mehrere Duplikatbilder angezeigt. [#367322]
  - HDX RealTime-Webcams werden für die folgenden Anwendungen nicht unterstützt:
    - Citrix GoToMeeting, wenn es auf Serverbetriebssystemmaschinen mit Windows 2012-Betriebssystemem gehostet wird. [#346430]
    - GoToMeeting (auf jeder Plattform), wenn die Webcam nach Beginn der Besprechung angeschlossen wird. [#346140]
    - Microsoft Lync 2013 und Adobe Connect mit VDAs unter Windows 8-, Windows 8.1- und Windows 2012-Betriebssystemen. [#340784, 348506, 459732]
    - Microsoft Office Communications Server-Videoanrufe (OCS), wenn die Webcam angeschlossen wird, wenn der Anruf bereits ausgeführt wird. [#370236]
    - Microsoft Silverlight. Dies ist ein zeitweiliges Problem. Um das Problem zu umgehen, aktivieren Sie auf dem Benutzergerät den Legacy-Codec, indem Sie einen DWORD-Registrierungsschlüsselnamen EnableDeepcompress\_Client unter HKEY\_CURRENT\_USER\Software\Citrix\HdxRealTime hinzufügen und auf 0 festlegen. [#379779]
    - 64-Bit-Videokonferenzanwendungen. Videokomprimierung für 64-Bit-Anwendungen wird nicht unterstützt. [#366515]
  - Bei der Verwendung von HDX 3D Pro mit dem XenDesktop 5.6 Feature Pack 1 VDA auf virtuellen Windows XP-Desktops ist das Kontrollkästchen "Fine Drawing (2D)" bei der ersten Verbindung ausgewählt und manchmal abgeblendet. Ursache sind Verzögerungen bei Registrierungsupdates, sodass während der Initialisierung in der Benutzeroberfläche des Konfigurationstools falsche Standardwerte angezeigt werden. [#353031]
- Als Workaround können Sie die Sitzung trennen und neu verbinden.

Sollte das Problem weiterhin auftreten, löschen Sie vorherige Sitzungsinformationen und löschen Sie die Einstellungen im folgenden Registrierungsschlüssel:

HKEY\_CURRENT\_USERS\Software\Citrix\HDX3D\BitmapRemotingConfig

- Unter Windows XP wird Citrix HDX-Webcam u. U. nicht erkannt. Installieren Sie Microsoft Visual C++ 2005 Service Pack 1 Redistributable Package über <http://www.microsoft.com/de-de/download/details.aspx?id=14431> und versuchen Sie es erneut, um dieses Problem zu beheben. [#382733]
  - Beim Anzeigen des Knotens "Grafikkarten" in der Geräte-Manager-Konsole wird die Standard-VGA-Grafikkarte in der Liste mit einem gelben Hinweissymbol angezeigt. Sie können diese Warnung ignorieren, da sie sich nicht auf die Funktionalität auswirkt. Diese Warnung wird angezeigt, da ein Legacymodell des XPDM-Anzeigetreibers (Standard-VGA-Grafikkarte) nicht geladen werden kann, wenn ein neuer Anzeigetreiber vom Typ WDDM (Citrix Anzeigetreiber) installiert ist. [#339390]
  - Benutzer stellen u. U. Probleme bei der Wiedergabe von Mediendateien auf Windows 8-Benutzergeräten fest. Dies liegt daran, dass das Standardprogramm für clientseitige Inhaltsabrufprotokolle, mit denen Mediendateien auf Benutzergeräte gestreamt werden, von diesem Produkt nicht registriert wird. Um dieses Problem für die Protokolle Microsoft Media Streaming (MMS) und Real Time Streaming (RTS) zu umgehen, ändern Sie das Standardprogramm für die Wiedergabe von Mediendateien von Windows Media Player zu Citrix CSF Handler. Für HTTP (Hypertext Transfer Protocol) ist kein Workaround möglich. [#328805]
  - Die TWAIN-Umleitung schlägt für gehostete, freigegebene Desktops und Anwendungen fehl. Dies ist ein Drittanbieterproblem bezüglich TWAIN-Anwendungen, die benötigen, dass bestimmte TWAIN-Binärdateien in bestimmten Pfaden sein müssen. [#300854, 340999]
- Kopieren Sie als Workaround auf der Windows Server 2012-Maschine mit dem VDA diese Dateien an die folgenden Speicherorte:
- Kopieren Sie die Datei "twain\_32.dll" in das Verzeichnis "\WINDOWS" des Benutzerprofils (z. B. Datei twain\_32.dll in den Ordner "%USERPROFILE%\Windows").
  - Kopieren Sie die Datei "twain\_32.dll.mui" in das Verzeichnis "\WINDOWS\en-US" des Benutzerprofils (z. B. Datei twain\_32.dll.mui in den Ordner "%USERPROFILE%\Windows\en-US").
- Die 64-Bit-Version von Windows Media Player oder QuickTime Player kann einige Videodateien nicht mit der serverseitigen Wiedergabe wiedergeben, wenn die HDX MediaStream Windows Media-Umleitung deaktiviert ist. Verwenden Sie als Workaround die 32-Bit-Version von Windows Media Player. [#384759]
- Im virtuellen Desktop ausgewählte universelle Druckserver-Drucker werden im Fenster "Geräte und Drucker" in der Windows-Systemsteuerung nicht angezeigt. In den Anwendungen stehen diese Drucker den Benutzern jedoch zum Drucken zur Verfügung. Dieses Problem tritt nur auf Windows Server 2012-, Windows 10 und Windows 8-Plattformen auf. [#335153]
  - Bei GPU-Passthrough und NVIDIA Kepler-Karten schlägt der erste Verbindungsversuch möglicherweise fehl, wenn Sie ein HDX 3D Pro-Benutzergerät mit drei oder vier Bildschirmen verwenden. Der zweite Verbindungsversuch sollte erfolgreich sein. [#422049]
  - Der Windows-Computer eines Benutzers reagiert nicht mehr, wenn eine GoToMeeting-Sitzung in einer Remote-PC-Zugriff-Sitzung mit einem Computer mit Intel Core i7-Prozessor gestartet wird und eine für USB-Umleitung konfigurierte Webcam verwendet wird. Wenn dieses Problem auftritt, starten Sie den Computer des Benutzers und die Remote-PC-Zugriff-Sitzung neu. Die Sitzung wird fortgesetzt, wo die Verbindung unterbrochen wurde. Sie vermeiden dieses Problem, indem Sie HDX-Videokomprimierung anstelle von USB-Umleitung verwenden. Weitere Informationen finden Sie unter <http://support.citrix.com/proddocs/topic/xendesktop-7/hd-new-graphics-video.html>. [#423284]
  - Benutzergeräte, auf denen Receiver für HTML5 ausgeführt wird, können möglicherweise keine Verbindung zu einer Serverbetriebssystemmaschine herstellen, auf der Windows Server 2012 R2

ausgeführt wird. So vermeiden Sie dieses Problem:

- Verwenden Sie eine vorhandene Maschine als die Windows-Serverbetriebssystemmaschine statt einer Maschine, die mit den Maschinenerstellungsdiensten oder Provisioning Services erstellt wurde.
- Wenn Sie mit den Maschinenerstellungsdiensten erstellte Maschinen verwenden, ändern Sie die Registrierung auf dem Masterimage für den Katalog und erstellen Sie folgenden DWORD-Wert: Achtung: Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.  
HKEY\_LOCAL\_MACHINE\Software\Citrix\GroupPolicy\Defaults\lcaPolicies\AcceptWebSocketsConnections = 1 HKEY\_LOCAL\_MACHINE\Software\Citrix\GroupPolicy\Defaults\lcaPolicies\AllowDaDa die WebSockets-Portnummer durch Bearbeiten der Registrierung festgelegt wird, brauchen Sie die Citrix Richtlinie für WebSockets-Verbindungen für diesen Katalog nicht zu aktivieren.

- Wenn Sie eine Maschine verwenden, die mit Provisioning Services erstellt wurde, folgen Sie den Empfehlungen unter [CTX139265](#). [#424064]

- Wenn HDX Flash-Umleitung in einer Konfiguration mit zwei Bildschirmen verwendet wird und das Fenster mit dem Flash-Inhalt im Vollbildmodus auf einem Bildschirm angezeigt wird, kann das Klicken auf eine beliebige Stelle auf dem Bildschirm dazu führen, dass das Fenster mit dem Flash-Inhalt den Fokus verliert und hinter das Sitzungsfenster verschoben wird. Dieses Verhalten ist bei HDX Flash-Umleitung beabsichtigt. In diesem Fall können Benutzer das Fenster mit dem Flash-Inhalt wieder sichtbar machen, indem sie den Modus für das Sitzungsfenster mit der Citrix Desktop Viewer-Symbolleiste vom Vollbild- in den Fenstermodus ändern. [#567132]

## Lizenzierungsprobleme

- Wenn die aktuelle Version des Lizenzservers nicht installiert ist, können beim Installieren des Delivery Controllers Fehler auftreten. Installieren Sie die aktuelle Version des Lizenzservers beim Upgrade auf Version 7.6 vor allen anderen Komponenten. [#510425]
- Wenn die Lizenzierung deinstalliert und neu installiert wird und ein Produktadministrator mit Lesezugriff versucht, die Lizenzinformationen in Studio anzuzeigen, wird der Fehler "Sie haben nicht die notwendige Berechtigung, um diesen Vorgang auszuführen" angezeigt. Ein Administrator, der nur Lesezugriff hat, hat keine Berechtigung, den Lizenzserver als vertrauenswürdige anzusehen. Als Workaround muss ein Volladministrator für die Lizenzierung den Lizenzserver im Knoten "Lizenzierung" authentifizieren. [#380982]
- Wenn Sie den Lizenzserver installiert haben, ohne ihn nach der Installation erfolgreich (mit dem Lizenzserverkonfigurationstool) zu konfigurieren, schlagen alle nachfolgenden Lizenzserverupdates fehl. Stellen Sie als Workaround sicher, dass jede Lizenzserverinstallation anschließend mit dem Lizenzserver-Konfigurationstool konfiguriert wird. [#377079]
- Beim Start der License Administration Console oder des Simple License Service wird möglicherweise eine leere Seite angezeigt, wenn in Internet Explorer die verstärkte Sicherheitskonfiguration aktiviert ist und die License Administration Console bzw. Simple License Service nicht zu den vertrauenswürdigen Sites gehören. Workaround: Deaktivieren Sie die verstärkte Sicherheitskonfiguration für Internet Explorer [#382429]
- Wenn der Port 8083 während der Produktinstallation oder dem Produktupgrade in Verwendung ist, schlagen die Lizenzserverkonfiguration und die Installation mit einem Lizenzserverkonfigurationsfehler fehl. Überprüfen Sie als Workaround das Ereignisprotokoll, um sicherzustellen, dass es sich wirklich um den Fehler "Port wird verwendet" handelt. Ist dies der Fall:
  1. Doppelklicken Sie zum Deinstallieren des Lizenzservers auf die Datei CTX\_Licensing.msi im Ordner "x64\Licensing" auf dem Produktinstallationsmedium.
  2. Führen Sie das Installationsprogramm erneut aus. Einige Komponenten werden als teilweise installiert angezeigt. Klicken Sie auf "Installieren" und das Installationsprogramm schließt die Installation ab und konfiguriert die erforderlichen Produktkomponenten.
  3. Installieren Sie den Lizenzserver manuell und geben Sie Portnummern an, die nicht von anderen Anwendungen auf der Maschine verwendet werden. [#390815]
- Die Benutzerliste für die License Administration Console und die Webseite für den Citrix Simple License Service unterstützt keine Nicht-ASCII-Zeichen in Benutzer-/Gruppennamen. Aufgrund dieser Einschränkung wird auf einem russischen Betriebssystem die Gruppe VORDEFINIERT\Administratoren nicht der Benutzerliste hinzugefügt, weil sie Nicht-ASCII-Zeichen enthält. Dies gilt für Neuinstallationen und Upgrades. Benutzer in der Gruppe VORDEFINIERT\Administratoren in einem früheren Release von XenDesktop und dem Simple License Service haben nach dem Upgrade keinen Zugriff auf die License Administration Console oder den Simple License Service. Als Workaround fügen Sie nach der Installation die russischen Benutzer-/Gruppennamen in einer ASCII-Zeichen-Version über die License Administration Console hinzu. Alternativ installieren Sie den Lizenzserver auf einem der anderen unterstützten Betriebssysteme. [#395305]
- Wenn Sie das Produkt installieren oder aktualisieren und permanente Lizenzen haben, werden die Lizenzen in Studio u. U. mit dem Ablaufdatum 01.01.2000 angezeigt. Sie können dieses Ablaufdatum ignorieren und die Desktops und Anwendungen starten. [#402975]

## Probleme beim lokalen App-Zugriff

- Wenn der lokale App-Zugriff in einer Windows 8- oder Windows Server 2012- Hostumgebung aktiviert ist, und die Erweiterung für eine auf dem Client gehostete App keine Dateitypzuordnung hat, schlägt die Umleitung der Dateitypzuordnung fehl. Der Benutzer wird aufgefordert, "Suchen Sie eine App im Store" oder "Weitere Optionen" auszuwählen. [#372834]  
Verwenden Sie als Workaround eine dieser Methoden auf dem VDA-Masterimage:
  - Benennen Sie den Registrierungswert DelegateExecute für HKEY\_CLASSES\_ROOT\Unbekannter\Shell openas\\DelegateExecute um.
  - Öffnen Sie eine Datei mit der Erweiterung im Editor. Die Umleitung der Dateitypzuordnung funktioniert bei weiteren Versuchen.
- Die URL-Umleitung ist standardmäßig von Microsoft auf Windows Server 2012 deaktiviert. Um sie zu aktivieren, deaktivieren Sie die verstärkte Sicherheitskonfiguration für Internet Explorer. [#356260]
- Änderungen an den lokalen App-Zugriff-Eigenschaften in einer Sitzung werden nicht automatisch übernommen. Als Workaround melden Sie sich ab und wieder an. [#357488]
- Wenn lokale App-Zugriff-Anwendungen auf Windows 8- oder Windows 2012-Plattformen gestartet werden können diese VDAs nicht von der Modern Shell gestartet werden. Workaround: Schließen Sie die lokale Anwendung und starten Sie dann die VDA-Anwendung. [#359670]
- Shellhook.dll wird nicht mit Receiver für Windows 3.4 und früheren Versionen geladen, wenn lokale Anwendungen gestartet werden. Ändern Sie als Workaround den Wert des Registrierungsschlüssels LocalApplnit\_DLLs unter HKLM\Software unter Microsoft Windows NT\\CurrentVersion\Windows auf 1. Wenn der Workaround nicht verwendet wird, funktioniert die Umleitung der Dateitypzuordnung vom Client zum Server nicht richtig. [#356130]
- Die URL-Umleitung schlägt u. U. Websites fehl, für die ein Popupblocker aktiviert ist. Um dieses Problem zu umgehen, deaktivieren Sie den Popublocker. Aus Sicherheitsgründen wird das Deaktivieren von Popublockern nicht empfohlen. [#371220]
- Wenn der Aero-Modus für eine VDA-Sitzung aktiviert ist, bestehen Inkonsistenzen zwischen den lokalen VDA-Anwendungen und den auf dem Client gehosteten Anwendungen (z. B. ALT + Tab; blinkende Einträge auf der Taskleiste, Sprungliste, Echtzeitschau). Deaktivieren Sie den Aero-Modus für Kompatibilität. [#361043]
- Die Flash-Umleitung hat Kompatibilitätsprobleme, z. B. einen schwarzen WMP-Bildschirm und ein Flash-Pseudocontainer außerhalb des Bereichs. Als Workaround sollten Sie die Flash-Umleitung deaktivieren. [#360182]
- Die Desktopgestaltungsumleitung für Grafiken ist deaktiviert, wenn der lokale App-Zugriff aktiviert ist. Deaktivieren Sie den lokalen App-Zugriff, um dieses Feature zu verwenden. [#377386]
- Wenn die Sitzungszuverlässigkeit und der lokale App-Zugriff aktiviert sind und die Netzwerkkonnektivität unterbrochen wird, wenn Sie auf der Produktsymbolleiste auf "Home" klicken und den Desktop des Clients anzeigen, wird in der VDA-Sitzung u. U. nicht der letzte Bildschirm vor der Unterbrechung der Netzwerkkonnektivität angezeigt. Stattdessen werden nur die Produktsymbolleiste und der Desktop des Clients angezeigt. Auf der Symbolleiste funktioniert nur die Option "Trennen". [#357769]
- Wenn der lokale App-Zugriff aktiviert ist und ein Benutzer vor oder während der Wiedergabe einer Mediendatei für die Sitzung Vollbild aktiviert, wird das Bild nur teilweise oder gar nicht angezeigt. Starten Sie die Sitzung neu, um das Problem zu beheben. [#402702]

## Probleme mit Desktop Lock

- Nach der Installation von Citrix Desktop Lock auf einer in eine Domäne eingebundenen Windows 8-Maschine, wird nach dem Neustart u. U. ein schwarzer Desktophintergrund und statt des normalen Startbildschirms das Fenster "Arbeitsplatz" angezeigt. Geben Sie zum Abmelden von der Maschine Logoff in der Adressleiste des Fensters ein und drücken Sie die Eingabetaste. Weitere Informationen zum Konfigurieren von Windows 8-Maschinen mit Desktop Lock finden Sie unter [Desktop Lock](#). [#329075]

- Wenn eine Windows 8- oder Windows 7-Maschine ausfällt und der Benutzer versucht, die Maschine mit Strg+Alt+Entf neu zu starten, wird das Dialogfeld "Windows-Sicherheit" auf dem lokalen Desktop angezeigt. Dieses Problem tritt auch auf, wenn die Citrix Desktop Lock-Software installiert und konfiguriert ist. Um dieses Problem zu lösen, wählen Sie "Task-Manager starten", um das Dialogfeld "Neu starten" anzuzeigen. [#337507]
- Citrix Desktop Lock leitet Adobe Flash-Inhalte nicht an Benutzergeräte in der Domäne weiter. Der Inhalt kann angezeigt werden, wird aber auf dem Server wiedergegeben, nicht lokal. Als Workaround können Sie die Adobe Flash-Umleitung für serverseitigen Inhaltsabruf konfigurieren, um den Inhalt vom Server an das Clientgerät zu übergeben. Das Problem tritt nicht bei Geräten auf, die keiner Domäne angehören, und auch nicht, wenn der Inhalt mit Desktop Viewer angezeigt wird. [#263092]
- Der VDA-Start auf einem Benutzergerät kann fehlschlagen, wenn auf dem Gerät Windows 7 mit Desktop Lock und Receiver für Windows Enterprise 3.4 ausgeführt wird. Das Problem kann auftreten, wenn der Benutzer mit dem Internet verbunden ist oder keine Verbindung hat. Die Fehlermeldung "Keine Internetkonnektivität" wird dann ggf. angezeigt. Dies ist ein Drittanbieterproblem von Microsoft. Eine mögliche Lösung finden Sie unter <http://technet.microsoft.com/en-us/library/cc766017>. [#408642]

#### Endbenutzer- und VDA-Probleme

- Auf dem Benutzergerät können Handleprobleme mit wfica32.exe auftreten, wenn Windows Media Player-Dateien fortlaufend in einer Windows 32-Bit-XP-Clientsitzung wiedergegeben werden. [#378146]
- Die Schaltfläche "Trennen" steht nicht für die Benutzer zur Verfügung. Als Workaround können Sie eine Verknüpfung zum Hilfsprogramm TSDIscon.exe bereitstellen, das im Betriebssystem enthalten ist. Benutzer können dann Sitzungen trennen. [#362937]
- Benutzer haben möglicherweise nach der Trennung weiterhin Zugriff auf den Remotecomputer. Dies tritt auf, wenn der Benutzer den Befehl "Trennen" im Startmenü auswählt. Nach der Auswahl von "Trennen" entsteht eine Verzögerung. Wenn der Benutzer während dieser Verzögerung Strg+Alt+Entf drückt, steht der VDA auf dem Remotecomputer für mehrere Minuten weiterhin zur Verfügung, bevor die Verbindung getrennt wird. Während dieser Zeit hängt VDA auf dem Benutzergerät, bis VDA getrennt wird. [#322301]
- Bei VDAs, die auf Windows 7- und Windows 8-Plattformen installiert sind, werden u. U. zwei Mauszeiger angezeigt, ein beweglicher und ein auf der Benutzeroberfläche arretierter. Dies ist ein Drittanbieterproblem mit dem Nvidia-Treiber. Als Workaround können Sie die Nvidia GRID-Technologie (früher VGX) deaktivieren. Führen Sie MontereyEnable.exe -disable -reset aus und starten die Maschine neu. [#307921]
- Das Feature "Microsoft Desktopgestaltung" verursacht Skalierbarkeitskosten für VDAs. Für Benutzer, die maximale Skalierbarkeit benötigen, sollte die Desktopgestaltung für alle Benutzer, die die Desktopgestaltungsumleitung nicht verwenden, über eine Microsoft Richtlinie deaktiviert werden. [#386602]
- In bestimmten Szenarios wird die Sitzung wiederholt gesperrt, wenn Benutzer eine gesperrte Sitzung lokal entsperren. Dieses Problem kann auftreten, wenn der Benutzer eine Tastatur lokal abschließt oder ausschaltet, die in einer ICA-Remotesitzung mit einer Remote-PC-Zugriff-Maschine verbunden ist. Als Workaround sollten die Benutzer die Sitzung remote neu starten, die Sitzung trennen und sich dann erneut von der Konsole anmelden. [#382554]
- Für VDAs vor 7 werden die Daten der Benutzer u. U. nicht in Director angezeigt, selbst wenn Sie die Windows-Remoteverwaltung (WinRM) für diese VDAs richtig konfiguriert haben. Starten Sie den WinRM-Dienst neu, um dieses Problem zu beheben. Die Daten werden wie erwartet in Director angezeigt. [#392047]
- In sicheren Umgebungen wird ein neuer VDA möglicherweise nicht bei einem Controller registriert. Vor allem bei der Installation von einem VDA, der eine lokale Sicherheitsrichtlinieneinstellung hat, die nur Administratoren den Zugriff auf den Computer vom Netzwerk geben, wird der VDA installiert, die Registrierung beim Controller schlägt jedoch fehl. Stattdessen wird eine Warnung angezeigt, dass die Benutzerrechte nicht richtig konfiguriert sind. Weitere Informationen finden Sie unter [CTX117248](#). [#336203]
- Benutzer von Receiver für Windows können sich nicht mit Passthrough-Authentifizierung an Stores anmelden, selbst wenn die Domänen-Passthrough-Authentifizierung im StoreFront-Authentifizierungsdienst aktiviert ist. Um dieses Problem zu beheben, führen Sie den Befehl "Set-BrokerSite -TrustRequestsSentToTheXmlServicePort \$True" in einer Windows PowerShell-Eingabeaufforderung auf dem Controller aus. [#330775]
- ICA-Roundtripüberprüfungen werden nicht unterstützt, wenn der Legacygrafikmodus mit einer Richtlinie für Windows Server 2012-VDAs angegeben ist. Für dieses Problem gibt es keinen Workaround. Hinweis: Windows 8-VDAs unterstützen den Legacymodus nicht, sind daher von diesem Problem nicht betroffen. [#394824]
- Wenn Sie bei Windows Server 2012 R2 oder einem älteren Domänencontroller vorgeben, dass Benutzer bei der nächsten Anmeldung ihr Kennwort ändern müssen, tritt ein Fehler auf. Entfernen Sie zur Umgehung des Problems das Microsoft-Update KB2883201. [#438725]
- Wenn sich Benutzer direkt über RDP mit VDAs verbinden, meldet der Controller manchmal zusätzliche Sitzungen, die nicht vorhanden sind, und diese nicht bestehenden Sitzungen werden in Studio als "Verbunden" angezeigt, bis der VDA neu gestartet wird. [#385823]
- Wenn für Citrix StoreFront unter dem Browser Firefox oder Chrome die Passthrough-Authentifizierung implementiert ist, werden die Benutzer beim Starten von Anwendungen aufgefordert, Anmeldeinformationen einzugeben. [#441487]
- Meldet sich unter Citrix Receiver für Web mit Domänen-Passthrough-Konfiguration ein Benutzer mit Smartcard bei einem Gerät an und startet dann einen veröffentlichten Desktop, kann die Verbindung fehlschlagen. Zur Problemlösung bearbeiten Sie die ICA-Standarddatei (z. B. C:\inetpub\wwwroot\Citrix\Store\App\_Data\default.ica) und fügen Sie dem Abschnitt [Application] die folgende Zeile hinzu:

```
DisableCtrlAltDel=False
[#452813]
```

#### Probleme bei Personal vDisk

Informationen zu bekannten Problemen bei Personal vDisk finden Sie unter [Personal vDisks](#).

#### Probleme mit Provisioning Services

Informationen zu Problemen bei Provisioning Services finden Sie unter [Info zu Provisioning Services 7.0, 7.1 und 7.6](#)

# Nicht in diesem Release enthaltene Features

May 10, 2016

Die folgenden Features werden derzeit nicht bereitgestellt oder werden nicht mehr unterstützt.

- **Für Fingereingabe optimierten Desktop starten:** Diese Einstellung wurde für Windows 10-Maschinen deaktiviert. Weitere Informationen finden Sie unter [Einstellungen der Richtlinie "Mobilerfahrung"](#).
- **SecureICA-Verschlüsselung unter 128 Bit:** In Releases vor 7.x war eine Verschlüsselung von Clientverbindungen für Basic-, 40-Bit-, 56-Bit- und 128-Bit-Verschlüsselung durch SecureICA möglich. In Releases ab Version 7 steht die SecureICA-Verschlüsselung nur für die 128-Bit-Verschlüsselung zur Verfügung.
- **Legacydrucken:** Die folgenden Druckfunktionen werden für Releases 7.x nicht unterstützt:
  - Abwärtskompatibilität für DOS-Clients und 16-Bit-Drucker, einschließlich Legacy-Clientdruckernamen.
  - Unterstützung für mit Windows 95 und Windows NT verbundene Drucker, einschließlich verbesserter erweiterter Druckereigenschaften und Win32FavorRetainedSetting.
  - Möglichkeit zum Aktivieren oder Deaktivieren automatisch gespeicherter und automatisch wiederhergestellter Drucker.
  - DefaultPrnFlag, eine Registrierungseinstellung für Server zum Aktivieren/Deaktivieren automatisch gespeicherter und automatisch wiederhergestellter Drucker, die in Benutzerprofilen auf dem Server gespeichert werden.
- **Secure Gateway:** In Releases vor 7.x wurden mit Secure Gateway sichere Verbindungen zwischen dem Server und den Benutzergeräten hergestellt. Die Sicherung externer Verbindungen erfolgt nun mit NetScaler Gateway.
- **Spiegeln von Benutzern:** In Releases vor 7.x steuerten Administratoren die Benutzer-zu-Benutzer-Spiegelung mit Richtlinien. In 7.x-Releases ist das Spiegeln von Endbenutzern ein integriertes Feature in Director, wobei die Windows-Remoteunterstützung den Administratoren das Spiegeln und Beheben von Problemen auf nahtlos bereitgestellten Anwendungen und virtuellen Desktops gestattet.
- **Energie- und Kapazitätsverwaltung:** In Releases vor 7.x wurde mit der Energie- und Kapazitätsverwaltung der Stromverbrauch reduziert und die Serverkapazität verwaltet. Microsoft Configuration Manager ist das Ersatztool für diese Funktion.
- **Flash v1-Umleitung:** Bei Clients, die nicht die Flash-Umleitung der zweiten Generation unterstützen (einschließlich Receiver für Windows vor Version 3.0, Receiver für Linux 11.100 und Citrix Online-Plug-In 12.1) erfolgt ein Fallback auf serverseitige Wiedergabe für Legacyfeatures der Flash-Umleitung. VDAs in 7.x-Releases unterstützen die Flash-Umleitungsfeatures der zweiten Generation.
- **Lokales Textecho:** Dieses Feature wurde mit früheren Windows-Anwendungstechnologien zur Beschleunigung der Anzeige eingegebenen Texts auf Benutzergeräten bei Verbindungen mit hoher Latenz eingesetzt. Aufgrund von Verbesserungen am Grafiksubsystem und HDX SuperCodec ist dieses Feature in 7.x-Releases nicht enthalten.
- **SmartAuditor:** In Releases vor 7.x konnte mit SmartAuditor die Bildschirmaktivität einer Benutzersitzung aufgezeichnet werden. Diese Komponente ist in 7.x-Releases nicht verfügbar. In Version 7.6 Feature Pack 1 wird sie durch die Sitzungsaufzeichnung ersetzt.
- **Single Sign-On:** Dieses Feature zur Gewährleistung der Kennwortsicherheit wird für Windows 8- und Windows Server 2012-Umgebungen nicht unterstützt. Es wird noch für Windows 2008 R2 und Windows 7 Umgebungen unterstützt, ist aber nicht in 7.x-Releases enthalten. Es ist auf der Downloadwebsite von Citrix verfügbar: <http://citrix.com/downloads>.
- **Oracle-Datenbankunterstützung:** 7.x-Releases benötigen eine SQL Server-Datenbank.
- **Systemüberwachung und -wiederherstellung (HMR):** In Releases vor 7.x konnten von HMR Tests auf den Servern einer Serverfarm durchgeführt werden, um deren Zustand zu überwachen und mögliche Integritätsrisiken zu ermitteln. In 7.x-Releases bietet Director eine zentrale Ansicht der Systemintegrität, da die Überwachungs- und Warnfunktionen für die ganze Infrastruktur innerhalb der Director-Konsole dargestellt werden.
- **Benutzerdefinierte ICA-Dateien:** Benutzerdefinierte ICA-Dateien wurden verwendet, um direkte Verbindungen von

Benutzergeräten (mit der ICA-Datei) zu einer bestimmten Maschine herzustellen. In 7.x-Releases ist dieses Feature standardmäßig deaktiviert, kann aber für die normale Verwendung mit einer lokalen Gruppe aktiviert werden, oder im Modus für hohe Verfügbarkeit verwendet werden, falls der Controller nicht mehr verfügbar ist.

- **Management Pack für System Center Operations Manager (SCOM) 2007:** Das Management Pack für die Überwachung der Aktivität der Farmen mit SCOM unterstützt 7.x-Releases nicht.
- **CNAME-Funktion:** Die CNAME-Funktion war in Versionen vor 7.x standardmäßig aktiviert. Bereitstellungen, die von CNAME-Einträgen für FQDN-Umleitung und von der Verwendung von NetBIOS-Namen abhängig sind schlagen möglicherweise fehl. In 7.x-Releases bietet stattdessen die automatische Aktualisierung von Delivery Controllern, durch die die Liste der Controller automatisch aktualisiert wird und VDAs automatisch benachrichtigt werden, wenn Controller der Site hinzugefügt und aus ihr entfernt werden. Das Feature für automatische Controller-Updates ist in den Citrix Richtlinien standardmäßig aktiviert, kann jedoch durch Erstellen einer Richtlinie deaktiviert werden. Alternativ können Sie die CNAME-Funktion in der Registrierung wieder aktivieren, um mit der vorhandenen Bereitstellung fortzufahren und FQDN-Umleitung und die Verwendung von NetBIOS-Namen zuzulassen. Weitere Informationen finden Sie unter [CTX137960](#).
- **Assistent für schnelles Bereitstellen:** In Studio-Releases vor 7.x konnte mit dieser Option eine vollständig installierte XenDesktop-Bereitstellung schnell bereitgestellt werden. Durch den neuen vereinfachten Installations- und Konfigurationsworkflow in XenDesktop 7.x ist der Assistent für schnelles Bereitstellen nicht mehr nötig.
- **Konfigurationsdatei für Remote PC-Dienst und PowerShell-Skript für die automatische Verwaltung:** Remote PC ist jetzt in Studio und den Controller integriert.
- **Workflow Studio:** In Releases vor 7.x war Workflow Studio die grafische Oberfläche für den Aufbau von Workflows für XenDesktop. Das Feature wird in 7.x-Releases nicht unterstützt.
- **Farbtiefe:** In Studio-Releases vor 7.6 wurde über diese Option auf der Seite "Benutzereinstellung" die Farbtiefe für Bereitstellungsgruppen festgelegt. In Version 7.6 kann die Farbtiefe für Bereitstellungsgruppen über das PowerShell-Cmdlet "New-BrokerDesktopGroup" oder "Set-BrokerDesktopGroup" festgelegt werden.
- **Starten nicht-veröffentlichter Programme bei Clientverbindung:** In Releases vor 7.x wurde über diese Citrix Richtlinieneinstellung angegeben, ob Startanwendungen oder veröffentlichte Anwendungen über ICA oder RDP auf dem Server gestartet werden sollten. In 7.x-Releases wird mit dieser Einstellung nur festgelegt, ob Startanwendungen oder veröffentlichte Anwendungen über RDP auf dem Server gestartet werden.
- **Desktop starten:** In Releases vor 7.x wird mit dieser Citrix Richtlinieneinstellung angegeben, ob Benutzer, die keine Administratoren sind, eine Verbindung zu einer Desktopsitzung herstellen dürfen. In 7.x-Releases müssen Benutzer ohne Administratorrechte zur Gruppe der Benutzer mit direktem Zugriff für eine VDA-Maschine gehören, um Verbindungen zu Sitzungen auf diesem VDA herzustellen. Die Einstellung **Desktop starten** ermöglicht Benutzern ohne Administratorrechte, die in der Gruppe der Benutzer mit direktem Zugriff eines VDAs sind, über eine ICA-Verbindung eine Verbindung zum VDA herzustellen. Die Einstellung **Desktop starten** hat keine Auswirkungen auf RDP-Verbindungen. Unabhängig von dieser Einstellung können Benutzer, die in der Gruppe der Benutzer mit direktem Zugriff eines VDAs sind, über eine RDP-Verbindung eine Verbindung zum VDA herstellen.

Nicht in Receiver enthaltene Features oder Features mit anderen Standardwerten

- **COM-Portzuordnung:** Mit der COM-Portzuordnung wurde der Zugriff auf COM-Ports auf dem Benutzergerät erlaubt oder verhindert. Die COM-Portzuordnung wurde zuvor standardmäßig aktiviert. In 7.x-Versionen von XenDesktop und XenApp ist die COM-Portzuordnung standardmäßig deaktiviert. Einzelheiten finden Sie unter [Konfigurieren von COM-Port- und LPT-Portumleitungseinstellungen in der Registrierung](#).
- **LPT-Portzuordnung:** Mit der LPT-Portzuordnung wird der Zugriff von Legacyanwendungen auf LPT-Ports gesteuert. Die LPT-Portzuordnung wurde zuvor standardmäßig aktiviert. In 7.x-Releases ist die LPT-Portzuordnung standardmäßig deaktiviert.



- **PCM-Audiocodec:** In 7.x-Releases wird der PCM-Audiocodec nur von HTML5-Clients unterstützt.
- **Unterstützung für Microsoft ActiveSync**
- **Proxyunterstützung für ältere Versionen**, dies umfasst:
  - Microsoft Internet Security und Acceleration (ISA) 2006 (Windows Server 2003).
  - Oracle iPlanet-Proxyserver 4.0.14 (Windows Server 2003).
  - Squid-Proxyserver 3.1.14 (Ubuntu Linux Server 11.10).



# Systemanforderungen

Nov 02, 2016

Dieser Abschnitt enthält folgende Themen:

[Sitzungsaufzeichnung - XenApp 7.6 FP1, FP2 und LTSR](#)

[Delivery Controller](#)

[Datenbank](#)

[Studio](#)

[Director](#)

[Virtual Delivery Agent \(VDA\) für Windows-Desktopbetriebssysteme](#)

[Virtual Delivery Agent \(VDA\) für Windows-Serverbetriebssysteme](#)

[Hosts/Virtualisierungsressourcen](#)

[Unterstützung Funktionsebenen für Active Directory](#)

[HDX: Desktopgestaltungsumleitung](#)

[HDX: Windows Media-Bereitstellung](#)

[HDX – Flash-Umleitung](#)

[HDX 3D Pro](#)

[HDX – Anforderungen für Videokonferenzen mit Videokomprimierung](#)

[HDX – weitere Anforderungen](#)

[Anforderungen für den universellen Druckserver](#)

[Weitere Anforderungen](#)

Die Systemanforderungen in diesem Dokument galten zum Zeitpunkt der Freigabe der Produktversion. Nicht in diesem Dokument aufgeführte Systemanforderungen (z. B. StoreFront, Hostsysteme, Receiver und Plug-Ins oder Provisioning Services) werden in der jeweiligen Dokumentation behandelt.

Wichtig: Vor Beginn einer Installation lesen Sie [Vorbereiten der Installation](#).

Sofern nicht anders angegeben, wird erforderliche Software (z. B. .NET und C++-Pakete), wenn sie nicht auf einer Maschine erkannt wird, automatisch bereitgestellt. Das Citrix Installationsmedium enthält außerdem einige erforderliche Softwarekomponenten.

Das Installationsmedium enthält mehrere Komponenten von Drittanbietern. Bevor Sie diese Citrix Software verwenden, überprüfen Sie, ob Sicherheitsupdates von Drittanbietern nötig sind und installieren Sie sie.

Die Speicherplatzangaben sind Schätzwerte und verstehen sich zuzüglich des für Produktimage, Betriebssystem und andere Software erforderlichen Speicherplatzes.

Wenn Sie die Kernkomponenten (Controller mit SQL Server Express, Studio, Director, StoreFront und Lizenzierung) auf einem Server installieren, benötigen Sie für die Evaluierung des Produkts mindestens 3 GB Arbeitsspeicher; mehr Speicher wird empfohlen, wenn Sie eine Umgebung für Benutzer ausführen. Die Leistung schwankt abhängig von der genauen Konfiguration, einschließlich der Anzahl der Benutzer, Anwendungen und Desktops sowie anderen Faktoren.

Wichtig: Verwenden Sie nach der Installation von XenApp auf einem Windows Server 2012 R2-System das Tool zum Aktivieren von Kerberos (XASsonKerb.exe), um die ordnungsgemäße Funktion der Citrix Kerberos-Authentifizierung sicherzustellen. Das Tool ist im Ordner "Support > Tools>XASsonKerb" auf dem Installationsmedium und für die Verwendung benötigen Sie lokale Administratorrechte. Damit Kerberos ordnungsgemäß funktioniert, führen Sie xassonkerb.exe –install über eine Eingabeaufforderung auf dem Server aus. Wenn Sie später ein Update ausführen, bei dem der Registrierungspfad HKLM\System\CurrentControlSet\Control\LSA\OSConfig geändert wird, führen Sie den Befehl noch einmal aus. Zum Anzeigen aller verfügbaren Tooloptionen führen Sie den Befehl mit dem Parameter –help aus.

## Sitzungsaufzeichnung - XenApp 7.6 FP1, FP2 und LTSR

Die Sitzungsaufzeichnung ist in Englisch, Französisch, Deutsch, Japanisch, Spanisch und vereinfachtem Chinesisch erhältlich. Alle Sitzungsaufzeichnungskomponenten, die miteinander eine Verbindung herstellen, müssen in der gleichen Sprachversion vorliegen; Installationen mit unterschiedlichen Sprachen werden nicht unterstützt.

Die englische Sprachversion der Sitzungsaufzeichnung wird unter den folgenden Betriebssystemen unterstützt: Englisch, Russisch, Chinesisch (traditionell) und Koreanisch. Die Sprachversionen Französisch, Deutsch, Japanisch, Spanisch und Chinesisch (vereinfacht) der Sitzungsaufzeichnung werden unter den entsprechenden Sprachversionen des Betriebssystems unterstützt.

### Feature Pack 1

## Verwaltungskomponenten der Sitzungsaufzeichnung

Sie können die Verwaltungskomponenten der Sitzungsaufzeichnung (Datenbank für die Sitzungsaufzeichnung, Sitzungsaufzeichnungsserver und Richtlinienkonsole) auf dem gleichen oder mehreren unterschiedlichen Servern installieren.

## Datenbank für die Sitzungsaufzeichnung

Unterstützte Betriebssysteme:

- Microsoft Windows Server 2012 R2
- Microsoft Windows Server 2012
- Microsoft Windows Server 2008 R2 mit Service Pack 1

Anforderungen:

- Folgende Pakete: Microsoft SQL Server 2008 R2 SP3 Shared Management Objects und Microsoft SQL Server 2008 R2 SP3 System CLR Types. Sie können die Pakete unter <http://www.microsoft.com/en-us/download/details.aspx?id=44272> herunterladen.
- Microsoft SQL Server 2014 (Enterprise und Express Editionen), Microsoft SQL Server 2012 (Enterprise und Express Editionen) mit Service Pack 2 oder Microsoft SQL Server 2008 R2 (Enterprise und Express Editionen) mit Service Pack 3
- .NET Framework Version 3.5 Service Pack 1

- .NET Framework Version 3.5 Service Pack 1

## Sitzungsaufzeichnungsserver

Unterstützte Betriebssysteme:

- Microsoft Windows Server 2012 R2
- Microsoft Windows Server 2012, Microsoft Windows Server 2008 R2 mit Service Pack 1

Anforderungen:

Vor der Sitzungsaufzeichnung müssen Sie einige Voraussetzungen installieren. Öffnen Sie den Server-Manager und fügen Sie die IIS-Rolle hinzu. Wählen Sie folgende Optionen aus:

- Anwendungsentwicklung - ASP.NET 3.5 (Andere Komponenten werden automatisch ausgewählt. Klicken Sie auf "Hinzufügen", um die erforderlichen Rollen zu akzeptieren.)
- Sicherheit – Windows-Authentifizierung
- Verwaltungstools – Kompatibilität mit IIS 6
  - IIS 6-Metabasiskompatibilität
  - IIS 6-WMI-Kompatibilität
  - IIS 6-Skriptingtools
  - IIS 6-Verwaltungskonsole
- NET Framework Version 3.5 Service Pack 1
- Wenn der Sitzungsaufzeichnungsserver HTTPS als Kommunikationsprotokoll verwendet, fügen Sie ein gültiges Zertifikat hinzu. Die Sitzungsaufzeichnung verwendet in der Standardeinstellung HTTPS; dies wird von Citrix empfohlen.
- Microsoft Message Queuing (MSMQ) mit deaktivierter Active Directory-Integration und aktivierter MSMQ-HTTP-Unterstützung.

## Richtlinienkonsole für die Sitzungsaufzeichnung

Unterstützte Betriebssysteme:

- Microsoft Windows Server 2012 R2
- Microsoft Windows Server 2012
- Microsoft Windows Server 2008 R2 mit Service Pack 1

Anforderungen:

- .NET Framework Version 3.5 Service Pack 1

## Sitzungsaufzeichnungsagent

Installieren Sie den Sitzungsaufzeichnungsagent auf jedem XenApp-Server, auf dem Sie Sitzungen aufzeichnen möchten.

Unterstützte Betriebssysteme:

- Microsoft Windows Server 2012 R2
- Microsoft Windows Server 2012
- Microsoft Windows Server 2008 R2 mit Service Pack 1

Anforderungen:

- Microsoft Message Queuing (MSMQ) mit deaktivierter Active Directory-Integration und aktivierter MSMQ-HTTP-Unterstützung
- .NET Framework Version 3.5 Service Pack 1

## Sitzungsaufzeichnungsplayer

Unterstützte Betriebssysteme:

- Microsoft Windows 8.1
- Microsoft Windows 8
- Microsoft Windows Server 7 mit Service Pack 1

Für optimale Ergebnisse sollten Sie den Sitzungsaufzeichnungsplayer auf einer Arbeitsstation installieren, die folgende Anforderungen erfüllt:

- Eine Bildschirmauflösung von 1024 x 768
- Eine Farbtiefe von mindestens 32 Bit
- Speicher: 1 GB RAM (Minimum) Größere RAM- und CPU-/GPU-Ressourcen können die Leistung bei der Wiedergabe grafikintensiver Aufzeichnungen verbessern, insbesondere, wenn die Aufzeichnungen viele Animationen enthalten.

Die Reaktionszeit bei der Suche hängt von der Größe der Aufzeichnung und der Computerhardware ab.

Anforderungen:

- .NET Framework Version 3.5 Service Pack 1

Feature Pack 2 und LTSR

## Verwaltungskomponenten der Sitzungsaufzeichnung

Sie können die Verwaltungskomponenten der Sitzungsaufzeichnung (Datenbank für die Sitzungsaufzeichnung, Sitzungsaufzeichnungsserver und Richtlinienkonsole) auf dem gleichen oder mehreren unterschiedlichen Servern installieren.

## Datenbank für die Sitzungsaufzeichnung

Unterstützte Betriebssysteme:

- Microsoft Windows Server 2012 R2
- Microsoft Windows Server 2012
- Microsoft Windows Server 2008 R2 mit Service Pack 1

Anforderungen:

- .NET Framework Version 3.5 Service Pack 1 (nur Windows Server 2008 R2) oder .NET Framework Version 4.5.1 oder 4.6.

## Sitzungsaufzeichnungsserver

Unterstützte Betriebssysteme:

- Microsoft Windows Server 2012 R2

- Microsoft Windows Server 2012, Microsoft Windows Server 2008 R2 mit Service Pack 1

Anforderungen:

Vor der Sitzungsaufzeichnung müssen Sie einige Voraussetzungen installieren. Öffnen Sie den Server-Manager und fügen Sie die IIS-Rolle hinzu. Wählen Sie folgende Optionen aus:

- Anwendungsentwicklung:
  - ASP.NET 4.5 unter Windows Server 2012 und Windows Server 2012 R2, ASP.NET unter Windows Server 2008 R2 (Andere Komponenten werden automatisch ausgewählt. Klicken Sie auf "Hinzufügen", um die erforderlichen Rollen zu akzeptieren.)
- Sicherheit – Windows-Authentifizierung
- Verwaltungstools – Kompatibilität mit IIS 6
  - IIS 6-Metabaskompatibilität
  - IIS 6-WMI-Kompatibilität
  - IIS 6-Skriptingtools
  - IIS 6-Verwaltungskonsolle
- .NET Framework Version 3.5 Service Pack 1 (nur Windows Server 2008 R2) oder .NET Framework Version 4.5.1 oder 4.6.
- Wenn der Sitzungsaufzeichnungsserver HTTPS als Kommunikationsprotokoll verwendet, fügen Sie ein gültiges Zertifikat hinzu. Die Sitzungsaufzeichnung verwendet in der Standardeinstellung HTTPS; dies wird von Citrix empfohlen.
- Microsoft Message Queuing (MSMQ) mit deaktivierter Active Directory-Integration und aktivierter MSMQ-HTTP-Unterstützung.

## Richtlinienkonsole für die Sitzungsaufzeichnung

Unterstützte Betriebssysteme:

- Microsoft Windows Server 2012 R2
- Microsoft Windows Server 2012
- Microsoft Windows Server 2008 R2 mit Service Pack 1

Anforderungen:

- .NET Framework Version 3.5 Service Pack 1 (nur Windows Server 2008 R2) oder .NET Framework Version 4.5.1 oder 4.6.

## Sitzungsaufzeichnungsagent

Installieren Sie den Sitzungsaufzeichnungsagent auf jedem XenApp-Server, auf dem Sie Sitzungen aufzeichnen möchten.

**Unterstützte Windows-Betriebssysteme:**

- Microsoft Windows Server 2012 R2
- Microsoft Windows Server 2012
- Microsoft Windows Server 2008 R2 mit Service Pack 1

**Anforderungen:**

- Microsoft Message Queuing (MSMQ) mit deaktivierter Active Directory-Integration und aktivierter MSMQ-HTTP-Unterstützung
- .NET Framework Version 3.5 Service Pack 1 (nur Windows Server 2008 R2) oder .NET Framework Version 4.5.1 oder 4.6.

# Sitzungsaufzeichnungsplayer

Unterstützte Betriebssysteme:

- Microsoft Windows 8.1
- Microsoft Windows 8
- Microsoft Windows Server 7 mit Service Pack 1

Für optimale Ergebnisse sollten Sie den Sitzungsaufzeichnungsplayer auf einer Arbeitsstation installieren, die folgende Anforderungen erfüllt:

- Eine Bildschirmauflösung von 1024 x 768
- Eine Farbtiefe von mindestens 32 Bit
- Speicher: 1 GB RAM (Minimum) Größere RAM- und CPU-/GPU-Ressourcen können die Leistung bei der Wiedergabe grafikintensiver Aufzeichnungen verbessern, insbesondere, wenn die Aufzeichnungen viele Animationen enthalten.

Die Reaktionszeit bei der Suche hängt von der Größe der Aufzeichnung und der Computerhardware ab.

Anforderungen:

- .NET Framework Version 3.5 Service Pack 1 oder .NET Framework Version 4.5.1 oder 4.6.

## Delivery Controller

Unterstützte Betriebssysteme:

- Windows Server 2012 R2, Standard und Datacenter Edition
- Windows Server 2012, Standard und Datacenter Edition
- Windows Server 2008 R2 SP1, Standard, Enterprise und Datacenter Edition

Anforderungen:

- Speicherplatz auf dem Datenträger: 100 MB Verbindungsleasing (standardmäßig aktiviert) steigert diese Anforderungen; die Dimensionierung hängt von der Anzahl der Benutzer und Anwendungen sowie dem Modus (RDS oder VDI) ab. Beispiel: 100.000 RDS Benutzer mit 100 vor kurzem verwendeten Anwendungen benötigen ca. 3 GB Speicherplatz für das Verbindungsleasing; Bereitstellungen mit mehr Anwendungen benötigen u. U. mehr Speicherplatz. Bei dedizierten VDI-Desktops benötigen 40.000 Desktops mindestens 400-500 MB. Auf jeden Fall sollten Sie mehrere GBs an zusätzlichem Speicherplatz bereitstellen.
- Microsoft .NET Framework 3.5.1 (nur Windows Server 2008 R2)
- Microsoft .NET Framework 4.5.1 (4.5.2 und 4.6 werden auch unterstützt).
- Microsoft .NET Framework 4.6.1
- Windows PowerShell 2.0 (in Windows Server 2008 R2 enthalten) oder 3.0 (in Windows Server 2012 R2 und Windows Server 2012 enthalten)
- Visual C ++ 2005, 2008 SP1 und 2010 Redistributable-Pakete.

## Datenbank

Unterstützte Versionen von Microsoft SQL Server für die Sitekonfigurationsdatenbank (die zunächst die Konfigurationsprotokollierungsdatenbank und die Überwachungsdatenbank enthält):

- SQL Server 2016, Express, Standard und Enterprise Edition.
- SQL Server 2014 bis SP2, Express, Standard und Enterprise Edition.
- SQL Server 2012 bis SP3, Express, Standard und Enterprise Edition. Standardmäßig wird SQL Server 2012 SP1 Express zusammen mit dem Controller installiert, wenn keine unterstützte SQL Server-Installation erkannt wird.

- SQL Server 2008 R2 SP2 und SP3, Express, Standard, Enterprise und Datacenter Edition.

Die folgenden Datenbankfeatures werden unterstützt (außer bei SQL Server Express, das nur den eigenständigen Modus unterstützt):

- SQL Server-Cluster-Instanzen
- SQL Server-Spiegelung
- SQL Server 2012-AlwaysOn-Verfügbarkeitsgruppen

Die Windows-Authentifizierung ist für Verbindungen zwischen dem Controller und der SQL Server-Datenbank erforderlich.

Weitere Informationen zu den aktuell unterstützten Versionen der Datenbanken finden Sie unter <http://support.citrix.com/article/CTX114501>.

## Studio

Unterstützte Betriebssysteme:

- Windows 8.1, Professional und Enterprise Edition
- Windows 8, Professional und Enterprise Edition
- Windows 7, Professional, Enterprise und Ultimate Edition
- Windows Server 2012 R2, Standard und Datacenter Edition
- Windows Server 2012, Standard und Datacenter Edition
- Windows Server 2008 R2 SP1, Standard, Enterprise und Datacenter Edition

Anforderungen:

- Speicherplatz auf der Datenträger: 75 MB
- Microsoft .NET Framework 4.6.1
- Microsoft .NET Framework 4.5.1 (4.5.2 und 4.6 werden auch unterstützt)
- Microsoft .NET Framework 3.5 SP1 (nur Windows Server 2008 R2 und Windows 7)
- Microsoft Management Console 3.0 (in allen unterstützten Betriebssystemen enthalten)
- Windows PowerShell 2.0 (in Windows 7 und Windows Server 2008 R2 enthalten) oder 3.0 (in Windows 8.1, Windows 8, Windows Server 2012 R2 und Windows Server 2012 enthalten)

## Director

Unterstützte Betriebssysteme:

- Windows Server 2012 R2, Standard und Datacenter Edition
- Windows Server 2012, Standard und Datacenter Edition
- Windows Server 2008 R2 SP1, Standard, Enterprise und Datacenter Edition

Anforderungen:

- Speicherplatz auf dem Datenträger: 50 MB
- Microsoft .NET Framework 4.5.1 (4.5.2 und 4.6 werden auch unterstützt).
- Microsoft .NET Framework 3.5 SP1 (nur Windows Server 2008 R2)
- Microsoft Internetinformationsdienste (IIS) 7.0 und ASP.NET 2.0. Stellen Sie sicher, dass der Static-Content-Rollendienst für die IIS-Serverrolle installiert ist. Wenn diese Komponenten nicht auf Ihrem Server installiert sind, werden Sie möglicherweise aufgefordert, das Windows Server-Installationsmedium einzulegen. Sie werden dann installiert.
- Unterstützte Browser zum Anzeigen von Director:
  - Internet Explorer 11 oder 10

Der Kompatibilitätsmodus wird für Internet Explorer nicht unterstützt. Für den Zugriff auf Director müssen Sie die empfohlenen Webbrowsereinstellungen verwenden. Akzeptieren Sie bei der Installation von Internet Explorer die Standardeinstellung zur Verwendung der empfohlenen Sicherheits- und Kompatibilitätseinstellungen. Wenn der Browser bereits ohne die empfohlenen Einstellungen installiert ist, navigieren Sie auf "Extras > Internetoptionen > Erweitert" > Zurücksetzen" und folgen Sie den Anweisungen.

- Firefox ESR (Extended Support Release)
- Chrome

## Virtual Delivery Agent (VDA) für Windows-Desktopbetriebssysteme

### Unterstützte Betriebssysteme:

- Windows 10, Enterprise Edition. Dies gilt für eigenständige VDA-Installationen von XenApp und XenDesktop 7.6 FP3, in der ein Standard-VDA verwendet wird. Folgendes wird in 7.6 FP3 unter Windows 10 nicht unterstützt:
  - HDX 3D Pro
  - GPU-Beschleunigung
  - Desktopgestaltungsumleitung
  - Legacygrafikmodus
  - Sicherer Start
  - Veröffentlichen universeller Windows-Apps mit VM-gehosteten Apps
  - Citrix Profilverwaltung
- Windows 8.1, Professional und Enterprise Edition
- Windows 8, Professional und Enterprise Edition
- Windows 7 SP1, Professional, Enterprise und Ultimate Edition

Für das Verwenden des Server-VDI-Features können Sie über die Befehlszeilenschnittstelle einen VDA für Windows-Desktopbetriebssysteme auf einem unterstützten Serverbetriebssystem installieren. Weitere Informationen finden Sie unter [Server VDI](#).

- Windows Server 2012 R2, Standard und Datacenter Edition
- Windows Server 2012, Standard und Datacenter Edition
- Windows Server 2008 R2 SP1, Standard, Enterprise und Datacenter Edition

### Anforderungen:

- Microsoft .NET Framework 4.6.1
- Microsoft .NET Framework 4.5.1 (4.5.2 und 4.6 werden auch unterstützt)
- Microsoft .NET Framework 3.5.1 (nur Windows 7)
- Microsoft Visual C++ 2005, 2008 und 2010 Runtimes (32 Bit und 64 Bit). Dies gilt für XenApp und XenDesktop 7.6, 7.6 FP1 und FP2.
- Microsoft Visual C++ 2008, 2010 und 2013 Runtime (32 Bit und 64 Bit). Dies gilt für eigenständige VDA-Installationen von XenApp und XenDesktop 7.6 FP3.

Remote-PC-Zugriff verwendet diesen VDA, den Sie auf physischen Büro-PCs installieren.

Mehrere Multimediabeschleunigungsfunktionen (z. B. HDX MediaStream-Windows Media-Umleitung) erfordern, dass Microsoft Media Foundation auf dem Computer installiert wird, auf dem der VDA installiert ist. Wenn Media Foundation nicht installiert ist, werden die Multimediabeschleunigungsfeatures nicht installiert und sind nicht funktionsfähig. Entfernen Sie Media Foundation nicht nach der Installation der Citrix Software von der Maschine, sonst können sich Benutzer nicht an der Maschine anmelden. Bei den meisten Editionen von Windows 8.1, Windows 8 und Windows 7 ist Media Foundation bereits installiert und kann nicht entfernt werden. Bei N-Editionen sind bestimmte medienrelevante Technologien nicht enthalten; Sie können die Software von Microsoft oder einem Drittanbieter beziehen.

Bei der VDA-Installation können Sie entscheiden, ob Sie die HDX 3D Pro-Version des VDAs für Windows-Desktopbetriebssysteme installieren möchten. Diese Version eignet sich besonders für die Verwendung mit DirectX- und OpenGL-gesteuerten Anwendungen sowie mit Rich Media-Inhalten wie Video.



## Virtual Delivery Agent (VDA) für Windows-Serverbetriebssysteme

Unterstützte Betriebssysteme:

- Windows Server 2012 R2, Standard und Datacenter Edition
- Windows Server 2012, Standard und Datacenter Edition
- Windows Server 2008 R2 SP1, Standard, Enterprise und Datacenter Edition

Das Installationsprogramm stellt die folgenden Anforderungen automatisch bereit, die auch auf den Citrix Installationsmedien in den Ordnern \Support zur Verfügung stehen:

- Microsoft .NET Framework 4.6.1
- Microsoft .NET Framework 4.5.1 (4.5.2 und 4.6 werden auch unterstützt)
- Microsoft .NET Framework 3.5.1 (nur Windows Server 2008 R2)
- Microsoft Visual C++ 2005, 2008 und 2010 Runtimes (32 Bit und 64 Bit). Dies gilt für XenApp und XenDesktop 7.6, 7.6 FP1 und FP2.
- Microsoft Visual C++ 2008, 2010 und 2013 Runtime (32 Bit und 64 Bit). Dies gilt für eigenständige VDA-Installationen von XenApp und XenDesktop 7.6 FP3.

Das Installationsprogramm installiert und aktiviert automatisch die Rollendienste für Remotedesktopdienste, wenn sie nicht bereits installiert und aktiviert sind.

Mehrere Multimediabeschleunigungsfunktionen (z. B. HDX MediaStream-Windows Media-Umleitung) erfordern, dass Microsoft Media Foundation auf dem Computer installiert wird, auf dem der VDA installiert ist. Wenn Media Foundation nicht installiert ist, werden die Multimediabeschleunigungsfeatures nicht installiert und sind nicht funktionsfähig. Entfernen Sie Media Foundation nicht nach der Installation der Citrix Software von der Maschine, sonst können sich Benutzer nicht an der Maschine anmelden. Bei den meisten Editionen von Windows Server 2012 R2, Windows Server 2012 und Windows Server 2008 R2 wird Media Foundation über den Server-Manager installiert (bei Windows Server 2012 R2 und Windows Server 2012: ServerMediaFoundation, bei Windows Server 2008 R2: DesktopExperience). Bei N-Editionen sind bestimmte medienrelevante Technologien nicht enthalten; Sie können die Software von Microsoft oder einem Drittanbieter beziehen.

## Hosts/Virtualisierungsressourcen

Unterstützte Plattformen:

- XenServer:
  - XenServer 6.5 SP1
  - XenServer 6.5
  - XenServer 6.2 SP1 und Hotfixes (Sie müssen SP1 anwenden, damit die Anwendung zukünftiger Hotfixes möglich ist)
  - XenServer 6.1
- VMware vSphere: Der "Linked Mode"-Betrieb von vSphere vCenter wird nicht unterstützt.
  - VMware vSphere 6.0 Update 1
  - VMware vSphere 6.0
  - VMware vSphere 5.5 Update 3
  - VMware vSphere 5.5 Update 2
  - VMware vSphere 5.5 Update 1
  - VMware vSphere 5.5
  - VMware vSphere 5.1 Update 3
  - VMware vSphere 5.1 Update 2
  - VMware vSphere 5.0 Update 3
  - VMware vSphere 5.0 Update 2
  - VMware vCenter 5.5-/6-Gerät

- System Center Virtual Machine Manager: Enthält alle Versionen von Hyper-V, die mit den unterstützten Versionen von System Center Virtual Machine Manager registriert werden können.
  - System Center Virtual Machine Manager 2012 R2
  - System Center Virtual Machine Manager 2012 SP1
  - System Center Virtual Machine Manager 2012
- Nutanix Acropolis 4.5: Mehrere XenApp- und XenDesktop-Features sind mit dieser Plattform nicht verfügbar. Details dazu finden Sie in [CTX202032](#). Weitere Informationen zur Verwendung des Produkts mit Acropolis finden Sie unter <https://portal.nutanix.com/#/page/docs>.

Sie können dieses Produkt auch in den folgenden Cloudumgebungen bereitstellen:

- Amazon Web Services (AWS)
  - Sie können Anwendungen und Desktops auf unterstützten Windows Server-Betriebssystemen bereitstellen.
  - SQL Server 2012 Enterprise ist für AWS nicht verfügbar.
  - AWS bietet keine Desktopbetriebssysteminstanzen.
  - Amazon Relational Database Service (RDS) wird nicht unterstützt.
  - Weitere Informationen finden Sie in der AWS-Dokumentation und unter [Deploy XenApp and XenDesktop 7.5 and 7.6 with Amazon VPC](#).
- Citrix CloudPlatform
  - Die unterstützte Mindestversion ist 4.2.1 mit Hotfixes 4.2.1-4.
  - Bereitstellungen wurden mit XenServer 6.2 (mit Service Pack 1 und Hotfix XS62ESP1003) und vSphere 5.1-Hypervisors getestet.
  - CloudPlatform unterstützt keine Hyper-V-Hypervisors.
  - CloudPlatform 4.3.0.1 unterstützt VMware vSphere 5.5.
  - Weitere Informationen sowie die Anforderungen für Linux-basierte Systeme finden Sie in der CloudPlatform-Dokumentation (einschließlich Versionshinweise für Ihre CloudPlatform-Version) und unter [XenApp und XenDesktop – Konzepte und Bereitstellung auf CloudPlatform](#).

Aktualisierte Informationen zur Hypervisorunterstützung finden Sie unter [CTX131239](#).

Die folgenden Kombinationen aus Virtualisierungsressourcen- und Speichertechnologien werden für Maschinenerstellungsdienste und für das Hinzufügen von Active Directory-Konten in VMs zur Laufzeit unterstützt: Mit einem Sternchen (\*) gekennzeichnete Kombinationen werden empfohlen.

| Virtualisierungsressource | Lokale Datenträger                             | NFS  | Blockspeicher                                | Speicherlink |
|---------------------------|------------------------------------------------|------|----------------------------------------------|--------------|
| XenServer                 | Ja                                             | Ja * | Ja                                           | Nein         |
| VMware                    | Ja (nicht vMotion oder dynamische Platzierung) | Ja * | Ja                                           | Nein         |
| Hyper-V                   | Ja                                             | Nein | Ja * (erfordert freigegebene Clustervolumes) | Nein         |

Das Wake-On-LAN-Feature von Remote-PC-Zugriff erfordert Microsoft System Center Configuration Manager 2012. Einzelheiten finden Sie unter [Configuration Manager und Wake-On-LAN-Feature für den Remote-PC-Zugriff](#).

Unterstützung Funktionsebenen für Active Directory

Die folgenden Funktionsebenen werden für Active Directory-Gesamtstrukturen und -Domänen unterstützt:

- Natives Windows 2000 (auf Domänencontrollern nicht unterstützt)
- Windows Server 2003
- Windows Server 2008
- Windows Server 2008 R2
- Windows Server 2012
- Windows Server 2012 R2

### HDX: Desktopgestaltungsumleitung

Das Windows-Benutzergerät bzw. der Thin Client muss Folgendes unterstützen oder enthalten:

- DirectX 9
- Pixel Shader 2.0 (in Hardware unterstützt)
- 32 Bit pro Pixel
- 1,5 GHz Prozessor (32-Bit oder 64-Bit)
- 1 GB RAM
- 128 MB Videospeicher auf der Grafikkarte oder einem integrierten Grafikprozessor

HDX überprüft das Windows-Gerät auf die erforderlichen GPU-Anforderungen und schaltet ggf. automatisch auf serverseitige Desktopgestaltung um. Geräte, die die erforderlichen GPU-Anforderungen erfüllen aber nicht die benötigte Prozessorgeschwindigkeit oder RAM-Spezifikationen haben, müssen in der GPO-Gruppe der Geräte erfasst werden, die von der Desktopgestaltungsumleitung ausgeschlossen sind.

Als Mindestausstattung ist eine verfügbare Bandbreite von 1,5 MBit/s erforderlich, empfohlen werden 5 MBit/s. Die Werte schließen End-to-End-Latenz ein.

### HDX: Windows Media-Bereitstellung

Für den clientseitigen Abruf von Windows Media-Inhalten, die Windows Media-Umleitung und die Windows Media-Multimediatranscodierung in Echtzeit werden folgende Clients unterstützt: Receiver für Windows, Receiver für iOS und Receiver für Linux.

Zur Verwendung des clientseitigen Inhaltsabrufs von Windows Media auf Windows 8-Geräten legen Sie Citrix Multimedia Redirector als Standardprogramm fest: Navigieren Sie zu Systemsteuerung > Programme > Standardprogramme > Standardprogramme festlegen, wählen Sie Citrix Multimedia Redirector und klicken Sie auf Dieses Programm als Standard festlegen oder auf Standards für dieses Programm auswählen.

Für die GPU-Transcodierung ist ein NVIDIA CUDA-fähiger GPU mit Compute Capability 1.1 oder höher erforderlich. Siehe <http://developer.nvidia.com/cuda/cuda-gpus>.

### HDX – Flash-Umleitung

Die folgenden Clients und Adobe Flash Player-Versionen werden unterstützt:

- Receiver für Windows: Die zweite Generation der Flash-Umleitungsfeatures erfordert Adobe Flash Player for Other Browsers, diese Version wird auch als NPAPI (Netscape Plugin Application Programming Interface) Flash Player bezeichnet.
- Receiver für Linux: Die zweite Generation der Flash-Umleitungsfeatures erfordert Adobe Flash Player for other Linux oder Adobe Flash Player for Ubuntu.
- Citrix Online Plug-In 12.1: Für Legacyfeatures der Flash-Umleitung wird Adobe Flash Player für Windows Internet Explorer benötigt (auch ActiveX Player genannt).

Die Hauptversionsnummer von Flash Player auf dem Benutzergerät muss größer oder gleich der Hauptversionsnummer von Flash Player auf dem Server sein. Wenn eine ältere Version oder gar keine Version von Flash Player auf dem Benutzergerät installiert ist, werden Flash-Inhalte auf dem Server wiedergegeben.

Die Maschinen, auf denen VDAs ausgeführt werden, erfordern Folgendes:

- Adobe Flash Player für Windows Internet Explorer (ActiveX Player)
- Internet Explorer Version 7, 8, 9, 10 und 11 (im Modus "Nicht-moderne-Oberfläche") Flash-Umleitung erfordert Internet Explorer auf dem Server; mit anderen Browsern werden Flash-Inhalte auf dem Server wiedergegeben.
- Der geschützte Modus muss in Internet Explorer deaktiviert sein (Extras > Internetoptionen > Registerkarte "Sicherheit" > Kontrollkästchen "Geschützten Modus aktivieren" deaktiviert). Starten Sie Internet Explorer neu, damit die Änderung wirksam wird.

## HDX 3D Pro

Wenn Sie einen VDA für Windows-Desktopbetriebssysteme installieren, können Sie die Version HDX 3D Pro installieren.

Auf der physischen bzw. virtuellen Maschine, auf der die Anwendung gehostet wird, kann GPU-Passthrough oder Virtual GPU (vGPU) verwendet werden:

- GPU-Passthrough steht mit Citrix XenServer zur Verfügung. GPU-Passthrough ist auch mit VMware vSphere und VMware ESX verfügbar und wird in diesem Zusammenhang als vDGA (virtual Direct Graphics Acceleration) bezeichnet.
- vGPU ist mit Citrix XenServer verfügbar. Weitere Informationen finden Sie unter [www.citrix.com/go/vGPU](http://www.citrix.com/go/vGPU) (Anmeldeinformationen für Citrix My Account erforderlich).

Als Minimalsausstattung für den Hostcomputer empfiehlt Citrix 4 GB RAM und vier virtuelle CPUs mit einer Taktfrequenz von 2,3 GHz.

Grafikprozessor (GPU):

- Im Hinblick auf CPU-basierte Komprimierung, einschließlich verlustfreier Komprimierung, unterstützt HDX 3D Pro alle Grafikkarten auf dem Hostcomputer, die mit der bereitgestellten Anwendung kompatibel sind.
- Für optimierten GPU-Framepufferzugriff mit NVIDIA GRID API erfordert HDX 3D Pro NVIDIA Quadro Smartcards mit den neuesten NVIDIA-Treibern. NVIDIA GRID liefert eine hohe Framerate und dadurch eine sehr interaktive Benutzererfahrung.
- Zu den Anforderungen für HDX 3D Pro zur Verwendung von vGPU mit XenServer gehören die Grafikkarten NVIDIA GRID K1 und K2.

Benutzergerät:

- HDX 3D Pro unterstützt alle Monitorauflösungen, die von dem GPU auf dem Hostcomputer unterstützt werden. Um mit den empfohlenen Minimalspezifikationen für Benutzergeräte und GPUs eine optimale Leistung zu erzielen, empfiehlt Citrix jedoch eine maximale Monitorauflösung für Benutzergeräte von 1920 x 1200 Pixeln für LAN-Verbindungen sowie von 1280 x 1024 Pixeln für WAN-Verbindungen.
- Als Mindestausstattung für Benutzergeräte empfiehlt Citrix mindestens 1 GB RAM und eine CPU mit einer Taktfrequenz von 1,6 GHz. Zur Verwendung des standardmäßigen Tiefenkomprimierungscodecs, der bei Verbindungen mit geringer Bandbreite erforderlich ist, ist eine leistungsfähigere CPU erforderlich, es sei denn, die Decodierung erfolgt in der Hardware. Zur Erzielung der optimalen Leistung empfiehlt Citrix die Ausstattung von Benutzergeräten mit mindestens 2 GB RAM und einer Dual-Core-CPU mit einer Taktfrequenz von mindestens 3 GHz.
- Bei Multimonitorzugriff empfiehlt Citrix Benutzergeräte mit Vierkern-CPU.
- Benutzergeräte benötigen keinen dedizierten GPU für den Zugriff auf Desktops oder Anwendungen, die mit HDX 3D Pro bereitgestellt werden.
- Citrix Receiver muss installiert sein.

## HDX – Anforderungen für Videokonferenzen mit Videokomprimierung

Unterstützte Clients: Citrix Receiver für Windows, Receiver für Mac und Receiver für Linux

Unterstützte Videokonferenzanwendungen:

- Citrix GoToMeeting HDFaces
- Adobe Connect
- Cisco WebEx
- IBM Sametime
- Microsoft Lync 2010 und 2013
- Für Microsoft Office Communicator:
- Google + Hangouts
- Media Foundation-basierte Videoanwendungen auf Windows 8.x, Windows Server 2012 und Windows Server 2012 R2
- Skype 6.7: Zum Verwenden von Skype auf einem Windows-Client bearbeiten Sie die Registrierung auf Client und Server folgendermaßen:
  - Clientregistrierungsschlüssel HKEY\_CURRENT\_USER\Software\Citrix\HdxRealTime  
Name: DefaultHeight, Typ: REG\_DWORD, Wert: 240  
  
Name: DefaultWidth, Typ: REG\_DWORD, Wert: 320
  - Serverregistrierungsschlüssel HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\Vd3d\Kompatibilität  
Name: skype.exe, Typ: REG\_DWORD, Wert: 0

Andere Anforderungen an Benutzergeräte:

- Geeignete Hardware für die Audiowiedergabe
- DirectShow-kompatible Webcam (Webcam-Standardeinstellungen verwenden). Hardware-verschlüsselungsfähige Webcams senken die clientseitige CPU-Nutzung.
- Webcam-Treiber (möglichst vom Kamerahersteller)

## HDX – weitere Anforderungen

UDP-Audio für Multistream-ICA unterstützt wird unter Receiver für Windows und Receiver für Linux 13 unterstützt.

Die Echounterdrückung wird unter Receiver für Windows unterstützt.

## Anforderungen für den universellen Druckserver

- Universeller Druckserver - Der universelle Druckserver umfasst Client- und Serverkomponenten. Die UPClient-Komponente ist in der VDA-Installation enthalten. Die UPServer-Komponente (die Sie auf jedem Druckserver installieren, auf dem freigegebene Drucker sind, die Sie mit dem universellen Druckertreiber von Citrix in Benutzersitzungen bereitstellen möchten) wird unter folgenden Systemen unterstützt:
  - Windows Server 2008 R2 SP1
  - Windows Server 2008 (32 Bit)
  - Windows Server 2012 R2 und 2012 Dies gilt nur für XenApp und XenDesktop ab Version 7.6 FP3.
- **XenApp und XenDesktop 7.6 FP3:** Die folgenden Voraussetzungen gelten für die Installation der UPServer-Komponente auf dem Druckserver:
  - Microsoft Visual Studio 2013 Runtime (32 Bit und 64 Bit)
  - Microsoft .NET Framework 4.5.1
  - CDF\_x64.msi für 64-Bit-Systeme bzw. CDF\_x86.msi für 32-Bit-Systeme
  - UpsServer\_x64.msi für 64-Bit-Systeme bzw. UpsServer\_x86.msi für 32-Bit-Systeme

Die UPClient-Komponente wird auf XenApp- und XenDesktop-Hosts installiert, die Netzwerksitzungsdrucker bereitstellen. Sie ist kompatibel mit Windows 10-Desktops und gehört zur VDA-Installation.

Die Benutzerauthentifizierung bei Druckvorgängen erfordert, dass der universelle Druckserver in der gleichen Domäne ist wie der Remotedesktopdienste-VDA.

## Sonstiges

- Citrix empfiehlt die Installation der Komponentensoftwareversionen auf dem Installationsmedium für dieses Release (bzw. ein Upgrade auf diese Versionen).
  - StoreFront erfordert 2 GB Arbeitsspeicher. Informationen zu Systemanforderungen finden Sie in der Dokumentation von StoreFront. StoreFront 2.6 ist die in diesem Release unterstützte Mindestversion.
  - Wenn Sie Provisioning Services mit diesem Release verwenden, wird als Mindestversion Provisioning Services 7.0 unterstützt.
  - Citrix Lizenzserver erfordert 40 MB Speicherplatz auf dem Datenträger. Informationen zu den Systemanforderungen finden Sie in der Dokumentation der Lizenzierung. Es wird nur Citrix Lizenzserver für Windows unterstützt. Die unterstützte Mindestversion ist 11.12.1.
- Die Microsoft-Gruppenrichtlinien-Verwaltungskontrolle ist erforderlich, wenn Sie Citrix Richtlinieninformationen in Active Directory und nicht in der Sitekonfigurationsdatenbank speichern. Weitere Informationen finden Sie in der Dokumentation von Microsoft.
- Standardmäßig wird zusammen mit einem VDA Receiver für Windows installiert. Informationen zu den Systemanforderungen für andere Plattformen finden Sie in der Dokumentation von Receiver für Windows.
- Receiver für Linux und Receiver für Mac stehen auf dem Produktinstallationsmedium zur Verfügung. Weitere Informationen zu Systemanforderungen finden Sie in der jeweiligen Dokumentation.
- Wenn Sie Access Gateway-Versionen vor 10.0 mit diesem Release verwenden, werden Windows 8.1- und Windows 8-Clients nicht unterstützt.
- Desktop Lock – unterstützte Betriebssysteme:
  - Windows 7 (einschließlich der Embedded Edition)
  - Windows XP Embedded
  - Windows VistaBenutzergeräte müssen mit einem LAN verbunden sein.

Unterstützte Receiver-Version: Citrix Receiver für Windows Enterprise 3.4-Paket (mindestens).

- Clientordnerumleitung – unterstützte Betriebssysteme:
  - Server: Windows Server 2008 R2 SP1, Windows Server 2012 und Windows Server 2012 R2
  - Client (mit der aktuellen Version von Citrix Receiver für Windows): Windows 7, Windows 8, und Windows 8.1
- Die Verwendung mehrerer Netzwerkkarten wird nicht unterstützt.
- Unterstützte Versionen sind im [App-V](#)-Artikel aufgeführt.

# Technische Übersicht

May 10, 2016

XenApp und XenDesktop sind Virtualisierungslösungen, die IT die Steuerung von virtuellen Maschinen, Anwendungen sowie von der Lizenzierung und Sicherheit ermöglichen und gleichzeitig Benutzern von überall Zugriff mit jedem Gerät bieten.

XenApp und XenDesktop ermöglichen Folgendes:

- Endbenutzer können Anwendungen und Desktops unabhängig vom Betriebssystem und von der Benutzeroberfläche eines Geräts ausführen.
- Administratoren können Netzwerke verwalten und Zugriff von ausgewählten Geräten oder allen Geräten gewähren oder einschränken.
- Administratoren können ein ganzes Netzwerk von einem Datacenter aus verwalten.

XenApp und XenDesktop haben eine einheitliche Architektur, die FlexCast Management Architecture (FMA) genannt wird. Die Hauptfunktion von FMA umfasst die Ausführung mehrerer Versionen von XenApp oder XenDesktop auf einer einzigen Site und die Bereitstellung von integriertem Provisioning.

## Hauptkomponenten von FMA

Eine typische XenApp- oder XenDesktop-Umgebung besteht aus einigen technischen Hauptkomponenten, die interagieren, wenn Benutzer Verbindungen mit Anwendungen und Desktops herstellen und Daten über die Siteaktivität erfassen.

### **Citrix Receiver**

Ein Softwareclient, der auf dem Benutzergerät installiert ist, stellt die Verbindung zur virtuellen Maschine über TCP-Port 80 oder 443 her und kommuniziert mit StoreFront über die StoreFront-Dienst-API.

### **StoreFront**

Die Benutzeroberfläche, mit der Benutzer authentifiziert werden, verwaltet Anwendungen und Desktops und hostet den Anwendungsstore. StoreFront kommuniziert mit dem Delivery Controller mit XML.

### **Delivery Controller**

Die zentrale Verwaltungskomponente einer XenApp- oder XenDesktop-Site. Sie besteht aus Diensten, die Ressourcen, Anwendungen und Desktops verwalten und die Last der Benutzerverbindungen optimieren und ausgleichen.

### **Virtual Delivery Agent (VDA)**

Agent, der auf Maschinen installiert ist, auf denen Windows Server- oder Windows-Desktopbetriebssysteme ausgeführt werden. Er gewährleistet, dass diese Maschinen und die von ihnen gehosteten Ressourcen Benutzern zur Verfügung gestellt werden können. Auf Windows-Serverbetriebssystemmaschinen ermöglichen VDAs das Hosten mehrerer Verbindungen für mehrere Benutzer und sind mit den Benutzern über einen der folgenden Ports verbunden:

- TCP-Port 80 oder Port 443, wenn SSL aktiviert ist
- TCP-Port 2598, wenn Citrix Gateway Protocol (CGP) aktiviert ist (für Sitzungszuverlässigkeit)
- TCP-Port 1494, wenn CGP deaktiviert ist oder wenn der Benutzer eine Verbindung über einen Legacyclient herstellt

### **Brokerdienst**

Ein Delivery Controller-Dienst, der protokolliert, welche Benutzer wo angemeldet sind sowie welche Sitzungsressourcen die Benutzer haben und ob Benutzer sich erneut mit vorhandenen Anwendungen verbinden müssen. Der Brokerdienst führt PowerShell aus und kommuniziert mit dem Broker Agent über TCP-Port 80. Er kann TCP-Port 443 nicht verwenden.

### **Broker Agent**

Ein Agent, der mehrere Plug-Ins hostet und Echtzeitdaten sammelt. Der Broker Agent ist auf dem VDA und mit dem Controller über TCP-Port 80 verbunden. Er kann TCP-Port 443 nicht verwenden.

## Monitor Service

Eine Komponente des Delivery Controllers, die historische Daten sammelt und diese standardmäßig in der Sitedatenbank speichert. Der Monitor Service kommuniziert über TCP-Port 80 oder 443.

## ICA-Datei/-Stack

Gebündelte Benutzerinformationen, die für die Verbindung mit dem VDA erforderlich sind.

## Sitedatenbank

Eine Microsoft SQL-Datenbank, die Daten für den Delivery Controller speichert, wie Siterichtlinien, Maschinenkataloge und Bereitstellungsgruppen.

## NetScaler Gateway

Eine Datenzugriffslösung, die sicheren Zugriff von innerhalb oder außerhalb der LAN-Firewall durch zusätzliche Anmeldeinformationen gewährleistet.

## Director

Ein webbasiertes Tool, das Administratoren den Zugriff auf Echtzeitdaten vom Broker Agent, historische Daten von der Sitedatenbank und HDX-Daten von NetScaler für die Problembehandlung und den Support bietet. Director kommuniziert mit dem Controller über TCP-Port 80 oder 443.

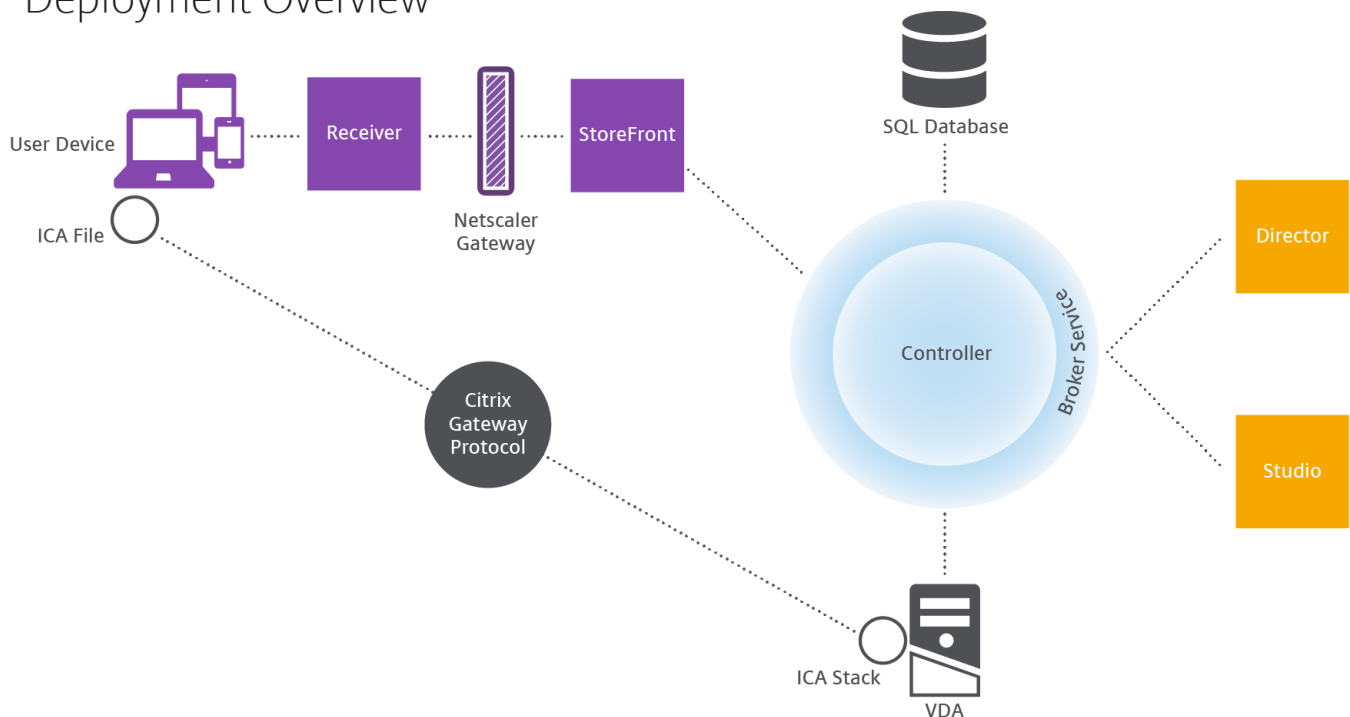
## Studio

Eine Verwaltungskonsole, mit der Administratoren Sites konfigurieren und verwalten und die Zugriff auf Echtzeitdaten vom Broker Agent ermöglicht. Studio kommuniziert mit dem Controller über TCP-Port 80.

## So funktionieren typische Bereitstellungen

XenApp- und XenDesktop-Sites bestehen aus Maschinen mit dedizierten Rollen, die Skalierbarkeit, hohe Verfügbarkeit und Failover gewährleisten und auf diese Weise eine inhärent sichere Lösung bieten. Eine XenApp- oder XenDesktop-Site besteht aus Windows-Server- und Desktopmaschinen mit installierten VDAs und dem Delivery Controller, der den Zugriff verwaltet.

## Deployment Overview





Durch den VDA können Benutzer Verbindungen mit Desktops und Anwendungen herstellen. Er ist auf Server- oder Desktopmaschinen im Datacenter für die meisten Bereitstellungsmethoden installiert, aber er kann auch auf physischen PCs für Remote-PC-Zugriff installiert werden.

Der Controller besteht aus unabhängigen Windows-Diensten, die Ressourcen, Anwendungen und Desktops verwalten und die Last der Benutzerverbindungen optimieren und ausgleichen. Jede Site hat einen oder mehrere Controller. Da Sitzungen von der Latenz, Bandbreite und Netzwerkzuverlässigkeit abhängig sind, sollten alle Controller idealerweise im gleichen LAN sein.

Benutzer greifen niemals direkt auf den Controller zu. Der VDA dient als Vermittler zwischen den Benutzern und dem Controller. Wenn Benutzer sich über StoreFront an der Site anmelden, werden ihre Anmeldeinformationen an den Brokerdienst weitergeleitet. Dieser ruft, basierend auf den festgelegten Richtlinien, die Profile der Benutzer und die für sie verfügbaren Ressourcen ab.

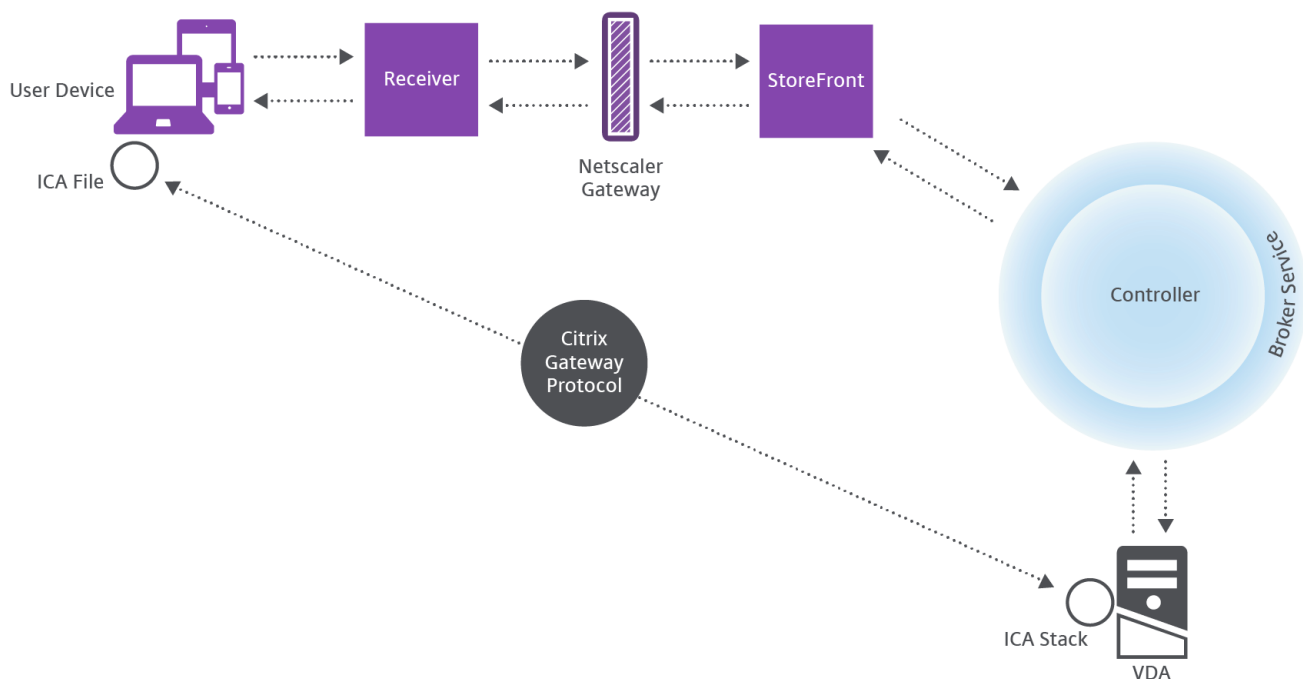
### Behandlung von Benutzerverbindungen

Zum Starten einer XenApp- oder XenDesktop-Sitzung stellt der Benutzer eine Verbindung über Citrix Receiver her (auf dem Benutzergerät installiert), oder über Receiver für Web (RFW).

In Receiver wählt der Benutzer den gewünschten physischen oder virtuellen Desktop oder die gewünschte virtuelle Anwendung.

Die Anmeldeinformationen des Benutzers werden über diesen Weg an den Controller geleitet, der durch Kommunikation mit dem Brokerdienst bestimmt, welche Ressourcen benötigt werden. Administratoren wird empfohlen, ein SSL-Zertifikat auf StoreFront zu installieren, sodass die von Receiver kommenden Anmeldeinformationen verschlüsselt werden.

### User connections



Der Brokerdienst bestimmt, auf welche Desktops und Anwendungen der Benutzer zugreifen kann.

Wenn die Anmeldeinformationen geprüft wurden, werden die Informationen zu verfügbaren Apps und Desktops über die

StoreFront-Receiver-Route an den Benutzer gesendet. Wenn der Benutzer Anwendungen oder Desktops aus dieser Liste auswählt, werden diese Informationen wieder an den Controller geleitet, der den richtigen VDA zum Hosten der einzelnen Anwendungen oder Desktops bestimmt.

Der Controller sendet eine Nachricht mit den Anmeldeinformationen des Benutzers sowie alle Daten zu dem Benutzer und der Verbindung an den VDA. Der VDA akzeptiert die Verbindung und sendet die Informationen über die gleiche Route an den Receiver zurück. Receiver bündelt die gesamten Informationen, die während der Sitzung generiert wurden, und erstellt eine ICA-Datei (Independent Computing Architecture) auf dem Benutzergerät, wenn Receiver lokal installiert ist, oder auf dem RFW, wenn der Zugriff über das Internet erfolgt. Wenn die Site ordnungsgemäß eingerichtet wurde, sind die Anmeldeinformationen während des gesamten Vorgangs verschlüsselt.

Die ICA-Datei wird auf das Benutzergerät kopiert und richtet eine direkte Verbindung zwischen dem Gerät und dem auf dem VDA ausgeführten ICA-Stack ein. Diese Verbindung umgeht die Verwaltungsinfrastruktur von Receiver, StoreFront und Controller.

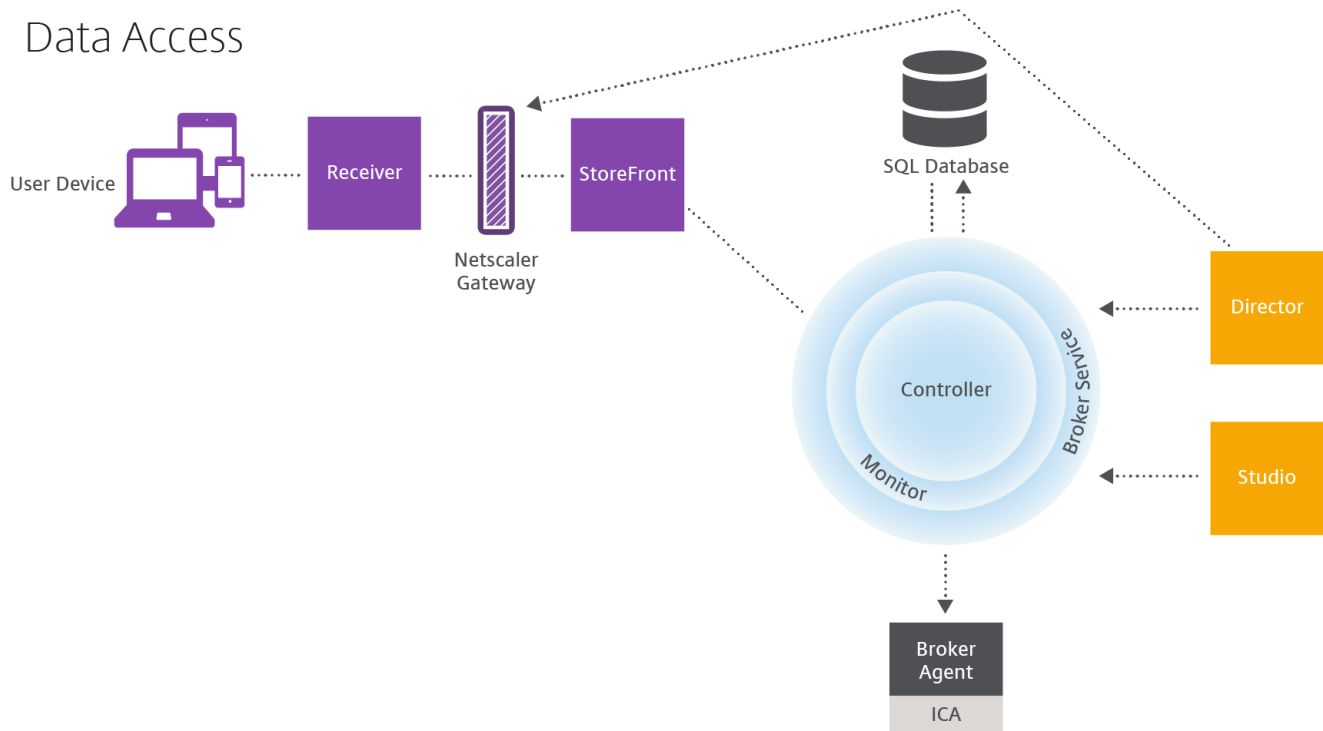
Die Verbindung zwischen Receiver und dem VDA verwendet das Citrix Gateway Protocol (CGP). Wenn eine Verbindung unterbrochen wird, kann der Benutzer bei aktivierter Sitzungszuverlässigkeit die Verbindung zum VDA wieder herstellen und muss sich nicht über die Verwaltungsinfrastruktur erneut anmelden. Die Sitzungszuverlässigkeit kann in Studio aktiviert oder deaktiviert werden.

Wenn der Client eine Verbindung zum VDA hergestellt hat, benachrichtigt der VDA den Controller, dass der Benutzer angemeldet ist, und der Controller sendet diese Informationen an die Sitedatenbank und beginnt mit der Protokollierung von Daten in der Überwachungsdatenbank.

### So funktioniert der Datenzugriff

Jede XenApp- oder XenDesktop-Sitzung produziert Daten, auf die die IT-Mitarbeiter über Studio oder Director zugreifen können. Mit Studio können Administratoren auf Echtzeitdaten aus dem Broker Agent zugreifen und damit Sites besser verwalten. Director bietet Zugriff auf dieselben Echtzeitdaten und historischen Daten, die in der Überwachungsdatenbank gespeichert sind, sowie auf HDX-Daten von NetScaler Gateway, die dem Helpdesk zum Support und zur Problembehandlung dienen.

## Data Access



Innerhalb des Controllers gibt der Brokerdienst Sitzungsdaten für jede Sitzung auf der virtuellen Maschine als Echtzeitdaten zurück. Der Überwachungsdienst erfasst ebenfalls die Echtzeitdaten und speichert sie als historische Daten in der Überwachungsdatenbank.

Studio kann nur mit dem Brokerdienst kommunizieren und hat daher lediglich Zugriff auf Echtzeitdaten. Director kommuniziert mit dem Brokerdienst (über ein Plug-In im Broker Agent), um auf die Sitedatenbank zuzugreifen.

Director kann zudem auf NetScaler Gateway zugreifen und Informationen zu HDX-Daten abrufen.

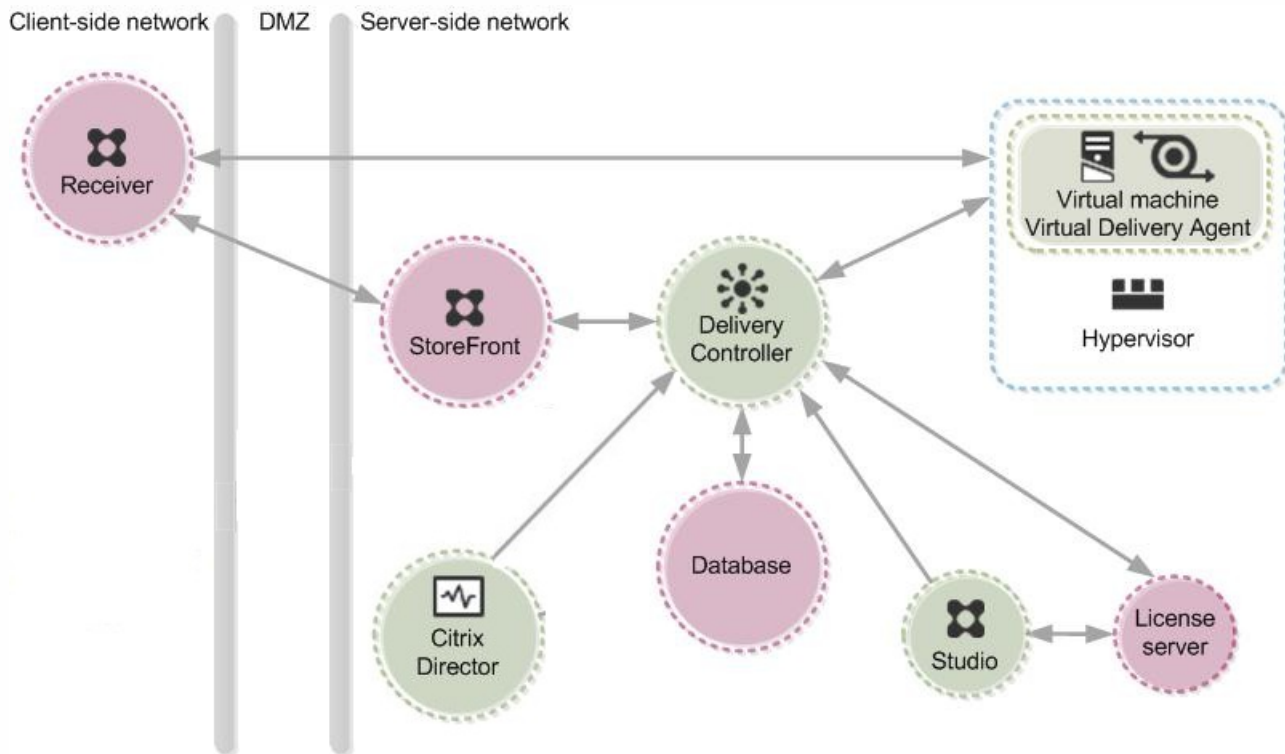
### Verwandter Inhalt

- [Konzepte und Komponenten](#)
- [Active Directory](#)
- [Fehlertoleranz](#)
- [Bereitstellungsmethoden](#)

# Konzepte und Komponenten

Jul 13, 2016

Diese Abbildung zeigt die wichtigsten Komponenten in einer typischen XenApp- oder XenDesktop-Bereitstellung, die als "Site" bezeichnet wird.



Die Abbildung enthält folgende Komponenten:

- **Delivery Controller:** Der Delivery Controller ist die zentrale Verwaltungskomponente einer XenApp- oder XenDesktop-Site. Jede Site hat einen oder mehrere Delivery Controller. Er muss auf mindestens einem Server im Datacenter installiert sein. (Um die Zuverlässigkeit und Verfügbarkeit der Site zu gewährleisten, installieren Sie den Controller auf mehreren Servern.) Der Controller besteht aus Diensten, die mit dem Hypervisor kommunizieren, um Anwendungen und Desktops zu verteilen, den Benutzerzugriff zu authentifizieren und zu verwalten, Verbindungen zwischen Benutzern und ihren virtuellen Desktops und Anwendungen zu vermitteln, Benutzerverbindungen zu optimieren und einen Lastausgleich für die Verbindungen auszuführen.

Die Daten dieser Dienste werden in der Sitedatenbank gespeichert.

Der Controller verwaltet den Zustand der Desktops, startet und hält sie basierend auf dem Bedarf und der administrativen Konfiguration an. In bestimmten Editionen ermöglicht der Controller die Installation der Profilverwaltung, mit der Sie personalisierte Einstellungen in virtualisierten oder physischen Windows-Umgebungen verwalten.

- **Datenbank:** Mindestens eine Microsoft SQL Server-Datenbank ist pro XenApp- oder XenDesktop-Site zum Speichern der Konfigurations- und Sitzungsinformationen erforderlich. Diese Datenbank speichert die Daten, die von den Diensten des Controllers gesammelt und verwaltet werden. Installieren Sie die Datenbank in Ihrem Datacenter und stellen Sie eine persistente Verbindung mit dem Controller sicher.
- **Virtual Delivery Agent (VDA):** Der VDA ist auf jeder physischen oder virtuellen Maschine der Site installiert, die Sie

Benutzern zur Verfügung stellen möchten. Durch ihn kann sich die Maschine beim Controller registrieren, sodass die Maschine und die darauf gehosteten Ressourcen Benutzern verfügbar gemacht werden können. VDAs erstellen und verwalten die Verbindung zwischen der Maschine und dem Benutzergerät, prüfen, ob eine Citrix Lizenz für den Benutzer oder die Sitzung verfügbar ist und wenden die für die Sitzung konfigurierten Richtlinien an. Der VDA übermittelt über den Broker Agent Sitzungsinformationen an den Brokerdienst auf dem Controller.

XenApp und XenDesktop enthalten VDAs für Windows-Server- und Windows-Desktopbetriebssysteme. Mit VDAs für Windows-Serverbetriebssysteme können mehrere Benutzer gleichzeitig eine Verbindung mit dem Server herstellen. Mit VDAs für Windows-Desktops kann jeweils nur ein Benutzer eine Verbindung zum Desktop herstellen.

- **StoreFront:** StoreFront authentifiziert Benutzer an Sites, die Ressourcen hosten und Stores von Desktops und Anwendungen verwalten, auf die Benutzer zugreifen. StoreFront hostet den Unternehmensanwendungsspeicher, über den Sie Benutzern Self-Service-Zugriff auf Desktops und Anwendungen gewähren, die Sie verfügbar machen. Außerdem verfolgt es die Anwendungsabonnements von Benutzern sowie Verknüpfungsdaten und andere Daten, um eine konsistente Benutzererfahrung über mehrere Geräte sicherzustellen.
- **Receiver:** Citrix Receiver wird auf Benutzergeräten und anderen Endpunkten installiert, u. a. virtuellen Desktops, und bietet Benutzern schnellen, sicheren Self-Service-Zugriff auf Dokumente, Anwendungen und Desktops von allen Geräten des Benutzers, u. a. Smartphones, Tablets und PCs. Receiver bietet bedarfsgesteuerten Zugriff auf Windows-, Web- und SaaS-Anwendungen. Bei Geräten, auf denen die Receiver-Software nicht installiert werden kann, ermöglicht Receiver für HTML5 eine Verbindung über einen HTML5-kompatiblen Webbrowser.
- **Studio:** Dies ist die Verwaltungskonsolle, mit der Sie die Bereitstellung konfigurieren und verwalten, wobei keine separaten Verwaltungskonsolen für die Bereitstellung von Anwendungen und Desktops benötigt werden. Studio bietet zahlreiche Assistenten, die Ihnen bei der Einrichtung Ihrer Umgebung, dem Erstellen der Arbeitslast zum Hosten von Anwendungen und Desktops sowie beim Zuweisen von Anwendungen und Desktops zu Benutzern behilflich sind. Sie können mit Studio auch Citrix Lizenzen für die Site zuweisen und verfolgen. Studio erhält die Informationen, die es anzeigt, vom Brokerdienst auf dem Controller.

- **Director:** Ein webbasiertes Tool, mit dem die Support- und Helpdesk-Teams eine XenDesktop-Umgebung überwachen, potenziell systembedrohende Probleme rechtzeitig behandeln und Unterstützung für Endbenutzer leisten können. Sie können mit einer Director-Bereitstellung Verbindungen zu mehreren XenApp- oder XenDesktop-Sites herstellen und diese überwachen.

Director zeigt Sitzungs- und Siteinformationen von den folgenden Quellen an:

- Echtzeit-Sitzungsdaten vom Brokerdienst auf dem Controller, einschließlich Daten, die der Brokerdienst vom Broker Agent auf dem VDA erhält.
- Historische Daten für die Site vom Überwachungsdienst auf dem Controller.
- Daten zum HDX-Datenverkehr (auch ICA-Datenverkehr genannt), die von HDX Insight auf dem NetScaler erfasst werden, wenn die Bereitstellung einen NetScaler und die XenApp- oder XenDesktop-Edition HDX Insights enthält.

Zudem können Sie über Microsoft Remoteunterstützung auch Benutzersitzungen anzeigen und steuern.

- **Lizenzserver:** Der Lizenzserver verwaltet die Produktlizenzen. Er kommuniziert mit dem Controller, um die Lizenzierung jeder Benutzersitzung zu verwalten, und mit Studio, um Lizenzdateien zuzuteilen. Sie müssen mindestens einen Lizenzserver zum Speichern und Verwalten Ihrer Lizenzdateien erstellen.
- **Hypervisor:** Der Hypervisor hostet die virtuellen Maschinen in der Site. Dies können virtuellen Maschinen sein, die Sie zum Hosten von Anwendungen und Desktops verwenden, und auch virtuellen Maschinen, die Sie zum Hosten von XenApp- und XenDesktop-Komponenten verwenden. Ein Hypervisor wird auf einem Hostcomputer installiert, der nur zur Ausführung des Hypervisors und dem Hosten virtueller Maschinen bestimmt ist. Der Citrix XenServer-Hypervisor ist in XenApp und XenDesktop enthalten, aber Sie können auch andere unterstützte Hypervisoren verwenden, z. B. Microsoft Hyper-V oder VMware vSphere.

Obwohl für die meisten Implementierungen von XenApp und XenDesktop ein Hypervisor erforderlich ist, benötigen Sie keinen für Remote-PC-Zugriff oder wenn Sie Provisioning Services (in bestimmten Editionen von XenApp und XenDesktop enthalten) statt MCS zum Bereitstellen von virtuellen Maschinen verwenden.

Diese zusätzlichen Komponenten, die nicht in der Abbildung oben enthalten sind, können auch in typischen XenApp- oder XenDesktop-Bereitstellungen enthalten sein:

- **Provisioning Services:** Provisioning Services ist eine optionale Komponente von XenApp und XenDesktop und steht in einigen Editionen zur Verfügung. Es bietet eine Alternative zu MCS für das Provisioning von virtuellen Maschinen. Während MCS Kopien eines Masterimages erstellt, streamt Provisioning Services das Masterimage zum Benutzergerät. Provisioning Services benötigen hierfür keinen Hypervisor, daher können Sie mit Provisioning Services physische Maschinen hosten. Wenn Provisioning Services in einer Site installiert ist, kommuniziert es mit dem Controller, um Benutzern Ressourcen bereitzustellen.
- **NetScaler Gateway:** Wenn Benutzer eine Verbindung von außerhalb der Unternehmensfirewall herstellen, können diese Verbindungen in diesem Release mit Citrix NetScaler Gateway (früher Access Gateway) mit SSL gesichert werden. NetScaler Gateway oder NetScaler VPX virtuelles Gerät ist ein SSL-VPN-Gerät, das in der DMZ (demilitarisierten Zone) bereitgestellt wird und sicheren, zentralen Zugriff über die Unternehmensfirewall bietet.
- **Citrix CloudBridge:** In Bereitstellungen, in denen virtuelle Desktops den Benutzern an Remotestandorten, wie Zweigstellen, bereitgestellt werden, kann mit Citrix CloudBridge (früher Citrix Branch Repeater oder WANScaler) die Leistung optimiert werden. Repeater erhöhen die Leistung in WAN-Netzwerken. Mit Repeatern im Netzwerk erleben Benutzer in Zweigstellen eine LAN-ähnliche Leistung über das WAN. CloudBridge kann bestimmten Teilen der Benutzererfahrung Priorität geben, damit sich beispielsweise die Benutzererfahrung in der Zweigstelle nicht verschlechtert, wenn eine große Datei oder ein großer Druckauftrag über das Netz gesendet wird. HDX WAN-Optimierung mit CloudBridge bietet Komprimierung mit Token sowie Entfernung von Datenduplikaten, wodurch die Bandbreitenanforderungen drastisch reduziert werden und die Leistung verbessert wird. Weitere Informationen finden Sie in der Dokumentation für Citrix CloudBridge.

## Einrichten und Zuweisen von Ressourcen: Maschinenkataloge und Bereitstellungsgruppen

Mit XenApp und XenDesktop richten Sie die Ressourcen, die Sie Benutzern bereitstellen möchten, mit Maschinenkatalogen ein; welche Benutzer Zugriff auf diese Ressourcen haben, wird mit Bereitstellungsgruppen festgelegt.

### Maschinenkataloge

Maschinenkataloge sind Sammlungen virtueller oder physischer Maschinen, die Sie als Einheit verwalten. Diese Maschinen und die Anwendungen oder virtuellen Desktops darauf sind die Ressourcen, die Sie den Benutzer bereitstellen. Auf allen Maschinen in einem Maschinenkatalog ist das gleiche Betriebssystem und der gleiche Virtual Desktop Agent (VDA) installiert. Sie verfügen auch über dieselben Anwendungen oder virtuellen Desktops. Normalerweise erstellen Sie ein Masterimage und verwenden es zum Erstellen von identischen virtuellen Maschinen im Katalog.

Beim Erstellen eines Maschinenkatalogs geben Sie den Maschinentyp und die Provisioningmethode für die Maschinen in diesem Katalog an.

### Maschinentypen

- **Windows-Serverbetriebssystemmaschinen:** Virtuelle oder physische Maschinen, die auf einem Windows-Serverbetriebssystem basieren und die mit XenApp veröffentlichte Apps (serverbasierte, gehostete Anwendungen) und mit XenApp veröffentlichte Desktops (servergehostete Desktops) bereitstellen. Mehrere Benutzer können gleichzeitig eine Verbindung mit diesen Maschinen herstellen.
- **Desktopbetriebssystemmaschinen:** Virtuelle oder physische Maschinen, die auf einem Windows-Desktopbetriebssystem basieren und die VDI-Desktops (Desktops, auf denen Windows-Desktopbetriebssysteme ausgeführt werden, die abhängig von den ausgewählten Optionen vollständig personalisiert werden können), VM-gehostete Apps (Anwendungen von Desktopbetriebssystemen) und gehostete physische Desktops bereitstellen. Nur jeweils ein Benutzer

kann eine Verbindung zu einem dieser Desktops herstellen.

- Remote-PC-Zugriff: Wenn Benutzergeräte auf einer Positivliste enthalten sind, können die Benutzer von einem beliebigen Gerät aus, auf dem Citrix Receiver ausgeführt wird, auf ihre Büro-PCs zugreifen. Remote-PC-Zugriff ermöglicht das Verwalten des Zugriffs auf Büro-PCs über die XenDesktop-Bereitstellung.

### Provisioningmethoden

- Maschinenerstellungsdienste (MCS): Eine Sammlung von Diensten, die bei Bedarf virtuelle Server und Desktops von einem Masterimage erstellen, die Speichernutzung optimieren und den Benutzern bei jeder Anmeldung eine virtuelle Maschine bereitstellen. Die Maschinenerstellungsdienste sind vollständig in Citrix Studio integriert und werden dort verwaltet.
- Provisioning Services: Ermöglicht ein Provisioning und erneutes Provisioning von Computern in Echtzeit von einem freigegebenen Datenträgerimage. Zielgeräte werden von Provisioning Services als eine Gerätesammlung verwaltet. Die Desktops und Anwendungen werden von einer Provisioning Services-vDisk bereitgestellt, deren Image von einem Masterzielgerät erstellt wurde. So können Sie die Verarbeitungsleistung der physischen Hardware oder der virtuellen Maschinen nutzen. Provisioning Services werden in einer eigenen Konsole verwaltet.
- Vorhandene Images: Bezieht sich auf Desktops und Anwendungen, die Sie bereits auf virtuelle Maschinen im Datacenter migriert haben. Sie müssen die Zielgeräte individuell oder kollektiv mit ESD-Tools (Electronic Software Distribution) verwalten.

### Bereitstellungsgruppen

Bereitstellungsgruppen sind Sammlungen von Benutzern, die Zugriff auf eine gemeinsame Gruppe von Ressourcen erhalten. Bereitstellungsgruppen enthalten Maschinen von Ihren Maschinenkatalogen und Active Directory-Benutzer, die Zugriff auf die Site haben. Oft ist es sinnvoll, Benutzer den Bereitstellungsgruppen mit ihrer Active Directory-Gruppe zuzuweisen, da sowohl Active Directory-Gruppen als auch Bereitstellungsgruppen Methoden sind, um Benutzer mit ähnlichen Anforderungen zu gruppieren.

Jede Bereitstellungsgruppe kann Maschinen aus mehreren Maschinenkatalogen enthalten und jeder Maschinenkatalog kann Maschinen für mehrere Bereitstellungsgruppen beitragen, aber eine Maschine kann jeweils nur zu einer Bereitstellungsgruppe gehören. Sie können über eine Bereitstellungsgruppe Anwendungen, Desktops oder beides bereitstellen.

Sie definieren, auf welche Ressourcen Benutzer in der Bereitstellungsgruppe zugreifen können. Beispiel: Sie möchten verschiedene Anwendungen verschiedenen Benutzern bereitstellen. Eine Methode ist die Installation aller bereitzustellenden Anwendungen auf dem Masterimage für einen Maschinenkatalog. Dann erstellen Sie in diesem Katalog genug Maschinen, um sie auf mehrere Bereitstellungsgruppen zu verteilen. Anschließend konfigurieren Sie jede Bereitstellungsgruppe so, dass sie einen anderen Teil der auf den Maschinen installierten Anwendungen bereitstellt.

### XenApp und XenDesktop 7.6 unterscheiden sich von XenApp 6.5 und früheren Versionen

Wenn Sie mit XenApp 6.5 und Vorversionen von XenApp vertraut sind, kann es nützlich sein, die Unterschiede zwischen XenApp 7.6 und XenDesktop 7.6 und den Vorversionen zu verstehen.

Obwohl die funktionellen Elemente nicht genau gleich sind, erleichtert die folgende Tabelle das Zuordnen der funktionellen Elemente von XenApp 6.5 und Vorversionen zu XenApp 7.6 und XenDesktop 7.6:

| XenApp 6.5 und Vorversionen               | XenApp und XenDesktop 7.6              |
|-------------------------------------------|----------------------------------------|
| Independent Management Architecture (IMA) | FlexCast Management Architecture (FMA) |
| Farm                                      | Site                                   |



| Workergruppe<br>XenApp 6.5 und Vorversionen                  | Maschinenkatalog<br>XenApp und XenDesktop 7.6                                                                                                                            |
|--------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                                              | Bereitstellungsgruppe                                                                                                                                                    |
| Worker                                                       | Virtual Delivery Agent (VDA)<br><br>Serverbetriebssystemmaschine, VDA für<br>Serverbetriebssystem<br><br>Desktopbetriebssystemmaschine, VDA für<br>Desktopbetriebssystem |
| Remotedesktopdienste (RDS) oder Terminaldienste-<br>Maschine | Serverbetriebssystemmaschine, VDA für<br>Serverbetriebssystem                                                                                                            |
| Zonen- und Datensammelpunkt                                  | Delivery Controller                                                                                                                                                      |
| Delivery Services Console                                    | Citrix Studio und Citrix Director                                                                                                                                        |
| Veröffentlichen von Anwendungen                              | Bereitstellen von Anwendungen                                                                                                                                            |
| Datenspeicher                                                | Datenbank                                                                                                                                                                |
| Lastauswertungsprogramm                                      | Lastverwaltungsrichtlinie                                                                                                                                                |
| Administrator                                                | Delegierter Administrator<br><br>Rolle<br><br>Geltungsbereich                                                                                                            |

XenApp 7.6 und XenDesktop 7.6 basieren auf der FlexCast Management Architecture (FMA). Die FMA ist eine dienstorientierte Architektur, die über Citrix Technologien übergreifend Interoperabilität und Verwaltungsmodularität ermöglicht. Die FMA bietet eine Plattform für die Anwendungsbereitstellung, Mobilität, Dienste, flexible Bereitstellung und Cloudverwaltung.

Die FMA ersetzt die Independent Management Architecture (IMA) in XenApp 6.5 und Vorversionen.

Dies sind die Hauptelemente der FMA gegenüber den Elementen von XenApp 6.5 und Vorversionen:

### **Bereitstellungssites**

Farmen waren die Objekte der obersten Ebene in XenApp 6.5 und Vorversionen. In XenApp 7.6 und XenDesktop 7.6 ist dies die Bereitstellungssite. Über Sites werden Benutzergruppen Anwendungen und Desktops angeboten.

Die FMA erfordert, dass Sie in einer Domäne sind, um eine Site bereitzustellen. Beispiel: Zum Installieren der Server muss Ihr Konto lokale Administratorrechte haben und in Active Directory ein Domänenbenutzer sein.



## **Maschinenkataloge und Bereitstellungsgruppen**

Maschinen, auf denen Anwendungen gehostet werden, gehörten in XenApp 6.5 und Vorversionen zu Workergruppen, um eine effiziente Verwaltung von Anwendungen und Serversoftware zu ermöglichen. Administratoren konnten für die Anwendungsverwaltung und für den Lastausgleich alle Maschinen in einer Workergruppe als eine Einheit behandeln. Ordner wurden verwendet, um Anwendungen und Maschinen zu organisieren.

In XenApp 7.6 und XenDesktop 7.6 verwenden Sie eine Kombination aus Maschinenkatalogen und Bereitstellungsgruppen, um Maschinen, den Lastausgleich und gehostete Anwendungen oder Desktops zu verwalten.

### **Virtual Delivery Agents**

In XenApp 6.5 und Vorversionen wurden auf Maschinen in Workergruppen Anwendungen für die Benutzer ausgeführt und die Maschinen kommunizierten mit Datensammelpunkten. In XenApp 7.6 und XenDesktop 7.6 kommuniziert der VDA mit Delivery Controllern, die die Benutzerverbindungen verwalten.

### **Delivery Controller**

In XenApp 6.5 und Vorversionen war ein Zonenmaster für Verbindungsanfragen von Benutzern und die Kommunikation mit Hypervisoren zuständig. In XenApp 7.6 und XenDesktop 7.6 verteilen und handhaben Controller in der Site Verbindungsanfragen.

XenApp 6.5 und Vorversionen ermöglichten Zonen das Aggregieren von Servern und Replizieren von Daten über WAN-Verbindungen. In XenApp 7.6 und XenDesktop 7.6 gibt es zwar keine exakte Entsprechung für Zonen, dennoch können Sie Benutzern WAN- und standortübergreifende Anwendungen bereitstellen. Sie können Bereitstellungssites für bestimmte geografische Standorte oder Datacenter erstellen und dann den Benutzern Zugriff auf mehrere Bereitstellungssites gewähren. App Orchestration mit XenApp 7.6 und XenDesktop 7.6 enthält Funktionen zum Verwalten mehrerer Sites in mehreren geografischen Bereichen.

### **Citrix Studio und Citrix Director**

Mit der Studio-Konsole können Sie Umgebungen konfigurieren und Benutzern Zugriff auf Anwendungen und Desktops gewähren. Studio ersetzt die Delivery Services Console in XenApp 6.5 und Vorversionen.

Administratoren verwenden Director, um die Umgebung zu überwachen, Benutzergeräte zu spiegeln und IT-Probleme zu behandeln. Zum Spiegeln von Benutzern muss Microsoft-Remoteunterstützung aktiviert sein; es ist standardmäßig aktiviert, wenn der VDA installiert ist.

### **Bereitstellen von Anwendungen**

In XenApp 6.5 und Vorversionen wurden mit dem Assistenten zur Anwendungsveröffentlichung Anwendungen vorbereitet und für Benutzer bereitgestellt. In XenApp 7.6 und XenDesktop 7.6 erstellen und fügen Sie Anwendungen mit Studio hinzu und stellen Sie den Benutzern in einer Bereitstellungsgruppe zur Verfügung. In Studio konfigurieren Sie zunächst eine Site, erstellen und geben Maschinenkataloge an und dann erstellen Sie Bereitstellungsgruppen in diesen Maschinenkatalogen. Mit Bereitstellungsgruppen wird festgelegt, welche Benutzer Zugriff auf die bereitgestellten Anwendungen haben.

### **Datenbank**

XenApp 7.6 und XenDesktop 7.6 speichern die Konfigurationsinformationen nicht im IMA-Datenspeicher. Stattdessen werden Konfigurations- und Sitzungsinformationen in einer Microsoft SQL Server-Datenbank gespeichert.

### **Lastverwaltungsrichtlinie**

In XenApp 6.5 und Vorversionen verwendeten Lastauswertungsprogramme vordefinierte Messungen zur Bestimmung der Last auf einer Maschine. Benutzerverbindungen konnten weniger ausgelasteten Maschinen zugeordnet werden.

In XenApp 7.6 und XenDesktop 7.6 erfolgt der Lastausgleich auf Maschinen mit Lastverwaltungsrichtlinien.

### **Delegierte Administratoren**

In XenApp 6.5 und Vorversionen wurden benutzerdefinierte Administratoren erstellt und ihnen wurden Berechtigungen auf der Basis von Ordnern und Objekten zugewiesen. In XenApp 7.6 und XenDesktop 7.6 basieren benutzerdefinierte Administratoren auf Rollen- und Geltungsbereichspaaren. Eine Rolle repräsentiert eine Stellenfunktion. Ihr sind definierte Berechtigungen zugewiesen, die eine Delegation erlauben. Ein Geltungsbereich steht für eine Sammlung von Objekten.

Vordefinierte Administratorrollen haben spezifische Berechtigungssätze, z. B. für Helpdesk, Anwendungen, Hosting und Kataloge. Beispiel: Helpdeskadministratoren können nur mit einzelnen Benutzern in bestimmten Sites arbeiten, während Volladministratoren die gesamte Bereitstellung überwachen und systemweite IT-Probleme behandeln können.

Der Übergang zu FMA bedeutet außerdem, dass einige in XenApp 6.5 und Vorversionen verwendete Features möglicherweise anders implementiert worden sind oder dass Sie sie eventuell durch andere Features, Komponenten oder Tools ersetzen müssen, um das gleiche Ziel zu erreichen.

| XenApp 6.5 und Vorversionen                                                                                                                                                          | XenApp und XenDesktop 7.6                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Sitzungsvorabstart und Sitzungsfortbestehen, die mit Richtlinieneinstellungen konfiguriert sind                                                                                      | <p>Sitzungsvorabstart und Sitzungsfortbestehen, die durch Bearbeiten der Bereitstellungsgruppeneinstellungen konfiguriert sind.</p> <p>Wie in XenApp 6.5 ermöglichen diese Features Benutzern einen schnellen Zugriff auf Anwendungen, indem Sitzungen gestartet werden, bevor sie angefordert werden (Sitzungsvorabstart), und aktiv bleiben, nachdem ein Benutzer alle Anwendungen geschlossen hat (Sitzungsfortbestehen). In XenApp und XenDesktop 7.6 aktivieren Sie die Features für bestimmte Benutzer, indem Sie diese Einstellungen für vorhandene Bereitstellungsgruppen konfigurieren. Sie <a href="#">Konfigurieren des Vorabstarts und des Fortbestehens von Sitzungen</a>.</p> |
| Unterstützung für nicht authentifizierte (anonyme) Benutzer erfolgt durch Zuweisen von Rechten für anonyme Benutzer beim Festlegen der Eigenschaften für veröffentlichte Anwendungen | Unterstützung für nicht authentifizierte (anonyme) Benutzer erfolgt durch Konfigurieren dieser Option, wenn Sie die Benutzereigenschaften einer Bereitstellungsgruppe festlegen. Siehe <a href="#">Benutzer</a> .                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| Lokaler Hostcache ermöglicht das Funktionieren eines Workerservers, selbst wenn eine Verbindung zum Datenspeicher nicht verfügbar ist                                                | Verbindungsleasing ermöglicht Benutzern die Wiederverbindung mit den zuletzt verwendeten Anwendungen und Desktops, selbst wenn die Sitedatenbank nicht verfügbar ist. Das Feature für Verbindungsleasing ergänzt die bewährten Methoden zum Bereitstellen hoher Verfügbarkeit bei SQL Server. Siehe <a href="#">Verbindungsleasing</a> .                                                                                                                                                                                                                                                                                                                                                    |
| Anwendungsstreaming                                                                                                                                                                  | App-V stellt gestreamte Anwendungen bereit, die mit Studio verwaltet werden.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| Webinterface                                                                                                                                                                         | Citrix empfiehlt, dass Sie StoreFront einsetzen.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| SmartAuditor                                                                                                                                                                         | Mit der Konfigurationsprotokollierung werden alle Sitzungsaktivitäten aus einer administrativen Perspektive erfasst. Sie können auch das von Citrix unterstützte Tool eines Drittanbieters zum Aufzeichnen von Sitzungen verwenden.                                                                                                                                                                                                                                                                                                                                                                                                                                                         |



# Active Directory

May 10, 2016

Active Directory ist zum Authentifizieren und Autorisieren erforderlich. Mit der Kerberos-Infrastruktur in Active Directory wird die Authentizität und Vertraulichkeit der Kommunikation zwischen den Delivery Controllern garantiert. Informationen zu Kerberos finden Sie in der Dokumentation von Microsoft.

Das Dokument [Systemanforderungen](#) enthält die unterstützten Funktionsebenen für Gesamtstruktur und Domäne. Zur Verwendung der Richtlinienmodellierung muss der Controller unter Windows 2003 oder höher bis Windows Server 2012 R2 ausgeführt werden. Dies wirkt sich nicht auf die Domänenfunktionsebene aus.

Dieses Produkt unterstützt Folgendes:

- Bereitstellungen, in denen die Benutzerkonten und Computerkonten in Domänen in einer einzigen Active Directory-Gesamtstruktur bestehen. Benutzer- und Computerkonten können in beliebigen Domänen in einer Gesamtstruktur bestehen. Alle Domänen- und Gesamtstrukturebenen werden in diesem Bereitstellungstyp unterstützt.
- Bereitstellungen, in denen die Benutzerkonten und die Computerkonten der Controller und virtuellen Desktops in unterschiedlichen Active Directory-Gesamtstrukturen bestehen. Bei diesem Bereitstellungstyp muss eine Vertrauensstellung zwischen den Domänen mit den Computerkonten der Controller und virtuellen Desktops und den Domänen mit den Benutzerkonten bestehen. Sie können Gesamtstruktur- oder externe Vertrauensstellungen verwenden. Alle Domänen- und Gesamtstrukturebenen werden in diesem Bereitstellungstyp unterstützt.
- Bereitstellungen, in denen die Computerkonten für Controller in einer Active Directory-Gesamtstruktur bestehen, die sich von den zusätzlichen Active Directory-Gesamtstrukturen mit den Computerkonten für die virtuellen Desktops unterscheidet. Bei diesem Bereitstellungstyp muss eine bidirektionale Vertrauensstellung zwischen den Domänen mit den Computerkonten der Controller und allen Domänen mit den Computerkonten der virtuellen Desktops bestehen. Bei diesem Bereitstellungstyp müssen alle Domänen mit Computerkonten für Controller oder virtuelle Desktops mindestens auf der Funktionsebene "Windows 2000 native" sein. Alle Funktionsebenen der Gesamtstruktur werden unterstützt.
- Beschreibbarer Domänencontroller. Schreibgeschützte Domänencontroller werden nicht unterstützt.

Virtual Delivery Agents (VDAs) können mit in Active Directory veröffentlichten Informationen die Controller ermitteln, bei denen sie sich registrieren können (Discovery). Diese Methode wird primär für Abwärtskompatibilität unterstützt und ist nur verfügbar, wenn die VDAs und die Controller in derselben Active Directory-Gesamtstruktur sind. Informationen über diese Discovery-Methode finden Sie unter [Active Directory-basierte Controller-Discovery](#) und [CTX118976](#).

## Bereitstellen in einer Active Directory-Umgebung mit mehreren Gesamtstrukturen

Hinweis: Diese Informationen gelten für Versionen ab XenDesktop 7.1 und XenApp 7.5. Sie gelten nicht für ältere Versionen von XenDesktop und XenApp.

Bei einer Active Directory-Umgebung mit mehreren Gesamtstrukturen und unidirektionalen oder bidirektionalen Vertrauensstellungen können Sie DNS-Weiterleitungen zur Suche und Registrierung von Namen verwenden. Mit dem Assistenten zum Zuweisen der Objektverwaltung können Sie den entsprechenden Active Directory-Benutzern das Erstellen von Computerkonten ermöglichen. Weitere Informationen zu diesem Assistenten finden Sie in der Dokumentation von Microsoft.

In der DNS-Infrastruktur sind keine Reverse-DNS-Zonen erforderlich, wenn die entsprechenden DNS-Weiterleitungen zwischen Gesamtstrukturen eingerichtet sind.

Der SupportMultipleForest-Schlüssel ist erforderlich, wenn der VDA und der Controller in unterschiedlichen Gesamtstrukturen eingerichtet sind, unabhängig davon, ob sich die Active Directory- und NetBIOS-Namen voneinander

unterscheiden. Der Schlüssel "SupportMultipleForest" ist nur auf dem VDA nötig. Mit den folgenden Informationen fügen Sie einen Registrierungsschlüssel hinzu:

Achtung: Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

- HKEY\_LOCAL\_MACHINE\Software\Citrix\VirtualDesktopAgent\SupportMultipleForest
  - Name: SupportMultipleForest
  - Typ: REG\_DWORD
  - Wert: 0x00000001 (1)

Sie müssen möglicherweise die DNS-Konfiguration umkehren, wenn sich der DNS-Namespace vom Active Directory-Namespace unterscheidet.

Wenn externe Vertrauensstellungen während des Setups vorhanden sind, ist der Registrierungsschlüssel "ListOfSIDs" erforderlich. Der Registrierungsschlüssel "ListOfSIDs" ist auch erforderlich, wenn der vollqualifizierte Domänenname (FQDN) für Active Directory sich vom DNS-FQDN unterscheidet oder die Domäne mit dem Domänencontroller einen anderen Netbios-Namen hat als der Active Directory-FQDN. Verwenden Sie zum Hinzufügen des Registrierungsschlüssels die folgenden Informationen:

- Für einen 32-Bit- oder 64-Bit-VDA verwenden Sie folgenden Registrierungsschlüssel:  
HKEY\_LOCAL\_MACHINE\Software\Citrix\VirtualDesktopAgent\ListOfSIDs
  - Name: ListOfSIDs
  - Typ: REG\_SZ
  - Daten: Sicherheits-ID (SID) der Controller

Wenn externe Vertrauensstellungen vorhanden sind, nehmen Sie die folgenden Änderungen auf dem VDA vor:

1. Navigieren Sie zur Datei: \Citrix\Virtual Desktop Agent\brokeragentconfig.exe.config.
2. Erstellen Sie eine Sicherungskopie der Datei.
3. Öffnen Sie die Datei in einem Textbearbeitungsprogramm, z. B. Editor.
4. Suchen Sie den Text allowNtlm="false" und ändern Sie den Text in allowNtlm="true".
5. Speichern Sie die Datei.

Nach dem Hinzufügen des Registrierungsschlüssels "ListOfSIDs" und der Bearbeitung der Datei brokeragent.exe.config starten Sie den Citrix Desktopdienst neu, um die Änderungen anzuwenden.

In der folgenden Tabelle werden die unterstützten Vertrauentypen aufgeführt:

| Vertrauentyp        | Transitivität   | Richtung                          | In diesem Release unterstützt |
|---------------------|-----------------|-----------------------------------|-------------------------------|
| Über-/untergeordnet | Transitiv       | Bidirektional                     | Ja                            |
| Strukturstamm       | Transitiv       | Bidirektional                     | Ja                            |
| Extern              | Nicht transitiv | Unidirektional oder bidirektional | Ja                            |
| Gesamtstruktur      | Transitiv       | Unidirektional oder bidirektional | Ja                            |

| <b>Vertrauentyp</b><br>Shortcut | <b>Transitivität</b><br>Transitiv | <b>Richtung</b><br>Unidirektional oder<br>bidirektional | <b>In diesem Release<br/>unterstützt</b><br>Ja |
|---------------------------------|-----------------------------------|---------------------------------------------------------|------------------------------------------------|
| Bereich                         | Transitiv oder nicht<br>transitiv | Unidirektional oder<br>bidirektional                    | Nein                                           |

Weitere Informationen über komplexe Active Directory-Umgebungen finden Sie unter [CTX134971](#).

# Fehlertoleranz

May 10, 2016

In diesem Dokument wird beschrieben, wie Sie die Fehlertoleranz in einer Bereitstellung erhöhen können, um die Verfügbarkeit essentiell wichtiger Anwendungen und Desktops zu jedem Zeitpunkt sicherzustellen.

## Konfigurieren der Datenbankfehlertoleranz

Alle Informationen werden in der Sitekonfigurationsdatenbank gespeichert; Delivery Controller kommunizieren nur mit der Datenbank und nicht untereinander. Ein Controller kann entfernt oder ausgeschaltet werden, ohne dass dies Auswirkungen auf die anderen Controller in der Site hat. Dies bedeutet jedoch, dass die Sitekonfigurationsdatenbank eine einzelne Fehlerquelle darstellt. Wenn der Datenbankserver ausfällt, funktionieren die bestehenden Verbindungen zu virtuellen Desktops weiter, bis der Benutzer sich entweder abmeldet oder die Verbindung mit dem virtuellen Desktop trennt. Wenn der Datenbankserver nicht erreichbar ist, können keine neuen Verbindungen hergestellt werden.

Citrix empfiehlt, dass Sie regelmäßig ein Backup der Datenbank durchführen, damit diese bei einem Ausfall des Datenbankservers von dem Backup wiederhergestellt werden kann. Außerdem gibt es einige Hochverfügbarkeitslösungen, die Sie berücksichtigen sollten, um automatisches Failover zu gewährleisten:

- **SQL-Spiegelung:** Dies ist die empfohlene Lösung. Das Spiegeln der Datenbank stellt sicher, dass der automatische Failoverprozess in nur wenigen Sekunden stattfindet, falls der aktive Datenbankserver ausfällt. Benutzer werden in der Regel also nicht beeinträchtigt. Diese Methode ist jedoch teurer als andere Lösungen, da vollständigen SQL-Serverlizenzen auf jedem Datenbankserver erforderlich sind. In gespiegelten Umgebungen kann SQL Server Express nicht verwendet werden.
- **Verwenden der Hochverfügbarkeitsfeatures des Hypervisors:** Bei dieser Methode wird die Datenbank als virtuelle Maschine bereitgestellt und die Hochverfügbarkeitsfeatures des Hypervisors werden verwendet. Diese Lösung ist billiger als das Spiegeln, da die bestehende Hypervisorsoftware verwendet wird und Sie zudem SQL Express verwenden können. Der automatische Failoverprozess ist jedoch langsamer, da eine neue Maschine u. U. eine Weile braucht, bis sie gestartet wird, und dadurch auch die Datenbank. Möglicherweise wird also der Dienst für Benutzer unterbrochen.
- **SQL-Clustering:** Mit dieser Technologie von Microsoft können Sie einem Server automatisch erlauben, die Aufgaben und Verantwortlichkeiten eines anderen, fehlgeschlagenen Servers zu übernehmen. Es ist jedoch etwas komplizierter, diese Lösung einzurichten. Zudem ist der automatische Failoverprozess in der Regel langsamer als mit anderen Lösungen, wie z. B. SQL-Spiegelung.
- **"AlwaysOn"-Verfügbarkeitsgruppen** sind eine Lösung für Hochverfügbarkeit und Notfallwiederherstellung für Unternehmen, die mit SQL Server 2012 eingeführt wurde. Damit können Sie die Verfügbarkeit für eine oder mehrere Benutzerdatenbanken maximieren. AlwaysOn-Verfügbarkeitsgruppen erfordern, dass die SQL Server-Instanzen auf Windows Server Failover Clustering-Knoten (WSFC) residieren. Weitere Informationen finden Sie unter [AlwaysOn Availability Groups \(SQL Server\)](#).

Hinweis: Die Installation eines Controllers auf einem Knoten in einer SQL-Clustering- oder SQL-Spiegelungsinstallation wird nicht unterstützt.

## Konfigurieren einer Site für die Verwendung einer gespiegelten Datenbank

Vor dem Erstellen der Site müssen zunächst vom Administrator alle Konfigurationsaufgaben mit den SQL Server-Verwaltungstools durchgeführt werden. Beim Ausführen des Assistenten zum Erstellen der Site werden dann die verbleibenden Aufgaben erledigt.

Bei einer gespiegelten Umgebung werden mindestens zwei SQL-Servermaschinen benötigt (im nachfolgenden Beispiel SQL Server A und SQL Server B). SQL Server Express kann nicht für Prinzipal- oder Spiegeldatenbanken verwendet werden.

Verwenden Sie zum Konfigurieren der SQL-Serverdatenbanken die Microsoft SQL Server-Verwaltungstools:

1. Installieren Sie die SQL-Serversoftware auf SQL-Server A und SQL-Server B
2. Erstellen Sie auf SQL-Server A die Datenbank, die als Prinzipaldatenbank verwendet wird (z. B. myDatabaseMirror).
  - Stellen Sie sicher, dass die Datenbank das vollständige Wiederherstellungsmodell und nicht das einfache Modell verwendet. (Das einfache Modell ist standardmäßig konfiguriert, dabei wird jedoch das Transaktionsprotokoll nicht gesichert.)
  - Verwenden Sie folgende Sortierreihenfolge bei der Erstellung der Datenbank: Latin1\_General\_100\_CI\_AS\_KS festgelegt, wobei Latin1\_General je nach Land variiert, z. B. Japanese\_100\_CI\_AS\_KS). Wenn diese Sortiereinstellung bei der Datenbankerstellung nicht angegeben wird, schlägt die anschließende Erstellung von Dienstschemas in der Datenbank fehl und es wird folgender Fehler (oder ein ähnlicher) angezeigt: "<Dienst>: Schema erfordert eine Datenbank, bei der Groß-/Kleinschreibung nicht beachtet wird." <Dienst> ist der Name des Diensts, dessen Schema erstellt wird.
  - Aktivieren Sie einen Read-Committed Snapshot (siehe [CTX137161](#)). Es ist wichtig, diese Aktivierung vor dem Spiegeln der Datenbank durchzuführen, um Fehler zu vermeiden.
3. Legen Sie auf SQL-Server A eine Sicherungskopie der Datenbank an und kopieren Sie sie auf SQL-Server B.
4. Stellen Sie auf SQL-Server B die Backupdatei wieder dort (SQL-Server B) her.
5. Beginnen Sie auf SQL-Server A mit der Spiegelung.

Der nächste Schritt hängt davon ab, ob der Citrix Administrator (der Person, die den Assistenten zum Erstellen der Site ausführt) auch über vollständige Datenbankberechtigungen verfügt:

- Wenn der Citrix Administrator Datenbankberechtigungen hat (Datenbank- und Citrix Administrator sind ein- und dieselbe Person), wird alles über Studio abgewickelt:
  1. Der Citrix Administrator erstellt mit Studio eine Site und gibt die Adressen der zuvor erstellten SQL-Serverdatenbank A und deren Namen (myDatabaseMirrorForXD) ein.
  2. Die Datenbankskripts werden automatisch angewendet und die Prinzipal- und die Spiegeldatenbank werden festgelegt.
- Wenn der Citrix Administrator keine Datenbankberechtigungen hat, muss der Citrix Administrator einen Datenbankadministrator hinzuziehen:
  1. Der Citrix Administrator erstellt mit Studio eine Site und gibt die Adressen der zuvor erstellten SQL-Serverdatenbank und deren Namen (myDatabaseMirrorForXD) ein.
  2. Durch Auswahl von Skript erstellen im Assistenten für die Siteerstellung werden ein Spiegelungs- und ein Primärskript generiert. Der Citrix Administrator gibt diese Skripts an den Datenbankadministrator, der sie dann anwendet (das Spiegelungsskript sollte zuerst angewendet werden). Der Datenbankadministrator muss den Citrix Administrator informieren, wenn die Aufgabe abgeschlossen ist.
  3. Der Citrix Administrator kann jetzt in Studio den Assistenten für die Siteerstellung abschließen. Die Prinzipal- und die Spiegeldatenbank sind eingerichtet.

Um die Spiegelung nach dem Erstellen der Site zu überprüfen, führen Sie das PowerShell-Cmdlet get-configdbconnection aus, damit sichergestellt ist, dass der Failoverpartner in der Verbindungszeichenfolge für die Spiegelung eingerichtet wurde.

Wenn Sie später einen Delivery Controller in einer gespiegelten Datenbankumgebung hinzufügen, verschieben oder entfernen möchten, lesen Sie die Anweisungen unter [Hinzufügen, Entfernen, Verschieben von Controllern und Verschieben eines VDAs](#).

## Sicherstellen des Zugriffs auf Desktops und Anwendungen bei Ausfall der Controller

Für den Fall, dass alle Delivery Controller einer Site ausfallen, können Sie den Virtual Delivery Agent so konfigurieren, dass er im Hochverfügbarkeitsmodus arbeitet, damit Benutzer weiterhin auf Desktops und Anwendungen zugreifen und diese



verwenden können. Im Hochverfügbarkeitsmodus akzeptiert der VDA direkte ICA-Verbindungen von Benutzern anstelle von durch den Controller vermittelten Verbindungen.

Dieses Feature ist nur zur Verwendung in dem seltenen Fall, dass die Kommunikation mit allen Controllern fehlschlägt, vorgesehen. Es ist keine Alternative zu anderen Hochverfügbarkeitslösungen. Weitere Informationen finden Sie unter [CTX127564](#).

### Bei Betriebsunfähigkeit der Datenbank

Das Feature für Verbindungsleasing ergänzt die bewährten Methoden zur hohen Verfügbarkeit bei SQL Server, da es Benutzern die Wiederverbindung mit den zuletzt verwendeten Anwendungen und Desktops ermöglicht, selbst wenn die Sitedatenbank nicht verfügbar ist. Einzelheiten finden Sie unter [Verbindungsleasing](#).

# Bereitstellungsmethoden

May 10, 2016

Es ist eine Herausforderung, die Anforderungen aller Benutzer mit einer Virtualisierungsbereitstellung erfüllen. Mit XenApp und XenDesktop können Administratoren die Benutzererfahrungen mit verschiedenen Methoden, die auch als FlexCast-Modelle bezeichnet werden, anpassen.

Diese Bereitstellungsmethoden, von denen jede ihre Vor- und Nachteile hat, bieten die beste Benutzererfahrung für jeden Anwendungsfall.

## Mobilisieren von Windows-Anwendungen auf Mobilgeräten

Touchscreen-Geräte, wie Tablets und Smartphones, sind jetzt Standard in Mobilität. Wenn Windows-basierte Anwendungen auf ihnen ausgeführt werden, die normalerweise auf großen Bildschirmen angezeigt werden und für volle Funktionalität Klicken mit der rechten Maustaste erfordern, können diese Geräte Probleme verursachen.

XenApp mit Citrix Receiver bietet eine sichere Lösung, sodass die Benutzer von Mobilgeräten Zugriff auf alle Funktionen ihrer Windows-basierten Apps haben, ohne dass diese kostenaufwendig für native mobile Plattformen umgeschrieben werden müssen.

Die Bereitstellungsmethode der mit XenApp veröffentlichten Anwendungen verwendet die HDX-Mobiltechnologie, die die mit der Mobilisierung von Windows-Anwendungen assoziierten Probleme löst. Mit dieser Methode können Windows-Anwendungen für die Toucheingabe umgestaltet werden, während Features wie Mehrfingerbewegungen, native Menüsteuerung, Kamera und GPS-Funktionen erhalten bleiben. Viele Touchfeatures sind nativ in XenApp und XenDesktop und zu ihrer Aktivierung sind keine Änderungen am Anwendungs Quellcode nötig.

Zu diesen Features gehören Folgende:

- Automatische Anzeige der Tastatur, wenn ein bearbeitbares Feld den Fokus erhält
- Größeres Auswahlsteuerelement ersetzt Windows-Kombinationsfeld-Steuerelement
- Mehrfingergesten, z. B. Vergrößern und Verkleinern mit zwei Fingern
- Trägheitssensitiver Bildlauf
- Touchpad oder direkte Cursornavigation

## Verringern der Aktualisierungskosten für PCs

Das Aktualisieren von physischen Maschinen ist eine große Aufgabe, der sich viele Unternehmen alle drei bis fünf Jahre stellen müssen, besonders, wenn ein Unternehmen bei Betriebssystemen und Anwendungen immer auf dem neuesten Stand sein muss. Wachsende Unternehmen sehen sich zudem hohen Kosten für das Hinzufügen von neuen Maschinen zum Netzwerk gegenüber.

Mit der Bereitstellungsmethode eines VDI mit Personal vDisk können Einzelbenutzern vollständig personalisierte Desktopbetriebssysteme auf jeder Maschine oder jedem Thin Client mit Serverressourcen bereitgestellt werden. Administratoren können virtuelle Maschinen erstellen, deren Ressourcen, z. B. Verarbeitung, Arbeitsspeicher und Speicher, sich im Datacenter des Netzwerks befinden.

Dadurch kann das Leben älterer Maschinen verlängert werden, Software ist auf dem aktuellen Stand und Ausfallzeiten bei Upgrades werden minimiert.

Sicheren Zugriff auf virtuelle Apps und Desktops für Vertragsunternehmen und Partner sicherstellen

Netzwerksicherheit ist ein wachsendes Problem, besonders bei der Arbeit mit Vertragsunternehmen, Partnern und gelegentlichen Mitarbeitern von Fremdfirmen, die Zugriff auf Unternehmensapps und Daten benötigen. Die Mitarbeiter benötigen möglicherweise auch leihweise Laptops oder andere Geräte, die zusätzliche Kosten verursachen.

Daten, Anwendungen und Desktops sind bei XenDesktop und XenApp hinter der Firewall des sicheren Netzwerks gespeichert, sodass Endbenutzer nur die Ein- und Ausgabevorgänge der Benutzergeräte wie Tastatureingaben, Mausklicks, Audio und Bildschirmaktualisierungen übermitteln. Durch die Verwaltung dieser Ressourcen in einem Datacenter bieten XenDesktop und XenApp eine sicherere Remotezugriffslösung als das übliche SSL VPN.

Bei einer VDI-Bereitstellung mit Personal vDisk können Administratoren Thin Clients oder die persönlichen Geräte von Benutzern nutzen, indem sie eine virtuelle Maschine auf einem Netzwerkserver erstellen und ein Betriebssystem für einzelne Benutzer anbieten. Auf diese Weise können IT-Abteilungen die Sicherheit gegenüber Mitarbeitern von Vertragsunternehmen aufrechterhalten, ohne zusätzliches teures Gerät kaufen zu müssen.

## Beschleunigte Migration

Beim Wechsel zu einem neuen Betriebssystem kann es eine Herausforderung sein, Legacy- und nicht kompatiblen Anwendungen bereitzustellen.

Mit VM-gehosteten Apps können Benutzer ältere Anwendungen über Citrix Receiver auf der aktualisierten virtuellen Maschine ohne Kompatibilitätsprobleme ausführen. Dadurch haben IT-Mitarbeiter mehr Zeit, Kompatibilitätsprobleme zu testen und zu beheben, Benutzern den Übergang zu erleichtern und die Effizienz bei Helpdeskanrufen zu erhöhen.

Zusätzliche Vorteile bei der Verwendung von XenDesktop während der Migration sind u. a.:

- Reduzierte Komplexität von Desktops
- Verbesserte Steuerung für IT-Mitarbeiter
- Erhöhte Flexibilität der Endbenutzer im Hinblick auf Geräte und Arbeitsplatz

## Virtualisierung professioneller 3-D-Grafikanwendungen für Designer und Techniker

Viele Designunternehmen und Herstellerfirmen sind stark auf professionelle 3-D-Grafikanwendungen angewiesen. Die Kosten der für diese Art Software erforderlichen leistungsfähigen Hardware sind für diese Unternehmen eine große finanzielle Belastung. Dazu kommen logistische Probleme durch die Freigabe großer Designdateien über FTP, E-Mail und andere Ad-hoc-Methoden.

Bei der XenDesktop-Bereitstellungsmethode für gehostete physische Desktops wird ein Desktopimage auf Arbeitsstationen und Bladeservern bereitgestellt, ohne dass Hypervisors benötigt werden, um grafikintensive 3-D-Anwendungen auf einem systemeigenen Betriebssystem auszuführen.

Alle Dateien werden in einem zentralen Datacenter im Netzwerk gespeichert und große Designdateien können schneller und sicherer für andere Benutzer im Netzwerk freigegeben werden, da die Dateien nicht zwischen Arbeitsstationen übertragen werden müssen.

## Transformation von Callcentern

Unternehmen mit großen Callcentern stehen vor der Herausforderung, zu Spitzenzeiten genügend Maschinen und zu anderen Zeiten nicht zu viele Maschinen im Einsatz zu haben.

Die gepoolte VDI-Bereitstellungsmethode bietet dynamischen Zugriff auf einen standardisierten Desktop für viele Benutzer zu minimalen Kosten. Die gepoolten Maschinen werden pro Sitzung nach dem Windhundprinzip zugeteilt.

Bei diesen virtuellen Maschinen fällt weniger alltägliche Verwaltung an, da während der Sitzung vorgenommene Änderungen

verworfen werden, wenn der Benutzer sich abmeldet. Dies erhöht auch die Sicherheit.

Von XenApp gehostete Desktops ist eine weitere Bereitstellungsmethode, die sich für Callcenter eignet. Bei dieser Methode werden mehrere Benutzerdesktops auf einem serverbasierten Betriebssystem gehostet.

Dies ist eine kosteneffizientere Methode als die gepoolte VDI-Bereitstellung, aber bei von XenApp gehosteten Desktops haben Benutzer keine Berechtigungen, um Anwendungen zu installieren, Systemeinstellungen zu ändern oder den Server neu zu starten.

# Neue Bereitstellungen

Nov 02, 2016

Erstellen einer XenApp- oder XenDesktop-Bereitstellung

1. Richten Sie die Virtualisierungsumgebung zum Hosten und Verwalten der Komponenten für die XenApp- oder XenDesktop-Umgebung ein.

Informationen zu unterstützten Versionen der Virtualisierungsplattformen, Verwaltungstools und Cloudbereitstellungslösungen finden Sie unter [Systemanforderungen](#).

Mit diesen Virtualisierungsplattformen können Sie Maschinen in der XenApp- oder XenDesktop-Umgebung hosten und verwalten:

- XenServer: Informationen zur Einrichtung und Verwendung von XenServer finden Sie unter [XenServer](#).
- VMware vSphere: Informationen zur Einrichtung und Verwendung von VMware vSphere mit XenApp oder XenDesktop finden Sie unter [Vorbereiten der Virtualisierungsumgebung: VMware](#).
- Hyper-V mit Microsoft System Center Virtualization Machine Manager (VMM): Informationen zur Einrichtung und Verwendung von Hyper-V mit VMM mit XenApp oder XenDesktop finden Sie unter [Vorbereiten der Virtualisierungsumgebung: Microsoft System Center Virtual Machine Manager](#).

Verwenden Sie Microsoft System Center Configuration Manager mit Citrix Connector 7.5 für System Center Configuration Manager 2012 zum Verwalten physischer und virtueller Maschinen in der XenApp- oder XenDesktop-Umgebung oder zum Aktivieren des Wake-On-LAN-Features von Remote-PC-Zugriff. Siehe [Vorbereiten der Verwendung von Microsoft System Center Configuration Manager](#).

Sie können mit diesen Cloudbereitstellungslösungen Produktkomponenten hosten und virtuelle Maschinen bereitstellen. Mit diesen Lösungen werden Rechenressourcen gepoolt, um öffentliche oder private Infrastructure-as-a-Service (IaaS)-Clouds sowie Hybrid-IaaS-Clouds zu erstellen.

- Amazon Web Services: Informationen finden Sie unter [Deploy XenApp and XenDesktop 7.5 and 7.6 with Amazon VPC](#).
- Citrix CloudPlatform: Informationen finden Sie unter [XenApp und XenDesktop – Konzepte und Bereitstellung auf CloudPlatform](#).

2. Richten Sie die nicht zur Citrix Infrastruktur gehörenden Komponenten ein, die für die XenApp- oder XenDesktop-Site erforderlich sind. Dazu gehört mindestens ein Domänencontroller mit Active Directory-Domänendiensten.
3. Installieren Sie die Citrix Komponenten für die XenApp- oder XenDesktop-Site. Sie können die Komponenten über eine assistentengestützte grafische Oberfläche installieren oder über eine Befehlszeilenoberfläche, die eine skriptbasierte Installation ermöglicht. Bei beiden Methoden werden die meisten Voraussetzungen automatisch installiert.
  1. Vor Beginn jeglicher Installation lesen Sie die Informationen zu den [Systemanforderungen](#). Lesen Sie außerdem die [Checkliste zur Installationsvorbereitung](#) und haken Sie diese ab.
  2. Installieren der Kernkomponenten: Delivery Controller, Citrix Studio, Citrix Director, Citrix Lizenzserver und Citrix StoreFront. Weitere Informationen zur Installation dieser Komponenten finden Sie unter [Installieren über die grafische Oberfläche](#) und [Installieren an der Befehlszeile](#).
  3. Erstellen Sie eine Site in Studio. Weitere Informationen finden Sie unter [Erstellen einer Site](#).
  4. Installieren Sie entweder auf dem künftig zur Erstellung virtueller Maschinen genutzten Masterimage oder direkt auf jeder Maschine einen Virtual Delivery Agent (VDA). Weitere Informationen zur Installation des VDAs finden Sie unter [Installieren über die grafische Oberfläche](#) und [Installieren an der Befehlszeile](#). Auch der Artikel [Installieren oder Entfernen von Virtual Delivery Agents mit Skripts](#) enthält ggf. nützliche Informationen.

Installieren Sie für Remote-PC-Zugriff-Bereitstellungen einen VDA für Desktopbetriebssysteme auf jedem Büro-PC. Citrix empfiehlt, dass Sie die Befehlszeilenschnittstelle des VDA-Installationsprogramms und vorhandene ESD-Methoden (Electronic Software Distribution) verwenden.

5. Sie können auch den universellen Druckserver auf den Druckservern in der Umgebung installieren. Weitere Informationen zur Installation des universellen Druckservers finden Sie unter [Installieren über die grafische Oberfläche](#) und [Installieren an der Befehlszeile](#).
4. Sie können auch zusätzlichen Citrix Komponenten in die XenApp- oder XenDesktop-Bereitstellung integrieren. Beispiel:
  - Provisioning Services ist eine optionale Komponente von XenApp und XenDesktop, mit der Maschinen durch das Streaming eines Masterimages auf die Zielgeräte bereitgestellt werden. Weitere Informationen finden Sie unter [Provisioning Services](#).
  - Citrix NetScaler Gateway ist eine sichere Anwendungszugriffslösung, die Administratoren durch Richtlinien auf Anwendungsebene und durch Aktionssteuerung ermöglicht, den Zugriff auf Anwendungen und Daten zu sichern. Weitere Informationen finden Sie unter [Citrix NetScaler Gateway](#).
  - Citrix CloudBridge bietet eine Reihe von Geräten, die die WAN-Leistung optimieren. Weitere Informationen finden Sie unter [Citrix CloudBridge](#).
5. Richten Sie die Ressourcen ein, die Sie den Benutzern bereitstellen. Ihre Vorgehensweise hängt von der Bereitstellungsmethode ab, die Sie verwenden. Nachfolgend ist die grundlegende Reihenfolge für die meisten Bereitstellungsmethoden aufgeführt:
  1. Erstellen Sie mit dem Verwaltungstool des Hypervisors ein Masterimage, das die Desktops oder Anwendungen definiert, die Sie bereitstellen möchten. Siehe [Vorbereiten eines Masterimages](#).
  2. Erstellen Sie einen Maschinenkatalog mit den physischen und virtuellen Maschinen des Masterimages. Informationen finden Sie unter [Erstellen eines Maschinenkatalogs](#).
    - Wenn Sie die Maschinenerstellungsdienste zum Bereitstellen von Maschinen verwenden, können Sie dem Maschinenkatalog in Studio Maschinen hinzufügen.
    - Wenn Sie Maschinen mit Provisioning Services bereitstellen, verwenden Sie die Provisioning Services-Konsole, um Maschinen dem Maschinenkatalog hinzuzufügen.
  3. Erstellen Sie in Studio eine Bereitstellungsgruppe, um anzugeben, welche Benutzer auf die Maschinen und die darauf installierten Anwendungen zugreifen können. Weitere Informationen finden Sie unter [Bereitstellungsgruppen](#).

# Vorbereiten der Installation

Jul 13, 2016

In den folgenden Tabellen finden Sie Listen mit Aufgaben, die ausgeführt und Aktionen, die berücksichtigt werden müssen, bevor Sie die Kernkomponenten (Delivery Controller, Citrix Studio, Citrix Director, Citrix Lizenzserver und StoreFront) und Virtual Delivery Agents (VDAs) installieren.

## Vorbereitungen der Kernkomponenten und allgemeine Installationsvorbereitungen

| • Beschreibung                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Zu Beginn:</p> <ul style="list-style-type: none"><li>• Wenn Sie sich mit dem Produkt nicht auskennen, lesen Sie die <a href="#">technische Übersicht</a> und verwandte Themen.</li><li>• Informieren Sie sich über<br/>— <i>bekannte Probleme</i><br/>, die bei der Installation auftreten können.</li><li>• Wenn Sie Komponenten in einer Cloudumgebung installieren, lesen Sie folgende Dokumentation:<ul style="list-style-type: none"><li>• <a href="#">Deploy XenApp and XenDesktop 7.5 and 7.6 with Amazon VPC</a> für Amazon Web Services;</li><li>• <a href="#">XenApp und XenDesktop – Konzepte und Bereitstellung auf CloudPlatform</a> für Citrix CloudPlatform.</li></ul></li><li>• Wenn Sie XenServer für die Virtualisierungsumgebung verwenden, finden Sie Anleitungen in der XenServer-Dokumentation.</li><li>• Wenn Sie <a href="#">VMware</a> oder <a href="#">Microsoft System Center Virtual Machine Manager</a> für die Virtualisierungsumgebung verwenden, lesen Sie auch die verknüpften Themen.</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <p>Legen Sie fest, wo die Komponenten installiert werden, und bereiten Sie die Maschinen und Betriebssysteme vor.</p> <ul style="list-style-type: none"><li>• Lesen Sie die <a href="#">Systemanforderungen</a> für die unterstützten Betriebssysteme und Versionen für Controller, Studio, Director, Virtualisierungsressourcen und VDAs. In den Dokumenten zu den Systemanforderungen für Citrix StoreFront und Citrix Lizenzserver finden Sie Angaben zu den unterstützten Plattformen.<ul style="list-style-type: none"><li>• Sie können die Kernkomponenten auf dem gleichen Server oder auf unterschiedlichen Servern installieren. Um beispielsweise eine kleinere Bereitstellung remote zu verwalten, können Sie Studio auf einer anderen Maschine installieren als den Server, auf dem der Controller installiert wurde. Um zukünftige Erweiterungen zu ermöglichen, sollten Sie die Komponenten auf verschiedenen Servern installieren, z. B. den Lizenzserver und Director auf unterschiedlichen Servern.</li><li>• Sie können den Delivery Controller und den Virtual Delivery Agent für Windows-Serverbetriebssysteme auf demselben Server installieren. Starten Sie den Installer und wählen Sie den Delivery Controller (sowie alle weiteren Kernkomponenten für diese Maschine). Starten Sie dann den Installer erneut und wählen Sie den Virtual Delivery Agent für Windows-Serverbetriebssysteme.</li><li>• Installieren Sie Studio nicht auf einem Server, auf dem XenApp 6.5 Feature Pack 2 für Windows Server 2008 R2 oder eine frühere Version von XenApp ausgeführt wird.</li></ul></li><li>• Stellen Sie sicher, dass für jedes Betriebssystem die neuesten Updates ausgeführt wurden.</li><li>• Stellen Sie sicher, dass bei allen Maschinen die Systemuhren synchronisiert sind. Die Synchronisierung wird für die Kerberos-Infrastruktur benötigt, die die Kommunikation zwischen den Maschinen sichert.</li><li>• Komponenten werden standardmäßig unter C:\Programme\Citrix installiert. Sie können einen anderen Speicherort während der Installation angeben. Stellen Sie jedoch sicher, dass dieser Speicherort den Netzwerkdienst ausführen kann.</li><li>• Die meisten Komponentenvoraussetzungen werden automatisch installiert. Ausnahmen hiervon finden Sie unter<br/>— <i>Systemanforderungen</i><br/>.</li></ul> |
| <p>Legen Sie den Installationsort der SQL-Serversoftware für die Sitekonfigurationsdatenbank fest.</p> <ul style="list-style-type: none"><li>• Standardmäßig wird SQL Server 2012 Express automatisch mit dem Controller auf dem Server installiert, sofern keine andere Instanz erkannt wird.</li></ul> <p>Die Standardinstallation verwendet die Standarddienstkonten und -privilegien von Windows. Informationen zu diesen Standards und dem Hinzufügen von Windows-Dienstkonten zur sysadmin-Rolle finden Sie in der Microsoft-Dokumentation. In dieser Konfiguration verwendet der Controller das Netzwerkdienstkonto. Der Controller erfordert keine weiteren SQL Server-Rollen oder -Berechtigungen.</p> <p>Bei Bedarf können Sie zum Ausblenden der Datenbankinstanz die Option <b>Instanz ausblenden</b> wählen. Geben Sie beim Konfigurieren der Datenbankadresse in Studio die statische Portnummer statt des Namens ein. Informationen zum Ausblenden einer Instanz des SQL Server-Datenbankmoduls finden Sie in der Dokumentation von Microsoft.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |

|   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|---|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| • | <p><b>Beschreibung</b></p> <ul style="list-style-type: none"> <li>Sie können auch eine unterstützte Version von SQL Server auf dem Server oder auf einem anderen Server separat installieren. In diesem Fall muss die SQL Server-Software nicht vor der Installation der Kernkomponenten installiert werden, aber sie muss installiert werden, bevor Sie die Site erstellen.</li> <li>Überprüfen Sie anhand der Dokumente unter <i>— Planen</i>, was bei Datenbanken zu berücksichtigen ist, und richten Sie andere unterstützte Redundanzinfrastrukturen ein.</li> </ul> <p>Wichtig: Die Windows-Authentifizierung ist für Verbindungen zwischen dem Controller und der SQL Server-Datenbank erforderlich.</p>                                                                                                                                                                                 |
|   | <p>Legen Sie fest, wie Ports geöffnet werden.</p> <p>Standardmäßig werden die folgenden Ports automatisch geöffnet, wenn der Windows-Firewalldienst ausgeführt wird, selbst wenn die Firewall nicht aktiviert ist. Sie können diese Standardaktion deaktivieren und die Ports manuell öffnen, wenn Sie die Firewall eines Drittanbieters oder keine Firewall verwenden, oder wenn Sie die Ports einfach selbst öffnen möchten.</p> <ul style="list-style-type: none"> <li>Controller: TCP 80, 443</li> <li>Director: TCP 80, 443</li> <li>Lizenzserver: TCP 7279, 8082, 8083, 27000</li> <li>StoreFront: TCP 80, 443</li> </ul> <p>Tipp: Umfassende Portinformationen finden Sie in <a href="#">CTX101810</a>. Informationen zu zusätzlichen Installationsoptionen finden Sie unter <a href="#">Installieren über die Befehlszeile</a>.</p>                                                     |
|   | <p>Konfigurieren Sie Ihre Active Directory-Domäne.</p> <ul style="list-style-type: none"> <li>Auf den Maschinen, auf denen Sie die Kernkomponenten installieren, müssen Sie sowohl ein Domänenbenutzer als auch ein lokaler Administrator sein.</li> <li>Versuchen Sie nicht, Komponenten auf einem Domänencontroller zu installieren.</li> <li>Das Dokument <i>— Systemanforderungen</i> enthält eine Liste der unterstützten Funktionsebenen. Weitere Informationen finden Sie in der Microsoft Dokumentation.</li> </ul> <p>Wenn Sie den Lizenzserver installieren, wird das Benutzerkonto automatisch zum Volladministrator auf dem Lizenzserver.</p>                                                                                                                                                                                                                                       |
|   | <p>Entscheiden Sie vor der Installation von Director, ob Sie die Spiegelungsfunktion von Director verwenden möchten, bei der Windows-Remoteunterstützung verwendet wird.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|   | <p>Gut zu wissen:</p> <ul style="list-style-type: none"> <li>Bei Installationsproblemen wird der Prozess angehalten und eine Fehlermeldung angezeigt. Komponenten, die erfolgreich installiert werden, bleiben gespeichert; sie müssen nicht neu installiert werden.</li> <li>Studio wird nach dem Abschluss der Installation automatisch gestartet. Sie können diese Aktion während der Installation deaktivieren.</li> <li>Beim Erstellen von Objekten vor, während und nach der Installation sollten Sie eindeutige Namen für jedes Objekt eingeben (z. B. Netzwerke, Gruppen, Kataloge, Ressourcen).</li> <li>Nach der Installation der Komponenten in Amazon Web Services (AWS) benötigen Sie die Werte für Region, Verfügbarkeitszone, VPC-Namen, Subnetzadressen, Domännennamen, Namen der Sicherheitsgruppen und Anmeldeinformationen, um mit Studio eine Site zu erstellen.</li> </ul> |

## VDA-Installationsvorbereitung

|   |                                                                                                                                                                                                                                                                                                                                                                      |
|---|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ✓ | <p><b>Beschreibung</b></p>                                                                                                                                                                                                                                                                                                                                           |
|   | <p>Wenn Sie einen VDA für Windows-Desktopbetriebssysteme installieren, legen Sie fest, ob die Version HDX 3D Pro installiert wird.</p> <p>Mit der Funktion HDX3D Pro werden Desktops und Anwendungen bereitgestellt, die am besten mit einem GPU mit Hardwarebeschleunigung funktionieren. Weitere Informationen finden Sie in der Dokumentation für HDX 3D Pro.</p> |
|   | <p>Legen Sie fest, wie der VDA verwendet wird.</p>                                                                                                                                                                                                                                                                                                                   |



|                                                                                   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-----------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | <p><b>Beschreibung</b></p> <p>Die Standardeinstellung basiert auf der Annahme, dass Sie ein Masterimage mit einem installierten VDA verwenden, um mit Maschinenerstellungsdiensten oder Provisioning Services andere virtuelle Maschinen zu erstellen. Sie können diese Standardeinstellung überschreiben, wenn Sie den VDA auf einer vorhandenen Maschine installieren möchten.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|                                                                                   | <p>Legen Sie fest, ob Sie Citrix Receiver für Windows (CitrixReceiver.exe) installieren.</p> <p>Sie können diese Standardaktion deaktivieren.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|                                                                                   | <p>Legen Sie fest, wie Ports geöffnet werden.</p> <p>Standardmäßig werden die folgenden Ports automatisch geöffnet, wenn der Windows-Firewalldienst ausgeführt wird, selbst wenn die Firewall nicht aktiviert ist. Sie können diese Standardaktion deaktivieren und die Ports manuell öffnen, wenn Sie die Firewall eines Drittanbieters oder keine Firewall verwenden, oder wenn Sie die Ports einfach selbst öffnen möchten.</p> <ul style="list-style-type: none"> <li>• Controller: TCP 80, 1494, 2598, 8008 <ul style="list-style-type: none"> <li>• Für die Kommunikation zwischen Benutzergeräten und virtuellen Desktops müssen Sie eingehendes TCP auf Ports 1494 und 2598 als Portausnahmen konfigurieren. Aus Sicherheitsgründen empfiehlt Citrix, diese registrierten Ports ausschließlich für das ICA-Protokoll und das Common Gateway Protocol zu verwenden.</li> <li>• Für die Kommunikation zwischen Controller und virtuellen Desktops müssen Sie den eingehenden Port 80 als Portausnahme konfigurieren.</li> </ul> </li> <li>• Windows-Remoteunterstützung: TCP 3389<br/>Wenn das Feature aktiviert ist, öffnet Windows diesen Port automatisch, selbst wenn Sie die Ports normalerweise manuell öffnen.</li> <li>• Echtzeitaudioübertragung: UDP 16500-16509</li> </ul> <p>Tipp: Vollständige Portinformationen finden Sie unter <a href="#">CTX101810</a>.</p> |
|                                                                                   | <p>Entscheiden Sie, wie die Speicherorte installierter Controller angegeben werden.</p> <ul style="list-style-type: none"> <li>• Manuell durch Eingabe des vollqualifizierten Domännennamens (FQDN) des Controllers. Sie können zwar einen Controller angeben, der zurzeit nicht in der Domäne ist, VDAs können jedoch nur eine Verbindung mit einem Controller in der Domäne herstellen. Zudem kann die Verbindung nur für Controller in der Domäne getestet werden.</li> <li>• Mit Active Directory: wenn der Controller in der Domäne eingebunden ist.</li> <li>• Lassen Sie den Controller durch Maschinenerstellungsdienste angeben.</li> <li>• Später: Durch erneutes Ausführen des Installationsprogramms, über Citrix Richtlinien, durch Einstellen der Registrierungswerte oder mit Active Directory-Organisationseinheiten.</li> </ul> <p>Citrix Gruppenrichtlinieneinstellungen, die die Controller-Speicherorte angeben, überschreiben die bei der Installation festgelegten Einstellungen.</p> <p>Nach der Angabe des Controller-Standorts können VDAs später mit der Funktion für automatische Updates aktualisiert werden, wenn weitere Controller installiert werden.</p>                                                                                                                                                                                           |
|                                                                                   | <p>Entscheiden Sie, ob Sie die folgenden Features verwenden möchten:</p> <ul style="list-style-type: none"> <li>• Leistung optimieren: Wenn diese Funktion aktiviert ist, wird das Optimierungstool für VDAs verwendet, die auf einer VM auf einem Hypervisor ausgeführt werden. Die VM-Optimierung umfasst das Deaktivieren von Offlinedateien, das Deaktivieren der Hintergrunddefragmentierung und die Verkleinerung der</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |

|                                                                                   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-----------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | <p><b>Beschreibung</b></p> <p>Ereignisprotokollgröße. Weitere Informationen finden Sie unter <a href="#">CTX125874</a>. Aktivieren Sie diese Option nicht, wenn Sie Remote-PC-Zugriff verwenden möchten. Standard = aktiviert</p> <ul style="list-style-type: none"> <li>• Windows-Remoteunterstützung: Wenn dieses Feature aktiviert ist, wird die Windows-Remoteunterstützung mit dem Feature zur Benutzerspiegelung von Director verwendet, und Windows öffnet automatisch TCP-Port 3389 in der Firewall, selbst wenn Sie die Firewallports normalerweise manuell öffnen. Standard = aktiviert</li> <li>• Echtzeitaudioübertragung für Audio: Wenn diese Funktion aktiviert ist, wird UDP für Audiopakete verwendet, wodurch die Audioleistung verbessert werden kann. Standard = aktiviert</li> <li>• Personal vDisk: (Nur für die Installation von VDA für Windows-Desktopbetriebssysteme auf einer VM.) Wenn dieses Feature aktiviert ist, können persönliche vDisks mit einem Masterimage verwendet werden. Weitere Informationen finden Sie unter <a href="#">Personal vDisks</a>. Standard = deaktiviert</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|                                                                                   | <p>Gut zu wissen:</p> <ul style="list-style-type: none"> <li>• Der Druckspoolerdienst ist auf dem Windows-Server standardmäßig aktiviert. Wenn Sie diesen Dienst deaktivieren, können Sie keinen VDA für Windows-Serverbetriebssysteme installieren. Daher müssen Sie sicherstellen, dass der Dienst aktiviert ist, bevor ein VDA installiert wird.</li> <li>• Das Installationsprogramm erkennt automatisch das Betriebssystem und erlaubt nur die Installation des für das System geeigneten VDA-Typs: VDA für Windows-Serverbetriebssysteme oder VDA für Windows-Desktopbetriebssysteme.</li> <li>• Die Profilverwaltung wird bei der Installation von VDA installiert.</li> <li>• Wenn Sie VDA installieren, wird automatisch eine neue lokale Benutzergruppe namens "Benutzer mit direktem Zugriff" erstellt. Bei VDA für Windows-Desktopbetriebssysteme gilt diese Gruppe nur für RDP-Verbindungen; bei VDA für Windows-Serverbetriebssysteme gilt sie für ICA- und RDP-Verbindungen.</li> <li>• Wenn Sie einen VDA für Windows-Serverbetriebssysteme installieren, werden die Rollendienste für die Remotedesktopdienste automatisch installiert und aktiviert, wenn sie nicht bereits installiert und aktiviert sind.</li> <li>• Für Remote-PC-Zugriff-Konfigurationen ist VDA für Windows-Desktopbetriebssystem auf jedem physischen Büro-PC zu installieren, auf den die Benutzer zugreifen.</li> <li>• Alternativ zum vollständigen ISO-Produktimage können Sie für die Installation von VDAs ein eigenständiges VDA-Installationspaket verwenden. Weitere Informationen finden Sie unter <a href="#">Installieren von VDAs mit dem eigenständigen Paket</a>.</li> </ul> |

## Virtual Desktop Agents unter Windows XP oder Windows Vista

Die aktuellen Virtual Delivery Agents (VDAs) werden unter Windows XP und Windows Vista nicht unterstützt. Darüber hinaus können einige Features dieses Releases (sowie anderer neuerer Releases) unter diesen Betriebssystemen nicht verwendet werden. Zur Nutzung der vollständigen Funktionalität dieses Releases empfiehlt Citrix, dass Sie Ihr Windows XP- oder Windows Vista-System durch Windows 7, Windows 8 oder Windows 10 ersetzen und dann Virtual Delivery Agent aus diesem Release installieren.

Wenn Maschinen mit Windows XP oder Windows Vista weiter betrieben werden müssen, können Sie auch eine ältere Virtual Desktop Agent-Version (5.6 FP1 mit bestimmten Hotfixes) installieren. Einzelheiten hierzu finden Sie unter [CTX140941](#).

Beachten Sie Folgendes:

- Die Kernkomponenten (Controller, Studio Director, StoreFront, Citrix Lizenzserver) können nicht auf einem Windows XP- oder Windows Vista-System installiert werden.
- Remote-PC-Zugriff wird auf Windows Vista-Systemen nicht unterstützt.
- Citrix Support für Windows XP endete zusammen mit dem erweiterten Support von Microsoft am 8. April 2014.

- Die Verwendung älterer VDAs kann sich auf die Verfügbarkeit von Features und die Registrierung des VDAs beim Controller auswirken (siehe auch [Hinweise zu heterogenen Umgebungen](#)).

# Vorbereiten der Virtualisierungsumgebung: VMware

Nov 02, 2016

Folgen Sie diesen Anweisungen, wenn Sie zur Bereitstellung von virtuellen Maschinen VMware verwenden.

## Installieren und Konfigurieren des Hypervisors

1. Installieren Sie vCenter Server und die Verwaltungstools. (Der "Linked Mode"-Betrieb von vSphere vCenter wird nicht unterstützt.)
2. Erstellen Sie ein VMware-Benutzerkonto mit den folgenden Mindestberechtigungen auf DataCenter-Ebene. Dieses Konto verfügt über Berechtigungen zum Erstellen neuer VMs und wird verwendet, um mit vCenter zu kommunizieren.

| SDK                                           | Benutzeroberfläche                                     |
|-----------------------------------------------|--------------------------------------------------------|
| Datastore.AllocateSpace                       | Datastore > Allocate space                             |
| Datastore.Browse                              | Datastore > Browse datastore                           |
| Datastore.FileManagement                      | Datastore > Low level file operations                  |
| Network.Assign                                | Network > Assign network                               |
| Resource.AssignVMToPool                       | Resource > Assign virtual machine to resource pool     |
| System.Anonymous, System.Read und System.View | Automatisch hinzugefügt.                               |
| Task.Create                                   | Tasks > Create task                                    |
| VirtualMachine.Config.AddRemoveDevice         | Virtual machine > Configuration > Add or remove device |
| VirtualMachine.Config.AddExistingDisk         | Virtual machine > Configuration > Add existing disk    |
| VirtualMachine.Config.AddNewDisk              | Virtual machine > Configuration > Add new disk         |
| VirtualMachine.Config.AdvancedConfig          | Virtual machine > Configuration > Advanced             |
| VirtualMachine.Config.CPUCount                | Virtual machine > Configuration > Change CPU Count     |
| VirtualMachine.Config.Memory                  | Virtual machine > Configuration > Memory               |

|                                                |                                                                                                                                                                        |
|------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| VirtualMachine.Config.RemoveDisk<br><b>SDK</b> | Virtual machine > Configuration > Remove disk<br><b>Benutzerschnittstelle</b>                                                                                          |
| VirtualMachine.Config.Resource                 | Virtual machine > Configuration > Change resource                                                                                                                      |
| VirtualMachine.Config.Settings                 | Virtual machine > Configuration > Settings                                                                                                                             |
| VirtualMachine.Interact.PowerOff               | Virtual machine > Interaction > Power Off                                                                                                                              |
| VirtualMachine.Interact.PowerOn                | Virtual machine > Interaction > Power On                                                                                                                               |
| VirtualMachine.Interact.Reset                  | Virtual machine > Interaction > Reset                                                                                                                                  |
| VirtualMachine.Interact.Suspend                | Virtual machine > Interaction > Suspend                                                                                                                                |
| VirtualMachine.Inventory.Create                | Virtual machine > Inventory > Create new                                                                                                                               |
| VirtualMachine.Inventory.CreateFromExisting    | Virtual machine > Inventory > Create from existing                                                                                                                     |
| VirtualMachine.Inventory.Delete                | Virtual machine > Inventory > Remove                                                                                                                                   |
| VirtualMachine.Inventory.Register              | Virtual machine > Inventory > Register                                                                                                                                 |
| VirtualMachine.Provisioning.Clone              | Virtual machine > Provisioning > Clone virtual machine                                                                                                                 |
| VirtualMachine.Provisioning.DiskRandomAccess   | Virtual machine > Provisioning > Allow disk access                                                                                                                     |
| VirtualMachine.Provisioning.GetVmFiles         | Virtual machine > Provisioning > Allow virtual machine download                                                                                                        |
| VirtualMachine.Provisioning.PutVmFiles         | Virtual machine > Provisioning > Allow virtual machine files upload                                                                                                    |
| VirtualMachine.Provisioning.DeployTemplate     | Virtual machine > Provisioning > Deploy template                                                                                                                       |
| VirtualMachine.Provisioning.MarkAsVM           | Virtual machine > Provisioning > Mark as virtual machine                                                                                                               |
| VirtualMachine.State.CreateSnapshot            | vSphere 5.0, Update 2 und vSphere 5.1, Update 1: Virtual machine > State > Create snapshot<br><br>vSphere 5.5: Virtual machine > Snapshot management > Create snapshot |

|                                                   |                                                                                                                                                                                                     |
|---------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>SDK</b><br>VirtualMachine.State.RemoveSnapshot | <b>Benutzeroberfläche</b><br>vSphere 5.0, Update 2 und vSphere 5.1, Update 1: Virtual machine > State > Remove snapshot<br><br>vSphere 5.5: Virtual machine > Snapshot management > Remove snapshot |
| VirtualMachine.State.RevertToSnapshot             | vSphere 5.0, Update 2 und vSphere 5.1, Update 1: Virtual machine > State > Revert to snapshot<br><br>vSphere 5.5: Virtual machine > Snapshot management > Revert to snapshot                        |

3. Wenn Sie die erstellten VMs mit Tags kennzeichnen wollen, fügen Sie die folgenden Berechtigungen dem Benutzerkonto hinzu:

|                           |                                   |
|---------------------------|-----------------------------------|
| <b>SDK</b>                | <b>Benutzeroberfläche</b>         |
| Global.ManageCustomFields | Global > Manage custom attributes |
| Global.SetCustomField     | Global > Set custom attribute     |

Um sicherzustellen, dass Sie ein sauberes Basisimage zum Erstellen neuer VMs verwenden, fügen Sie Tags den VMs hinzu, die mit den Maschinenerstellungsdiensten erstellt wurden. Damit schließen Sie sie aus der Liste der für Basisimages verfügbaren virtuellen Maschinen aus.

## Beschaffen und Importieren eines Zertifikats

Um die vSphere-Kommunikation zu schützen, empfiehlt Citrix die Verwendung von HTTPS statt HTTP. HTTPS benötigt digitale Zertifikate. Citrix empfiehlt die Verwendung eines digitalen Zertifikats, das von einer Zertifizierungsstelle unter Berücksichtigung der Sicherheitsrichtlinie Ihrer Organisation erstellt wurde.

Wenn Sie kein digitales Zertifikat verwenden können, das von einer Zertifizierungsstelle ausgestellt wurde, können Sie das mit VMware installierte selbstsignierte Zertifikat verwenden, vorausgesetzt, die Sicherheitsrichtlinie Ihrer Organisation lässt dies zu. Fügen Sie das VMware vCenter-Zertifikat jedem Controller hinzu. Führen Sie diese Schritte aus:

1. Fügen Sie den vollqualifizierten Domännennamen (FQDN) des Computers, auf dem vCenter Server ausgeführt wird, der Hostdatei auf dem Server im Verzeichnis %SystemRoot%/WINDOWS/system32/Drivers/etc/ hinzu. Dieser Schritt ist nur erforderlich, wenn der FQDN des Computers, auf dem vCenter Server ausgeführt wird, nicht bereits im Domännennamensystem vorhanden ist.
2. Rufen Sie das vCenter-Zertifikat mit einer der folgenden Methoden ab:
  - Führen Sie auf dem vCenter-Server folgende Schritte aus:
    1. Kopieren Sie die Datei rui.crt vom vCenter-Server zu einem Speicherort, auf den Ihre Delivery Controller zugreifen können.
    2. Navigieren Sie auf dem Controller zu dem Speicherort des exportierten Zertifikats und öffnen Sie die Datei rui.crt.
  - Laden Sie das Zertifikat über einen Webbrowser herunter. Bei Verwendung von Internet Explorer müssen Sie (abhängig von Ihrem Benutzerkonto) ggf. in Internet Explorer mit der rechten Maustaste klicken und "Als Administrator ausführen" wählen, um das Zertifikat herunterzuladen und zu installieren.

1. Öffnen Sie einen Webbrowser und stellen Sie eine sichere Webverbindung zu dem vCenter-Server her, z. B. <https://server1.domain1.com>
2. Akzeptieren Sie die Sicherheitswarnungen.
3. Klicken Sie auf die Adressleiste, in der der Zertifikatsfehler angezeigt wird.
4. Zeigen Sie das Zertifikat an und klicken Sie auf die Registerkarte Details.
5. Wählen Sie In Datei kopieren und exportieren Sie im CER-Format; geben Sie an entsprechender Stelle einen Namen an.
6. Speichern Sie das exportierte Zertifikat.
7. Navigieren Sie auf den Speicherort des exportierten Zertifikats und öffnen Sie die CER-Datei.
- Direkter Import von Internet Explorer, der als Administrator ausgeführt wird:
  1. Öffnen Sie einen Webbrowser und stellen Sie eine sichere Webverbindung mit dem vCenter-Server her, z. B. <https://server1.domain1.com>.
  2. Akzeptieren Sie die Sicherheitswarnungen.
  3. Klicken Sie auf die Adressleiste, in der der Zertifikatsfehler angezeigt wird.
  4. Zeigen Sie das Zertifikat an.
- Importieren des Zertifikats auf jedem Controller in den Zertifikatspeicher:
  1. Klicken Sie auf Zertifikat installieren, wählen Sie Lokaler Computer aus und klicken Sie dann auf Weiter.
  2. Wählen Sie Alle Zertifikate in folgendem Speicher speichern aus und klicken Sie dann auf Durchsuchen.
  3. Bei Verwendung von Windows Server 2008 R2:
    1. Aktivieren Sie das Kontrollkästchen Physischen Speicher anzeigen.
    2. Erweitern Sie Vertrauenswürdige Personen.
    3. Wählen Sie Lokaler Computer.
    4. Klicken Sie auf Weiter und dann auf Fertig stellen.
  - Bei Verwendung von Windows Server 2012 oder Windows Server 2012 R2:
    1. Wählen Sie Vertrauenswürdige Personen und klicken Sie auf OK.
    2. Klicken Sie auf Weiter und dann auf Fertig stellen.

Wichtig: Wenn Sie den Namen des vSphere-Servers nach der Installation ändern, müssen Sie ein neues selbstsigniertes Zertifikat auf diesem Server erstellen, bevor Sie das neue Zertifikat importieren.

## Erstellen einer Master-VM

Verwenden Sie eine Master-VM zur Bereitstellung von Benutzerdesktops und Anwendungen.

1. Installieren Sie einen VDA auf der Master-VM unter Auswahl der Option zur Desktopoptimierung, wodurch die Leistung verbessert wird.
2. Erstellen Sie einen Snapshot der Master-VM, um diesen als Sicherungskopie zu verwenden. Weitere Informationen finden Sie unter [Vorbereiten eines Masterimages](#).

## Erstellen virtueller Desktops

Wenn Sie VM mit Studio erstellen statt einen vorhandenen Maschinenkatalog auszuwählen, geben Sie beim Einrichten der Hostinginfrastruktur für die Erstellung virtueller Desktops die folgenden Informationen ein:

1. Wählen Sie den VMware vSphere-Hosttyp aus.
2. Geben Sie die Adresse des Zugriffspunkts für das vCenter SDK (<https://vmware.example.com/sdk>) ein.
3. Geben Sie die Anmeldeinformationen für das zuvor eingerichtete VMware-Konto ein, das Berechtigungen zum Erstellen neuer VMs hat. Geben Sie den Benutzernamen im Format Domäne/Benutzername ein.

# Vorbereiten der Virtualisierungsumgebung: Microsoft System Center Virtual Machine Manager

May 10, 2016

Befolgen Sie die nachfolgenden Anweisungen, wenn Sie Hyper-V mit Microsoft System Center Virtual Machine Manager (VMM) zur Bereitstellung von virtuellen Maschinen verwenden.

Dieses Release unterstützt Folgendes:

- VMM 2012: Bietet bessere Verwaltungsfunktionen und ermöglicht die Verwaltung des gesamten virtualisierten Datacenters sowie der virtuellen Maschinen. Dieses Release organisiert das Patchen von Clusterhosts sowie die Integration mit Windows Server Update Services und ermöglicht somit das Erstellen eines Basisplans für Patches, die jeder Host benötigt.
- VMM 2012 SP1: Bietet Leistungsverbesserungen für Maschinenerstellungsdienste (MCS), wenn Sie SMB 3.0 auf Dateiservern mit freigegebenen Clustervolumes und Storage Area Networks (SANs) verwenden. Diese Dateifreigaben bieten kostengünstiges Zwischenspeichern und reduzierte E/A im SAN-Speicher mit verbesserter Leistung.
- VMM 2012 R2: Ermöglicht die angemessene Verwaltung von wichtigen Windows Server 2012 R2-Funktionen, wie das Ausführen von VM-Snapshots, dynamische VHDX-Größenanpassung und Speicherplätze.

Dieses Release unterstützt nur virtuelle Maschinen der ersten Generation mit VMM 2012 R2. Virtuelle Maschinen der zweiten Generation werden weder für Bereitstellungen mit Maschinenerstellungsdiensten (MCS) noch mit Provisioning Services unterstützt. Mit MCS oder Provisioning Services erstellte VM der zweiten Generation erscheinen nicht in der Auswahlliste für eine Master-VM. Auf ihnen ist der sichere Start standardmäßig aktiviert, wodurch eine richtige Funktionsweise des VDAs verhindert wird.

## Aktualisieren von VMM

- Upgrade von VMM 2012 auf VMM 2012 SP1 oder VMM 2012 R2  
Informationen zu den Anforderungen von VMM- und Hyper-V-Hosts finden Sie unter <http://technet.microsoft.com/en-us/library/gg610649.aspx>. Informationen zu den Anforderungen für die VMM-Konsole finden Sie unter <http://technet.microsoft.com/en-us/library/gg610640.aspx>.

Ein gemischter Hyper-V-Cluster wird nicht unterstützt. Ein gemischter Cluster ist beispielsweise einer, bei dem eine Hälfte des Clusters Hyper-V 2008 und die andere Hälfte Hyper-V 2012 ausführen.

- Upgrade von VMM 2008 R2 auf VMM 2012 SP1  
Wenn Sie ein Upgrade von XenDesktop 5.6 unter VMM 2008 R2 ausführen, halten Sie folgende Reihenfolge ein, um die Ausfallzeit von XenDesktop möglichst kurz zu halten:
  1. Upgrade von VMM auf 2012 (jetzt XenDesktop 5.6 und VMM 2012)
  2. Upgrade von XenDesktop auf die aktuelle Version (jetzt aktuelle XenDesktop-Version und VMM 2012)
  3. Upgrade von VMM von 2012 auf 2012 SP1 (jetzt aktuelle XenDesktop-Version und VMM 2012 SP1)
- Upgrade von VMM 2012 SP1 auf VMM 2012 R2  
Wenn Sie ein Upgrade von XenDesktop oder XenApp 7 unter VMM 2012 SP1 ausführen, halten Sie folgende Reihenfolge ein, um die Ausfallzeit von XenDesktop möglichst kurz zu halten:
  1. Upgrade von XenDesktop oder XenApp auf die aktuelle Version (jetzt aktuelle XenDesktop-/XenApp-Version und VMM 2012 SP1)
  2. Upgrade von VMM 2012 SP1 auf 2012 R2 (jetzt aktuelle XenDesktop- oder XenApp-Version und VMM 2012 R2)

## Zusammenfassung von Installation und Konfiguration

1. Installieren und konfigurieren Sie einen Hypervisor.
  1. Installieren Sie Microsoft Hyper-V Server und VMM auf Ihren Servern. Alle Delivery Controller müssen in derselben Gesamtstruktur sein wie die VMM-Server.
  2. Installieren Sie die System Center VMM-Konsole auf allen Controllern.
  3. Überprüfen Sie die folgenden Kontoinformationen:
    - Das Konto, das Sie zum Festlegen von Hosts in Studio verwenden, ist ein VMM-Administrator oder delegierter VMM-Administrator für die relevanten Hyper-V-Maschinen. Wenn dieses Konto nur über die delegierte Administratorrolle in VMM verfügt, werden die Speicherdaten in Studio beim Erstellen des Hosts nicht aufgeführt.
    - Das Benutzerkonto, das für die Studio-Integration verwendet wird, muss auch Mitglied der lokalen Administratorsicherheitsgruppe auf



jedem Hyper-V-Server sein, um zur Lebenszyklusverwaltung von VM (z. B. VM erstellen, aktualisieren und löschen) berechtigt zu sein.

Hinweis: Die Installation des Controllers direkt auf einem Server, auf dem Hyper-V ausgeführt wird, wird nicht unterstützt.

2. Erstellen Sie eine Master-VM.

1. Installieren Sie einen Virtual Delivery Agent auf der Master-VM und wählen Sie die Option zur Desktopoptimierung aus. Dies verbessert die Leistung.
2. Erstellen Sie einen Snapshot der Master-VM, um diesen als Sicherungskopie zu verwenden.

Weitere Informationen finden Sie unter [Vorbereiten eines Masterimage](#).

3. Erstellen Sie virtuelle Desktops. Wenn Sie MCS zum Erstellen von VMs beim Erstellen einer Site oder Verbindung verwenden.

1. Wählen Sie den Typ des Microsoft-Virtualisierungshosts aus.
2. Geben Sie die Adresse als vollqualifizierten Domännennamen des Hostservers ein.
3. Geben Sie die Anmeldeinformationen für das zuvor erstellte Administratorkonto ein, das Berechtigungen zum Erstellen von neuen VMs enthält.
4. Wählen Sie im Dialogfeld Hostdetails den Cluster oder eigenständigen Host aus, der beim Erstellen der neuen VMs verwendet werden soll. Wichtig: Sie müssen auch dann zu einem Cluster oder eigenständigen Host navigieren und diesen auswählen, wenn Sie eine Bereitstellung mit einem einzelnen Hyper-V-Host verwenden.

### MCS auf SMB 3-Dateifreigaben

Bei Maschinenkatalogen, die mit MSC auf SMB 3-Dateifreigaben für VM-Speicher erstellt wurden, müssen Sie darauf achten, dass die Anmeldeinformationen die nachfolgenden Anforderungen erfüllen, damit Aufrufe von der Hypervisor Communications Library (HCL) des Controllers die Verbindung mit dem SMB-Speicher herstellen:

- Die VMM-Benutzeranmeldeinformationen müssen vollständigen Lese-/Schreibzugriff auf den SMB-Speicher umfassen.
- Speichervorgänge auf dem virtuellen Datenträger werden bei Vorgängen im Lebenszyklus der VM über den Hyper-V-Server mit den VMM-Anmeldeinformationen durchgeführt.

Wenn Sie SMB als Speicher verwenden, aktivieren Sie das Feature "CredSSP" (Credential Security Support Provider) vom Controller auf den einzelnen Hyper-V-Maschinen, wenn Sie VMM 2012 SP1 mit Hyper-V unter Windows Server 2012 verwenden. Weitere Informationen finden Sie unter [CTX137465](#).

Über eine standardmäßige Remote-PowerShell V3-Sitzung verwendet die HCL CredSSP zum Öffnen einer Verbindung mit der Hyper-V-Maschine. Dieses Feature übergibt mit Kerberos verschlüsselte Benutzeranmeldeinformationen an die Hyper-V-Maschine. Die PowerShell-Befehle in dieser Sitzung auf der Remotemaschine mit Hyper-V werden dann unter Verwendung der angegebenen Anmeldeinformationen (in diesem Fall, derer des VMM-Benutzers) ausgeführt, sodass eine ordnungsgemäße Kommunikation mit dem Speicher gewährleistet wird.

Die folgenden Tasks verwenden PowerShell-Skripts der HCL, die an die Hyper-V-Maschine zur Verwendung mit SMB 3.0-Speicher gesendet werden.

- **Konsolidieren des Masterimages:** Ein Masterimage erstellt ein neues MCS-Provisioningschema (Maschinenkatalog). Die Master-VM wird durch dieses Schema geklont und vereinfacht, damit sie zum Erstellen neuer VM aus dem neu erstellten Datenträger bereit ist (die Abhängigkeit zur ursprünglichen Master-VM wird entfernt).

ConvertVirtualHardDisk im Namespace root\virtualization\v2

Beispiel:

```
$ims = Get-WmiObject -class $class -namespace "root\virtualization\v2"; $result = $ims.ConvertVirtualHardDisk($diskName, $vhdaText) $result
```

- **Erstellen eines differenzierenden Datenträgers:** erstellt einen differenzierenden Datenträger aus dem Masterimage, das durch Konsolidierung des Masterimages generiert wurde. Der differenzierende Datenträger wird dann an eine neue VM angeschlossen.

CreateVirtualHardDisk im Namespace root\virtualization\v2

Beispiel:

```
$ims = Get-WmiObject -class $class -namespace "root\virtualization\v2"; $result = $ims.CreateVirtualHardDisk($vhdaText); $result
```

- **Upload von Identitätsdisks:** Von der HCL kann die Identitätsdisk nicht direkt in den SMB-Speicher hochgeladen werden. Daher muss die Identitätsdisk von der Hyper-V-Maschine hochgeladen und in den Speicher kopiert werden. Da die Hyper-V-Maschine die Disk nicht auf dem Controller lesen kann, muss sie von der HCL zuerst wie folgt über die Hyper-V-Maschine kopiert werden:
  1. Upload der Identitätsdisk durch die HCL auf die Hyper-V-Maschine über die Administratorfreigabe.
  2. Die Disk wird von der Hyper-V-Maschine über ein PowerShell-Skript, das in der Remote-PowerShell-Sitzung ausgeführt wird, in den SMB-Speicher kopiert. Auf der Hyper-V-Maschine wird ein Ordner erstellt, dessen Berechtigungen nur für den VMM-Benutzer gesperrt sind (über

die remote PowerShell-Verbindung).

3. Die HCL löscht die Datei aus der Administratorfreigabe.

4. Wenn der Identitätsdisk-Upload durch die HCL auf die Hyper-V-Maschine abgeschlossen ist, werden die Identitätsdisks von der Remote-PowerShell-Sitzung in den SMB-Speicher kopiert und dann aus der Hyper-V-Maschine gelöscht.

Falls der Ordner der Identitätsdisk gelöscht wird, wird er neu erstellt, damit er zur Wiederverwendung verfügbar ist.

- **Download von Identitätsdisks:** Wie beim Upload wird die Identitätsdisk über die Hyper-V-Maschine an die HCL übergeben. Beim folgenden Prozess wird, falls noch nicht vorhanden, ein Ordner erstellt, der nur VMM-Benutzerbenutzerberechtigungen auf dem Hyper-V-Server hat.
  1. Die Disk wird von der Hyper-V-Maschine aus dem SMB-Speicher in den lokalen Hyper-V-Speicher kopiert, und zwar über ein PowerShell-Skript, das in der Remote-PowerShell V3-Sitzung ausgeführt wird.
  2. Die HCL liest die Disk aus der Administratorfreigabe der Hyper-V-Maschine in den Speicher.
  3. Die HCL löscht die Datei aus der Administratorfreigabe.

- **Erstellen einer persönlichen vDisk:** Wenn der Administrator die VM in einem Personal vDisk-Maschinenkatalog erstellt, muss eine leere Disk (PvD) erstellt werden.

Der Aufruf zum Erstellen einer leeren Disk erfordert keinen direkten Zugriff auf den Speicher. Wenn Sie PvDs auf anderen Speichern als dem Haupt- oder dem Betriebssystemdatenträger haben, verwenden Sie Remote-PowerShell zum Erstellen der PvD in einem Ordner mit dem gleichen Namen wie die VM, aus dem sie erstellt wurde. Verwenden Sie Remote-PowerShell nicht mit CSV oder LocalStorage. VMM-Befehlsfehler werden vermieden, wenn zuerst das Verzeichnis und dann die leere Disk erstellt werden.

Führen Sie auf der Hyper-V-Maschine `mkdir` an dem Speicher aus.

# Vorbereiten der Verwendung von Microsoft System Center Configuration Manager

May 10, 2016

Bei Sites, in denen der Zugriff auf Anwendungen und Desktops auf physischen Geräten mit System Center Configuration Manager (Configuration Manager) 2012 verwaltet wird, kann diese Verwendung über die Integrationsoptionen auf XenApp bzw. XenDesktop ausgeweitet werden.

- **Citrix Connector 7.5 für Configuration Manager 2012:** Citrix Connector bildet eine Brücke zwischen Configuration Manager und XenApp bzw. XenDesktop. Mit Citrix Connector können Sie alltägliche Vorgänge in den mit Configuration Manager verwalteten physischen Umgebungen und den mit XenApp bzw. XenDesktop verwalteten virtuellen Umgebungen übergreifend vereinheitlichen. Weitere Informationen zum Connector finden Sie unter [Citrix Connector 7.5 für System Center Configuration Manager 2012](#).
- **Configuration Manager Wake Proxy-Feature:** Unabhängig davon, ob in der Umgebung Citrix Connector verwendet wird, wird für das Wake-On-LAN-Feature für den Remote-PC-Zugriff Configuration Manager benötigt. Weitere Informationen finden Sie unter [Microsoft System Center Configuration Manager und Remote-PC-Zugriff-Wake-On-LAN](#).
- **XenApp- und XenDesktop-Eigenschaften:** XenApp- und XenDesktop-Eigenschaften ermöglichen die Identifizierung virtueller Citrix Desktops für die Verwaltung durch Configuration Manager. Diese Eigenschaften werden automatisch von Citrix Connector verwendet, sie können aber auch manuell konfiguriert werden, wie im folgenden Abschnitt beschrieben.

## Eigenschaften

Eigenschaften stehen Microsoft System Center Configuration Manager 2012 und 2012 R2 für die Verwaltung virtueller Desktops zur Verfügung.

Boolesche Eigenschaften in Configuration Manager 2012 können als 1 oder 0 statt True oder False angezeigt werden.

Die Eigenschaften stehen in der Klasse Citrix\_virtualDesktopInfo im Namespace Root\Citrix\DesktopInformation zur Verfügung. Die Namen der Eigenschaften stammen vom Anbieter für Windows-Verwaltungsinstrumentation (WMI).

| Eigenschaft        | Beschreibung                                                                                                                                                                                     |
|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| AssignmentType     | Legt den Wert auf IsAssigned fest. Gültige Werte: <ul style="list-style-type: none"><li>• ClientIP</li><li>• ClientName</li><li>• Keine</li><li>• User: legt IsAssigned auf True fest.</li></ul> |
| BrokerSiteName     | Site, gibt den gleichen Wert wie HostIdentifier zurück.                                                                                                                                          |
| DesktopCatalogName | Dem Desktop zugewiesener Maschinenkatalog                                                                                                                                                        |
| DesktopGroupName   | Dem Desktop zugewiesene Bereitstellungsgruppe                                                                                                                                                    |
|                    |                                                                                                                                                                                                  |

| HostIdentifier<br>Eigenschaft                                                 | Site, gibt den gleichen Wert wie BrokerSiteName zurück.<br>Beschreibung                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| IsAssigned                                                                    | True = ordnet den Desktop einem Benutzer zu; False = zufälliger Desktop.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| IsMasterImage                                                                 | Ermöglicht Entscheidungen bezüglich der Umgebung. Beispielsweise können Sie Anwendungen auf dem Masterimage und nicht auf den bereitgestellten Maschinen installieren, besonders dann, wenn diese Maschinen auf Bootmaschinen in einem fehlerfreien Zustand sind. Gültige Werte: <ul style="list-style-type: none"> <li>• True auf einer VM, die als Masterimage verwendet wird (dieser Wert wird während der Installation basierend auf einer Auswahl festgelegt).</li> <li>• Cleared auf einer VM, die von diesem Image bereitgestellt wird.</li> </ul> |
| IsVirtualMachine                                                              | True für eine virtuelle Maschine, False für eine physische Maschine                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| OSChangesPersist                                                              | False, wenn das Betriebssystemimage des Desktops bei jedem Neustart in einen fehlerfreien Zustand versetzt wird, andernfalls True                                                                                                                                                                                                                                                                                                                                                                                                                         |
| PersistentDataLocation                                                        | Der Speicherort, an dem Configuration Manager persistente Daten speichert. Benutzer haben hierauf keinen Zugriff.                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| PersonalDiskDriveLetter                                                       | Bei einem Desktop mit einer persönlichen vDisk ist dies der Laufwerksbuchstabe, den Sie der persönlichen vDisk zuweisen.                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| BrokerSiteName,<br>DesktopCatalogName,<br>DesktopGroupName,<br>HostIdentifier | Festgelegt, wenn der Desktop sich beim Controller registriert; bei Desktops, die noch nicht vollständig registriert sind, NULL                                                                                                                                                                                                                                                                                                                                                                                                                            |

Zum Sammeln der Eigenschaften führen Sie eine Hardwareinventur in Configuration Manager durch. Zum Anzeigen der Eigenschaften verwenden Sie den Ressourcen-Explorer von Configuration Manager. In diesen Fällen können die Namen Leerzeichen enthalten oder vom Eigenschaftsnamen etwas abweichen. **BrokerSiteName** kann z. B. als "Broker Site Name" angezeigt werden. Informationen zu den folgenden Aufgaben finden Sie unter [Citrix WMI Properties and System Center Configuration Manager 2012](#):

- Konfigurieren von Configuration Manager zum Sammeln von Citrix WMI-Eigenschaften vom Citrix VDA
- Erstellen abfragebasierter Gerätesammlungen mit Citrix WMI-Eigenschaften
- Erstellen globaler Bedingungen basierend auf Citrix WMI-Eigenschaften
- Verwenden globaler Bedingungen zum Definieren von Anforderungen für Anwendungsbereitstellungstypen

Sie können in der Microsoft-Klasse CCM\_DesktopMachine im Namespace Root\ccm\_vdi auch Microsoft-Eigenschaften verwenden. Weitere Informationen finden Sie in der Dokumentation von Microsoft.

Configuration Manager und Wake-On-LAN-Feature für den Remote-PC-Zugriff

Informationen zur Planung und Bereitstellung von Remote-PC-Zugriff finden Sie unter [Remote-PC-Zugriff](#) und [Bereitstellen von Remote-PC-Zugriff für Benutzer](#).

Zum Konfigurieren des Wake-On-LAN-Features von Remote-PC-Zugriff führen Sie die folgenden Schritte aus, bevor Sie einen VDA auf den Büro-PCs installieren und mit Studio die Remote-PC-Zugriff-Bereitstellung erstellen oder aktualisieren:

- Konfigurieren Sie Configuration Manager 2012 im Unternehmen und stellen Sie dann den Configuration Manager-Client auf allen Remote-PC-Zugriff-Maschinen bereit. Warten Sie, bis der geplante SCCM-Bestandszyklus ausgeführt wurde (oder erzwingen Sie manuell eine Ausführung). Die Zugriffsanmeldeinformationen, die Sie in Studio zum Konfigurieren der Verbindung mit Configuration Manager angeben, müssen Sammlungen im Geltungsbereich und die Rolle "Remotetoolsverantwortlicher" umfassen.
- Für die Unterstützung von Intel Active Management Technology (AMT) gilt Folgendes:
  - Die unterstützte Mindestversion auf dem PC ist AMT 3.2.1.
  - Stellen Sie den PC für die Verwendung von AMT mit Zertifikaten und zugehörigen Provisioningvorgängen bereit.
- Für die Unterstützung von Configuration Manager Wake-Proxy bzw. von Magic Packet:
  - Konfigurieren Sie Wake-On-LAN in den BIOS-Einstellungen aller PCs.
  - Zur Unterstützung von Configuration Manager Wake Proxy aktivieren Sie die entsprechende Option in Configuration Manager. Stellen Sie für jedes Subnetz des Unternehmens mit PCs, auf denen das Wake-On-LAN-Feature von Remote-PC-Zugriff verwendet wird, sicher, dass mindestens drei Maschinen als Sentinelmaschinen fungieren können.
  - Zur Unterstützung von Magic Packet konfigurieren Sie Netzwerkrouter und Firewalls so, dass Magic Packets entweder per subnetzgesteuertem Broadcast oder Unicast gesendet werden können.

Nach der Installation des VDAs auf Büro-PCs aktivieren oder deaktivieren Sie die Energieverwaltung beim Erstellen der Remote-PC-Zugriff-Bereitstellung in Studio.

- Wenn Sie die Energieverwaltung aktivieren, geben Sie Verbindungsdetails an, d. h. Adresse und Anmeldeinformationen von Configuration Manager sowie einen Namen.
- Wenn Sie die Energieverwaltung nicht aktivieren, können Sie eine Energieverwaltungsverbindung (Configuration Manager) später hinzufügen und dann einen Remote-PC-Zugriff-Maschinenkatalog zur Aktivierung der Energieverwaltung bearbeiten und die neue Energieverwaltungsverbindung angeben.

Sie können eine Energieverwaltungsverbindung bearbeiten, um die Verwendung von Configuration Manager Wake Proxy und Magic Packets zu konfigurieren und die Paketübertragungsmethode zu ändern.

# Installieren über die grafische Oberfläche

May 10, 2016

Vor Beginn einer Installation führen Sie die unter [Vorbereiten der Installation](#) beschriebenen Schritte durch.

Starten Sie die grafische Oberfläche des Installationsprogramms:

1. Laden Sie das Produktpaket herunter und entzippen Sie es. Optional können Sie die ISO-Datei auch auf DVD brennen.
2. Melden Sie sich mit einem lokalen Administratorkonto am Server an, auf dem Sie die Komponenten installieren.
3. Legen Sie die DVD in das Laufwerk ein oder stellen Sie die ISO-Datei bereit. Doppelklicken Sie auf Automatische Auswahl, wenn das Installationsprogramm nicht automatisch gestartet wird, oder klicken Sie auf das bereitgestellte Laufwerk.
4. Wählen Sie die Komponente aus, die Sie installieren möchten:
  - Wenn Sie ein Neueinsteiger sind, wählen Sie Delivery Controller. Anschließend können Sie den Delivery Controller installieren und wahlweise auch Studio, Director, Lizenzserver und StoreFront auf demselben Server.
  - Wenn Sie bereits einige Komponenten installiert haben und Ihre Bereitstellung erweitern möchten, klicken Sie in der rechten Spalte auf die zu installierende Komponente. Diese Spalte enthält Kernkomponenten und den universellen Druckserver, den Sie auf Ihrem Druckserver installieren können.
  - Klicken Sie zum Installieren eines Virtual Delivery Agent (VDA) auf den verfügbaren VDA-Eintrag. Das Installationsprogramm weiß, welcher VDA für das Betriebssystem, auf dem Sie das Installationsprogramm ausführen, der richtige ist.

Später können Sie einen installierten VDA auf folgende Weise anpassen:

1. Klicken Sie in Windows im Dialogfeld zum Hinzufügen oder Entfernen von Programmen mit der rechten Maustaste auf Citrix Virtual Delivery Agent <Versionsnummer> und wählen Sie Ändern.
2. Wählen Sie Virtual Delivery Agent-Einstellungen anpassen. Wenn das Installationsprogramm startet, können Sie die Controlleradressen ändern sowie den TCP/IP-Port für die Registrierung beim Controller (Standard = 80) festlegen und ob Portausnahmen für die Windows-Firewall automatisch geöffnet werden sollen.

Sie können Komponenten auch mit der grafischen Oberfläche aktualisieren. Weitere Informationen finden Sie unter [Upgrade einer Bereitstellung](#).

Alternativ zum vollständigen ISO-Produktimage können Sie für die Installation von VDAs ein eigenständiges VDA-Installationspaket verwenden. Weitere Informationen finden Sie unter [Installieren von VDAs mit dem eigenständigen Paket](#).

In XenApp und XenDesktop 7.6 FP3 enthält das UPS-Paket aktualisierte Versionen des eigenständigen UPS-Clients und der Serverkomponenten. Installationsanweisungen finden Sie im Abschnitt "Installieren des Citrix universellen Druckservers (UPS)" in [Bereitstellen von Druckern](#).

# Installieren an der Befehlszeile

Nov 02, 2016

Verwenden Sie die Befehlszeilenoberfläche für Folgendes:

- Installieren einer oder mehrerer der Kernkomponenten: Delivery Controller, Citrix Studio, Citrix Director, Lizenzserver und StoreFront.
- Installieren von Virtual Delivery Agent (VDA) auf einem Masterimage bzw. auf einer virtuellen oder physischen Maschine.  
Sie können auch die Skripts anpassen, die auf dem Medium sind, und sie zum Installieren und Entfernen von VDAs in Active Directory verwenden.
- Anpassen bereits installierter VDAs.
- Installieren eines universellen Druckers, der Netzwerksitzungsdrucker bereitstellt. Der Controller hat bereits die universelle Druckerverfunktionalität; Sie müssen nur den universellen Druckserver auf den Druckservern in der Umgebung installieren.

Sie haben außerdem folgende Möglichkeiten:

- Entfernen zuvor installierter Komponenten aus dieser Version mit den Optionen /remove oder /removeall. Einzelheiten finden Sie unter [Entfernen von Komponenten](#).
- Führen Sie ein Upgrade von Komponenten durch (siehe [Upgrade einer Bereitstellung](#)).

Sie müssen der Originaladministrator sein oder verwenden Sie "Als Administrator ausführen", um den Fortschritt der Befehlsausführung und die Rückgabewerte anzuzeigen. Weitere Informationen finden Sie in der Microsoft-Befehlsdokumentation.

Wichtig: Vor Beginn einer Installation führen Sie die unter [Vorbereiten der Installation](#) beschriebenen Schritte durch.  
Installieren von Kernkomponenten über die Befehlszeile

Führen Sie im Setupverzeichnis \x64\XenDesktop auf dem Medium den Befehl XenDesktopServerSetup.exe aus. Die folgende Tabelle beschreibt die Befehlsoptionen.

Hinweis: Geben Sie zum Installieren von XenApp die Option "/xenapp" in der Befehlszeile ein. Geben Sie zum Installieren von XenDesktop nicht die Option "/xenapp" in der Befehlszeile ein.

| Option                                | Beschreibung                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|---------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| /help oder /h                         | Zeigt die Hilfe für Befehle an.                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| /quiet oder /passive                  | Während der Installation wird keine Benutzeroberfläche angezeigt. Der einzige Hinweis auf den Installationsvorgang ist im Windows Task-Manager. Wenn diese Option ausgelassen wird, wird die grafische Oberfläche gestartet.                                                                                                                                                                                                                                                                         |
| /logpath Pfad                         | Speicherort der Protokolldateien. Der angegebene Ordner muss bereits vorhanden sein. Er wird nicht vom Installationsprogramm erstellt. Standard = "%TEMP%\Citrix\XenDesktop Installer"                                                                                                                                                                                                                                                                                                               |
| /noreboot                             | Verhindert einen Neustart nach der Installation. (Bei den meisten Kernkomponenten ist ein Neustart in der Standardeinstellung nicht aktiviert).                                                                                                                                                                                                                                                                                                                                                      |
| /remove                               | Entfernt die mit /components angegebenen Kernkomponenten.                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| /removeall                            | Entfernt alle installierten Kernkomponenten.                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| /xenapp                               | Installiert XenApp. Wenn diese Option ausgelassen wird, wird XenDesktop installiert.                                                                                                                                                                                                                                                                                                                                                                                                                 |
| /configure_firewall                   | Öffnet alle Ports in der Windows-Firewall, die für die installierten Komponenten erforderlich sind, wenn der Windows-Firewalldienst ausgeführt wird, selbst wenn die Firewall nicht aktiviert ist. Wenn Sie die Firewall eines Drittanbieters verwenden oder keine Firewall verwenden, müssen die Ports manuell geöffnet werden.                                                                                                                                                                     |
| /components component [component] ... | (erforderlich). Durch Trennzeichen getrennte Liste der zu installierenden oder zu entfernenden Komponenten. Gültige Werte: <ul style="list-style-type: none"><li>• CONTROLLER - Controller</li><li>• DESKTOPSTUDIO - Studio</li><li>• DESKTOPDIRECTOR - Director</li><li>• LICENSESERVER - Citrix Lizenzierung</li><li>• STOREFRONT - StoreFront</li></ul> Wenn diese Option ausgelassen wird, werden alle Komponenten installiert (bzw. entfernt, wenn die Option /remove ebenfalls angegeben ist). |
| /installdir Verzeichnis               | Vorhandenes leeres Verzeichnis, in dem die Komponenten installiert werden. Standard: C:\Programme\Citrix                                                                                                                                                                                                                                                                                                                                                                                             |
| /tempdir Verzeichnis                  | Das Verzeichnis, das die temporären Dateien während der Installation enthält. Standard = C:\Windows\Temp.                                                                                                                                                                                                                                                                                                                                                                                            |
| /nosql                                | Verhindert die Installation von Microsoft SQL Server Express auf dem Server, auf dem Sie den Controller installieren. Wenn diese Option ausgelassen wird, wird SQL Server Express installiert.                                                                                                                                                                                                                                                                                                       |
| /no_remote_assistance                 | (Gilt nur für die Installation von Director.) Verhindert die Installation und Aktivierung der Windows-Remoteunterstützung.                                                                                                                                                                                                                                                                                                                                                                           |

Beispiel: Mit dem folgenden Befehl werden ein XenDesktop-Controller, Studio, die Citrix Lizenzierung und SQL Server Express auf dem Server installiert. Für die Komponentenkommunikation erforderliche Ports werden automatisch geöffnet.

\x64\XenDesktop Setup\XenDesktopServerSetup.exe /components controller,desktopstudio,licenseserver /configure\_firewall

Mit dem folgenden Befehl werden ein XenApp-Controller, Studio und SQL Server Express auf dem Server installiert. Für die Komponentenkommunikation erforderliche Ports werden automatisch geöffnet.

\x64\XenDesktop Setup\XenDesktopServerSetup.exe /xenapp /components controller,desktopstudio /configure\_firewall

Installieren von VDA über die Befehlszeile

Bei der Installation von VDA für die Verwendung mit Remote-PC-Zugriff geben Sie nur Optionen ein, die auf physischen Maschinen (nicht VMs oder Masterimages) gültig und für VDAs für Windows-Desktopbetriebssysteme relevant sind.

Führen Sie auf dem Produktmedium im XenDesktop-Setupverzeichnis den Befehl `XenDesktopVdaSetup.exe` aus. Die folgende Tabelle beschreibt die Befehlsoptionen. Wenn nicht anders angegeben, gelten die Optionen für physische und virtuelle Maschinen sowie für VDAs für Windows-Desktopbetriebssysteme und VDAs für Windows-Serverbetriebssysteme.

| Option                                                         | Beschreibung                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|----------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>/h</code> oder <code>/help</code>                        | Zeigt die Hilfe für Befehle an.                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| <code>/quiet</code> oder <code>/passive</code>                 | Während der Installation wird keine Benutzeroberfläche angezeigt. Der einzige Hinweis auf den Installations- und Konfigurationsvorgang ist im Windows Task-Manager. Wenn diese Option ausgelassen wird, wird die grafische Oberfläche gestartet.                                                                                                                                                                                                                      |
| <code>/logpath</code> Pfad                                     | Speicherort der Protokolldateien. Der angegebene Ordner muss bereits vorhanden sein. Er wird nicht vom Installationsprogramm erstellt. Standard = <code>"%TEMP%\CitrixXenDesktop Installer"</code>                                                                                                                                                                                                                                                                    |
| <code>/noreboot</code>                                         | Verhindert einen Neustart nach der Installation. VDA steht erst nach einem Neustart vollständig zur Verfügung.                                                                                                                                                                                                                                                                                                                                                        |
| <code>/remove</code>                                           | Entfernt die mit <code>/components</code> angegebenen Komponenten.                                                                                                                                                                                                                                                                                                                                                                                                    |
| <code>/removeall</code>                                        | Entfernt alle installierten VDA-Komponenten.                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <code>/reconfig</code>                                         | Passt die zuvor konfigurierten VDA-Einstellungen an, wenn der Befehl mit den Optionen <code>/portnumber</code> , <code>/Controller</code> oder <code>/enable_hdx_ports</code> verwendet wird. Wenn Sie diese Option ohne die Option <code>/quiet</code> angeben, wird die grafische Oberfläche zum Anpassen von VDA gestartet.                                                                                                                                        |
| <code>/portnumber</code> port                                  | (Gilt nur, wenn die Option <code>/reconfig</code> angegeben wurde.) Portnummer für die Kommunikation zwischen VDA und dem Controller. Der zuvor konfigurierte Port wird deaktiviert, es sei denn, es handelt sich um Port 80.                                                                                                                                                                                                                                         |
| <code>/components</code><br><code>component[.component]</code> | Durch Trennzeichen getrennte Liste der zu installierenden oder zu entfernenden Komponenten. Gültige Werte: <ul style="list-style-type: none"> <li>VDA: Installation von VDA</li> <li>PLUGINS: Installation von Citrix Receiver für Windows (CitrixReceiver.exe)</li> </ul> Wenn diese Option ausgelassen wird, werden alle Komponenten installiert.                                                                                                                   |
| <code>/installdir</code> Verzeichnis                           | Vorhandenes leeres Verzeichnis, in dem die Komponenten installiert werden. Standard: <code>C:\Programme\Citrix</code>                                                                                                                                                                                                                                                                                                                                                 |
| <code>/tempdir</code> Verzeichnis                              | Das Verzeichnis für die temporären Dateien während der Installation. (Diese Option ist bei der grafischen Oberfläche nicht verfügbar) Standard = <code>C:\Windows\Temp</code> .                                                                                                                                                                                                                                                                                       |
| <code>/site_guid</code> guid                                   | GUID (Globally Unique Identifier) der Website Active Directory Organisationseinheit (OU). Dabei wird ein virtueller Desktop einer Site zugeordnet, wenn Active Directory für die Discovery verwendet wird (das Feature für automatische Updates ist die empfohlene und Discovery-Standardmethode). Die Site-GUID ist eine Site-Eigenschaft, die in Studio angezeigt wird. Geben Sie nicht beide Optionen, <code>/site_guid</code> und <code>/controllers</code> , an. |
| <code>/controllers</code> "controller<br>[controller] [...]"   | Durch Leerzeichen getrennte, vollqualifizierte Domännennamen (FQDNs) der Controller, mit denen VDA kommunizieren kann; von Anführungszeichen umschlossen. Geben Sie nicht beide Optionen, <code>/site_guid</code> und <code>/controllers</code> , an.                                                                                                                                                                                                                 |
| <code>/xa_server_location</code> url                           | URL des Servers für Windows Serveranwendungen.                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <code>/enable_remote_assistance</code>                         | Aktiviert Windows-Remoteunterstützung für die Verwendung mit Director. Wenn Sie diese Option angeben, wird der TCP-Port 3389 durch Windows in der Firewall geöffnet, selbst wenn die Option <code>/enable_hdx_ports</code> ausgelassen wird.                                                                                                                                                                                                                          |
| <code>/enable_hdx_ports</code>                                 | Öffnet die erforderlichen Ports für den Controller und die angegebenen Features (Windows-Remoteunterstützung, Echtzeitübertragung und Optimieren) in der Windows Firewall, wenn der Windows-Firewalldienst erkannt wird, selbst wenn die Firewall nicht aktiviert ist. Wenn Sie eine andere oder keine Firewall verwenden, müssen die Ports manuell geöffnet werden.                                                                                                  |
| <code>/optimize</code>                                         | Aktiviert die Optimierung für VDAs, die auf einer VM auf einem Hypervisor ausgeführt werden. Die VM-Optimierung umfasst das Deaktivieren von Offlinedateien, das Deaktivieren der Hintergrunddefragmentierung und die Verkleinerung der Ereignisprotokollgröße. Geben Sie diese Option für Remote-PC-Zugriff nicht an. Weitere Informationen zum Optimierungstool finden Sie unter <a href="#">CTX125874</a> .                                                        |
| <code>/baseimage</code>                                        | (Gilt nur für die Installation von VDA für Windows-Desktopbetriebssysteme auf einer VM.) Ermöglicht das Verwenden von persönlichen vDisks mit einem Masterimage. Weitere Informationen finden Sie unter <a href="#">Personal vDisks</a> .                                                                                                                                                                                                                             |
| <code>/enable_hdx_3d_pro</code>                                | Installiert VDA für HDX 3D Pro. Weitere Informationen finden Sie in der Dokumentation für HDX 3D Pro.                                                                                                                                                                                                                                                                                                                                                                 |
| <code>/enable_real_time_transport</code>                       | Aktiviert oder deaktiviert die Verwendung von UDP für Audiopakete (Echtzeittransport für Audio). Das Aktivieren dieses Features kann die Audioleistung verbessern. Verwenden Sie die Option <code>"/enable_hdx_ports"</code> , wenn Sie möchten, dass die UDP-Ports automatisch bei Erkennung des Windows-Firewalldiensts geöffnet werden.                                                                                                                            |



| Option                                               | Beschreibung                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| /image                                               | Installiert VDA auf einer VM. Richtet den VDA als Masterimage ein.                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| /virtualmachine                                      | (Gilt nur für die Installation von VDA auf einer VM.) Überschreibt das Erkennen einer physischen Maschine durch den Installer. Dabei werden BIOS-Informationen an die VMs weitergegeben, sodass sie als physische Maschinen erscheinen.                                                                                                                                                                                                                                                            |
| /nodesktopexperience                                 | (Gilt nur für die Installation von VDA für Windows-Serverbetriebssysteme.) Verhindert das Aktivieren von Enhanced Desktop Experience. Dieses Feature wird auch über die Citrix Richtlinieneinstellung "Enhanced Desktop Experience" gesteuert.                                                                                                                                                                                                                                                     |
| /nocitrixwddm                                        | (Gilt nur für Windows 7-Maschinen, die keine WDDM-Treiber enthalten.) Deaktiviert die Installation des Citrix WDDM-Treibers.                                                                                                                                                                                                                                                                                                                                                                       |
| /servervdi                                           | Installiert einen VDA für Windows-Desktopbetriebssysteme auf einem unterstützten Windows Server. Lassen Sie diese Option aus, wenn Sie einen VDA für Windows-Serverbetriebssysteme auf einem Windows Server installieren. Bevor Sie diese Option verwenden, lesen Sie den Artikel <a href="#">Server-VDI</a> .<br><br><b>Hinweis:</b> Fügen Sie die Option /masterimage hinzu, wenn Sie den VDA auf einem Image installieren, und verwenden Sie MCS zum Erstellen von Server-VMs von diesem Image. |
| /installwithsecurebootenabled                        | Ermöglicht die VDA-Installation, wenn der sichere Start aktiviert ist. Wenn diese Option ausgelassen wird, gibt eine Warnung an, dass der sichere Start für eine erfolgreiche VDA-Installation deaktiviert werden muss.                                                                                                                                                                                                                                                                            |
| /exclude "Personal vDisk","Machine Identity Service" | (Gilt nur bei Upgrades von einem älteren 7.x-VDA auf einem physischen Computer.) Schließt Personal vDisk und den Maschinenidentitätsdienst vom Upgrade aus. Informationen zur erweiterten Verwendung dieser Option finden Sie unter <a href="#">CTX140972</a> .                                                                                                                                                                                                                                    |

Mit dem folgenden Befehl wird beispielsweise ein VDA für Windows-Desktopbetriebssysteme und Citrix Receiver am Standardspeicherort auf einer VM installiert. VDA wird als Masterimage verwendet. Der VDA registriert sich anfänglich am Controller auf dem "Contr-Main" genannten Server in der Domäne "mydomain" und verwendet Personal vDisks, das Optimierungsfeature und die Windows-Remoteunterstützung.

XenDesktop SetupXenDesktopVdaSetup.exe /quiet /components vda,plugins /controllers "Contr-Main.mydomain.local" /enable\_hdx\_ports /optimize /masterimage /baseimage /enable\_remote\_assistant

Mit dem folgenden Befehl wird VDA für Windows-Desktopbetriebssysteme und Citrix Receiver am Standardspeicherort des Büro-PCs installiert, der mit Remote-PC-Zugriff verwendet wird. Nach der Installation von VDA wird die Maschine nicht neu gestartet. Ein Neustart ist jedoch erforderlich, bevor VDA verwendet werden kann. Der VDA registriert sich anfänglich am Controller auf dem "Contr-East" genannten Server in der Domäne "mydomain" und verwendet UDP für Audiopakete. HDX-Ports werden geöffnet, wenn der Windows-Firewalldienst erkannt wird.

XenDesktop SetupXenDesktopVdaSetup.exe /quiet /components vda,plugins /controllers "Contr-East.mydomain.local" /enable\_hdx\_ports /enable\_real\_time\_transport /noreboot

Alternativ zum vollständigen ISO-Produktimage können Sie für die Installation von VDAs ein eigenständiges VDA-Installationspaket verwenden. Weitere Informationen finden Sie unter [Installieren von VDAs mit dem eigenständigen Paket](#).

Anpassen eines VDAs über die Befehlszeile

Nachdem VDA installiert wurde, können Sie einige Einstellungen anpassen. Führen Sie auf dem Produktmedium vom Setupverzeichnis \x64\XenDesktop den Befehl XenDesktopVdaSetup.exe aus und legen Sie dabei eine oder mehrere der oben beschriebenen Optionen fest:

- /reconfigure: Diese Option ist zum Anpassen von VDA erforderlich
- /h oder /help
- /quiet
- /noreboot
- /controller
- /portnumber port
- /enable\_hdx\_ports

Installieren des universellen Druckservers über die Befehlszeile

- Hinweise:**
- Das Bereitstellen des universellen Druckservers auf 32-Bit-Betriebssystemen wird nicht unterstützt.
  - Die Benutzerauthentifizierung bei Druckvorgängen erfordert, dass der universelle Druckserver in der gleichen Domäne ist wie der Remotedesktopdienste-VDA.
  - Die UPClient-Komponente ist in der VDA-Installation enthalten und Sie müssen die Clientkomponente nicht manuell installieren.

Installieren des universellen Druckservers an der Befehlszeile

- Auf einem Druckserver unter Windows 2008 R2 SP1, Windows Server 2012 oder Windows Server 2012 R2 führen Sie **UpsServer\_x64.msi** im Verzeichnis \x64\Universal Print Server\ des LTSR-Images aus.

Wenn Sie den universellen Druckserver über die Befehlszeile installieren, empfiehlt es sich, die Befehlszeilenoption /ENABLE\_CEIP auf 1 festzulegen, um am [Citrix Programm zur Verbesserung der Benutzerfreundlichkeit \(CEIP\)](#) teilzunehmen.

Beispiel:

Code

KOPIEREN

```
msiexec /i UpsServer.msi ENABLE_CEIP=1
```

Wenn Sie am Programm teilnehmen, werden anonyme Statistiken und Nutzungsinformationen an Citrix gesendet, um die Qualität und Leistung der Produkte zu verbessern.

# Erstellen einer Site

May 10, 2016

Eine Site ist der Name, den Sie einer Produktbereitstellung geben. Sie umfasst die Delivery Controller und die anderen Kernkomponenten, VDAs, Verbindungen mit virtuellen Ressourcen (falls verwendet) sowie die Maschinenkataloge und Bereitstellungsgruppen, die Sie erstellen und verwalten. Eine Site muss nicht unbedingt einem geografischen Standort entsprechen, obwohl dies der Fall sein kann. Sie erstellen die Site nach der Installation der Komponenten und bevor Sie Maschinenkataloge und Bereitstellungsgruppen erstellen.

In der folgenden Tabelle finden Sie die Aufgaben, die Sie abschließen müssen, und Punkte, die Sie berücksichtigen sollten, bevor Sie den Assistenten für die Siteerstellung in Studio starten.

✓

Beschreibung

Legen Sie fest, welchen Sitetyp Sie erstellen möchten:

Anwendungs- und Desktopbereitstellungssite: Wenn Sie eine Anwendungs- und Desktopbereitstellungssite erstellen möchten, haben Sie die Wahl zwischen einer vollständigen Bereitstellung (empfohlen) oder einer leeren Site. (Leere Sites sind nur teilweise konfiguriert und werden normalerweise von erfahrenen Benutzern erstellt.)

Remote-PC-Zugriff-Site: Ermöglicht bestimmten Benutzern den Remotezugriff auf ihre Büro-PCs über eine sichere Verbindung. Wenn Sie das Wake-On-LAN-Feature von Remote-PC-Zugriff verwenden möchten, führen Sie die unter [Configuration Manager und Wake-On-LAN-Feature für den Remote-PC-Zugriff](#) beschriebenen Aufgaben durch.

Wenn Sie zu diesem Zeitpunkt eine Bereitstellung für die Anwendungs- und Desktopbereitstellung erstellen, können Sie eine Remote-PC-Zugriff-Bereitstellung später hinzufügen. Ebenso können Sie einer Remote-PC-Zugriff-Bereitstellung später eine vollständige Bereitstellung hinzufügen.

Die Siteerstellung umfasst das Erstellen der Sitekonfigurationsdatenbank. Stellen Sie sicher, dass die SQL Server-Software installiert ist, bevor Sie eine Site erstellen.

Zum Erstellen der Datenbank müssen Sie ein lokaler Administrator und ein Domänenbenutzer sein. Sie müssen zudem entweder SQL Server-Berechtigungen haben oder Sie können Skripts generieren, die dann vom Datenbankadministrator ausgeführt werden.

Berechtigungen: Sie benötigen die folgenden Berechtigungen zum Einrichten der Datenbank. Die Berechtigungen können explizit konfiguriert oder über eine Active Directory-Gruppenmitgliedschaft erworben werden:

| Vorgang             | Zweck                                                                                     | Serverrolle        | Datenbankrolle |
|---------------------|-------------------------------------------------------------------------------------------|--------------------|----------------|
| Datenbank erstellen | Erstellen einer geeigneten leeren Datenbank                                               | dbcreator          |                |
| Schema erstellen    | Erstellen aller dienstspezifischen Schemas und Hinzufügen des ersten Controllers zur Site | securityadmin<br>* | db_owner       |

| ✓                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | Beschreibung                              | Zweck                                                                                                                       | Serverrolle     | Datenbankrolle |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|-----------------|----------------|
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Controller hinzufügen                     | Hinzufügen eines weiteren Controllers (zusätzlich zum ersten) zur Site                                                      | securityadmin * | db_owner       |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Controller hinzufügen (Spiegelungsserver) | Hinzufügen einer Controller-Anmeldung zu dem Datenbankserver, der derzeit die Spiegelrolle einer gespiegelten Datenbank hat | securityadmin * |                |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Schema aktualisieren                      | Anwenden von Aktualisierungen oder Hotfixes auf das Schema                                                                  |                 | db_owner       |
| <p>* Zwar ist die securityadmin-Serverrolle technisch restriktiver als die sysadmin-Serverrolle, aber in der Praxis ist sie als gleichwertig anzusehen.</p> <p>Wenn Sie diese Vorgänge mit Studio ausführen, muss das Benutzerkonto Mitglied der sysadmin-Serverrolle sein.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                                           |                                                                                                                             |                 |                |
| <p>Wenn Ihre Studio-Anmeldeinformationen diese Berechtigungen nicht umfassen, werden Sie aufgefordert, Benutzeranmeldeinformationen für SQL Server einzugeben.</p> <ul style="list-style-type: none"> <li>• <b>Skripts:</b> Wenn der Datenbankserver gesperrt ist und Sie nicht über die erforderlichen SQL Server-Berechtigungen verfügen, kann der Assistent für die Siteerstellung zwei Datenbankskripts erstellen, eines zum Einrichten der Datenbank und das zweite zur Verwendung in einer Spiegelungsumgebung. Nachdem Sie die Skripterstellung angefordert haben, geben Sie die erstellten Skripts dem Datenbankadministrator (oder einer Person mit den erforderlichen SQL Server-Berechtigungen) zur Ausführung auf dem Datenbankserver und der gespiegelten Datenbank (falls erforderlich). Nachdem das Skript ausgeführt und die Datenbank erfolgreich erstellt wurde, können Sie die Erstellung der Site abschließen.</li> </ul>                                                                                                                                                                                                                                                                                       |                                           |                                                                                                                             |                 |                |
| <p>Erwägen Sie, ob Sie das 30-tägige kostenlose Probeabo, bei dem die Lizenzdateien später hinzugefügt werden können, oder ob Sie vorhandene Lizenzen verwenden. Sie können Lizenzdateien im Assistenten für die Siteerstellung hinzufügen oder herunterladen.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                                           |                                                                                                                             |                 |                |
| <p>Konfigurieren Sie die Umgebung der Virtualisierungsressource (Hostumgebung).</p> <p>Wenn Sie XenServer verwenden:</p> <ul style="list-style-type: none"> <li>• Weitere Informationen finden Sie in der XenServer-Dokumentation.</li> <li>• Sie müssen die Anmeldeinformationen eines VM-Hauptadministrators oder eines höherrangigen Benutzers eingeben.</li> <li>• Citrix empfiehlt die Verwendung von HTTPS zur Sicherung der Kommunikation mit XenServer. Um HTTPS zu verwenden, müssen Sie das auf XenServer installierte Standard-SSL-Zertifikat, durch ein Zertifikat einer vertrauenswürdigen Zertifizierungsstelle ersetzen; siehe <a href="#">CTX128656</a>.</li> <li>• Sie können hohe Verfügbarkeit konfigurieren, wenn diese auf XenServer aktiviert ist. Citrix empfiehlt, dass Sie alle Server im Pool auswählen, um die Kommunikation mit XenServer zu ermöglichen, wenn der Poolmaster ausfällt. Die Auswahl kann unter "Edit High Availability" auf dem hinzugefügten Host erfolgen.</li> <li>• Sie können auch einen GPU-Typ und eine GPU-Gruppe oder die Passthrough-Authentifizierung auswählen, wenn der XenServer vGPU unterstützt. Es wird angezeigt, ob die Auswahl dedizierte GPU-Ressourcen</li> </ul> |                                           |                                                                                                                             |                 |                |

|                                                                                   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | <p>umfasst.<br/><b>Beschreibung</b></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|                                                                                   | <p>Bei der Verwendung von VMware konsultieren Sie die Produktdokumentation sowie <a href="#">Vorbereiten der Virtualisierungsumgebung: VMware</a>.</p> <p>Bei Verwendung von Hyper-V konsultieren Sie die Produktdokumentation und <a href="#">Vorbereiten der Virtualisierungsumgebung: Microsoft System Center Virtual Machine Manager</a>.</p> <p>Entscheiden Sie, ob Sie Maschinenerstellungsdienste (MCS) oder andere Tools zum Erstellen von VMs auf den Virtualisierungsressourcen verwenden möchten.</p> <p>Entscheiden Sie, ob Sie freigegebenen oder lokalen Speicher verwenden. Freigegebener Speicher ist über das Netzwerk verfügbar. Wenn Sie einen freigegebenen Speicher verwenden, können Sie die Verwendung von IntelliCache aktivieren, um die Last auf dem Speichergerät zu reduzieren. Informationen finden Sie unter <a href="#">Verwenden von IntelliCache für XenServer-Verbindungen</a>.</p> <p>Entscheiden Sie, ob persönliche vDisks verwendet werden und ob diese freigegebenen oder lokalen Speicher verwenden. Persönliche vDisks können den gleichen oder anderen Speicher als die VMs verwenden.</p> <p>Wenn Sie Produktkomponenten in einer Cloudumgebung installiert haben, benötigen Sie beim Konfigurieren der ersten Verbindung den API-Schlüssel und die Werte des geheimen Schlüssels. Sie können die Schlüsseldatei mit diesen Werten aus AWS oder CloudPlatform exportieren und anschließend in den Assistenten für die Siteerstellung importieren.</p> <p>Wenn Sie eine Site für eine Cloudbereitstellung erstellen, benötigen Sie zudem die in AWS konfigurierten Werte für Region, Availability Zone, VPC-Name, Subnetzadressen, Domänenname, Namen der Sicherheitsgruppen und Anmeldeinformationen.</p> |
|                                                                                   | <p>Entscheiden Sie, ob Sie App-V-Veröffentlichung verwenden möchten, und konfigurieren Sie ggf. die entsprechenden Ressourcen.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|                                                                                   | <p>Gut zu wissen:</p> <ul style="list-style-type: none"> <li>Für das Erstellen einer Remote-PC-Zugriff-Site gilt Folgendes: <ul style="list-style-type: none"> <li>Es werden automatisch ein Maschinenkatalog unter dem Namen "Remote-PC-Zugriff-Maschinen" und eine Bereitstellungsgruppe unter dem Namen "Remote-PC-Zugriff-Desktops" erstellt.</li> <li>Sie müssen Benutzer bzw. Benutzergruppen angeben. Es gibt keine Standardaktion zum automatischen Hinzufügen aller Benutzer.</li> <li>Sie können das Wake-On-LAN-Feature (Energieverwaltung) aktivieren und die Adresse und Anmeldeinformationen für Microsoft System Center Configuration Manager (ConfigMgr) sowie einen Verbindungsnamen angeben.</li> </ul> </li> <li>Der Benutzer, der eine Site erstellt, ist Volladministrator dieser Site. Weitere Informationen finden Sie unter <a href="#">Delegierte Administration</a>.</li> <li>Wenn eine leere Datenbank erstellt wird, hat sie Standardattribute mit folgenden Ausnahmen: <ul style="list-style-type: none"> <li>Die Sortierreihenfolge wird auf Latin1_General_100_CI_AS_KS festgelegt, wobei Latin1_General je nach Land variiert, z. B. Japanese_100_CI_AS_KS). Wenn diese Sortiereinstellung bei der Datenbankerstellung nicht angegeben wird, schlägt die anschließende Erstellung der Dienstschemas in der Datenbank fehl und ein Fehler wie "Schema erfordert eine Datenbank, bei der Groß-/Kleinschreibung nicht beachtet wird." wird angezeigt. (Wenn eine Datenbank manuell erstellt wird, kann eine beliebige Sortierreihenfolge</li> </ul> </li> </ul>                                                                                                                                                         |

|   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|---|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ✓ | verwendet werden, wenn sie Groß- und Kleinschreibung sowie Akzente und japanische Schriftzeichen erkennt. Der Name der Sortierreihenfolge endet normalerweise mit _CI_AS_KS.)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|   | <b>Beschreibung</b> <ul style="list-style-type: none"> <li>• Der Wiederherstellungsmodus ist auf "Einfach" eingestellt. Für die Verwendung als gespiegelte Datenbank ändern Sie den Wiederherstellungsmodus zu "Vollständig".</li> <li>• Wenn Sie eine Sitekonfigurationsdatenbank erstellen, speichert diese auch Konfigurationsänderungen, die vom Konfigurationsprotokollierungsdienst aufgezeichnet werden, sowie Trend- und Leistungsdaten, die vom Überwachungsdienst verwendet und in Citrix Director angezeigt werden. Wenn Sie diese Features verwenden und Daten von mehr als sieben Tagen speichern, empfiehlt Citrix, nach der Erstellung einer Site die Datenbanken für die Konfigurationsprotokollierung und die Überwachung (sekundäre Datenbanken) an unterschiedlichen Orten zu speichern.</li> <li>• Bei Verwendung von Leerzeichen im Namen der Überwachungsdatenbank oder einer Sitekonfigurationsdatenbank, die die Überwachungsdatenbank enthält, treten Fehler auf, wenn auf die Datenbank zugegriffen wird. Weitere Informationen finden Sie unter <a href="#">CTX200325</a>.</li> <li>• Am Ende des Assistenten für die Siteerstellung werden Sie gefragt, ob Sie sich an dem Programm zur Verbesserung der Benutzerfreundlichkeit von Citrix beteiligen möchten. Wenn Sie diesem Programm beitreten, werden anonyme Statistik- und Nutzungsinformationen an Citrix gesendet. Weitere Informationen finden Sie unter <a href="#">Informationen zum Citrix Programm zur Verbesserung der Benutzerfreundlichkeit</a>.</li> </ul> |

Starten Sie Studio, falls es nicht geöffnet ist. Geben Sie nach dem Erstellen einer Site im mittleren Bereich Folgendes an:

- Den Sitetyp und den Namen der Site.
- Datenbankinformationen. Wenn Sie während der Controller-Installation die Installation der SQL Server Express-Standarddatenbank angefordert haben, werden einige Informationen bereits bereitgestellt. Wenn Sie einen Datenbankserver verwenden, der auf einem anderen Server installiert ist, geben Sie den Datenbankserver und -namen ein:

| Datenbanktyp                 | Eingabe                     | Auswirkungen dieser Datenbankkonfiguration                                                                                |
|------------------------------|-----------------------------|---------------------------------------------------------------------------------------------------------------------------|
| Eigenständig oder gespiegelt | Servername                  | Die Standardinstanz wird verwendet und SQL Server verwendet den Standardport.                                             |
|                              | servername\INSTANCENAME     | Eine benannte Instanz wird verwendet und SQL Server verwendet den Standardport.                                           |
|                              | servername,port-number      | Die Standardinstanz wird verwendet und SQL Server verwendet einen benutzerdefinierten Port. (Das Komma ist erforderlich.) |
| Sonstiges                    | cluster-name                | Eine geclusterte Datenbank.                                                                                               |
|                              | availability-group-listener | Eine stets aktivierte Datenbank.                                                                                          |

Nachdem Sie auf Weiter klicken und die Benachrichtigung erhalten, dass die Dienste keine Verbindung zu einer Datenbank herstellen konnten, geben Sie an, dass sie von Studio erstellt werden soll. Wenn Sie keine Berechtigung zum Bearbeiten der Datenbank haben, verwenden Sie Datenbankskript erstellen. Die Skripts müssen ausgeführt werden, bevor die Site vollständig erstellt wird.

- Lizenzserveradresse wie folgt: Name:[Port], wobei Name ein vollqualifizierter Domänenname (FQDN), eine NetBIOS- oder IP-Adresse ist. Das empfohlene Format ist FQDN. Wenn Sie die Portnummer auslassen, ist der Standardport 27000. Sie können erst fortfahren, wenn eine erfolgreiche Verbindung zum Lizenzserver hergestellt wurde.
- (Nur Remote-PC-Zugriff-Sites.) Energieverwaltungsinformationen, einschließlich ConfigMgr-Verbindungsinformationen.
- Informationen für die Verbindung mit der Virtualisierungsressource und Speicherinformationen. Wenn Sie keine Ressource verwenden oder Studio für die Verwaltung von Benutzerdesktops nutzen, die auf dedizierten Blade-PCs gehostet werden, wählen Sie den Verbindungstyp Keine.
- Angaben zum App-V-Verwaltungsserver und App-V-Veröffentlichungsserver.
- (Nur Remote-PC-Zugriff-Sites.) Benutzer- und Maschinenkonteninformationen.
  - Benutzerinformationen. Klicken Sie auf Benutzer hinzufügen. Wählen Sie Benutzer und Benutzergruppen aus und klicken Sie dann auf Benutzer hinzufügen.
  - Informationen zu Maschinenkonten Klicken Sie auf Maschinenkonten hinzufügen. Wählen Sie die Maschinenkonten aus und klicken Sie dann auf Maschinenkonten hinzufügen. Klicken Sie auf Organisationseinheiten hinzufügen. Wählen Sie die Domäne und die Organisationseinheiten und geben Sie an, ob Elemente in Unterordnern eingeschlossen werden. Klicken Sie auf Organisationseinheiten hinzufügen.

Sie können einen HTML-Bericht der Testergebnisse für die Site anzeigen. So führen Sie die Tests aus

1. Klicken Sie in Studio oben im linken Bereich auf den Eintrag Studio ().
2. Klicken Sie im mittleren Bereich auf Site testen.

# Installieren oder Entfernen von Virtual Delivery Agents mit Skripts

May 10, 2016

Das Installationsmedium enthält Beispielskripts, um Virtual Delivery Agents (VDAs) für Gruppen von Maschinen in Active Directory zu installieren, zu aktualisieren oder zu entfernen. Sie können die Skripts auch auf einzelne Maschinen anwenden und sie zum Verwalten von Masterimages einsetzen, die von den Maschinenerstellungsdiensten und Provisioning Services verwendet werden.

Erforderliche Zugriffsberechtigungen:

- Für die Skripts ist Lesezugriff für "Jeder" auf der Netzwerkfreigabe erforderlich, auf der der VDA-Installationsbefehl ist. Der Installationsbefehl ist beim vollständigen Produkt-ISO-Image "XenDesktopVdaSetup.exe" und beim eigenständigen Installationsprogramm "VDAWorkstationSetup.exe" bzw. "VDAServerSetup.exe".
- Die Protokolldetails werden auf jeder lokalen Maschine gespeichert. Sollen die Ergebnisse auch zentral zur Überprüfung und Analyse protokolliert werden, benötigen die Skripts Lese- und Schreibzugriff auf der Netzwerkfreigabe für "Jeder".

Um die Ergebnisse der Skriptausführung zu überprüfen, müssen Sie die zentrale Protokollfreigabe untersuchen. Erfasst werden das Skriptprotokoll, das Installationsprogrammprotokoll und die MSI-Installationsprotokolle. Jeder Installations- oder Deinstallationsvorgang wird in einem Ordner mit Zeitstempel aufgezeichnet. Am Präfix "PASS" oder "FAIL" im Ordnername ist ersichtlich, ob der Vorgang erfolgreich war oder nicht. Sie können herkömmliche Verzeichnissuchprogramme verwenden, um schnell eine fehlerhafte Installation oder Deinstallation im zentralen Protokoll zu finden, statt lokal auf den Zielmaschinen zu suchen.

Wichtig: Vor Beginn einer Installation führen Sie die unter [Vorbereiten der Installation](#) beschriebenen Schritte durch.

1. Besorgen Sie sich das Beispielskript InstallVDA.bat aus \Support\AdDeploy\ auf dem Installationsmedium. Citrix empfiehlt, dass Sie ein Backup der ursprünglichen Skriptdatei anlegen, bevor Sie sie ändern.
2. Bearbeiten Sie das Skript:
  - Geben Sie die Version des zu installierenden VDA an: SET DESIREDVERSION. Beispielsweise kann die Version 7 als 7.0 angegeben werden. Der vollständige Wert befindet sich auf dem Installationsmedium in der Datei ProductVersion.txt (z. B. 7.0.0.3018); eine vollständige Übereinstimmung ist jedoch nicht erforderlich.
  - Geben Sie die Netzwerkfreigabe an, von der das Installationsprogramm aufgerufen wird. Verweisen Sie auf den Stamm des Layouts (der höchste Punkt der Struktur): die geeignete Version des Installationsprogramms (32 Bit oder 64 Bit) wird automatisch aufgerufen, wenn das Skript ausgeführt wird. Beispiel: SET DEPLOYSHARE=\\fileservers1\share1.
  - Geben Sie optional einen Netzwerkfreigabeort zum Speichern der zentralen Protokolle an. Beispiel: SET LOGSHARE=\\fileservers1\log1).
  - Geben Sie die VDA-Konfigurationsoptionen wie unter [Installieren an der Befehlszeile](#) beschrieben ein. Die Optionen /quiet und /noreboot sind standardmäßig im Skript enthalten und sind erforderlich: SET COMMANDLINEOPTIONS=/QUIET /NOREBOOT.
3. Weisen Sie mit den Startskripts für Gruppenrichtlinien das Skript der Organisationseinheit in Active Directory zu, in der die Maschinen sind. Diese Organisationseinheit sollte nur Maschinen enthalten, auf denen Sie VDA installieren möchten. Wenn die Maschinen in der Organisationseinheit neu gestartet werden, wird das Skript auf allen ausgeführt und VDA auf jeder Maschine installiert, deren Betriebssystem unterstützt wird.

1. Besorgen Sie sich das Beispielskript UninstallVDA.bat aus \Support\AdDeploy\ auf dem Installationsmedium. Citrix empfiehlt, dass Sie ein Backup der ursprünglichen Skriptdatei anlegen, bevor Sie sie ändern.
2. Bearbeiten Sie das Skript.
  - Geben Sie die Version des zu entfernenden VDAs an: SET CHECK\_VDA\_VERSION. Beispielsweise kann die Version 7 als 7.0 angegeben werden. Der vollständige Wert befindet sich auf dem Installationsmedium in der Datei ProductVersion.txt (z. B. 7.0.0.3018); eine vollständige Übereinstimmung ist jedoch nicht erforderlich.
  - Geben Sie optional einen Netzwerkfreigabeort zum Speichern der zentralen Protokolle an.
3. Weisen Sie mit den Startskripts für Gruppenrichtlinien das Skript der Organisationseinheit in Active Directory zu, in der die Maschinen sind. Diese Organisationseinheit sollte nur Maschinen enthalten, von denen Sie VDA entfernen möchten. Wenn die Maschinen in der Organisationseinheit neu gestartet werden, wird das Skript auf allen ausgeführt und VDA auf jeder Maschine entfernt.

Das Skript generiert interne Protokolldateien, die den Skriptausführungsverlauf beschreiben. Das Skript kopiert das Protokoll Kickoff\_VDA\_Startup\_Skript innerhalb von Sekunden nachdem die Bereitstellung auf der Maschine gestartet wurde in die zentrale Protokollfreigabe, sodass Sie den Gesamtvorgang überprüfen können. Wenn dieses Protokoll nicht wie vorgesehen in die Protokollfreigabe kopiert wird, können Sie zur weiteren Fehlerbehebung die lokale Maschine untersuchen, denn durch das Skript werden zwei Debugging-Protokolldateien im Ordner %temp% auf jeder Maschine gespeichert:

- Kickoff\_VDA\_Startup\_Script\_.log
- VDA\_Install\_ProcessLog\_.log

Überprüfen Sie den Inhalt dieser Protokolle, um Folgendes für das Skript sicherzustellen:

- Es wird wie erwartet ausgeführt.
- Das Zielbetriebssystem wird korrekt erkannt.
- Der Verweis auf ROOT von DEPLOYSHARE ist korrekt konfiguriert (enthält die Datei "AutoSelect.exe").
- Die Authentifizierung bei den Freigaben DEPLOYSHARE und LOG ist möglich.



# Installieren von VDAs mit dem eigenständigen Paket

Nov 02, 2016

Statt des vollständigen XenApp- oder XenDesktop-ISO-Produktimages können Sie zum Installieren von Virtual Delivery Agents (VDAs) ein eigenständiges VDA-Installationspaket verwenden. Das kleinere Paket ist besser für Bereitstellungen auf physischen Maschinen oder an Remotestandorten geeignet, da es über Electronic Software Delivery (ESD)-Pakete als limitierte Bereitstellung oder lokale Kopie bereitgestellt wird.

Das eigenständige VDA-Paket ist in erster Linie für Bereitstellungen über die Befehlszeile (unbeaufsichtigte Installation) gedacht, da es die gleichen Befehlszeilenparameter wie der Befehl `XenDesktopVdaSetup.exe` unterstützt, der vom vollständigen Produktinstallationsprogramm verwendet wird. Das Paket bietet zudem eine grafische Oberfläche, die dem VDA-Installer des vollständigen ISO-Produktimages gleicht.

Es gibt zwei selbstextrahierende eigenständige VDA-Pakete: eines zur Installation auf unterstützten Serverbetriebssystemmaschinen und eines für unterstützte Workstation- bzw. Desktopbetriebssystemmaschinen.

Die unterstützten Betriebssystemen für VDAs sowie andere Anforderungen, die vor der Installation erfüllt sein müssen, finden Sie unter [Systemanforderungen für XenApp und XenDesktop 7.6](#). Unter [Vorbereiten der Installation](#) finden Sie Details zu den während der VDA-Installation von Ihnen anzugebenden Informationen und zu treffenden Entscheidungen.

**XenApp und XenDesktop 7.6 FP3:** Das VDA-Paket stellt auf der Maschine nicht vorhandene Voraussetzungen automatisch bereit, einschließlich Visual C++ 2008, 2010 und 2013 Runtimes (32-Bit und 64-Bit-Versionen) und .NET Framework 4.5.1.

Die .NET - Anforderung wird vom Installationsprogramm ausgelassen, wenn auf dem Desktop, auf dem Sie den VDA installieren möchten, Windows 10 installiert ist, das .NET 4.6 enthält.

**XenApp und XenDesktop 7.6, 7.6 FP1 und 7.6 FP2:** Das VDA-Paket stellt auf der Maschine nicht vorhandene Voraussetzungen automatisch bereit, einschließlich Microsoft Visual C++ 2005, 2008 und 2010 Runtimes (32-Bit und 64-Bit-Versionen) und .NET Framework 4.5.1.

Bei der Installation auf einer unterstützten Serverbetriebssystemmaschine werden die Rollendienste für die Remotedesktopdienste (RDS) installiert und aktiviert, bevor der VDA installiert wird. Sie können die Voraussetzungen vor der Installation des VDAs jedoch auch selbst installieren.

Ausnahme: Stellen Sie sicher, dass auf Maschinen mit Windows Server 2008 R2 und Windows 7 mindestens .NET 3.5.1 installiert ist, bevor Sie mit der Installation von VDA beginnen.

## Info über Neustarts

- Bei der VDA-Installation ist zum Abschluss ein Neustart erforderlich.
- Um während der Installation möglichst wenige zusätzliche Neustarts durchführen zu müssen, stellen Sie vor Beginn der VDA-Installation sicher, dass .NET Framework 4.5.1 oder 4.5.2 installiert ist. Installieren und aktivieren Sie auf Windows-Serverbetriebssystemmaschinen vor der VDA-Installation zudem die Rollendienste für Remotedesktopdienste. (Andere Voraussetzungen erfordern normalerweise keine Neustarts, daher können Sie das Installationsprogramm einfach laufen lassen.)
- Wenn Sie vor der VDA-Installation nicht die Voraussetzungen installieren und dann die Option `/noreboot` für die Befehlszeileninstallation festlegen, müssen Sie die Neustarts ausführen. Wenn Sie beispielsweise die Voraussetzungen

automatisch bereitstellen, wartet das Installationsprogramm nach der Installation von RDS auf einen Neustart. Führen Sie nach dem Neustart den Befehl erneut aus, damit die Installation des VDAs fortgesetzt wird.

Wenn Sie an der grafischen Oberfläche oder der Befehlszeilenschnittstelle die Option zum Ausführen des Pakets verwenden, werden die Dateien des Pakets in den Ordner "Temp" extrahiert. Zum Extrahieren in den Ordner "Temp" wird auf der Maschine mehr Speicherplatz beansprucht, als wenn Sie das vollständige ISO-Produktimage verwenden. Dateien, die in den Ordner "Temp" extrahiert wurden, werden nicht automatisch gelöscht, aber Sie können sie nach Abschluss der Installation manuell löschen (aus C:\Windows\Temp\Ctx-\*, wobei \* ein willkürlicher GUID ist).

Sie können auch mit dem Hilfsprogramm eines Drittanbieters, das CAB-Archivdateien aus EXE-Dateien extrahieren kann (z. B. 7-Zip), die Dateien in ein Verzeichnis Ihrer Wahl extrahieren und dann den Befehl "XenDesktopVdaSetup.exe" ausführen. In 7.6 FP3 können Sie den Befehl /extract mit einem absoluten Pfad verwenden. Weitere Informationen finden Sie unter "Verwendung" im Abschnitt weiter unten.

Wenn in der Bereitstellung Microsoft System Center Configuration Manager verwendet wird, schlägt eine VDA-Installation u. U. scheinbar mit Exitcode 3 fehl, obwohl der VDA erfolgreich installiert wird. Sie können diese irreführende Meldung vermeiden, indem Sie die Installation mit einem CMD-Script umschließen oder die Erfolgscodes im Configuration Manager-Paket ändern. Weitere Informationen finden Sie in dieser [Forumsdiskussion](#).

**Citrix Display Only Driver in XenApp und XenDesktop 7.6 FP3**

Der Citrix Display Only Driver (DOD) ist der einzige installierte und unterstützte Anzeigetreiber auf dem XenDesktop Standard-VDA unter Windows 10.

Der Citrix DOD hat keine GPU-Unterstützung, selbst wenn ein GPU oder vGPU vorhanden ist. Alle Wiedergabevorgänge werden vom Microsoft Basic Renderer in der Software mit der CPU ausgeführt. Der Citrix DOD bietet keine Unterstützung für die Desktopgestaltungsumleitung (DCR). Der Citrix DOD ist nicht auf XenApp installiert und wird nicht unterstützt.

Wichtig: Für die Installation benötigen Sie erhöhte Administratorrechte vor Beginn der Installation oder verwenden Sie die Option "Als Administrator ausführen".

- 1. Laden Sie das entsprechende Paket auf die Maschine herunter, auf der Sie den VDA installieren möchten. Zum Zugriff auf die Downloadsite benötigen Sie Citrix Kontoanmeldeinformationen.

| Wo installieren Sie den VDA?                                            | Dieses Paket herunterladen |
|-------------------------------------------------------------------------|----------------------------|
| Auf einer unterstützten Serverbetriebssystemmaschine                    | VDAServerSetup.exe         |
| Auf einer unterstützten Workstation- bzw. Desktopbetriebssystemmaschine | VDAServerSetup.exe         |

Verwenden Sie für Einzelbenutzer-Einzelserverbereitstellungen (z. B. beim Bereitstellen von Windows Server 2012 für einen Benutzer zur Webentwicklung) das Paket VDAWorkstationSetup.exe. Weitere Informationen finden Sie unter [Server-VDI](#).

- 2. Installieren Sie den VDA mit der grafischen Oberfläche oder der Befehlszeilenschnittstelle.  
Nicht vergessen: Für die Installation benötigen Sie erhöhte Administratorrechte vor Beginn der Installation oder verwenden Sie die Option **Als Administrator ausführen**.  
**Installieren über die grafische Oberfläche:**

Deaktivieren Sie die Benutzerkontensteuerung (UAC), klicken Sie mit der rechten Maustaste auf das Downloadpaket und wählen Sie **Als Administrator ausführen**. Das Installationsprogramm wird gestartet und der Installationsassistent wird ausgeführt. Zum Abschluss des Assistenten ist ein Neustart erforderlich, bevor Sie den VDA in einer Site verwenden können. (Der Assistent ist derselbe wie der, der beim vollständigen ISO-Produktimage den VDA installiert.)

### Installieren über die Befehlszeilenoberfläche:

Sie haben zwei Optionen:

- **Extrahieren Sie die Dateien aus dem Paket und führen Sie XenDesktopVdaSetup.exe aus.**

**XenApp und XenDesktop 7.6 FP3:** Verwenden Sie zum Extrahieren der Dateien vor der Installation /extract mit dem absoluten Pfad, z. B.:

```
.\VDAWorkstationSetup.exe /extract %temp%\CitrixVDAInstallMedia
```

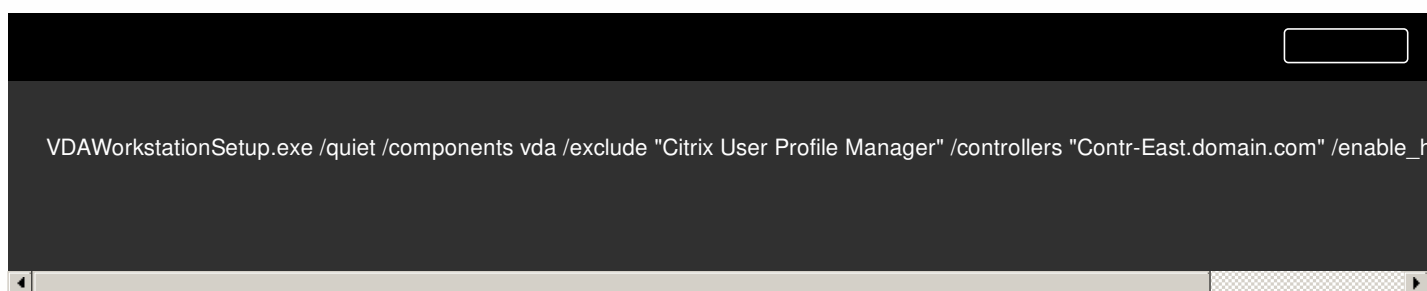
Führen Sie dann in einem separaten Befehl XenDesktopVdaSetup.exe in dem Verzeichnis mit dem extrahierten Inhalt aus. Weitere Informationen zu Parametern finden Sie unter [Installieren an der Befehlszeile](#) und [CTX140972](#).

**XenApp und XenDesktop 7.6, 7.6 FP1 und 7.6 FP2:** Verwenden Sie zum Extrahieren der Dateien vor der Installation das Hilfsprogramm eines Drittanbieters, das CAB-Archivdateien aus EXE-Dateien extrahieren kann (z. B. 7-Zip). Führen Sie dann in einem separaten Befehl XenDesktopVdaSetup.exe in dem Verzeichnis mit dem extrahierten Inhalt aus. Weitere Informationen zu Parametern finden Sie unter [Installieren an der Befehlszeile](#) und [CTX140972](#).

- **Führen Sie das Paket aus.**

Führen Sie das Downloadpaket aus als wäre es der Befehl XenDesktopVdaSetup.exe. Weitere Informationen zu Parametern finden Sie unter [Installieren an der Befehlszeile](#) und [CTX140972](#).

Beispielsweise wird mit dem gebräuchlichsten Installationsbefehl für Remote-PC-Zugriff ein VDA auf einem physischen Büro-PC ohne Citrix Receiver oder Citrix Profile Manager installiert. Nach der Installation des VDA wird die Maschine nicht automatisch neu gestartet. Ein Neustart ist jedoch erforderlich, bevor der VDA verwendet werden kann. Zu Beginn wird der VDA auf dem Server 'Contr-East' beim Controller registriert. Wenn der Windows-Firewalldienst erkannt wird, werden Ports geöffnet.



## Hinweis

Das Ausschließen der Citrix Profilverwaltung von der Installation (über die Option `/exclude 'Citrix User Profile Manager'`) hat Auswirkungen auf die Überwachung und Problembehandlung von VDAs mit Citrix Director. Auf den Seiten "Benutzerdetails" und "Endpunkte" treten Fehler in den Bereichen "Personalisierung" und "Anmeldedauer" auf. Auf den Seiten "Dashboard" und "Trends" werden im Bereich "Durchschnittliche Anmeldedauer" nur Daten für Maschinen angezeigt, auf denen Profilverwaltung installiert ist.

Selbst wenn Sie eine Lösung zur Benutzerprofilverwaltung eines Drittanbieters verwenden, wird empfohlen, den Citrix Profilverwaltungsdienst zu installieren und auszuführen, damit die Überwachung und Problembehandlung in Citrix Director möglich ist.

ist (das Aktivieren des Citrix Profilverwaltungsdiensts ist nicht erforderlich).

# Maschinenkataloge

May 10, 2016

Sammlungen von physischen oder virtuellen Maschinen werden als Einheit ("Sitzungsmaschinenkatalog") verwaltet. Bei vielen Bereitstellungen wird ein Masterimage oder eine Vorlage auf dem Host erstellt und dann im Maschinenkatalog als Basis für Citrix Tools (z. B. Maschinenerstellungsdienste oder Provisioning Services) zum Erstellen von VMs verwendet. Ein Katalog kann auch physische Maschinen enthalten.

Wenn Sie einen Maschinenkatalog erstellt haben, werden automatisch Tests ausgeführt, um sicherzustellen, dass er richtig konfiguriert wurde. Wenn die Tests abgeschlossen sind, können Sie einen Testbericht anzeigen. Sie können diese Tests bei Bedarf auch später über Citrix Studio site-name im Studio-Navigationsbereich ausführen.

Wenn die Tests abgeschlossen sind, erstellen Sie eine [Bereitstellungsgruppe](#) .

# Erstellen eines Maschinenkatalogs

May 10, 2016

Wenn Sie Citrix Tools (Maschinenerstellungsdienste oder Provisioning Services) zum Erstellen von VM für die Bereitstellung verwenden möchten, erstellen Sie auf dem Host-Hypervisor ein Masterimage oder eine Vorlage. Erstellen Sie dann den Maschinenkatalog.

Stellen Sie sicher, dass der Host genügend Prozessoren, Arbeitsspeicher und Speicher zur Unterbringung der Maschinen hat, die Sie erstellen möchten.

Das Masterimage enthält das Betriebssystem, nicht virtualisierte Anwendungen, den VDA und andere Software. VMs werden in einem Maschinenkatalog basierend auf einem zuvor erstellten und beim Erstellen des Katalogs angegebenen Masterimage erstellt.

Gut zu wissen:

- Masterimages werden auch als Klonimage, Golden Image oder Basisimage bezeichnet.
- Bei Cloudbereitstellungen werden statt Masterimages Vorlagen verwendet. Informationen zu Vorlagen finden Sie
  - in Amazon Web Services unter [Deploy XenApp and XenDesktop 7.5 and 7.6 with Amazon VPC](#)
  - in Citrix CloudPlatform unter [XenApp und XenDesktop – Konzepte und Bereitstellung auf CloudPlatform](#).

Wenn Sie Provisioning Services verwenden, können Sie entweder ein Masterimage oder einen physischen Computer als Masterzielgerät verwenden.

- Bei Remote-PC-Zugriff-Maschinenkatalogen werden keine Masterimages verwendet.
- Hinweise zur Microsoft KMS-Aktivierung bei Verwendung der Maschinenerstellungsdienste:
  - Wenn Ihre Bereitstellung 7.x-VDA mit einem XenServer 6.1- oder 6.2-Host, einem vSphere-Host oder einem Microsoft System Center Virtual Machine Manager-Host enthält, müssen Sie Microsoft Windows oder Microsoft Office nicht manuell zurücksetzen.
  - Wenn die Bereitstellung einen 5.x-VDA mit einem XenServer 6.0.2-Host enthält, lesen Sie [CTX128580](#).

Wichtig: Wenn Sie Provisioning Services oder Maschinenerstellungsdienste verwenden, führen Sie auf den Masterimages nicht Sysprep aus.

1. Erstellen Sie mit dem Verwaltungstool des Hypervisors ein neues Masterimage und installieren Sie dann das Betriebssystem sowie alle Service Packs und Updates  
Die Anzahl der vCPUs und die Größe des Speichers sind zu diesem Zeitpunkt nicht wichtig, da Sie diese Werte ändern können, wenn Sie den Maschinenkatalog erstellen. Konfigurieren Sie jedoch den für Desktops und Anwendungen erforderlichen Speicherplatz auf der Festplatte, da dieser Wert später nicht im Katalog geändert werden kann.
2. Stellen Sie sicher, dass die Festplatte am Gerätestandort 0 verbunden ist. Dieser Standort ist in den meisten Standardmasterimagevorlagen automatisch konfiguriert; in einigen benutzerdefinierten Vorlagen ist dies jedoch nicht unbedingt der Fall.
3. Installieren und konfigurieren Sie die folgende Software auf dem Masterimage:
  - Integrationstools für den Hypervisor (z. B. XenServer-Tools, Hyper-V-Integrationsdienste oder VMware-Tools). Wenn Sie diesen Schritt auslassen, funktionieren die Anwendungen und Desktops unter Umständen nicht richtig.
  - Einen VDA für Windows-Serverbetriebssysteme oder für Windows-Desktopbetriebssysteme (Citrix empfiehlt die Installation der neuesten Version, damit die neuesten Features verfügbar sind). Aktivieren Sie während der Installation

die Optimierungsoption zur Verbesserung der Leistung durch Neukonfigurierung bestimmter Windows-Features.

- Tools von Drittanbietern, zum Beispiel Antivirensoftware oder Agents zur elektronischen Softwareverteilung. Konfigurieren Sie Dienste wie Windows Update mit den für Benutzer und Maschinentyp geeigneten Einstellungen.
  - Anwendungen von Drittanbietern, die Sie nicht virtualisieren möchten. Citrix empfiehlt das Virtualisieren von Anwendungen, da dies die Kosten erheblich senkt, denn das Masterimage muss nach dem Hinzufügen oder Neukonfigurieren einer Anwendung nicht aktualisiert werden. Außerdem belegen weniger installierte Anwendungen weniger Platz auf Masterimage-Festplatten, wodurch Speicherkosten eingespart werden.
  - App-V-Clients mit den empfohlenen Einstellungen, wenn Sie App-V-Anwendungen veröffentlichen möchten.
  - Wenn Sie Maschinenerstellungsdienste verwenden und Microsoft Windows in lokalisierter Version ausführen möchten, installieren Sie die Gebietsschemas und Sprachpakete. Wenn ein Snapshot beim Provisioning erstellt wird, verwenden die bereitgestellten VMs die installierten Gebietsschemas und Sprachpakete.
4. Bei Verwendung von Provisioning Services erstellen Sie eine VHD-Datei für die vDisk von Ihrem Masterzielgerät, bevor Sie das Masterzielgerät in eine Domäne einbinden.
  5. Fügen Sie das Masterimage der Domäne hinzu, der Desktops und Anwendungen angehören werden, und stellen Sie sicher, dass das Masterimage auf dem Host verfügbar ist, auf dem die Maschinen erstellt werden.
  6. Citrix empfiehlt, dass Sie einen Snapshot des Masterimages erstellen und benennen, damit es künftig identifiziert werden kann. Wenn Sie ein Masterimage anstelle eines Snapshots beim Erstellen eines Maschinenkatalogs angeben, erstellt Studio automatisch einen Snapshot, der jedoch nicht umbenannt werden kann.

**Vorbereiten eines Masterimages für GPU-fähige Maschinen auf XenServer** Wenn Sie für Ihre Hostinginfrastruktur XenServer verwenden, benötigen GPU-fähige Maschinen ein dediziertes Masterimage. Diese VMs erfordern Videotreiber, die GPUs unterstützen und so konfiguriert sind, dass die VMs Software verwenden können, die den GPU für Vorgänge verwendet.

1. Erstellen Sie in XenCenter eine VM mit Standard-VGA sowie Netzwerken und einer vCPU.
2. Aktualisieren Sie die VM-Konfiguration so, dass der GPU (entweder Passthrough oder vGPU) verwendet werden kann.
3. Installieren Sie ein unterstütztes Betriebssystem und aktivieren Sie RDP.
4. Installieren Sie XenServer-Tools und NVIDIA-Treiber.
5. Deaktivieren Sie die VNC-Verwaltungskontrolle (Virtual Network Computing), um die Leistung zu optimieren, und starten Sie anschließend die VM neu.
6. Sie werden aufgefordert, RDP zu verwenden. Installieren Sie mit RDP den VDA und starten Sie dann die VM neu.
7. Optional können Sie einen Snapshot der VM erstellen und als Vorlage für andere GPU-Masterimages verwenden.
8. Installieren Sie mit RDP kundenspezifische Anwendungen, die in XenCenter konfiguriert werden und GPU-Funktionen verwenden.

Bevor Sie den Assistenten zum Erstellen von Maschinenkatalogen starten, lesen Sie die nachfolgenden Anweisungen, damit Sie wissen, welche Optionen Sie auswählen und welche Informationen Sie angeben müssen. Je nach Ihrer Umgebung und der von Ihnen getroffenen Auswahl werden einige Elemente des Assistenten möglicherweise nicht angezeigt oder heißen anders.

In Studio:

- Wenn Sie eine Site, jedoch noch keinen Maschinenkatalog erstellt haben, führt Studio Sie zum richtigen Startpunkt zur Erstellung eines Maschinenkatalogs.
- Wenn Sie bereits einen Maschinenkatalog erstellt haben und einen weiteren erstellen möchten, wählen Sie im Studio-Navigationsbereich Maschinenkataloge und dann im Aktionsbereich Maschinenkatalog erstellen.

Der Assistent führt Sie durch die nachfolgend beschriebenen Einstellungen.

- Betriebssystem

Jeder Katalog enthält nur Maschinen eines Typs:

- Windows-Serverbetriebssystem: Ein Maschinenkatalog für Windows-Serverbetriebssysteme bietet Desktops und Anwendungen, die von mehreren Benutzern verwendet werden können.
- Windows-Desktopbetriebssystem: Ein Maschinenkatalog für Windows-Desktopbetriebssysteme bietet Desktops und Anwendungen, die einzelnen Benutzern zugewiesen werden.
- Remote-PC-Zugriff: Ein Remote-PC-Zugriff-Katalog bietet Benutzern Remotezugriff auf ihre physischen Büro-Desktopmaschinen. Bei Remote-PC-Zugriff wird VPN nicht für die Sicherheit benötigt.

Amazon Web Services (AWS) unterstützt nur Kataloge für Serverbetriebssystemmaschinen (und Server-VDI, siehe [Server-VDI](#)), nicht aber solche für Desktopbetriebssystemmaschinen oder Remote-PC-Zugriff-Maschinen.

- Maschinenverwaltung

Geben Sie an, ob für Maschinen in dem Katalog Energieverwaltung über Studio ausgeführt werden soll:

- Maschinen mit Energieverwaltung über Studio oder über eine Cloudumgebung bereitgestellte Maschinen (z. B. VM oder Blade-PC). Diese Option ist nur verfügbar, wenn Sie bereits eine Hypervisor- oder Cloudumgebungsverbindung konfiguriert haben. In aller Regel wird eine Verbindung beim Erstellen der Site konfiguriert. Wenn dies nicht der Fall ist, können Sie später eine neue Verbindung erstellen und dann den Maschinenkatalog bearbeiten.
- Maschinen ohne Energieverwaltung (z. B. physische Maschinen).

Geben Sie an, welches Tool Sie für die Bereitstellung von Maschinen verwenden möchten:

- Maschinenerstellungsdienste (MCS): verwendet ein Masterimage oder eine Vorlage zum Erstellen und Verwalten virtueller Maschinen.
  - MCS ist für physische Maschinen nicht verfügbar.
  - Bei Maschinenkatalogen in Cloudumgebungen wird MCS verwendet.
- Provisioning Services: verwaltet Zielgeräte als Gerätesammlung. Eine als Image eines Masterzielgeräts erstellte Provisioning Services-vDisk liefert Desktops und Anwendungen.
- Anderes: Tool, mit dem bereits Maschinen im Datacenter verwaltet werden. Citrix empfiehlt die Verwendung von Microsoft System Center Configuration Manager oder einer anderen Drittanbieteranwendung, um sicherzustellen, dass die Maschinen in dem Katalog konsistent sind.

- Desktoperfahrung

Führen Sie für Maschinenkataloge mit Desktopbetriebssystemmaschinen, die für die Bereitstellung von Desktops verwendet werden sollen, folgende Schritte aus:

- Geben Sie an, ob Benutzer bei jeder Anmeldung mit einem neuen (zufälligen) Desktop oder immer mit dem gleichen (statischen) Desktop verbunden werden.
- Wenn Benutzer mit dem gleichen Desktop verbunden werden, geben Sie an, wie Änderungen, die Benutzer an dem Desktop vornehmen, behandelt werden sollen. Änderungen können auf einer eigenen Personal vDisk oder auf dem lokalen VM-Datenträger des Benutzers gespeichert werden oder sie können gelöscht werden. (Wenn Sie sich für die Speicherung auf einer Personal vDisk entscheiden, geben Sie den Laufwerksbuchstaben und die Größe später im Assistenten an.)

- Masterimage oder Maschinenvorlage

Wählen Sie das Masterimage (keine Cloud) oder die Maschinenvorlage (Cloud), das bzw. die Sie zuvor erstellt haben.

Wichtig: Wenn Sie Provisioning Services oder Maschinenerstellungsdienste verwenden, führen Sie auf den Masterimages nicht Sysprep aus.

- Sicherheit

Cloudumgebungen: Wählen Sie mindestens eine Sicherheitsgruppe für die virtuellen Maschinen. Diese werden nur angezeigt, wenn die Verfügbarkeitszone Sicherheitsgruppen unterstützt. Entscheiden Sie, ob die Maschinen Hardware gemeinsam oder dedizierte Hardware verwenden sollen.



- VMs oder Gerätesammlung oder Virtuelle Maschinen und Benutzer

Legen Sie fest, wie viele virtuelle Maschinen erstellt werden sollen. Sie können wählen, wie viele virtuelle CPUs und wie viel Arbeitsspeicher (in MB) jede Maschine haben soll. Jede VM hat eine 32-GB-Festplatte. Dieser Wert ist im Masterimage festgelegt und kann im Katalog nicht geändert werden.

Wenn Sie zuvor festgelegt haben, dass die Änderungen der Benutzer an Desktops auf separaten Personal vDisks gespeichert werden sollen, geben Sie deren Größe in Gigabyte und den Laufwerksbuchstaben an.

Wenn Sie mehrere Netzwerkkarten (NICs) verwenden möchten, weisen Sie jeder Karte ein virtuelles Netzwerk zu. Sie können beispielsweise einer Karte ein bestimmtes sicheres Netzwerk und einer anderen ein häufiger verwendetes Netzwerk zuweisen. In diesem Assistenten können Sie auch Netzwerkkarten hinzufügen und entfernen.

- Maschinenkonten

Remote-PC-Zugriff-Kataloge: Geben Sie die hinzuzufügenden Active Directory-Maschinenkonten oder Organisationseinheiten an, die Benutzern oder Benutzergruppen entsprechen.

Sie können eine zuvor konfigurierte Energieverwaltungsverbindung auswählen oder die Energieverwaltung nicht verwenden. Wenn Sie die Energieverwaltung verwenden möchten, jedoch noch keine geeignete Verbindung konfiguriert wurde, können Sie die Verbindung später erstellen und dann die Energieverwaltungseinstellungen des Maschinenkatalogs entsprechend bearbeiten.

- Computerkonten

Jede Maschine im Maschinenkatalog benötigt ein Active Directory-Computerkonto. Geben Sie an, ob neue Konten erstellt oder vorhandene Konten verwendet werden sollen, und geben Sie den Speicherort für diese Konten an. Wenn Sie vorhandene Konten verwenden, stellen Sie sicher, dass Sie genügend unbenutzte Computerkonten für die Maschinen haben, die erstellt werden sollen.

Sie können Active Directory nach vorhandenen Konten durchsuchen oder eine CSV-Datei mit den Kontonamen importieren. Die importierte Datei muss folgendes Format aufweisen:

[ADComputerAccount] ADcomputeraccountname.domain ...

Bei Katalogen mit physischen oder vorhandenen Maschinen wählen Sie vorhandene Konten aus oder importieren Sie diese, und weisen Sie jeder Maschine sowohl ein Active Directory-Computerkonto als auch ein Benutzerkonto zu.

Bei Maschinen, die mit Provisioning Services erstellt wurden, werden Computerkonten für Zielgeräte anders verwaltet. Weitere Informationen hierzu finden Sie in der Dokumentation zu Provisioning Services.

Geben Sie außerdem das Kontobenennungsschema für die zu erstellenden Maschinen an. Hashzeichen (#) im Schema repräsentieren sequenzielle Zahlen bzw. Buchstaben, die in den zusätzlichen, von Ihnen eingegebenen Namenstext eingeschlossen werden.

- Name und Beschreibung

Auf der letzten Seite des Assistenten geben Sie den Namen und die Beschreibung für den Maschinenkatalog ein. Diese Informationen werden in Studio angezeigt.

# Verwalten von Maschinenkatalogen

May 10, 2016

Bei Katalogen für zufällige Maschinen können Sie Benutzerdesktops verwalten, indem Sie globale Änderungen (z. B. Windows-Updates, Updates der Antivirensoftware, Betriebssystemupdates oder Konfigurationsänderungen) auf das Masterimage anwenden. Ändern Sie anschließend den Maschinenkatalog so, dass dieser das aktualisierte Masterimage verwendet, damit Benutzern bei der nächsten Anmeldung der aktualisierte Desktop zur Verfügung steht. Sie können wichtige Änderungen für eine große Benutzerzahl in einem Arbeitsgang vornehmen.

Bei statischen und Remote-PC-Zugriff-Maschinenkatalogen müssen Sie Updates an Benutzerdesktops außerhalb von Studio entweder einzeln oder kollektiv mit Bereitstellungssoftware von Drittanbietern verwalten. Bei Maschinen, die mit Provisioning Services erstellt wurden, werden Updates an Benutzerdesktops über vDisk verteilt.

Citrix empfiehlt, vor dem Durchführen von Updates Kopien oder Snapshots der Masterimages zu speichern. In der Datenbank wird von jedem Masterimage eines Maschinenkatalogs ein historischer Datensatz beibehalten. Masterimages dürfen nicht gelöscht, verschoben oder umbenannt werden. Bei Problemen mit Updates, die auf den Benutzerdesktops bereitgestellt wurden, können Sie das Masterimage auf die vorherige Version zurücksetzen und die Downtime für Benutzer minimieren.

Vorbereitungen:

- Stellen Sie sicher, dass der Virtualisierungshost genügend Prozessoren, Arbeitsspeicher und Speicher zur Unterbringung der zusätzlichen Maschinen hat.
- Stellen Sie sicher, dass Sie genügend ungenutzte Active Directory-Computerkonten haben. Wenn Sie vorhandene Konten verwenden, wird die Anzahl der Maschinen, die Sie hinzufügen können, durch die Anzahl der verfügbaren Konten beschränkt.
- Wenn Sie Active Directory-Computerkonten für die zusätzlichen Maschinen mit Studio erstellen, müssen Sie die erforderlichen Domänenadministratorrechte haben.

1. Wählen Sie im Studio-Navigationsbereich Maschinenkataloge aus.
2. Wählen Sie einen Maschinenkatalog und dann im Aktionsbereich Maschinen hinzufügen.
3. Legen Sie die Anzahl der hinzuzufügenden virtuellen Maschinen fest.
4. Wenn Sie angeben, dass neue Active Directory-Konten erstellt werden sollen (dieser Schritt ist erforderlich, wenn nicht genügend Konten für die Anzahl der hinzuzufügenden VMs vorhanden sind):
  - Wählen Sie die Domäne und den Speicherort, wo die Konten erstellt werden sollen.
  - Legen Sie ein Kontobenennungsschema mit Hashmarkierungen zur Kennzeichnung der Platzierung sequenzieller Ziffern bzw. Buchstaben fest (ein Name darf nicht mit einer Ziffer beginnen). Beispiel: Das Benennungsschema "PC-Vertrieb-##" (und Aktivierung von 0-9) bewirkt eine Benennung der Computerkonten als "PC-Vertrieb-01", "PC-Vertrieb-02", "PC-Vertrieb-03" usw.

Wenn Sie die Verwendung vorhandener Active Directory-Konten anfordern, führen Sie folgende Schritte aus:

- Navigieren Sie zu den Konten oder klicken Sie auf "Importieren" und geben Sie eine CSV-Datei mit Kontonamen an. Stellen Sie sicher, dass Sie ausreichend Konten für die hinzuzufügenden Maschinen haben.
- Da diese Konten von Studio verwaltet werden, gestatten Sie Studio, die Kennwörter für alle Konten zurückzusetzen, oder geben Sie das Kontokennwort (muss für alle Konten gleich sein) an.

Die Maschinen werden in einem Hintergrundprozess erstellt, der beim Erstellen einer großen Zahl von Maschinen lange

dauern kann. Die Maschinenerstellung wird fortgesetzt, selbst wenn Sie Studio schließen.

1. Wählen Sie im Studio-Navigationsbereich Maschinenkataloge aus.
2. Wählen Sie einen Katalog und dann im Aktionsbereich Maschinenkatalog bearbeiten.
3. (Nur bei Remote-PC-Zugriff-Katalogen) Auf der Seite Energieverwaltung können Sie die Energieverwaltungseinstellungen des Remote-PC-Zugriff-Katalogs ändern und eine Energieverwaltungsverbindung auswählen. Verwenden Sie die Seite Organisationseinheiten zum Hinzufügen oder Entfernen von Organisationseinheiten.  
Ändern Sie auf der Seite Beschreibung die Beschreibung des Maschinenkatalogs.

1. Wählen Sie im Studio-Navigationsbereich Maschinenkataloge aus.
2. Wählen Sie einen Katalog und dann im Aktionsbereich Maschinenkatalog umbenennen.
3. Geben Sie den neuen Namen ein.

Vor dem Löschen eines Maschinenkatalogs müssen Sie Folgendes sicherstellen:

- Alle Benutzer sind abgemeldet und es werden keine getrennten Sitzungen ausgeführt.
- Der Wartungsmodus wird für alle Maschinen in dem Katalog aktiviert und dann werden alle Maschinen heruntergefahren.
- Der Katalog ist keiner Bereitstellungsgruppe zugeordnet.

1. Wählen Sie im Studio-Navigationsbereich Maschinenkataloge aus.
2. Wählen Sie einen Katalog und dann im Aktionsbereich Maschinenkatalog löschen.
3. Geben Sie an, ob die Maschinen in dem Katalog gelöscht werden sollen. Wenn die Maschinen gelöscht werden sollen, geben Sie an, ob die zugehörigen Computerkonten in Active Directory beibehalten, deaktiviert oder gelöscht werden sollen.

Wenn Sie eine Maschine aus einem Maschinenkatalog löschen, können Benutzer nicht mehr darauf zugreifen. Vor dem Löschen von Maschinen müssen Sie Folgendes sicherstellen:

- Die Benutzerdaten wurden gesichert oder werden nicht mehr benötigt.
- Alle Benutzer sind abgemeldet. Durch das Aktivieren des Wartungsmodus wird verhindert, dass Benutzer eine Verbindung mit einer Maschine herstellen.
- Es sind keine Desktops eingeschaltet oder angehalten.

1. Wählen Sie im Studio-Navigationsbereich "Maschinenkataloge" aus.
2. Wählen Sie einen Katalog und dann im Aktionsbereich "Maschinen anzeigen".
3. Wählen Sie eine oder mehrere Maschinen aus und klicken Sie dann im Aktionsbereich auf "Wartungsmodus einschalten".
4. Wählen Sie im Aktionsbereich "Löschen" aus.
5. Wählen Sie aus, ob die Maschinen wirklich gelöscht werden sollen. Wenn Sie die Maschinen löschen, legen Sie fest, wie mit den zugehörigen Active Directory-Computerkonten verfahren werden soll:

| Im Maschinenkatalog | In Active Directory |
|---------------------|---------------------|
| Beibehalten         | Nicht ändern        |
| Entfernen           | Nicht entfernen     |

|                                  |                                |
|----------------------------------|--------------------------------|
| Im Maschinenkatalog<br>Entfernen | In Active Directory<br>Disable |
| Entfernen                        | Löschen                        |

Zum Verwalten von Active Directory-Konten in einem Maschinenkatalog haben Sie folgende Möglichkeiten:

- Freigeben nicht verwendeter Maschinenkonten durch Entfernen von Active Directory-Computerkonten aus Desktopbetriebssystem- und Serverbetriebssystemmaschinenkatalogen. Diese Konten können dann für andere Maschinen verwendet werden.
  - Fügen Sie Konten hinzu, damit beim Hinzufügen weiterer Maschinen zum Katalog Computerkonten bereit stehen.
1. Wählen Sie im Studio-Navigationsbereich Maschinenkataloge aus.
  2. Wählen Sie einen Maschinenkatalog und dann im Aktionsbereich Active Directory-Konten verwalten.
  3. Entscheiden Sie, ob Sie Computerkonten hinzufügen oder löschen möchten.
    - Wenn Sie Konten hinzufügen, werden Sie aufgefordert, anzugeben, wie mit den Kennwörtern verfahren werden soll: Setzen Sie entweder alle zurück oder geben Sie ein für alle Konten geltendes Kennwort ein. Sie können die Kennwörter zurückzusetzen, wenn Sie die aktuellen Kennwörter nicht kennen. Zum Zurücksetzen von Kennwörtern müssen Sie die entsprechende Berechtigung haben. Wenn Sie ein Kennwort eingeben, wird das Kennwort von Konten beim Importieren geändert.
    - Beim Löschen eines Kontos werden Sie aufgefordert, festzulegen, ob das Konto in Active Directory beibehalten, deaktiviert oder gelöscht werden soll.

Durch Aktualisieren eines Masterimages werden Änderungen auf alle Desktops und Anwendungen in einem Maschinenkatalog angewendet, die mit dem Masterimage erstellt wurden. Die Verwaltung allgemeiner Aspekte über ein einzelnes Masterimage ermöglicht eine schnelle Bereitstellung systemweiter Änderungen, z. B. Windows-Updates oder Konfigurationsänderungen auf einer Vielzahl von Desktops.

Nach dem Vorbereiten und Testen eines neuen/aktualisierten Masterimages auf dem Host (siehe [Vorbereiten eines Masterimages](#)) ändern Sie den Maschinenkatalog so, dass er es verwendet.

Beachten Sie Folgendes:

- Citrix empfiehlt, vor dem Durchführen von Updates Kopien oder Snapshots der Masterimages zu speichern. In der Datenbank wird von jedem Masterimage eines Maschinenkatalogs ein historischer Datensatz beibehalten. Bei Problemen mit Updates, die auf den Benutzerdesktops bereitgestellt wurden, können Sie das Masterimage auf die vorherige Version zurücksetzen und die Downtime für Benutzer minimieren. Masterimages dürfen nicht gelöscht, verschoben oder umbenannt werden, da ansonsten ein Katalog nicht auf ihre Verwendung zurückgesetzt werden kann. Obwohl Studio einen Snapshot erstellen kann, empfiehlt Citrix, dass Sie einen Snapshot mit der Hypervisor-Verwaltungskonsolle erstellen und dann den Snapshot in Studio auswählen. Dadurch können Sie statt eines automatisch erstellten Namens einen aussagekräftigen Namen und eine Beschreibung zuweisen.
- Bei GPU-Masterimages können Sie das Masterimage nur über die XenServer XenCenter-Konsolle ändern.
- Bei Maschinenkatalogen, die Provisioning Services verwenden, müssen Sie eine neue vDisk veröffentlichen, um Änderungen auf den Katalog anzuwenden. Informationen hierzu finden Sie in der Dokumentation zu Provisioning Services.

- Nach dem Aktualisieren des Masterimages müssen Sie die Computer über Studio neu starten, damit die Änderungen wirksam werden und den Benutzern zur Verfügung stehen. Dies kann automatisch passieren, z. B. wenn ein Benutzer sich von einem Desktop abmeldet oder als Teil eines konfigurierten Neustartzeitplans. Alternativ können Sie eine Maschine über Studio neu starten.

1. Wählen Sie im Studio-Navigationsbereich Maschinenkataloge aus.
2. Wählen Sie einen Maschinenkatalog und dann im Aktionsbereich Maschinen aktualisieren.
3. Wählen Sie auf der Seite Masterimage den Host und das neue/aktualisierte Masterimage aus.
4. Geben Sie auf der Seite Rolloutstrategie an, wie das neue oder aktualisierte Masterimage auf Benutzermaschinen angewendet wird: beim nächsten Herunterfahren oder sofort.
  - Wenn Sie das Image beim nächsten Herunterfahren aktualisieren, können Sie Benutzer vom Update benachrichtigen.
  - Wenn Sie das Image sofort aktualisieren, können Sie angeben, ob alle Maschinen zur gleichen Zeit oder in bestimmten Abständen neu gestartet werden. Sie können Benachrichtigung an die Benutzer senden, und zwar 1 Minute, 5 oder 15 Minuten, bevor sie abgemeldet werden und die Maschine neu gestartet wird.

1. Wählen Sie im Studio-Navigationsbereich Maschinenkataloge aus.
2. Wählen Sie den Maschinenkatalog aus und wählen Sie dann Rollback für Maschinenupdate im Aktionsbereich.
3. Geben Sie an, wie das zurückgesetzte Masterimage auf die Desktops von Benutzern angewendet wird (siehe oben).

Diese Rollbackstrategie wird nur auf Desktops angewendet, die zurückgesetzt werden müssen. Benutzer von Desktops, die nicht mit dem neuen/aktualisierten Masterimage aktualisiert wurden, durch das das Rollback erforderlich wurde (z. B. weil sich der Benutzer nicht abgemeldet hat), erhalten keine Nachricht und die Abmeldung wird nicht erzwungen.

Aktualisieren Sie den Maschinenkatalog nach dem Upgrade der VDAs auf den Maschinen auf eine neuere Version. Citrix empfiehlt das Upgrade aller VDAs auf die aktuelle Version, damit alle Zugriff auf die neuesten Features haben.

Hinweis: Für Windows XP- und Windows Vista-Maschinen muss eine ältere VDA-Version verwendet werden. Die Verwendung der neuesten Produktfeatures ist hier nicht möglich. Wenn Sie diese Maschinen nicht auf ein derzeit unterstütztes Windows-Betriebssystem aktualisieren können, empfiehlt Citrix, sie in einem separaten Maschinenkatalog zu führen. Weitere Informationen finden Sie unter [VDAs auf Maschinen mit Windows XP oder Windows Vista](#) und [Unterstützung gemischter VDAs](#).

Führen Sie vor dem Upgrade eines Maschinenkatalogs folgende Schritte durch:

- Wenn Sie Provisioning Services verwenden, aktualisieren Sie die VDA-Version in der Provisioning Services Console.
- Starten Sie die aktualisierten Maschinen, damit sie sich bei dem Controller registrieren. Auf diese Weise kann Studio feststellen, dass die Maschinen im Maschinenkatalog aktualisiert werden müssen.

1. Wählen Sie im Studio-Navigationsbereich Maschinenkataloge aus.
2. Wählen Sie den Maschinenkatalog aus. Auf der Registerkarte "Details" im unteren Bereich werden Versionsinformationen angezeigt.
3. Wählen Sie Katalog aktualisieren.
  - Wenn Studio erkennt, dass für den Katalog ein Upgrade erforderlich ist, wird eine Meldung angezeigt. Folgen Sie den Anweisungen.
  - Kann eine Maschine nicht aktualisiert werden, wird eine Meldung mit einer Erläuterung der Ursache des Problems angezeigt. Citrix empfiehlt, dass Sie alle Maschinenprobleme beheben, bevor Sie den Maschinenkatalog aktualisieren, damit alle Maschinen einwandfrei funktionieren.

Bevor Sie das Upgrade eines mit Provisioning Services erstellten Maschinenkatalogs rückgängig machen, ändern Sie die VDA-Version in der Provisioning Services Console.

1. Wählen Sie im Studio-Navigationsbereich Maschinenkataloge aus.
2. Wählen Sie den Maschinenkatalog aus. Auf der Registerkarte "Details" im unteren Bereich werden Versionsinformationen angezeigt.
3. Wählen Sie Rückgängig machen und folgen Sie den Anweisungen.

# Bereitstellungsgruppen

May 10, 2016

Eine Bereitstellungsgruppe ist eine Sammlung von Maschinen aus einem oder mehreren Maschinenkatalogen. Die Bereitstellungsgruppe gibt an, welche Benutzer diese Maschinen verwenden können und die für die Benutzer verfügbaren Anwendungen.

Erstellen Sie zuerst eine Bereitstellungsgruppe. Später können Sie die anfänglichen Einstellungen ändern und zusätzliche Einstellungen konfigurieren.

So erstellen Sie eine Bereitstellungsgruppe

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie im Aktionsbereich Bereitstellungsgruppe erstellen aus. Der Assistent führt Sie durch die nachfolgend beschriebenen Einstellungen.

Wählen Sie einen Maschinenkatalog und die Anzahl der Maschinen aus, die Sie aus dem Katalog verwenden möchten.

- Mindestens eine Maschine in dem ausgewählten Katalog muss unbenutzt bleiben.
- Ein Maschinenkatalog kann in mehreren Bereitstellungsgruppen angegeben werden, eine Maschine kann jedoch nur in einer Bereitstellungsgruppe verwendet werden.
- Eine Bereitstellungsgruppe kann mehrere Maschinenkataloge verwenden, diese Kataloge müssen dieselben Maschinentypen enthalten (Serverbetriebssystemmaschinen, Desktopbetriebssystemmaschinen oder Remote-PC-Zugriff-Maschinen). Sie können also in einer Bereitstellungsgruppe oder in einem Maschinenkatalog nicht verschiedene Maschinentypen mischen.
- Sie können auch keine Bereitstellungsgruppe erstellen, die Desktopbetriebssystemmaschinen aus einem Maschinenkatalog für statische Desktops und Maschinen aus einem für zufällige Desktops konfigurierten Maschinenkatalog enthält.
- Jede Maschine in einem Remote-PC-Zugriff-Maschinenkatalog wird automatisch einer Bereitstellungsgruppe zugewiesen.

Der Typ gibt an, was eine Bereitstellungsgruppe bietet: nur Desktops, nur Anwendungen oder beides. Bereitstellungsgruppen mit statischen Desktopbetriebssystemmaschinen können nicht gleichzeitig Desktops und Anwendungen anbieten.

Geben Sie die Benutzer und Benutzergruppen an, die die Anwendungen und/oder Desktops in der Bereitstellungsgruppe verwenden können.

Es gibt zwei Benutzertypen: authentifizierte und nicht authentifizierte Benutzer (nicht authentifizierte Benutzer werden auch als "anonyme" Benutzer bezeichnet). Sie können einen oder beide Typen konfigurieren.

- **Authentifiziert:** Die Benutzer und Gruppenmitglieder, die Sie namentlich festlegen, müssen in StoreFront oder Citrix Receiver für den Zugriff auf Anwendungen und Desktops Anmeldeinformationen (z. B. Smartcard oder Benutzername und Kennwort) angeben.
- **Nicht authentifiziert (anonym):** Bei Bereitstellungsgruppen mit Serverbetriebssystemmaschinen können Sie ein

Kontrollkästchen aktivieren, um den Zugriff auf Anwendungen und Desktops ohne Angabe von Anmeldeinformationen in StoreFront oder Citrix Receiver zuzulassen. Wenn Benutzer beispielsweise über Kioske auf Anwendungen zugreifen, werden ggf. für die Anwendung Anmeldeinformationen benötigt, nicht aber für Citrix Zugriffsportal und -Tools. Eine Gruppe anonymer Benutzer wird erstellt, wenn Sie den VDA installieren.

- Damit nicht authentifizierten Benutzern Zugriff erteilt werden kann, muss auf jeder Maschine in der Bereitstellungsgruppe ein VDA für Windows-Serverbetriebssysteme (mindestens Version 7.6) installiert sein. Wenn nicht authentifizierte Benutzer aktiviert sind, müssen Sie einen StoreFront-Store ohne Authentifizierung haben.
- Nicht authentifizierte Benutzerkonten werden bei Bedarf beim Start einer Sitzung erstellt und "AnonXYZ" genannt (XYZ ist ein eindeutiger dreistelliger Wert).
- Für Benutzersitzungen ohne Authentifizierung gilt ein Standardleerlaufzeitout von 10 Minuten. Beim Trennen der Verbindung mit dem Client erfolgt automatisch die Abmeldung. Wiederverbindung, Roaming zwischen Clients und Workspace Control werden nicht unterstützt.

In der folgenden Tabelle werden die Optionen erläutert.

| Zugriff aktivieren für                                           | Benutzer und Benutzergruppen hinzufügen/zuweisen? | Kontrollkästchen "Nicht authentifizierte Benutzer zulassen" aktivieren? |
|------------------------------------------------------------------|---------------------------------------------------|-------------------------------------------------------------------------|
| Nur authentifizierte Benutzer                                    | Ja                                                | Nein                                                                    |
| Nur nicht authentifizierte Benutzer                              | Nein                                              | Ja                                                                      |
| Sowohl authentifizierte als auch nicht authentifizierte Benutzer | Ja                                                | Ja                                                                      |

Bei Desktopgruppen mit Desktopbetriebssystemmaschinen können Sie Benutzerdaten (eine Liste der Benutzer) nach dem Erstellen der Bereitstellungsgruppe importieren. Siehe [Importieren und Exportieren von Benutzerlisten](#).

Die auf einer aus dem Masterimage, einer Vorlage im Maschinenkatalog oder auf dem App-V-Verwaltungsserver erstellten Maschine erkannten Anwendungen werden in einer Liste angezeigt. Wählen Sie die Anwendungen aus, die der Bereitstellungsgruppe hinzugefügt werden.

Sie können Anwendungen außerdem manuell hinzufügen (erstellen). Sie müssen den Pfad zur ausführbaren Datei, das Arbeitsverzeichnis, optionale Befehlszeilenargumente und Anzeigenamen für Administratoren und Benutzer angeben.

Die Anwendungseigenschaften können Sie ändern. Informationen hierzu finden Sie unter [Ändern der Anwendungseigenschaften](#).

Sie können keine Anwendungen für Remote-PC-Zugriff-Bereitstellungsgruppen erstellen.

Standardmäßig werden hinzugefügte Anwendungen in einem Ordner mit dem Namen Applications abgelegt. Ordner erleichtern die Verwaltung, wenn eine große Zahl Anwendungen vorliegt. Sie können einen anderen Ordner angeben, wenn Sie die Anwendung hinzufügen, es ist jedoch einfacher, Ordner später zu verwalten. Weitere Informationen finden Sie unter [Verwalten von Anwendungsordnern](#).

Wenn Sie zwei Anwendungen mit dem gleichen Namen den gleichen Benutzern bereitstellen, ändern Sie in Studio die



Eigenschaft Anwendungsname (Benutzer), sonst wird den Benutzern der Name in Receiver doppelt angezeigt.

Wählen Sie StoreFront-URLs bzw. fügen Sie sie hinzu, die von Citrix Receiver verwendet wird, der auf jeder Maschine in der Bereitstellungsgruppe installiert ist. Sie können StoreFront-Serveradressen auch später festlegen, indem Sie im Navigationsbereich Konfiguration > StoreFront auswählen. Wenn Sie den StoreFront-Server hinzufügen, fügen Sie am Ende der URL "/Discovery" hinzu.

# Einstellungen

May 10, 2016

In den folgenden Dokumenten wird beschrieben, wie Sie die meisten der Einstellungen, die Sie für Bereitstellungsgruppen festlegen und aktualisieren können, konfigurieren und verwalten.

- [Anwendungen](#)
- [Maschinen](#)
- [Remote-PC-Zugriff](#)
- [Sitzung](#)
- [Benutzer](#)

Die nachfolgenden Informationen beziehen sich auf Einstellungen, die nicht in diesen Dokumenten behandelt werden.

Bevor Sie eine Bereitstellungsgruppe des Typs "nur Anwendungen" oder "Desktops und Anwendungen" in eine Bereitstellungsgruppe vom Typ "nur Desktops" ändern, löschen Sie alle Anwendungen aus der Bereitstellungsgruppe.

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie eine Bereitstellungsgruppe und dann im Aktionsbereich Bereitstellungsgruppe bearbeiten.
3. Wählen Sie auf der Seite Bereitstellungstyp den Bereitstellungstyp, den Sie der Bereitstellungsgruppe zuweisen möchten.

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie eine Bereitstellungsgruppe und dann im Aktionsbereich Bereitstellungsgruppe bearbeiten.
3. Auf der Seite Grundeinstellungen können Sie folgende Einstellungen ändern:

| Einstellung                  | Beschreibung                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Beschreibung                 | Text in StoreFront, der Benutzern angezeigt wird                                                                                                                                                                                                                                                                                                                                                                                                            |
| Kontrollkästchen "Aktiviert" | Zeigt an, ob die Bereitstellungsgruppe aktiviert ist.                                                                                                                                                                                                                                                                                                                                                                                                       |
| Desktops pro Benutzer        | Nur Desktopbetriebssystemmaschinen: die maximale Anzahl freigegebener Desktops, die ein Benutzer gleichzeitig aktiv haben darf. Bei Bereitstellungen mit Zuweisung bei der ersten Verwendung gibt dieser Wert an, wie viele Desktops Benutzer sich selbst zuweisen können.                                                                                                                                                                                  |
| Zeitzone                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Secure ICA aktivieren        | Schützt die Kommunikation zwischen Maschinen in der Bereitstellungsgruppe mit SecureICA, wodurch das ICA-Protokoll verschlüsselt wird (Standardverschlüsselungsgrad ist 128 Bit, dieser kann mit dem SDK geändert werden). Citrix empfiehlt die Verwendung zusätzlicher Verschlüsselungsmethoden, z. B. SSL/TLS-Verschlüsselung, wenn Datenübertragungen über öffentliche Netzwerke stattfinden. Bei SecureICA wird die Datenintegrität auch nicht geprüft. |

Führen Sie ein Upgrade einer Bereitstellungsgruppe durch, nachdem Sie ein Upgrade der VDAs auf deren Maschinen durchgeführt haben.

Hinweis: Wenn Sie ältere VDA-Versionen weiterverwenden müssen, sind neuere Produktfeatures ggf. nicht verfügbar. Weitere Informationen finden Sie unter [Upgrade einer Bereitstellung](#).

Führen Sie vor dem Upgrade der Bereitstellungsgruppe folgende Schritte durch:

- Wenn Sie Provisioning Services verwenden, aktualisieren Sie die VDA-Version in der Provisioning Services Console.
  - Starten Sie die Maschinen mit dem neuen VDA, damit sie sich bei dem Controller registrieren können. Dadurch wird Studio darüber informiert, welche Elemente in der Bereitstellungsgruppe aktualisiert werden müssen.
1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
  2. Wählen Sie die Bereitstellungsgruppe und dann im Aktionsbereich Bereitstellungsgruppe aktualisieren.

Vor dem Starten des Upgrades wird in Studio gemeldet, welche Maschinen nicht aktualisiert werden können (falls es solche gibt) und warum. Sie können das Upgrade dann abbrechen, die Ursachen beheben und das Upgrade erneut starten.

Wenn das Upgrade der Bereitstellungsgruppe abgeschlossen ist, können Sie Maschinen auf ihren vorherigen Zustand zurücksetzen, indem Sie die Bereitstellungsgruppe und dann im Aktionsbereich Rückgängig machen wählen.

# Maschinen

Jan 05, 2017

Sofern nicht anders angegeben werden die folgenden Schritte für alle Typen von Bereitstellungsgruppen (Serverbetriebssysteme, Desktopbetriebssysteme und Remote-PC-Zugriff) unterstützt.

Hinweis: Dieser Vorgang wird für Remote-PC-Zugriff-Maschinen nicht unterstützt.

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie die Bereitstellungsgruppe und dann im Aktionsbereich Maschinen anzeigen.
3. Wählen Sie die Maschine und anschließend im Aktionsbereich eine der folgenden Optionen (einige Optionen sind je nach Maschinenzustand ggf. nicht verfügbar):
  - Herunterfahren erzwingen: Die Maschine wird zwingend abgeschaltet und die Liste der Maschinen wird aktualisiert.
  - Neu starten: Das Betriebssystem wird heruntergefahren und die Maschine wird dann neu gestartet. Wenn das Betriebssystem diese Aufgaben nicht ausführen kann, bleibt die Maschine im aktuellen Zustand.
  - Anhalten: hält die Maschine an, ohne sie herunterzufahren, und die Liste der Maschinen wird aktualisiert.
  - Herunterfahren: Das Betriebssystem wird aufgefordert, herunterzufahren.

Wenn die Maschine nicht innerhalb von 10 Minuten heruntergefahren wird, wird sie ausgeschaltet. Wenn Windows versucht, während des Herunterfahrens Updates zu installieren, besteht die Gefahr, dass die Maschine ausgeschaltet wird, bevor die Updates abgeschlossen sind.

Hinweis: Citrix empfiehlt, dass Sie das Herunterfahren von Desktopbetriebssystemmaschinen durch Benutzer während einer Sitzung nicht zulassen. Einzelheiten finden Sie in der Microsoft-Dokumentation zu Richtlinien.

Hinweis: Die Energieverwaltung ist nur bei virtuellen Desktopbetriebssystemmaschinen, nicht aber bei physischen Maschinen (einschließlich Remote-PC-Zugriff-Maschinen) möglich. Desktopbetriebssystemmaschinen mit GPU-Funktionen können nicht angehalten werden, sodass Energieverwaltungsvorgänge fehlschlagen. Entsprechende Informationen zu Serverbetriebssystemmaschinen finden Sie unter [Erstellen eines Neustartzeitplans](#).

Maschinen können einen der folgenden Zustände aufweisen:

| Bereitstellungsgruppe | Status                                                                  |
|-----------------------|-------------------------------------------------------------------------|
| Zufällig              | Zufällig zugewiesen und in Verwendung                                   |
|                       | Nicht zugewiesen und nicht verbunden                                    |
| Statisch (zugewiesen) | Dauerhaft zugeordnet und in Verwendung                                  |
|                       | Dauerhaft zugewiesen und nicht verbunden (aber bereit für Verbindungen) |
|                       | Nicht zugewiesen und nicht verbunden                                    |

Statische Bereitstellungsgruppen enthalten im Normalbetrieb sowohl dauerhaft zugewiesene als auch nicht zugewiesene

Maschinen. Anfangs sind alle Maschinen nicht zugewiesen (außer beim Erstellen der Bereitstellungsgruppe manuell zugewiesene Maschinen). Wenn Benutzer eine Verbindung herstellen, werden Maschinen dauerhaft zugewiesen. Die Energieverwaltung ist bei nicht zugewiesenen Maschinen in den Bereitstellungsgruppen vollständig, bei dauerhaft zugewiesenen Maschinen nur teilweise möglich.

- **Pools und Puffer:** Unter einem Pool versteht man bei zufälligen Bereitstellungsgruppen und nicht zugewiesenen Maschinen in statischen Bereitstellungsgruppen eine Gruppe nicht zugewiesener (oder temporär zugewiesener) Maschinen, die eingeschaltet bleiben und mit denen Benutzer eine Verbindung herstellen können (eine Maschine ist direkt nach der Anmeldung verfügbar). Die Poolgröße (d. h. die Zahl der Maschinen, die eingeschaltet bleiben) kann abhängig von der Tageszeit konfiguriert werden. (Verwenden Sie zum Konfigurieren des Pools bei statischen Bereitstellungsgruppen das SDK.)

Ein Puffer ist eine zusätzliche Gruppe nicht zugewiesener Maschinen, die eingeschaltet werden, wenn die Anzahl der Maschinen im Pool unter einen Schwellenwert (Prozentsatz der Größe der Bereitstellungsgruppe) fällt. Für große Bereitstellungsgruppen wird unter Umständen eine große Zahl Maschinen eingeschaltet, wenn der Schwellenwert unterschritten wird. Planen Sie die Größe Ihrer Bereitstellungsgruppen daher sorgfältig oder passen Sie die Standardpuffergröße mit dem SDK an.

- **Energiestatustimer:** Sie können mit den Energiestatustimern Maschinen anhalten, wenn die Verbindung eine bestimmte Zeit lang getrennt war. Maschinen werden zum Beispiel automatisch außerhalb der Bürostunden angehalten, wenn die Verbindung mindestens 10 Minuten lang getrennt war. Zufällige Maschinen oder Maschinen mit persönlichen vDisks werden bei Abmeldung des Benutzers automatisch heruntergefahren, es sei denn, Sie konfigurieren die Bereitstellungsgruppeneigenschaft "ShutdownDesktopsAfterUse" im SDK.  
Sie können Timer für Werktage und Wochenenden sowie für Spitzen- und Nebenzeiten konfigurieren.
- **Teilweise Energieverwaltung bei dauerhaft zugewiesenen Maschinen:** Bei dauerhaft zugewiesenen Maschinen können Sie Energiestatustimer, aber keine Pools oder Puffer einrichten. Die Maschinen werden zu Beginn jeder Spitzenzeit eingeschaltet und zu Beginn eines Zeitraums mit geringer Auslastung ausgeschaltet. Es ist keine Feinsteuerung der Zahl der Maschinen möglich, die als Ausgleich für verwendete Maschinen verfügbar werden (im Gegensatz zu nicht zugewiesenen Maschinen).

Einstellen der Energieverwaltung für Desktopbetriebssystemmaschinen

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie eine Bereitstellungsgruppe und dann im Aktionsbereich Bereitstellungsgruppe bearbeiten.
3. Wählen Sie auf der Seite Energieverwaltung im Dropdownmenü Energieverwaltung für Maschinen die Option Wochentage aus. (Wochentage umfassen standardmäßig die Tage von Montag bis Freitag.)
4. Wählen Sie bei zufälligen Bereitstellungsgruppen unter Maschinen einschalten die Option Bearbeiten und geben Sie die Poolgröße während der Werktage an. Wählen Sie anschließend die Anzahl der einzuschaltenden Maschinen.
5. Legen Sie unter Spitzenzeiten die Zeiträume für Spitzen- und Nebenzeiten für jeden Tag fest.
6. Stellen Sie die Energiestatustimer für Spitzen- und Nebenzeiten an Werktagen ein:
  - Geben Sie für Während Spitzenzeiten > Wenn getrennt die Verzögerung (in Minuten) ein, nach der getrennte Maschinen in der Bereitstellungsgruppe angehalten werden, und klicken Sie auf Anhalten.
  - Geben Sie für Während Nicht-Spitzenzeiten > Wenn getrennt die Verzögerung (in Minuten) ein, nach der abgemeldete Maschinen in der Bereitstellungsgruppe heruntergefahren werden, und klicken Sie auf Herunterfahren. Dieser Timer ist für Bereitstellungsgruppen mit zufälligen Maschinen nicht verfügbar.
8. Wählen Sie im Dropdownmenü Energieverwaltung für Maschinen die Option Wochenende aus und konfigurieren Sie die Spitzenzeiten und Energiestatustimer für Wochenenden.

Verwenden Sie das SDK für Folgendes:

- Herunterfahren anstelle von Anhalten von Maschinen basierend auf Energiestatustimern, oder wenn Timer auf Abmeldungen anstatt von Verbindungstrennungen reagieren sollen
- Ändern der Standardeinstellungen für Werktage und Wochenende

Hinweis: Neustartzeitpläne können nur für Serverbetriebssystemmaschinen verwendet werden. Informationen zu Desktopbetriebssystemmaschinen finden Sie unter [Energieverwaltung für Maschinen](#).

So konfigurieren Sie einen Neustartzeitplan

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie eine Bereitstellungsgruppe und dann im Aktionsbereich Bereitstellungsgruppe bearbeiten.
3. Führen Sie auf der Seite Neustartzeitplan folgende Schritte durch:
  - Legen Sie mit Maschinen neu starten fest, wie oft die Maschinen neu gestartet werden.
  - Geben Sie mit den Feldern für Erste Gruppe neu starten um die Uhrzeit (Stunde und Minute im 24-Stunden-Format) an, zu der der erste Server den Neustart beginnen soll.
  - Geben Sie über das Dropdownmenü Neustart weiterer Gruppen alle an, ob alle Server gleichzeitig neu gestartet werden sollen oder, falls nicht, wie viel Zeit für den Neustart der einzelnen Server in der Bereitstellungsgruppe eingeplant werden soll.  
 Beispiel: Angenommen, eine Bereitstellungsgruppe hat fünf Server, die Zeit für Erste Gruppe neu starten um: ist auf 13:00 Uhr festgelegt und der Wert für Neustart weiterer Gruppen alle: ist auf "1 Stunde" festgelegt. Die Dauer (60 Minuten) wird durch die Anzahl der Maschinen (fünf) geteilt; das ergibt ein Neustartintervall von 12 Minuten. Die Neustartzeiten sind also 13:00 Uhr, 13:12 Uhr, 13:24 Uhr, 13:36 Uhr und 13:48 Uhr. So können alle fünf Maschinen einen Neustart vor dem Ende des angegebenen Zeitraums (1 Stunde) ausführen.
  - Geben Sie an, ob eine bestimmte Zeit vor deren Abmeldung eine Nachricht an die Benutzer gesendet werden soll. Die Benachrichtigung wird entsprechend der für die einzelnen Server berechneten Neustartzeit gesendet, wie im Beispiel beschrieben.

Sie können kein automatisiertes Einschalten oder Herunterfahren in Studio durchführen.

Wenn Sie vorübergehend verhindern möchten, dass neue Verbindungen mit Maschinen hergestellt werden, können Sie den Wartungsmodus für eine oder alle Maschinen in einer Bereitstellungsgruppe aktivieren. Das ist beispielsweise vor dem Anwenden von Patches oder der Verwendung von Verwaltungstools nützlich.

- Wenn sich eine Serverbetriebssystemmaschine im Wartungsmodus befindet, können Benutzer eine Verbindung mit vorhandenen Sitzungen herstellen, aber keine neuen Sitzungen starten.
- Bei einer Desktopbetriebssystemmaschine (oder einem Computer mit Remote-PC-Zugriff) im Wartungsmodus können Benutzer keine Verbindung herstellen. Aktuelle Verbindungen bleiben bis zur Trennung oder Abmeldung erhalten.

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie eine Bereitstellungsgruppe aus.
3. Zum Aktivieren des Wartungsmodus für alle Maschinen in der Bereitstellungsgruppe wählen Sie im Aktionsbereich Wartungsmodus einschalten.

So aktivieren Sie den Wartungsmodus für eine Maschine

1. Wählen Sie im Aktionsbereich Maschinen anzeigen.
2. Wählen Sie eine Maschine aus und wählen Sie dann im Aktionsbereich Wartungsmodus einschalten.
4. Zum Deaktivieren des Wartungsmodus für eine oder alle Maschinen in einer Bereitstellungsgruppe befolgen Sie die Anweisungen oben unter Auswahl der Option Wartungsmodus ausschalten im Aktionsbereich.

Einstellungen für Windows-Remotedesktopverbindungen wirken sich auch darauf aus, ob eine Serverbetriebssystemmaschine im Wartungsmodus ist. Der Wartungsmodus ist in folgenden Fällen aktiviert:

- Der Wartungsmodus für den Server wie oben beschrieben aktiviert.
- Die Remotedesktopverbindung wurde auf "Keine Verbindung mit diesem Computer zulassen" festgelegt.
- Remotedesktopverbindung ist nicht auf "Keine Verbindung mit diesem Computer zulassen" festgelegt und für den Anmeldemodus der Remotehostkonfiguration gilt eine der folgenden Optionen:
  - Wiederverbindungen zulassen, doch neue Anmeldungen verhindern
  - Neue Verbindungen zulassen, doch Neuanmeldungen bis zum Neustart des Servers verweigern

Hinweis: Sie können nur Desktopbetriebssystemmaschinen neu zuweisen, nicht aber Serverbetriebssystemmaschinen oder Maschinen, die mit Provisioning Services erstellt wurden.

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie eine Bereitstellungsgruppe aus.
3. So weisen Sie mehrere Maschinen neu zu
  1. Wählen Sie im Aktionsbereich Bereitstellungsgruppe bearbeiten aus.
  2. Wählen Sie auf der Seite Maschinenzuteilung (Benutzerzuweisung) die Maschinen aus und geben Sie die neuen Benutzer an.
4. So weisen Sie eine Maschine neu zu
  1. Wählen Sie im Aktionsbereich Maschinen anzeigen.
  2. Wählen Sie eine Maschine und dann im Aktionsbereich Benutzer ändern.
  3. Fügen Sie einen Benutzer hinzu oder entfernen Sie den Benutzer.

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie eine Bereitstellungsgruppe und dann im Aktionsbereich Bereitstellungsgruppe bearbeiten.
3. Legen Sie auf der Seite Benutzereinstellungen einen Wert für "Desktops pro Benutzer" fest.

Hinweis: Sie können Tags nur für Desktopbetriebssystemmaschinen verwenden.

Anhand von Tags können Sie eine Maschinensuche eingrenzen oder den Zugriff auf Maschinen beschränken. Sie können eine beliebige Anzahl von Tags mit beliebiger Länge hinzufügen.

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie eine Bereitstellungsgruppe und wählen Sie dann im Aktionsbereich Maschinen anzeigen.
3. Wählen Sie eine Maschine aus.
4. Zum Hinzufügen von Tags wählen Sie im Aktionsmenü Tag hinzufügen und geben Sie ein oder mehrere Tags ein. Trennen Sie die Angaben durch Semikola (;).

Wählen Sie zum Ändern oder Entfernen von Tags im Aktionsmenü Tags bearbeiten und führen Sie die Änderungen durch.

Hinweis: Die Lastverwaltung ist nur bei Serverbetriebssystemmaschinen möglich.

Bei der Lastverwaltung wird die Serverlast gemessen und festgelegt, welcher Server unter den aktuellen Umgebungsbedingungen auszuwählen ist. Diese Auswahl basiert auf folgenden Faktoren:

- **Wartungsmodusstatus des Servers:** Eine Serverbetriebssystemmaschine wird nur für den Lastausgleich berücksichtigt, wenn der Wartungsmodus für sie deaktiviert ist. (Detaillierte Informationen finden Sie unter [Unterbinden der Benutzerverbindung mit Maschinen \(Wartungsmodus\)](#).)

- Der **Serverlastindex** bestimmt, mit welcher Wahrscheinlichkeit ein Server, der Serverbetriebssystemmaschinen bereitstellt, Verbindungen erhält. Der Index basiert auf einer Kombination von Lastauswertungskriterien: Anzahl der Sitzungen sowie Einstellungen für Leistungswerte (z. B. CPU-, Datenträger- und Speichernutzung). Die Lastauswertungskriterien werden in den Richtlinieneinstellungen für die Lastverwaltung festgelegt.
- Sie können den Lastindex in Director, über die Suche in Studio und im SDK überwachen.
- In Studio ist die Serverlastindexspalte standardmäßig ausgeblendet. Zum Anzeigen der Spalte wählen Sie eine Maschine und dann mit der rechten Maustaste eine Spaltenüberschrift und wählen Sie dann Spalte auswählen. Klicken Sie in der Kategorie "Maschine" auf Serverlastindex.
- Verwenden Sie im SDK das Cmdlet "Get-BrokerMachine".

Ein Serverlastindex von 10.000 bedeutet, dass der Server voll ausgelastet ist. Wenn keine anderen Server verfügbar sind, erhalten die Benutzer beim Starten einer Sitzung u. U. eine Meldung, dass der Desktop oder die Anwendung zurzeit nicht verfügbar ist.

- **Richtlinieneinstellung "Toleranzwert für gleichzeitige Anmeldungen"** Maximale Anzahl von gleichzeitigen Anmeldeanforderungen am Server. (Diese Einstellung entspricht der Lastdrosselung in XenApp-Versionen vor 7.5.)

Wenn alle Server den Toleranzwert für gleichzeitige Anmeldungen erreichen oder überschreiten, wird die nächste Anmeldeanforderung dem Server mit der niedrigsten Anzahl ausstehender Anmeldungen zugewiesen. Wenn mehrere Server dieses Kriterium erfüllen, wird der Server mit dem niedrigsten Lastindex ausgewählt.

Weitere Informationen finden Sie unter

— *Referenz für Richtlinieneinstellungen*

.

Beim Entfernen einer Maschine wird diese aus der Bereitstellungsgruppe gelöscht, jedoch nicht aus dem Maschinenkatalog, den die Bereitstellungsgruppe verwendet. Die Maschinen sind daher für Zuweisungen zu anderen Bereitstellungsgruppen weiter verfügbar.

Maschinen müssen heruntergefahren werden, bevor sie entfernt werden können. Wenn Sie vorübergehend verhindern möchten, dass Benutzer eine Verbindung mit der Maschine herstellen, während Sie sie löschen, setzen Sie die Maschine in den Wartungsmodus, bevor Sie sie herunterfahren.

Wenn Sie eine Maschine einem anderen Benutzer zuweisen, denken Sie daran, dass Maschinen persönliche Daten enthalten können. Es empfiehlt sich ggf. ein Reimaging der Maschine.

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie eine Bereitstellungsgruppe und wählen Sie dann im Aktionsbereich Maschinen anzeigen.
3. Stellen Sie sicher, dass die Maschine heruntergefahren ist.
4. Wählen Sie im Aktionsbereich Aus Bereitstellungsgruppe entfernen.

Alle Änderungen zum Einschränken des Zugriffs auf Maschinen in einer Bereitstellungsgruppe haben Vorrang vor zuvor durchgeführten Einstellungen, unabhängig von der verwendeten Methode. Sie haben folgende Möglichkeiten:

- Einschränken des Zugriffs für Administratoren über Geltungsbereiche für die delegierte Administration. Sie können einen Geltungsbereich erstellen und zuweisen, in dem Administratoren auf alle Anwendungen zugreifen können, und einen zweiten Geltungsbereich, der nur den Zugriff auf spezifische Anwendungen zulässt. Weitere Informationen finden Sie in der Dokumentation zur delegierten Administration.
- Einschränken des Zugriffs für Benutzer über SmartAccess-Richtlinienausdrücke, mit denen über NetScaler Gateway



hergestellte Benutzerverbindungen gefiltert werden.

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
  2. Wählen Sie die Bereitstellungsgruppe und dann im Aktionsbereich Bereitstellungsgruppe bearbeiten.
  3. Wählen Sie auf der Seite Zugriffsrichtlinie die Option Über NetScaler Gateway hergestellte Verbindungen aus.
  4. Wenn Sie nur einen Teil dieser Verbindungen auswählen möchten, wählen Sie Verbindungen, auf die mindestens einer der folgenden Filter zutrifft. Legen Sie dann die NetScaler Gateway-Site fest und fügen Sie SmartAccess-Richtlinienausdrücke für zulässige Benutzerzugriffsszenarios hinzu, bzw. bearbeiten oder löschen Sie diese. Weitere Informationen finden Sie in der Dokumentation zu NetScaler Gateway.
- Einschränken des Zugriffs für Benutzer über Ausschlussfilter für mit dem SDK festgelegte Zugriffsrichtlinien. Zugriffsrichtlinien werden auf Bereitstellungsgruppen angewendet, um Verbindungen genauer zu definieren. Sie können beispielsweise den Maschinenzugriff für eine Untergruppe von Benutzern einschränken und zulässige Benutzergeräte festlegen. Mit Ausschlussfiltern können Zugriffsrichtlinien weiter angepasst werden. Aus Sicherheitsgründen können Sie beispielsweise den Zugriff für eine Untergruppe der Benutzer oder Geräte verweigern. Ausschlussfilter sind in der Standardeinstellung deaktiviert.

Wenn Sie beispielsweise den Zugriff von einem Lernlabor im Subnetz des Unternehmensnetzwerks auf eine spezifische Bereitstellungsgruppe verhindern möchten, unabhängig davon, wer die Maschinen im Labor nutzt, verwenden Sie folgenden Befehl: `Set-BrokerAccessPolicy -Name VPDesktops_Direct -ExcludedClientIPFilterEnabled $True` -

Sie können das Sternchen (\*) als Platzhalter für alle Tags, die mit dem gleichen Richtlinienausdruck beginnen, verwenden. Wenn Sie beispielsweise auf einer Maschine das Tag "VPDesktops\_Direct" hinzufügen und auf einer anderen das Tag "VPDesktops\_Test", wird der Filter durch Festlegen des Tags auf "VPDesktops\_\*" im Skript "Set-BrokerAccessPolicy" auf beide Maschinen angewendet.

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie die Bereitstellungsgruppe und wählen Sie dann im Aktionsbereich Maschinen anzeigen.
3. Wählen Sie eine Maschine und dann im Aktionsbereich Maschinen aktualisieren.
  - Zum Auswählen eines anderen Masterimages wählen Sie Masterimage. Wählen Sie dann einen Snapshot. Durch Erweitern eines ausgewählten Snapshots werden die zugehörigen Masterimages angezeigt.
  - Zum Anwenden der Änderungen und Benachrichtigen der Benutzer der Maschine wählen Sie Rolloutbenachrichtigung für Endbenutzer. Geben Sie anschließend Folgendes an:
    - Zeitpunkt der Aktualisierung des Masterimages: jetzt oder beim nächsten Neustart
    - Neustart-Verteilungszeit: alle Maschinen zur gleichen Zeit oder zeitlich versetzt
    - Ob und wann Benutzer über den Neustart benachrichtigt werden sowie die Meldung, die sie erhalten sollen

# Anwendungen

May 10, 2016

So fügen Sie einer Bereitstellungsgruppe eine Anwendung hinzu:

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie die Bereitstellungsgruppe aus.
3. Klicken Sie im Aktionsbereich auf Anwendungen hinzufügen.

Die auf einer aus dem Masterimage, einer Vorlage im Maschinenkatalog oder auf dem App-V-Verwaltungsserver erstellten Maschine erkannten Anwendungen werden in einer Liste angezeigt. Wählen Sie eine oder mehrere Anwendungen aus, um Sie der Bereitstellungsgruppe hinzuzufügen.

Sie können Anwendungen außerdem manuell hinzufügen (erstellen). Sie müssen den Pfad zur ausführbaren Datei, das Arbeitsverzeichnis, optionale Befehlszeilenargumente und Anzeigenamen für Administratoren und Benutzer angeben.

Sie können die Anwendungseigenschaften ändern (siehe unten).

Standardmäßig werden hinzugefügte Anwendungen in einem Ordner mit dem Namen Applications abgelegt. Weitere Informationen über Anwendungsordner finden Sie weiter unten.

Gehen Sie zum Duplizieren, Deaktivieren, Umbenennen und Löschen einer Anwendung und zum Bearbeiten von Anwendungstags folgendermaßen vor:

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie die Registerkarte "Anwendungen" im mittleren Bereich und dann die Anwendung aus.
3. Wählen Sie die relevante Aufgabe im Aktionsbereich.

Gut zu wissen:

- Wenn Sie eine Anwendung duplizieren, wird diese automatisch umbenannt und neben die ursprüngliche Anwendung platziert.
- Wenn Sie eine Anwendung löschen, wird diese aus der Bereitstellungsgruppe, jedoch nicht aus dem Masterimage entfernt.
- Anleitungen zum Verschieben einer Anwendung in einen anderen Anwendungsordner finden Sie unten.

So ändern Sie die Eigenschaften einer Anwendung

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie die Registerkarte "Anwendungen" im mittleren Bereich und dann die Anwendung aus.
3. Wählen Sie im Aktionsbereich Eigenschaften.

Sie können Folgendes anzeigen und ändern:

| Eigenschaft    | Seite           |
|----------------|-----------------|
| Anwendungsname | Identifizierung |
|                |                 |

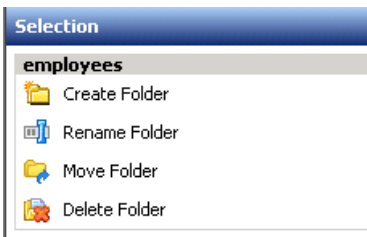
| Kategorie in Receiver<br>Eigenschaft      | Bereitstellung<br>Seite  |
|-------------------------------------------|--------------------------|
| Befehlszeilenargumente                    | Speicherort              |
| Beschreibung                              | Identifizierung          |
| Dateinamenerweiterungen                   | Dateitypzuordnung        |
| Dateitypzuordnungen                       | Dateitypzuordnung        |
| Symbol                                    | Bereitstellung           |
| Schlüsselwörter für StoreFront            | Identifizierung          |
| Pfad zur ausführbaren Datei               | Speicherort              |
| Verknüpfung auf dem Desktop des Benutzers | Bereitstellung           |
| Sichtbarkeit                              | Sichtbarkeit beschränken |
| Arbeitsverzeichnis                        | Speicherort              |

Anwendungsänderungen werden evtl. für aktuelle Anwendungsbenutzer erst wirksam, wenn diese ihre Sitzung abmelden.

Standardmäßig werden hinzugefügte Anwendungen in einem Ordner mit dem Namen

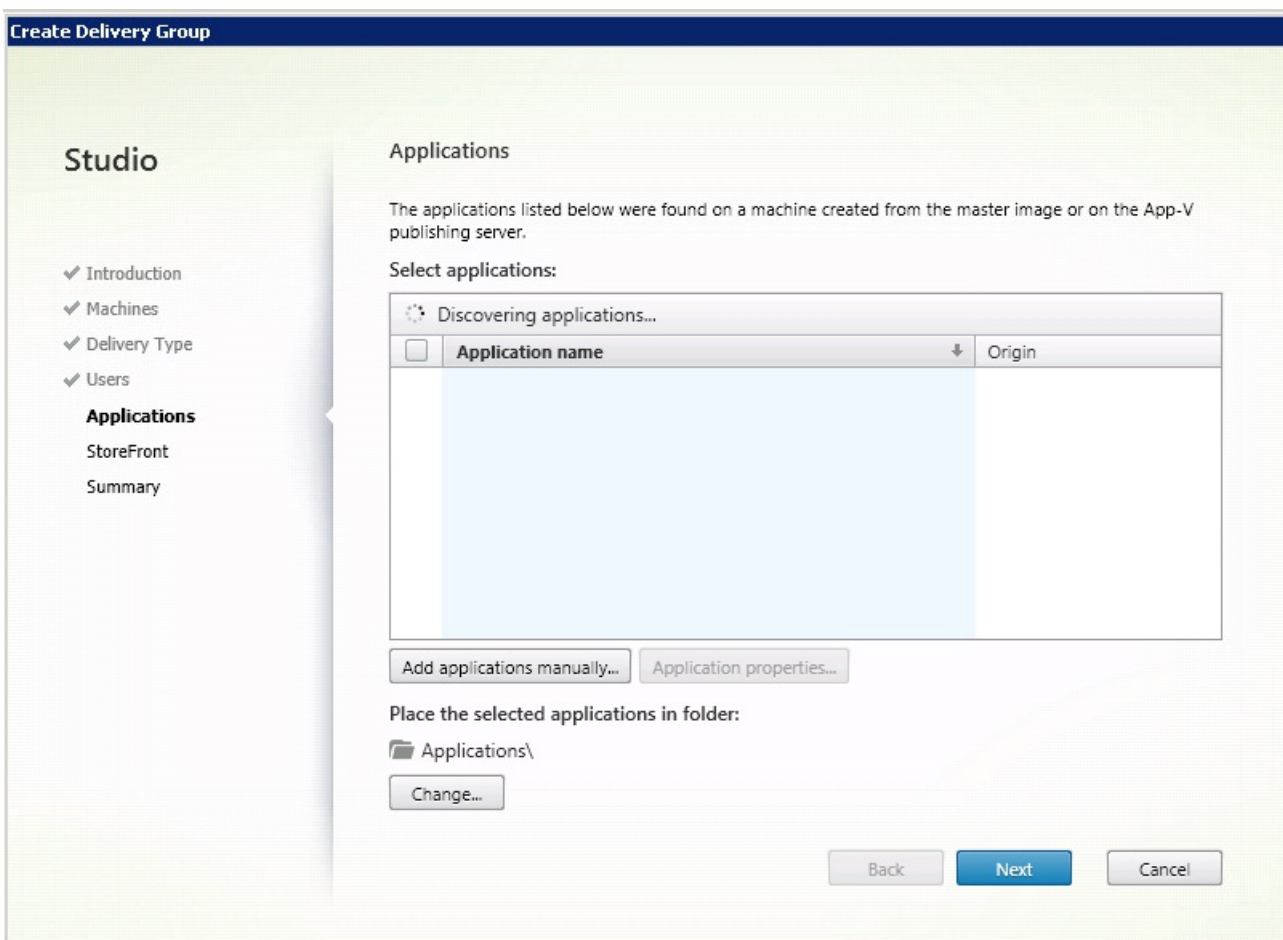
— *Applications*

abgelegt. Sie haben folgende Möglichkeiten:



- Erstellen zusätzlicher Ordner und anschließendes Verschieben von Anwendungen in die neuen Ordner.
  - Ordner können bis zu fünffach verschachtelt werden.
  - Ordner müssen keine Anwendungen enthalten, leere Ordner sind zugelassen.
  - Ordner werden alphabetisch aufgelistet, es sei denn, Sie verschieben sie oder geben beim Erstellen einen anderen Speicherort an.
  - Sie können mehr als einen Ordner mit dem gleichen Namen haben, sofern jeder einen anderen übergeordneten Ordner hat. Sie können mehr als eine Anwendung mit dem gleichen Namen haben, sofern jede in einem anderen Ordner ist.
- Verschieben eines Ordners auf derselben Ebene oder in eine andere Ebene. Das Verschieben ist mit Drag & Drop am einfachsten.
- Umbenennen oder Löschen eines von Ihnen erstellten Ordners. Sie können den Ordner "Applications" nicht umbenennen oder löschen. Sie können aber alle Anwendungen in diesem Ordner in andere von Ihnen erstellte Ordner verschieben.

Sie können auch Ordner für Anwendungen beim Erstellen oder Bearbeiten einer Bereitstellungsgruppe erstellen.

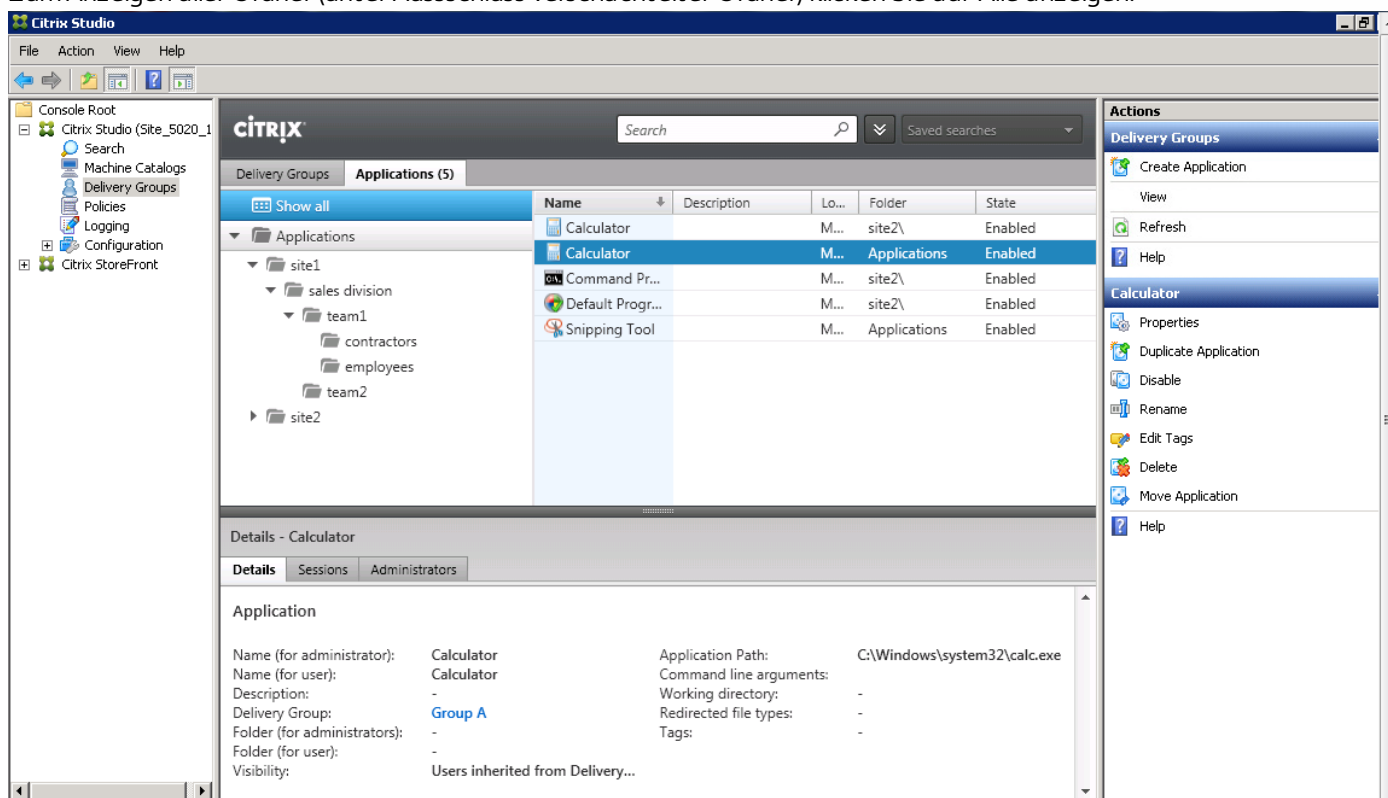


Zum Entfernen, Umbenennen und Löschen eines Ordners, der Anwendungen enthält, benötigen Sie für alle enthaltenen Anwendungen die Berechtigung zum Anzeigen von Anwendungen in Ordnern und die Berechtigung zum Bearbeiten der Anwendungseigenschaften. Weitere Informationen finden Sie unter [Delegierte Administration](#).

Tipp: Die folgenden Anweisungen gelten für den Aktionsbereich in Studio. Alternativ können Sie Kontextmenüs oder Drag & Drop verwenden. Wenn Sie einen Ordner am falschen Speicherort erstellen oder ihn dorthin verschieben, können Sie ihn per Drag & Drop an den korrekten Speicherort ziehen.

Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen und dann im mittleren Bereich die Registerkarte Anwendungen.

- Zum Anzeigen aller Ordner (unter Ausschluss verschachtelter Ordner) klicken Sie auf Alle anzeigen.



- So erstellen Sie einen Ordner
  1. Zum Erstellen des Ordners auf der höchsten Ebene (d. h. nicht als Unterordner) wählen Sie den obersten Anwendungsordner aus. Zum Erstellen des Ordners als Unterordner eines vorhandenen Ordners wählen Sie diesen Ordner aus.
  2. Wählen Sie im Aktionsbereich Ordner erstellen. Geben Sie einen Namen für die Ordner bestehend aus 1-64 Zeichen ein. Leerstellen sind zugelassen.
- So verschieben Sie einen Ordner
  1. Wählen Sie den Ordner und dann im Aktionsbereich Ordner verschieben. (Sie können immer nur einen Ordner verschieben, es sei denn, der Ordner enthält Unterordner.)
  2. Zum Verschieben des Ordners auf die höchste Ebene (d. h. nicht als Unterordner) wählen Sie den obersten Anwendungsordner aus. Zum Verschieben des Ordners unter einen vorhandenen Ordner wählen Sie diesen Ordner aus.
- Zum Umbenennen eines Ordners wählen Sie den Ordner und dann im Aktionsbereich Ordner umbenennen. Geben Sie einen aus 1-64 Zeichen bestehenden neuen Namen ein.
- Zum Löschen eines Ordners wählen Sie den Ordner und dann im Aktionsbereich Ordner löschen. Beim Löschen eines Ordners, der Anwendungen und andere Ordner enthält, werden diese Objekte auch gelöscht. Beim Löschen einer Anwendung wird die Anwendungszuweisung aus der Bereitstellungsgruppe aber nicht aus der Maschine entfernt.

- Zum Verschieben von Anwendungen in einen Ordner wählen Sie eine oder mehrere Anwendungen und dann im Aktionsbereich Anwendung verschieben. Wählen Sie den Ordner aus.

Zum Hinzufügen oder Verschieben von Anwendungen in Ordner über den Assistenten zum Erstellen einer Bereitstellungsgruppe wählen Sie die entsprechenden Anwendungen auf der Seite Anwendungen und anschließend Ändern.

- Zum Verschieben der Anwendung in einen vorhandenen Ordner wählen Sie diesen Ordner.
- So verschieben Sie die Anwendung in einen neuen Ordner
  - Zum Erstellen eines Ordners auf höchster Ebene (d. h. nicht als Unterordner) wählen Sie den obersten Anwendungsordner und dann Neuer Ordner. Geben Sie einen aus 1-64 Zeichen bestehenden Namen für den Ordner ein. Leerstellen sind zulässig.
  - Zum Erstellen eines verschachtelten Ordners unter einem vorhandenen Ordner (nicht dem Ordner "Applications") wählen Sie den vorhandenen Ordner und dann Neuer Ordner. Geben Sie einen aus 1-64 Zeichen bestehenden Namen für den Ordner ein. Leerstellen sind zulässig.

# Benutzer

May 10, 2016

Es gibt zwei Benutzertypen: authentifizierte und nicht authentifizierte Benutzer (nicht authentifizierte Benutzer werden auch als "anonyme" Benutzer bezeichnet). Sie können einen oder beide Typen konfigurieren.

- **Authentifiziert:** Die Benutzer und Gruppenmitglieder, die Sie namentlich festlegen, müssen in StoreFront oder Citrix Receiver für den Zugriff auf Anwendungen und Desktops Anmeldeinformationen (z. B. Smartcard oder Benutzername und Kennwort) angeben.
- **Nicht authentifizierte (anonym):** Bei Bereitstellungsgruppen mit Serverbetriebssystemmaschinen können Sie ein Kontrollkästchen aktivieren, um den Zugriff auf Anwendungen und Desktops ohne Angabe von Anmeldeinformationen in StoreFront oder Citrix Receiver zuzulassen. Wenn Benutzer beispielsweise über Kioske auf Anwendungen zugreifen, werden ggf. für die Anwendung Anmeldeinformationen benötigt, nicht aber für Citrix Zugriffsportal und -Tools. Eine Gruppe anonymer Benutzer wird erstellt, wenn Sie den VDA installieren.
  - Damit nicht authentifizierte Benutzern Zugriff erteilt werden kann, muss auf jeder Maschine in der Bereitstellungsgruppe ein VDA für Windows-Serverbetriebssysteme (mindestens Version 7.6) installiert sein. Wenn nicht authentifizierte Benutzer aktiviert sind, müssen Sie einen StoreFront-Store ohne Authentifizierung haben.
  - Nicht authentifizierte Benutzerkonten werden bei Bedarf beim Start einer Sitzung erstellt und "AnonXYZ" genannt (XYZ ist ein eindeutiger dreistelliger Wert).
  - Für Benutzersitzungen ohne Authentifizierung gilt ein Standardleerlaufzeitlimit von 10 Minuten. Beim Trennen der Verbindung mit dem Client erfolgt automatisch die Abmeldung. Wiederverbindung, Roaming zwischen Clients und Workspace Control werden nicht unterstützt.

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie eine Bereitstellungsgruppe und dann im Aktionsbereich Bereitstellungsgruppe bearbeiten.
3. In der folgenden Tabelle werden die Optionen erläutert.

| Zugriff aktivieren für                                           | Benutzer und Benutzergruppen hinzufügen/zuweisen? | Kontrollkästchen "Nicht authentifizierte Benutzer zulassen" aktivieren? |
|------------------------------------------------------------------|---------------------------------------------------|-------------------------------------------------------------------------|
| Nur authentifizierte Benutzer                                    | Ja                                                | Nein                                                                    |
| Nur nicht authentifizierte Benutzer                              | Nein                                              | Ja                                                                      |
| Sowohl authentifizierte als auch nicht authentifizierte Benutzer | Ja                                                | Ja                                                                      |

Bei Desktopgruppen mit Desktopbetriebssystemmaschinen können Sie Benutzerdaten (eine Liste der Benutzer) nach dem Erstellen der Bereitstellungsgruppe importieren. Siehe [Importieren und Exportieren von Benutzerlisten](#) unten.

Bei Bereitstellungsgruppen mit physischen Desktopbetriebssystemmaschinen können Sie Benutzerinformationen nach dem Erstellen der Bereitstellungsgruppe aus einer CSV-Datei importieren. Sie können Benutzerinformationen auch in eine CSV-

Datei exportieren. Die CSV-Datei kann Daten aus einer vorherigen Produktversion enthalten.

Die erste Zeile der CSV-Datei muss durch Trennzeichen getrennte Spaltenüberschriften (in beliebiger Reihenfolge) enthalten, z. B. ADComputerAccount, AssignedUser, VirtualMachine und HostId. Die nachfolgenden Zeilen enthalten durch Trennzeichen getrennte Daten. Die Einträge unter ADComputerAccount können allgemeine Namen, IP-Adressen Distinguished Names oder Domänen-/Computernamenpaare sein.

So importieren oder exportieren Sie Benutzerinformationen

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie eine Bereitstellungsgruppe und dann im Aktionsbereich Bereitstellungsgruppe bearbeiten.
3. Klicken Sie auf der Seite Maschinenzuteilung auf die Schaltfläche Liste importieren bzw. Liste exportieren und navigieren Sie zum Speicherort der Datei.



# Sitzungen

Nov 02, 2016

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie eine Bereitstellungsgruppe und wählen Sie dann im Aktionsbereich Maschinen anzeigen.
3. Zum Abmelden einer Sitzung wählen Sie die Sitzung oder den Desktop und dann im Aktionsbereich Abmelden. Die Sitzung wird geschlossen und die Maschine steht nun anderen Benutzern zur Verfügung, sofern sie nicht einem bestimmten Benutzer zugewiesen ist.

Zum Trennen einer Sitzung wählen Sie die Sitzung oder den Desktop und dann im Aktionsbereich Trennen. Anwendungen werden weiter ausgeführt und die Maschine bleibt dem Benutzer zugewiesen. Der Benutzer kann eine Verbindung mit derselben Maschine wiederherstellen.

Zum Senden einer Nachricht an die Benutzer wählen Sie die Sitzung, die Maschine oder den Benutzer und dann im Aktionsbereich Nachricht senden. Geben Sie die Nachricht ein.

Sie können die Energiestatustimer für Desktopbetriebssystemmaschinen so konfigurieren, dass nicht genutzte Sitzungen automatisch verarbeitet werden. Einzelheiten finden Sie unter [Energieverwaltung für Maschinen](#).

Hinweis: Diese Features werden nur auf Serverbetriebssystemmaschinen unterstützt.

Vorabstart und Fortbestehen von Sitzungen ermöglichen einen schnellen Zugriff durch Benutzer auf Anwendungen, indem Sitzungen gestartet werden, bevor sie angefordert werden, und aktiv bleiben, nachdem ein Benutzer alle Anwendungen geschlossen hat.

In der Standardeinstellung werden diese Features nicht verwendet, d. h. eine Sitzung startet, wenn ein Benutzer eine Anwendung startet, und bleibt so lange aktiv, bis die letzte Anwendung der Sitzung geschlossen wird.

Überlegungen:

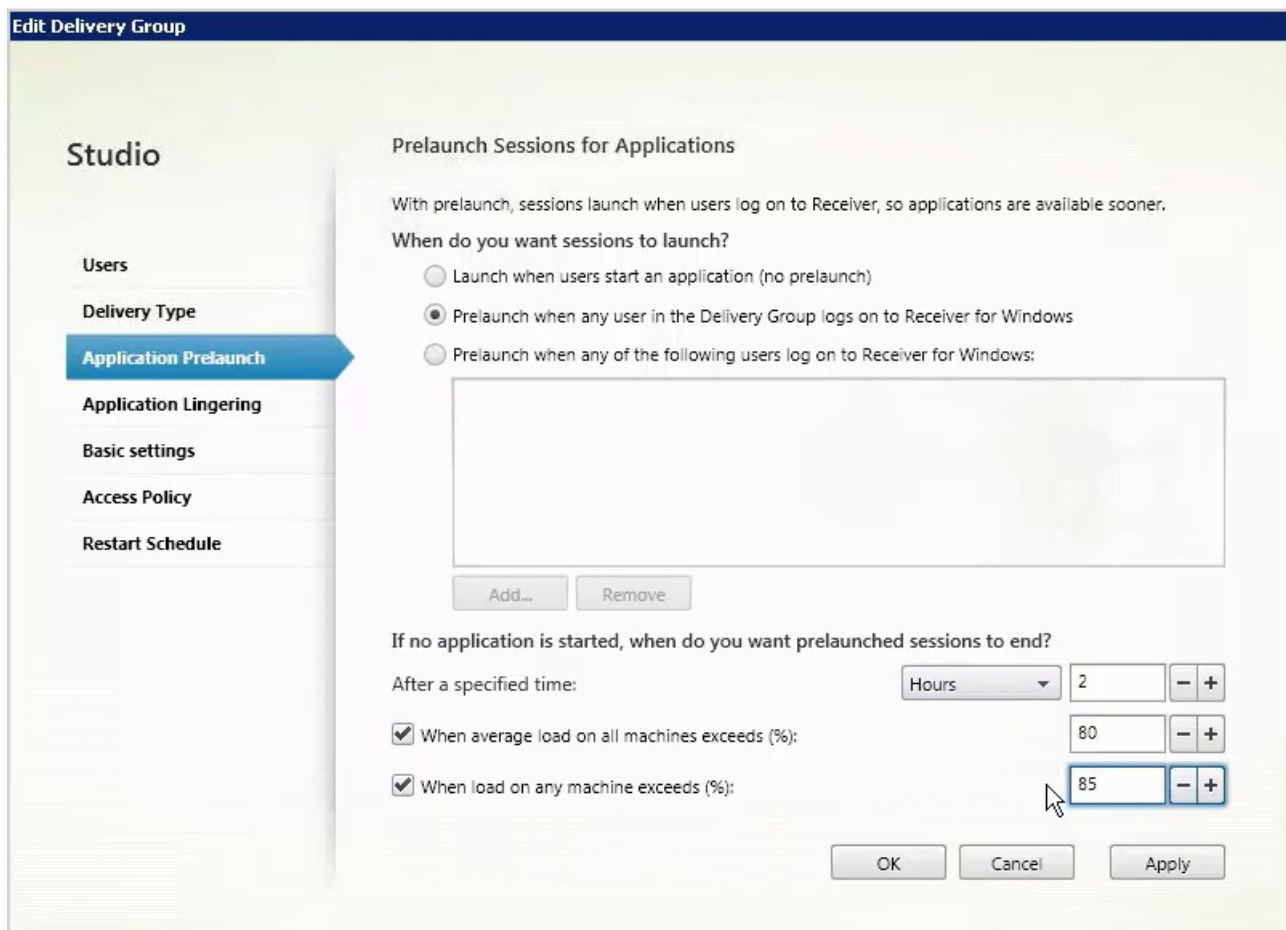
- Die Bereitstellungsgruppe muss Anwendungen unterstützen und auf den Maschinen muss ein VDA für Serverbetriebssysteme in mindestens Version 7.6 ausgeführt werden.
- Sitzungsvorabstart wird nur bei Verwendung von Citrix Receiver für Windows unterstützt. Sitzungsfortbestehen wird bei Verwendung von Citrix Receiver für Windows und Citrix Receiver für Web unterstützt. Es ist eine zusätzliche Receiver-Konfiguration erforderlich. Anweisungen hierzu finden Sie in den eDocs für Ihre Receiver für Windows-Version. Suchen Sie dort nach "Sitzungsvorabstart".

Hinweis: Receiver für HTML5 wird nicht unterstützt.

- Bei Verwendung des Sitzungsvorabstarts gilt Folgendes:
  - Unabhängig von den Administratoreinstellungen funktioniert Vorabstart nicht, wenn die Maschine eines Endbenutzers in den Modus "Anhalten" oder "Ruhezustand" versetzt wird.
  - Ein Vorabstart kann erfolgen, wenn der Endbenutzer seine Maschine/Sitzung sperrt. Wenn sich der Endbenutzer jedoch von Citrix Receiver anmeldet, wird die Sitzung beendet und ein Vorabstart ist nicht möglich.
- Vorab gestartete und fortbestehende Sitzungen verbrauchen eine Lizenz, jedoch nur wenn sie verbunden sind. Nicht genutzte vorab gestartete und fortbestehende Sitzungen werden standardmäßig nach 15 Minuten getrennt. Dieser Wert kann über das PowerShell-Cmdlet "New/Set-BrokerSessionPreLaunch" konfiguriert werden.
- Eine sorgfältige Planung und Überwachung der Aktivitätsmuster von Benutzern ist wichtig, damit diese Features so eingerichtet werden können, dass sie einander ergänzen. In einer optimalen Konfiguration besteht ein Gleichgewicht zwischen dem Vorteil einer schnelleren Anwendungsverfügbarkeit für Benutzer und den durch den Verbrauch von Lizenzen und die fortdauernde Zuteilung von Ressourcen entstehenden Kosten.
- Sie können den Vorabstart von Sitzungen auch für eine spezifische Uhrzeit in Receiver konfigurieren.

#### Aktivieren des Vorabstarts von Sitzungen

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie eine Bereitstellungsgruppe und dann im Aktionsbereich Bereitstellungsgruppe bearbeiten.
3. Aktivieren Sie den Vorabstart von Sitzungen, indem Sie auf der Seite Anwendungsvorabstart auswählen, wann Sitzungen gestartet werden sollen:
  - Wenn Benutzer eine Anwendung starten. Dies ist die Standardeinstellung. Der Vorabstart von Sitzungen ist deaktiviert.
  - Wenn ein Benutzer der Bereitstellungsgruppe sich an Receiver für Windows anmeldet.
  - Wenn ein beliebiger Benutzer einer Liste mit Benutzern und Bereitstellungsgruppen sich an Receiver für Windows anmeldet. Bei Auswahl dieser Option müssen Sie auch die Benutzer oder Benutzergruppen festlegen.



4. Eine vorab gestartete Sitzung wird durch eine normale Sitzung ersetzt, wenn der Benutzer eine Anwendung startet. Wenn der Benutzer keine Anwendung startet (d. h. die vorab gestartete Sitzung wird nicht verwendet), wird durch die folgenden Einstellungen bestimmt, wie lange die Sitzung aktiv bleibt. Informationen zu diesen Einstellungen finden Sie unter

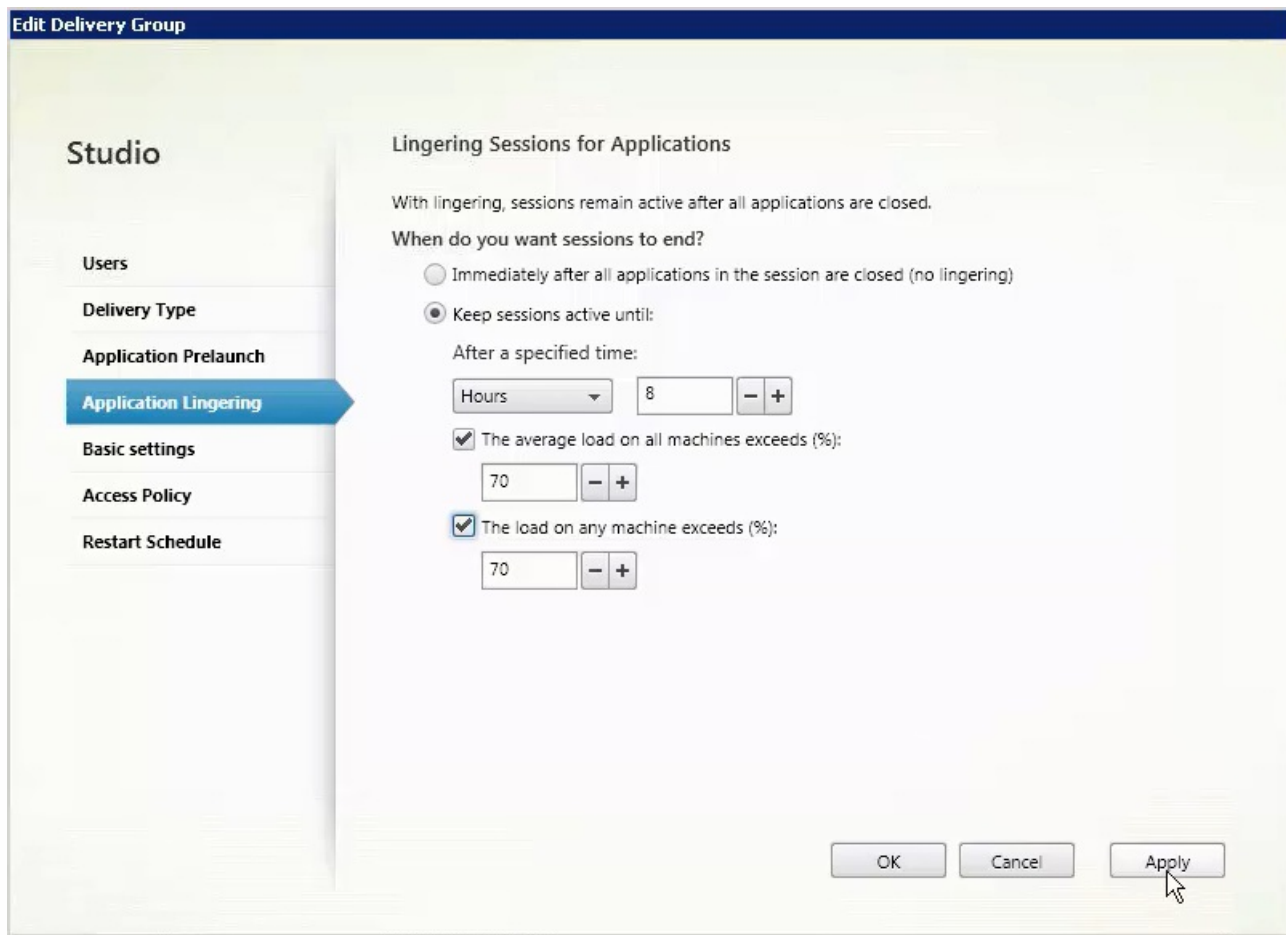
— *Dauer des Aktivbleibens nicht genutzter vorab gestarteter und fortbestehender Sitzungen*

weiter unten.

- Ablauf eines vorgegebenen Zeitintervalls. Das Zeitintervall können Sie ändern (1-99 Tage, 1-2376 Stunden oder 1-142.560 Minuten).
  - Wenn die durchschnittliche Last auf allen Maschinen in der Bereitstellungsgruppe einen bestimmten Prozentsatz (1-99 %) übersteigt.
  - Wenn die Last auf einer Maschine in der Bereitstellungsgruppe einen bestimmten Prozentsatz (1-99 %) übersteigt.
- Eine vorab gestartete Sitzung bleibt also bis zum Eintreten eines der folgenden Ereignisse aktiv: ein Benutzer startet eine Anwendung, das vorgegebene Zeitintervall läuft ab oder der angegebene Lastschwellenwert wird überschritten.

Aktivieren von Sitzungsfortbestehen

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie eine Bereitstellungsgruppe und dann im Aktionsbereich Bereitstellungsgruppe bearbeiten.
3. Aktivieren Sie auf der Seite Anwendungsfortbestehen das Sitzungsfortbestehen durch Aktivieren des Optionsfelds Sitzungen bleiben aktiv bis.



4. Mehrere Einstellungen wirken sich darauf aus, wie lange eine Sitzung aktiv bleibt, wenn der Benutzer keine weitere Anwendung startet. Informationen zu diesen Einstellungen finden Sie unter *— Dauer des Aktivbleibens vorab gestarteter und fortbestehender Sitzungen* weiter unten.
- Ablauf eines vorgegebenen Zeitintervalls. Das Zeitintervall können Sie ändern (1-99 Tage, 1-2376 Stunden oder 1-142.560 Minuten).
  - Wenn die durchschnittliche Last auf allen Maschinen in der Bereitstellungsgruppe einen bestimmten Prozentsatz (1-99 %) übersteigt.
  - Wenn die Last auf einer Maschine in der Bereitstellungsgruppe einen bestimmten Prozentsatz (1-99 %) übersteigt. Eine fortbestehende Sitzung bleibt also bis zum Eintreten eines der folgenden Ereignisse aktiv: ein Benutzer startet eine Anwendung, das vorgegebene Zeitintervall läuft ab oder der angegebene Lastschwellenwert wird überschritten.

**Dauer des Aktivbleibens nicht genutzter vorab gestarteter und fortbestehender Sitzungen** Wie lange eine nicht genutzte Sitzung aktiv bleibt, wenn der Benutzer keine Anwendung startet, kann über ein Timeout oder über Serverlast-Schwellenwerte angegeben werden. Sie können alle Parameter konfigurieren und die Sitzung wird jeweils durch das zuerst auftretende Ereignis beendet.

- Timeout: Ein konfiguriertes Timeout gibt die Anzahl der Minuten, Stunden oder Tage an, die eine nicht genutzte vorab gestartete oder fortbestehende Sitzung aktiv bleibt. Wenn Sie ein zu kurzes Timeout konfigurieren, werden vorab gestartete Sitzungen beendet, bevor der Benutzer in den Genuss des schnelleren Anwendungszugriffs kommt. Ist das Timeout zu lang, werden eingehende Benutzerverbindungen möglicherweise abgewiesen, da der Server nicht genügend Ressourcen hat.

Sie können dieses Timeout über das SDK (Cmdlet "New/Set-BrokerSessionPreLaunch"), nicht aber über Studio deaktivieren. Wenn Sie das Timeout deaktivieren, wird es für die betreffende Bereitstellungsgruppe in Studio und im

Assistenten zum Bearbeiten von Bereitstellungsgruppen nicht angezeigt.

- **Schwellenwerte:** Das automatische Beenden vorab gestarteter und fortbestehender Sitzungen auf der Basis der Serverlast gewährleistet, dass Sitzungen so lange wie möglich geöffnet bleiben (vorausgesetzt, es sind Serverressourcen verfügbar). Nicht genutzte vorab gestartete und fortbestehende Sitzungen verursachen keine Abweisung von Verbindungen, da sie automatisch beendet werden, wenn Ressourcen für neue Benutzersitzungen benötigt werden. Sie können zwei Schwellenwerte konfigurieren: die durchschnittliche Last aller Server der Bereitstellungsgruppe und die höchste Last eines Servers in der Bereitstellungsgruppe (beides in Prozent). Wird ein Schwellenwert überschritten, werden jeweils die Sitzungen beendet, die sich am längsten im Zustand "vorab gestartet" bzw. "fortbestehend" befinden. Das Beenden erfolgt einzeln im Minutentakt bis die Last unter den Schwellenwert fällt. (Solange der Schwellenwert überschritten ist, werden keine neuen Sitzungen vorab gestartet.)

Server mit VDAs, die nicht beim Controller registriert sind, und Server im Wartungsmodus gelten als voll ausgelastet. Bei einem ungeplanten Ausfall werden vorab gestartete und fortbestehende Sitzungen automatisch beendet, um Kapazität freizugeben.

# Mit XenApp veröffentlichte Anwendungen und Desktops

May 10, 2016

Verwenden Sie Serverbetriebssystemmaschinen zum Bereitstellen von mit XenApp veröffentlichten Anwendungen und Desktops.

In dieser Tabelle wird beschrieben, welche Situationen, Benutzer und Überlegungen für die Verwendung der Bereitstellungsmethode gelten.

|                                  |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Anwendungsfall</b>            | <p><b>Ziel</b></p> <p>Kostengünstige, serverbasierte Bereitstellung, um die Kosten für die Bereitstellung von Anwendungen für eine große Anzahl von Benutzern gering zu halten, und gleichzeitig eine sichere High-Definition-Benutzererfahrung zu bieten.</p> <p><b>Benutzer</b></p> <p>Benutzer führen vordefinierte Aufgaben aus, es wird keine Personalisierung oder kein Offlinezugriff auf Anwendungen benötigt. Hierzu können aufgabenorientierte Mitarbeiter, wie z. B. Callcenter- und Einzelhandelsarbeitskräfte gehören, oder Benutzer, die Arbeitsstationen gemeinsam verwenden.</p> <p><b>Anwendungstypen</b></p> <p>Beliebige Anwendung.</p> |
| <b>Vorteile und Überlegungen</b> | <p><b>Vorteile</b></p> <p>Verwaltbare und skalierbare Lösung für das Datenzentrum.</p> <p>Kosteneffektivste Lösung für die Anwendungsbereitstellung.</p> <p>Gehostete Anwendungen werden zentral verwaltet und Benutzer können die Anwendung nicht ändern, wodurch eine konsistente, sichere und zuverlässige Benutzererfahrung bereitgestellt wird.</p> <p><b>Überlegungen</b></p> <p>Benutzer müssen online sein, um auf ihre Anwendungen zuzugreifen.</p>                                                                                                                                                                                               |
| <b>Benutzererfahrung</b>         | <p>Benutzer fordern eine oder mehrere Anwendungen von StoreFront über ihr Startmenü oder eine von Ihnen vorgegebene URL an.</p> <p>Anwendungen werden virtuell bereitgestellt und in High Definition auf Benutzergeräten angezeigt.</p> <p>Abhängig von den Profileinstellungen werden Benutzeränderungen gespeichert, wenn die</p>                                                                                                                                                                                                                                                                                                                        |

|                                                                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                                                     | <p>Anwendungssitzung des Benutzers beendet wird. Andernfalls werden die Änderungen werden gelöscht.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <p><b>Verarbeiten, Hosten und Bereitstellen von Anwendungen</b></p> | <p><b>Prozess</b></p> <p>Die Anwendungsverarbeitung findet auf den Hostingmaschinen statt, nicht auf den Benutzergeräten.</p> <p>Die Hostingmaschine kann eine physische oder eine virtuelle Maschine sein.</p> <p><b>Host</b></p> <p>Anwendungen und Desktops sind auf einer Serverbetriebssystemmaschine gespeichert.</p> <p>Maschinen werden über Maschinenkataloge verfügbar gemacht.</p> <p><b>Bereitstellung</b></p> <p>Maschinen in Maschinenkatalogen sind in Bereitstellungsgruppen organisiert, die Benutzergruppen dieselben Anwendungen bereitstellen.</p> <p>Serverbetriebssystemmaschinen unterstützen Folgendes:</p> <ul style="list-style-type: none"> <li>• Desktop- und Anwendungsbereitstellungsgruppen, die sowohl Desktops als auch Anwendungen hosten.</li> <li>• Anwendungsbereitstellungsgruppen, die nur Anwendungen hosten</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| <p><b>Sitzungsverwaltung und -zuweisung</b></p>                     | <p><b>Sitzungen</b></p> <p>Auf Serverbetriebssystemmaschinen werden mehrere Sitzungen auf einer einzelnen Maschinen ausgeführt, über die mehrere Anwendungen und Desktops an mehrere, gleichzeitig verbundene Benutzer bereitgestellt werden. Jeder Benutzer benötigt eine einzelne Sitzung, um die gehosteten Anwendungen auszuführen.</p> <p>Beispiel: Ein Benutzer meldet sich an und fordert eine Anwendung an. Eine der Sitzungen auf dieser Maschine ist für die anderen Benutzer nicht mehr verfügbar. Ein zweiter Benutzer meldet sich an und fordert eine Anwendung an, die von dieser Maschine gehostet wird. Eine zweite Sitzung auf derselben Maschine ist damit jetzt nicht verfügbar. Wenn beide Benutzer weitere Anwendungen anfordern, werden keine zusätzlichen Sitzungen benötigt, da ein Benutzer mehrere Anwendungen in der gleichen Sitzung ausführen kann. Wenn zwei weitere Benutzer sich anmelden und Desktops anfordern, und zwei Sitzungen auf derselben Maschine verfügbar sind, hostet diese eine Maschine nun vier Sitzungen für vier verschiedene Benutzer.</p> <p><b>Zufällige Maschinenzuweisung</b></p> <p>In der Bereitstellungsgruppe, der ein Benutzer zugewiesen ist, wird eine Maschine auf einem Server mit der geringsten Last ausgewählt. Ein Computer mit Sitzungsverfügbarkeit wird nach dem Zufallsprinzip zugewiesen und stellt einem Benutzer bei der Anmeldung Anwendungen bereit.</p> |

So stellen Sie mit XenApp veröffentlichte Anwendungen bereit

1. Installieren Sie die bereitzustellenden Anwendungen auf einem Masterimage mit einem unterstützten Windows-Serverbetriebssystem.
2. Erstellen Sie einen Maschinenkatalog für dieses Masterimage oder aktualisieren Sie einen vorhandenen Katalog mit dem Masterimage.
3. Erstellen Sie eine Anwendungsbereitstellungsgruppe zum Bereitstellen der Desktops für Benutzer.
4. Wählen Sie in der Liste der installierten Anwendungen die Anwendung aus, die Sie bereitstellen möchten.

So stellen Sie mit XenApp veröffentlichte Desktops bereit

1. Installieren Sie Apps auf einem Masterimage mit einem unterstützten Windows-Serverbetriebssystem.
2. Erstellen Sie einen Maschinenkatalog für dieses Masterimage oder aktualisieren Sie einen vorhandenen Katalog mit dem Masterimage.
3. Erstellen Sie eine Desktopbereitstellungsgruppe zum Bereitstellen der Desktops für Benutzer.



# VM-gehostete Apps

May 10, 2016

Verwenden Sie Desktopbetriebssystemmaschinen zum Bereitstellen von VM-gehosteten Apps.

In dieser Tabelle wird beschrieben, welche Situationen, Benutzer und Überlegungen für die Verwendung der Bereitstellungsmethode gelten.

|                                  |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Anwendungsfall</b>            | <p><b>Ziel</b></p> <p>Eine clientbasierte Anwendungsbereitstellungslösung, die eine sichere, zentrale Verwaltung bietet und eine große Anzahl von Benutzern pro Hostserver (oder Hypervisor) unterstützt und Benutzern Anwendungen bereitstellt, die problemlos in High Definition angezeigt werden.</p> <p><b>Benutzer</b></p> <p>Interne und externe Auftragnehmer, Drittanbietern und andere vorläufige Teammitglieder.</p> <p>Die Benutzer benötigen keinen Offlinezugriff auf gehostete Anwendungen.</p> <p><b>Anwendungstypen</b></p> <p>Anwendungen, die möglicherweise nicht gut mit anderen Anwendungen funktionieren oder mit dem Betriebssystem interagieren, z. B. .NET Framework. Dieser Typ von Anwendungen eignet sich gut für das Hosting auf virtuellen Maschinen.</p> <p>Anwendungen, die auf älteren Betriebssystemen (z. B. Windows XP oder Windows Vista) und älteren Architekturen (z. B. 32 Bit oder 16 Bit) ausgeführt werden. Durch Isolieren jeder Anwendung auf einer eigenen virtuellen Maschine werden die anderen Benutzer bei Ausfall einer Maschine nicht beeinträchtigt.</p> |
| <b>Vorteile und Überlegungen</b> | <p><b>Vorteile</b></p> <p>Anwendungen und Desktops auf dem Masterimage werden sicher verwaltet, gehostet und auf Maschinen im Datenzentrum ausgeführt, womit eine kosteneffektivere Lösung für die Anwendungsbereitstellung bereitgestellt wird.</p> <ul style="list-style-type: none"><li>• Die Benutzer können bei der Anmeldung willkürlich einer Maschine in einer Bereitstellungsgruppe zugewiesen werden, die für das Hosting einer Anwendung konfiguriert ist.</li><li>• Sie können auch einem einzelnen Benutzer eine einzelne Maschine für die Anwendungsbereitstellung jedes Mal statisch zuweisen, wenn sich der Benutzer anmeldet. Bei statisch zugewiesenen Maschinen kann der Benutzer eigene Anwendungen auf der virtuellen Maschine installieren und verwalten.</li></ul> <p><b>Überlegungen</b></p> <p>Das Ausführen mehrerer Sitzungen auf Desktopbetriebssystemmaschinen wird nicht unterstützt. Daher beansprucht jeder Benutzer bei der Anmeldung eine einzelne Maschine innerhalb einer Bereitstellungsgruppe und der Zugriff auf die Anwendungen muss online</p>                       |

|                                                              |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|--------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                                              | <p>erfolgen.</p> <p>Bei dieser Methode werden die Serverressourcen für die Verarbeitung von Anwendungen sowie der Speicher für die persönlichen vDisks der Benutzer erhöht.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>Benutzererfahrung</b>                                     | Die gleiche nahtlose Anwendungserfahrung wie mit gehosteten, freigegebenen Anwendungen auf Serverbetriebssystemmaschinen.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <b>Verarbeiten, Hosten und Bereitstellen von Anwendungen</b> | <p><b>Prozess</b></p> <p>Wie bei Serverbetriebssystemmaschinen, nur dass es sich um virtuelle Desktopbetriebssystemmaschinen handelt.</p> <p><b>Host</b></p> <p>Wie bei Serverbetriebssystemmaschinen, nur dass es sich um virtuelle Desktopbetriebssystemmaschinen handelt.</p> <p><b>Bereitstellung</b></p> <p>Wie bei Serverbetriebssystemmaschinen, allerdings können Desktopbetriebssystemmaschinen nur in einer Desktop-Bereitstellungsgruppe bestehen.</p>                                                                                                                                                                                                                                                                                                  |
| <b>Sitzungsverwaltung und -zuweisung</b>                     | <p><b>Sitzungen</b></p> <p>Desktopbetriebssystemmaschinen führen eine Desktopsitzung von einer Maschine aus. Nur beim Zugriff auf Anwendungen: Ein Benutzer kann mehrere Anwendungen verwenden (und ist nicht auf eine Anwendung eingeschränkt), da das Betriebssystem jede Anwendung als eine neue Sitzung ansieht.</p> <p><b>Zufällige und statische Maschinenzuweisungen</b></p> <p>Nach der Anmeldung können Benutzer in der Bereitstellungsgruppe, der sie zugewiesen wurden, auf Folgendes zugreifen:</p> <ul style="list-style-type: none"> <li>• Statisch zugewiesene Maschine, sodass sich der Benutzer jedes Mal an derselben Maschine anmeldet.</li> <li>• Zufällig zugewiesene Maschine, die je nach Sitzungsverfügbarkeit ausgewählt wird.</li> </ul> |

So stellen Sie VM-gehostete Apps bereit:

1. Installieren Sie die bereitzustellenden Anwendungen auf einem Masterimage mit einem unterstützten Windows-Desktopbetriebssystem.
2. Erstellen Sie einen Maschinenkatalog für dieses Masterimage oder aktualisieren Sie einen vorhandenen Katalog mit dem Masterimage.  
Entscheiden Sie beim Definieren der Desktopdarstellung für den Maschinenkatalog, ob Benutzer bei jeder Anmeldung mit einer neuen VM oder mit derselben VM verbunden werden.
3. Erstellen Sie eine Anwendungsbereitstellungsgruppe zum Bereitstellen der Desktops für Benutzer.
4. Wählen Sie in der Liste der installierten Anwendungen die Anwendung aus, die Sie bereitstellen möchten.



# VDI-Desktops

May 10, 2016

Verwenden Sie Desktopbetriebssystemmaschinen zum Bereitstellen von VDI-Desktops.

VDI-Desktops werden auf virtuellen Maschinen gehostet und bieten jedem Benutzer ein Desktopbetriebssystem.

VDI-Desktops benötigen mehr Ressourcen als mit XenApp veröffentlichte Desktops, aber es ist nicht erforderlich, dass die auf ihnen installierten Anwendungen serverbasierte Betriebssysteme unterstützen. Abhängig vom ausgewählten Typ des VDI-Desktops kann der Desktop einzelnen Benutzern zugewiesen werden und von den Benutzern in hohem Maße personalisiert werden.

Beim Erstellen eines Maschinenkatalogs für VDI-Desktops erstellen Sie einen der folgenden Desktoptypen:

- Zufällige, nicht beständige Desktops (gepoolte VDI-Desktops): Jedes Mal, wenn ein Benutzer sich anmeldet, um einen dieser Desktops zu verwenden, wird eine Verbindung zu einem dynamisch ausgewählten Desktop in einem Pool von Desktops erstellt, die alle auf einem Masterimage basieren. Alle Änderungen an dem Desktop gehen verloren, wenn die Maschine neu gestartet wird.
- Statischer, nicht beständiger Desktop: Wenn sich ein Benutzer zum ersten Mal anmeldet, um einen dieser Desktops zu verwenden, wird ihm ein Desktop aus einem Pool von Desktops zugewiesen, die alle auf einem Masterimage basieren. Daraufhin wird dem Benutzer jedes Mal, wenn er sich anmeldet, um einen der Desktops zu verwenden, derselbe Desktop wie beim ersten Mal zugewiesen. Alle Änderungen an dem Desktop gehen verloren, wenn die Maschine neu gestartet wird.
- Statisch und permanent (VDI-Desktop mit persönlicher vDisk): Im Gegensatz zu anderen Typen der VDI-Desktops können diese Desktops vollständig personalisiert werden. Wenn sich ein Benutzer zum ersten Mal anmeldet, um einen dieser Desktops zu verwenden, wird ihm ein Desktop aus einem Pool von Desktops zugewiesen, die alle auf einem Masterimage basieren. Daraufhin wird dem Benutzer jedes Mal, wenn er sich anmeldet, um einen der Desktops zu verwenden, derselbe Desktop wie beim ersten Mal zugewiesen. Änderungen am Desktop bleiben beim Neustart der Maschine erhalten, da sie auf einer persönlichen vDisk gespeichert werden.

So stellen Sie VDI-Desktops bereit:

1. Erstellen Sie ein Masterimage mit einem unterstützten Windows-Desktopbetriebssystem.
2. Erstellen Sie einen Maschinenkatalog für dieses Masterimage oder aktualisieren Sie einen vorhandenen Katalog mit dem Masterimage.  
Entscheiden Sie beim Definieren der Desktopdarstellung für den Maschinenkatalog, ob Benutzer bei jeder Anmeldung mit einer neuen VM oder mit derselben VM verbunden werden, und legen Sie fest, ob Änderungen am Desktop beibehalten werden.
3. Erstellen Sie eine Desktopbereitstellungsgruppe zum Bereitstellen der Desktops für Benutzer.

# Remote-PC-Zugriff

Nov 02, 2016

Mit Remote-PC-Zugriff kann ein Endbenutzer sich remote von jedem Standort aus an einem physischen Windows-PC im Büro anmelden. Der Virtual Delivery Agent (VDA) wird auf dem Büro-PC installiert. Er registriert sich bei dem Delivery Controller und verwaltet die HDX-Verbindung zwischen dem PC und den Clientgeräten des Endbenutzers. Remote-PC-Zugriff unterstützt ein Self-Service Modell. Nach der Einrichtung der Positivliste der Maschinen, die für Benutzer zugänglich sind, können die entsprechenden Benutzer ihre Büro-PCs einer Site ohne Eingriff des Administrators hinzufügen. Citrix Receiver wird auf ihrem Clientgerät ausgeführt und ermöglicht Zugriff auf alle Anwendungen und Daten auf dem Büro-PC von der Remote-PC-Zugriff-Desktopsitzung aus.

Ein Benutzer kann mehrere Desktops haben, einschließlich mehrere physische PCs oder eine Kombination von physischen PCs und virtuellen Desktops.

Hinweis: Energiesparmodus und Ruhezustand werden für Remote PC nicht unterstützt. Remote-PC-Zugriff gilt nur für XenDesktop-Lizenzen. Sitzungen verbrauchen Lizenzen genauso wie andere XenDesktop-Sitzungen.

Überlegungen zu Active Directory:

- Bevor Sie die Remote-PC-Bereitstellungssite konfigurieren, richten Sie die Organisationseinheiten und Sicherheitsgruppen ein und erstellen Sie dann Benutzerkonten. Verwenden Sie diese Konten, um die Benutzer für die Bereitstellungsgruppen anzugeben, die für Remote-PC-Zugriff verwendet werden.
- Wenn Sie Active Directory modifizieren, nachdem eine Maschine dem Maschinenkatalog hinzugefügt wurde, wird diese Zuweisung von Remote-PC-Zugriff nicht erneut ausgewertet. Sie können, falls erforderlich, Maschinen manuell einem anderen Katalog zuweisen.
- Wenn Sie Organisationseinheiten verschieben oder löschen, sind die für Remote-PC-Zugriff verwendeten möglicherweise nicht mehr aktuell. VDAs sind u. U. (wenn überhaupt) nicht mehr dem am besten geeigneten Maschinenkatalog bzw. der am besten geeigneten Bereitstellungsgruppe zugewiesen.

Überlegungen zu Maschinenkatalogen und Bereitstellungsgruppen:

- Eine Maschine kann nur jeweils einem Maschinenkatalog und einer Bereitstellungsgruppe zugewiesen sein.
- Sie können Maschinen in einen oder mehrere Remote PC-Zugriff-Maschinenkataloge einfügen.
- Wenn Sie Maschinenkonten für den Maschinenkatalog auswählen, wählen Sie die niedrigste zutreffende Organisationseinheit, um einen möglichen Konflikt mit Maschinen eines anderen Maschinenkatalogs zu vermeiden. Beispiel: Im Fall von Bank/Bankbeamte/Kassierer, wählen Sie Kassierer.
- Sie können alle Maschinen eines Remote PC-Maschinenkatalogs über eine oder mehrere Bereitstellungsgruppen zuweisen. Wenn eine Benutzergruppe beispielsweise bestimmte Richtlinieninstellungen und eine andere Gruppe andere Einstellungen benötigt, können Sie die Benutzer unterschiedlichen Bereitstellungsgruppen zuweisen und so die HDX-Richtlinien nach Bereitstellungsgruppe filtern.
- Wenn die Benutzerbetreuung in Ihrer IT-Infrastruktur auf dem geografischen Standort, der Abteilung oder einer anderen Kategorie beruht, können Sie Maschinen und Benutzer entsprechend gruppieren und so die delegierte Administration ermöglichen. Stellen Sie sicher, dass jeder Administrator über Berechtigungen für die relevanten Maschinenkataloge und die entsprechenden Bereitstellungsgruppen hat.
- Erstellen Sie für Benutzer mit Büro-PCs mit Windows XP einen separaten Maschinenkatalog und eine separate Bereitstellungsgruppe für diese Systeme. Aktivieren Sie bei der Auswahl von Maschinenkonten für diesen Katalog in Studio das Kontrollkästchen für Maschinen, die unter Windows XP ausgeführt werden.

Überlegungen zur Bereitstellung:

- Sie können eine Remote-PC-Zugriff-Bereitstellung erstellen und dann später konventionelle Virtual Desktop

Infrastructure-Desktops (VDI) oder Anwendungen hinzufügen. Sie können auch Remote-PC-Zugriff-Desktops einer vorhandenen VDI-Bereitstellung hinzufügen.

- Erwägen Sie eine Aktivierung der Windows-Remoteunterstützung bei der Installation des VDA auf Büro-PCs. Diese Option ermöglicht es den Helpdeskteams, die Director verwenden, Benutzersitzungen mit der Windows-Remoteunterstützung anzuzeigen und zu steuern.
- Legen Sie fest, wie Sie den VDA auf jedem Büro-PC bereitstellen möchten. Citrix empfiehlt, eine elektronische Softwareverteilung zu verwenden, z. B. Active Directory-Skripts und Microsoft System Center Configuration Manager. Das Installationsmedium enthält Active Directory-Beispielskripts.
- Die Funktionalität für sicheren Start wird zurzeit nicht unterstützt. Deaktivieren Sie sicheren Start, wenn Sie planen, den Workstation-VDA bereitzustellen.
- Jeder Büro-PC muss in der Domäne per Netzkabel verbunden sein.
- Windows 7 Aero wird für Büro-PCs unterstützt, ist aber nicht erforderlich.
- Tastatur und Maus müssen direkt am PC oder Laptop angeschlossen werden, nicht am Bildschirm oder an anderen Komponenten, die ausgeschaltet werden können. (Wenn Sie Eingabegeräte an solche Komponenten (z. B. Bildschirme) anschließen müssen, sollten diese nicht ausgeschaltet werden.)
- Wenn Sie Smartcards verwenden, lesen Sie [Smartcards](#).
- Remote-PC-Zugriff kann auf den meisten Laptop-Computern verwendet werden. Für einen besseren Zugriff und die optimale Verbindung konfigurieren Sie die Energiesparoptionen bei Laptops wie die eines Desktop-PCs. Beispiel:
  - Deaktivieren Sie den Ruhezustand.
  - Deaktivieren Sie den Standbymodus.
  - Legen Sie die Aktion bei Schließen des Deckels auf Keine Aktion fest.
  - Stellen Sie die Aktion Beim Drücken des Netzschalters auf Herunterfahren.
  - Deaktivieren Sie die Energiesparfunktionen der Grafikkarte.
  - Deaktivieren Sie die Energiesparfunktionen der Netzwerkkarte.
  - Deaktivieren Sie Akku-Energiespartechnologien.

Folgendes wird für Remote-PC-Zugriff-Geräte nicht unterstützt:

- An-/Ausdocken des Laptops.
  - KVM-Switches oder andere Komponenten, die eine Sitzung trennen.
  - Hybrid-PCs (einschließlich All-in-One- und NVIDIA Optimus-Laptops/-PCs) und Surface Pro/Book.
  - Installieren Sie Citrix Receiver auf jedem Clientgerät mit Remotezugriff auf den Büro-PC.
  - Mehrere Benutzer mit Remotezugriff auf denselben Büro-PC sehen in Receiver dasselbe Symbol. Wenn ein Benutzer eine Remoteanmeldung am PC vornimmt, wird diese Ressource anderen Benutzern als nicht verfügbar angezeigt.
  - Standardmäßig wird eine Remotesitzung des Benutzers automatisch getrennt, wenn ein lokaler Benutzer eine Sitzung auf dieser Maschine (durch Drücken von Strg + Alt + Entf) initiiert. Fügen Sie den folgenden Registrierungseintrag auf dem Büro-PC hinzu und starten Sie dann die Maschine neu, um diese automatische Aktion zu verhindern.
- Achtung: Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

HKLM\SOFTWARE\Citrix\PortICA\RemotePC "SasNotification"=dword:00000001

Zur weiteren Anpassung des Verhaltens dieses Features:

HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\PortICA\RemotePC

- RpcMode (dword)

- RpcTimeout (dword)

RpcMode:

1 bedeutet, dass immer der Remotebenutzer gewinnt, wenn er nicht innerhalb des angegebenen Timeouts auf die Meldung reagiert.

2 bedeutet, dass der lokale Benutzer immer gewinnt. Wenn diese Einstellung nicht festgelegt wird, gewinnt standardmäßig der Remotebenutzer.

RpcTimeout:

Timeout in Sekunden für den Benutzer, bis automatisch entschieden wird, welche Art von Modus erzwungen werden soll. Wenn diese Einstellung nicht festgelegt wird, gilt der Standardwert von 30 Sekunden. Als Mindestwert sollte hier 30 Sekunden festgelegt werden. Der Benutzer muss den Computer neu starten, damit die Änderungen in Kraft treten.

Wenn der lokale Benutzer den Zugriff auf die Konsole erzwingen möchte, kann er innerhalb von 10 Sekunden zwei Mal Strg + Alt + Entf drücken, um lokal auf die Remotesitzung zuzugreifen und eine Verbindungstrennung zu erzwingen.

Wenn ein lokaler Benutzer nach der Registrierungsänderung und dem Maschinenneustart für die Anmeldung am PC Strg+Alt+Entf drückt und der Computer von einem Remotebenutzer verwendet wird, wird der Remotebenutzer gefragt, ob er die Verbindung des lokalen Benutzers zulässt oder verweigert. Bei der Zulassung der Verbindung wird die Sitzung des Remotebenutzers getrennt.

Folgende XenDesktop-Features werden für Remote-PC-Zugriff-Bereitstellungen nicht unterstützt:

- Erstellen von Masterimages und virtuellen Maschinen
- Bereitstellen gehosteter Anwendungen
- Personal vDisks
- Clientordnerumleitung

Remote-PC-Zugriff unterstützt Wake-On-LAN, sodass physische PCs remote eingeschaltet werden können. Dieses Feature ermöglicht es Benutzern, ihre Büro-PCs ausgeschaltet zu lassen, wenn diese nicht verwendet werden, um Energiekosten zu sparen. Außerdem ist ein Remotezugriff möglich, wenn Maschinen unabsichtlich ausgeschaltet wurden, z. B. bei witterungsbedingten Problemen.

Im XenDesktop 7.6 Feature Pack 3 veröffentlichte Citrix ein experimentelles Wake-On-LAN-SDK. Mit dem SDK können Sie oder der Drittanbieter einer Wake-On-LAN-Lösung ohne System Center 2012 R2 einen Connector erstellen. Weitere Informationen finden Sie unter <http://support.citrix.com/article/CTX202272>.

Das Wake-On-LAN-Feature von Remote-PC-Zugriff wird auf Folgendem unterstützt:

- PCs, die Intel Active Management Technology (AMT) unterstützen.
- PCs, auf denen Wake-On-LAN im BIOS aktiviert ist.

Sie müssen Microsoft System Center Configuration Manager (ConfigMgr) 2012 konfigurieren, um das Wake-On-LAN-Feature zu verwenden. ConfigMgr bietet Zugriff auf AMT-Energiebefehle für den Computer und Unterstützung für den Aktivierungsproxy sowie Magic Packets. Wenn Sie anschließend mit Studio eine Bereitstellung für den Remote-PC-Zugriff erstellen (bzw. wenn Sie eine andere Energieverwaltungsverbindung für den Remote-PC-Zugriff hinzufügen), aktivieren Sie die Energieverwaltung und geben Sie ConfigMgr-Zugriffsinformationen an.

Beachten Sie außerdem Folgendes:

- Die Verwendung der AMT-Energieverwaltung wird aus Gründen der Sicherheit und Sitzungszuverlässigkeit zwar bevorzugt, es werden jedoch auch zwei Methoden ohne AMT unterstützt: ConfigMgr-Aktivierungsproxy und Magic Packets.
- Nur AMT-fähige Maschinen: Das Wake-On-LAN-Feature unterstützt auch die Aktionen "Herunterfahren erzwingen" und "Neustart erzwingen" von Studio und Director. Außerdem ist in StoreFront und Receiver eine Neustartaktion verfügbar.

Weitere Informationen finden Sie unter [Configuration Manager und Wake-On-LAN-Feature für den Remote-PC-Zugriff](#) und [Bereitstellen von Remote-PC-Zugriff für Benutzer](#).



# Bereitstellen von Remote-PC-Zugriff für Benutzer

May 10, 2016

Über Remote-PC-Zugriff haben Desktopbenutzer sicheren Zugriff auf Ressourcen auf dem Büro-PC und genießen gleichzeitig die Vorteile der Citrix HDX-Technologie.

Hinweis: Remote-PC-Zugriff gilt nur für XenDesktop-Lizenzen.

1. Zur Verwendung des Energieverwaltungsfeatures von Remote-PC-Zugriff (auch "Remote-PC-Zugriff-Wake-On-LAN" genannt), führen Sie die Konfigurationsaufgaben auf den PCs und in Microsoft System Center Configuration Manager (ConfigMgr) vor dem Erstellen der Remote-PC-Zugriff-Bereitstellung in Studio durch. Siehe [Configuration Manager und Wake-On-LAN-Feature für den Remote-PC-Zugriff](#).
2. Beim Erstellen der ersten Remote-PC-Zugriff-Bereitstellung können Sie die Energieverwaltung für die Maschinen im Standardmaschinenkatalog für Remote-PC-Zugriff aktivieren oder deaktivieren. Wenn Sie die Energieverwaltung aktivieren, geben Sie ConfigMgr-Verbindungsinformationen an. Geben Sie dann Benutzer und Computerkonten an. Weitere Informationen finden Sie unter [Erstellen einer Site](#). Die Erstellung einer Remote-PC-Bereitstellung schließt eine zukünftige VDI-Verwendung für die Site nicht aus.  
Bei der Erstellung einer Remote-PC-Zugriff-Bereitstellung wird ein Standardmaschinenkatalog unter dem Namen  
— *Remote-PC-Zugriff-Maschinen*  
und eine Standardbereitstellungsgruppe unter dem Namen  
— *Remote-PC-Zugriff-Desktops*  
erstellt.
3. Beim Erstellen eines anderen Maschinenkatalogs für die Verwendung mit Remote-PC-Zugriff gilt Folgendes:
  - Betriebssystem: Wählen Sie Remote-PC-Zugriff und wählen Sie eine Energieverwaltungsverbindung aus. Das Verwenden der Energieverwaltung ist nicht obligatorisch. Wenn keine Energieverwaltungsverbindungen konfiguriert sind, können Sie nach dem Abschließen des Assistenten für die Erstellung eines Maschinenkatalogs eine erstellen (Verbindungstyp = Microsoft Configuration Manager-Wake-On-LAN) und dann den Maschinenkatalog unter Angabe der neuen Verbindung bearbeiten.
  - Maschinenkonten: Sie können angezeigte Maschinenkonten oder Organisationseinheiten wählen oder Maschinenkonten bzw. Organisationseinheiten hinzufügen.
4. Installieren Sie den VDA auf dem Büro-PC, der für lokalen und Remotezugriff verwendet wird. Normalerweise wird der VDA automatisch mit dem Verwaltungssoftwarepaket bereitgestellt. Bei Test- oder kleinen Bereitstellungen können Sie jedoch den VDA manuell auf jedem Büro-PC installieren.  
Sobald der VDA installiert ist, wird der nächste Domänenbenutzer, der sich auf dem Büro-PC an einer Konsolensitzung anmeldet (lokal oder über RDP), automatisch dem Remote-PC-Desktop zugewiesen. Wenn in der Konsolensitzung weitere Domänenbenutzer angemeldet werden, werden sie der Desktopbenutzerliste unter Berücksichtigung der eingerichteten Beschränkungen ebenfalls hinzugefügt.  
Hinweis: Zum Verwenden von RDP-Verbindungen außerhalb Ihrer XenApp- bzw. XenDesktop-Umgebung müssen Sie Benutzer oder Gruppen zur Gruppe der Benutzer mit direktem Zugriff hinzufügen.
5. Weisen Sie die Benutzer an, auf jedem Clientgerät, das sie für den Remotezugriff auf den Büro-PC verwenden, Citrix Receiver herunterzuladen und zu installieren. Citrix Receiver ist unter <http://www.citrix.com> oder in den Anwendungsverteilungssystemen für unterstützte Mobilgeräte verfügbar.

Sie können eine Energieverwaltungsverbindung zum Konfigurieren der erweiterten Einstellungen bearbeiten. Sie können Folgendes aktivieren:

- ConfigMgr-Aktivierungsproxy
- Wake-On-LAN-Pakete (Magic Packets). Wenn Sie Wake-On-LAN-Pakete aktivieren, können Sie eine Wake-On-LAN-Übertragungsmethode auswählen: subnetzgesteuertes Broadcast oder Unicast.

Der PC verwendet AMT-Energiebefehle (sofern unterstützt) und alle aktivierten erweiterten Einstellungen. Wenn der PC keine AMT-Befehle verwendet, werden die erweiterten Einstellungen verwendet.

Der Delivery Controller schreibt die folgenden Diagnoseinformationen über Remote-PC-Zugriff in das Windows-Anwendungsereignisprotokoll. Informationsmeldungen werden nicht eingeschränkt. Fehlermeldungen werden durch Löschen doppelter Nachrichten eingeschränkt.

- 3300 (Informationsmeldung): Maschine zum Katalog hinzugefügt
- 3301 (Informationsmeldung): Maschine der Bereitstellungsgruppe hinzugefügt
- 3302 (Informationsmeldung): Maschine dem Benutzer zugewiesen
- 3303 (Fehler): Ausnahme

Wenn die Energieverwaltung für Remote-PC-Zugriff aktiviert ist, können Maschinen, die sich in einem anderen Subnetz als der Controller befinden, ggf. nicht per subnetzgesteuertes Broadcast gestartet werden. Wenn Sie eine subnetzübergreifende Energieverwaltung mit subnetzgesteuertem Broadcast benötigen und AMT nicht unterstützt wird, versuchen Sie es mit dem Aktivierungsproxy oder Unicast (stellen Sie sicher, dass diese Einstellungen in den erweiterten Eigenschaften der Energieverwaltungsverbindung aktiviert sind).

# Verwalten von Remote-PC-Zugriff-Bereitstellungsgruppen

May 10, 2016

Wenn eine Maschine eines Remote-PC-Zugriff-Maschinenkatalogs keinem Benutzer zugewiesen wurde, weist Studio vorübergehend die Maschine einer Bereitstellungsgruppe zu, die dem Maschinenkatalog zugeordnet ist. Diese temporäre Zuweisung enthält Informationen, die für die spätere Zuweisung an den geeigneten Benutzer benötigt werden. Die Zuweisung von Bereitstellungsgruppe zu Maschinenkatalog ist mit einem Prioritätswert verbunden.

Die Priorität bestimmt, welcher Bereitstellungsgruppe eine Maschine bei der Registrierung beim System oder wenn ein Benutzer eine Maschinenzuweisung benötigt, zugewiesen wird. Je geringer der Wert, desto höher die Priorität. Wenn ein Remote-PC-Zugriff-Maschinenkatalog mehrere Bereitstellungsgruppenzuweisungen hat, wird die mit der höchsten Priorität vom System ausgewählt. Sie können die Priorität mit dem PowerShell-SDK festlegen.

Beim Erstellen eines Remote-PC-Zugriff-Maschinenkatalogs wird dieser einer Bereitstellungsgruppe zugeordnet. Dies bedeutet, dass dem Maschinenkatalog später hinzugefügte Maschinenkonten oder Organisationseinheiten in der Bereitstellungsgruppe hinzugefügt werden können. Die Zuordnung kann deaktiviert oder aktiviert werden.

1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen aus.
2. Wählen Sie eine Remote-PC-Zugriff-Bereitstellungsgruppe aus.
3. Wählen Sie im Abschnitt "Details" die Registerkarte "Kataloge" und dann einen Remote-PC-Zugriff-Maschinenkatalog.
4. Um eine Zuordnung hinzuzufügen oder wiederherzustellen, wählen Sie Desktops hinzufügen. Zum Entfernen einer Zuordnung wählen Sie Zuordnung entfernen.

# App-V

May 10, 2016

Mit Microsoft Application Virtualization (App-V) können Sie Anwendungen als Dienste bereitstellen, aktualisieren und unterstützen. Benutzer können auf Anwendungen zugreifen, ohne sie auf ihren Geräten installieren zu müssen. App-V und Microsoft User State Virtualization (USV) ermöglichen den Zugriff auf Anwendungen und Daten unabhängig vom Standort oder von der Internetverbindung.

Die folgende Tabelle enthält eine Liste der unterstützten Versionen. (Der App-V-Client 4.6.2 wird nicht mehr unterstützt.)

| App-V           | XenDesktop-/XenApp-Version                                           |                          |
|-----------------|----------------------------------------------------------------------|--------------------------|
|                 | Delivery Controller                                                  | VDA                      |
| 5.0             | XenDesktop 7 bis aktuelle Version<br>XenApp 7.5 bis aktuelle Version | 7.0 bis aktuelle Version |
| 5.0 SP1         | XenDesktop 7 bis aktuelle Version<br>XenApp 7.5 bis aktuelle Version | 7.0 bis aktuelle Version |
| 5.0 SP2         | XenDesktop 7 bis aktuelle Version<br>XenApp 7.5 bis aktuelle Version | 7.1 bis aktuelle Version |
| 5.0 SP3 und 5.1 | XenDesktop 7.6<br>XenApp 7.6                                         | 7.6.300                  |

Der Offlinezugriff auf Anwendungen wird vom unterstützten App-V-Client nicht unterstützt. Die Unterstützung der App-V-Integration umfasst die Verwendung von SMB-Freigaben für Anwendungen. Das HTTP-Protokoll wird nicht unterstützt.

Anwendungen sind nahtlos verfügbar, ohne dass Vorkonfigurationen oder Änderungen an den Einstellungen des Betriebssystems vorgenommen werden müssen. App-V enthält die folgenden Komponenten:

- **Verwaltungsserver:** Bietet eine zentrale Konsole zum Verwalten der App-V-Infrastruktur und zum Bereitstellen von virtuellen Anwendungen für den App-V-Desktop Client und den Remotedesktopdienste-Client. Der App-V-Verwaltungsserver führt das vom Administrator benötigte Authentifizieren, Anfordern und Bereitstellen von Sicherheit, Messungen, Überwachung und Sammeln von Daten durch. Der Server verwendet Active Directory und unterstützende Tools zum Verwalten von Benutzern und Anwendungen.
- **Veröffentlichungsserver:** Stellt App-V-Clients Anwendungen für bestimmte Benutzer bereit und hostet das virtuelle Anwendungspaket für das Streaming. Die Pakete werden vom Verwaltungsserver abgerufen.
- **Client:** Ruft virtuelle Anwendungen ab, veröffentlicht die Anwendungen auf dem Client und erstellt und verwaltet automatisch virtuelle Umgebungen zur Laufzeit auf Windows-Geräten. Der App-V-Client wird auf dem Virtual Desktop Agent installiert und speichert benutzerspezifische virtuelle Anwendungseinstellungen, wie Registrierungs- und Dateiänderungen, in den Benutzerprofilen.

Sie können App-V-Anwendungen von Serverbetriebssystem- und Desktopbetriebssystem-Bereitstellungsgruppen starten:

- Über Citrix Receiver
- Vom Startmenü
- Über den App-V-Client und Citrix Receiver
- Gleichzeitig von mehreren Benutzern auf mehreren Geräten
- Über Citrix StoreFront

Geänderte App-V-Anwendungseigenschaften werden implementiert, wenn die Anwendung gestartet wird. Beispiel: Bei Anwendungen mit einem geänderten Anzeigenamen oder einem angepassten Symbol wird die Modifikation angezeigt, wenn Benutzer die Anwendung starten.

Es besteht keine Änderung in der Leistung von App-V-Anwendungen, wenn eine Desktop- und Anwendungsbereitstellungsgruppe in eine Nur-Anwendungsbereitstellungsgruppe geändert wird.

Es wird nur eine serverbasierte App-V-Bereitstellung unterstützt, in der ein Administrator mit einem App-V-Verwaltungsserver und -Veröffentlichungsserver App-V-Anwendungen verwaltet.

So stellen Sie App-V-Anwendungen bereit

1. Stellen Sie App-V bereit (Anweisungen siehe <http://technet.microsoft.com/en-us/virtualization/hh710199>).
2. Veröffentlichen Sie die App-V-Anwendungen auf dem App-V-Verwaltungsserver. Konfigurieren von Einstellungen, u. a. Berechtigungen und Dateitypzuordnung Diese Einstellungen sind bereits vorhanden, wenn Sie App-V bereits bereitgestellt haben.
3. Sie können auch die Einstellungen des App-V-Veröffentlichungservers ändern. Weitere Informationen finden Sie weiter unten.
4. Installieren Sie den App-V-Client auf VDAs.
5. Während der Siteerstellung in Studio geben Sie die URLs des App-V-Veröffentlichungs- und App-V-Managementsservers mit Portnummern an. Diese Server werden automatisch von den Bereitstellungsgruppen verwendet.
6. Installieren Sie den App-V-Client auf dem Masterimage für Maschinenkataloge. Konfigurieren Sie den Client mit Einstellungen, u. a. ShareContentStoreMode und EnablePackageScripts. (Sie müssen den App-V-Veröffentlichungsserver nicht im Masterimage konfigurieren, da er beim Anwendungsstart konfiguriert wird.)
7. Wählen Sie die App-V-Anwendungen beim Erstellen der Bereitstellungsgruppe aus.

Die Anwendungen sind jetzt verfügbar.

Sie können nach dem Erstellen einer Site die Informationen zum App-V-Server angeben oder ändern. Wählen Sie im Studio-Navigationsbereich Konfiguration > App-V-Veröffentlichung und dann Einträge im Aktionsbereich aus. Sie können App-V-Veröffentlichung mit URLs von Portnummern für die App-V-Verwaltungs- und -Veröffentlichungsserver hinzufügen. Sie können diese Adressen auch bearbeiten oder entfernen. Wenn Sie App-V-Anwendungen aktualisieren, wird angezeigt, ob ein Problem bei der Verbindung mit einem Server aufgetreten ist, und Einträge für nicht mehr verfügbare Anwendungen werden entfernt.

Zum Ändern der Veröffentlichungsseinstellungen empfiehlt Citrix die Verwendung der SDK-Cmdlets auf dem Controller.

- Geben Sie `Get-CtxAppvServerSetting -AppVPublishingServer` ein, um die Einstellungen des Veröffentlichungservers anzuzeigen.
- Um sicherzustellen, dass App-V-Anwendungen richtig gestartet werden, geben Sie `Set-CtxAppvServerSetting -UserRefreshonLogon 0` ein.

Mit dem folgenden Cmdlet werden die Einstellungen des App-V-Veröffentlichungservers auf dem Controller geändert. Nicht alle Parameter sind obligatorisch.

Set-CtxAppVServerSetting -AppVPublishingServer -UserRefreshOnLogon -UserRefreshEnabled -UserRefreshInterval -UserRefreshIntervalUnit -GlobalRefreshOnLogon -GlobalRefreshEnabled -GlobalRefreshInterval -GlobalRefreshIntervalUnit

Hinweis: Wenn Sie zuvor GPO-Richtlinieneinstellungen für die Verwaltung der Veröffentlichungsservereinstellungen verwendet haben, werden die App-V-Integrationsseinstellungen, einschließlich der vorherigen Cmdlet-Einstellungen, von den GPO-Einstellungen überschrieben. Dies kann dazu führen, dass der Start der App-V-Anwendung fehlschlägt. Citrix empfiehlt, dass Sie alle GPO-Richtlinieneinstellungen entfernen und die gleichen Einstellungen mit dem SDK konfigurieren.

- Wenn Sie die Adresse des App-V-Verwaltungsservers und des Veröffentlichungsservers in Studio festlegen und beim Testen der Verbindung ein Fehler auftritt, prüfen Sie Folgendes:
    1. App-V-Server ist eingeschaltet: Senden Sie entweder einen Ping-Befehl oder prüfen Sie in IIS-Manager, ob für jeden Server als Zustand "Gestartet" und "Wird ausgeführt" angezeigt wird.
    2. PowerShell Remoting ist auf dem App-V-Server aktiviert. Ist dies nicht der Fall, folgen Sie den Anweisungen unter <http://technet.microsoft.com/en-us/magazine/ff700227.aspx>.
    3. Der App-V-Server wurde Active Directory hinzugefügt.  
Sind Studio-Maschine und App-V-Server in verschiedenen Active Directory-Domänen, zwischen denen keine Vertrauensbeziehung besteht, führen Sie über die PowerShell-Konsole auf der Studio-Maschine winrm Config/client '@(TrustedHosts="<App-V-Server FQDN>")' aus. Wird "TrustedHosts" über das Gruppenrichtlinienobjekt verwaltet, wird eine Fehlermeldung angezeigt, die besagt, dass die Konfigurationseinstellung "TrustedHosts" nicht geändert werden kann, da sie von Richtlinien gesteuert wird, und dass die Richtlinie auf "Nicht konfiguriert" festgelegt werden müsste, damit die Konfigurationseinstellung geändert werden kann. Wenn diese Meldung angezeigt wird, fügen Sie einen Eintrag für den App-V-Servernamen in der TrustedHosts-Richtlinie im Gruppenrichtlinienobjekt hinzu (Administrative Vorlagen > Windows-Komponenten > Windows-Remoteverwaltung (WinRM) > WinRM-Client).
    4. Der Studio-Administrator ist gleichzeitig App-V-Serveradministrator.
    5. Auf dem App-V-Server ist die Dateifreigabe aktiviert. Geben Sie hierfür in Windows-Explorer oder über den Befehl "Ausführen" Folgendes ein: \\<App-V-Server FQDN>
    6. Der App-V-Server hat dieselben Dateifreigabeberechtigungen wie der App-V-Administrator. Fügen Sie hierzu auf dem App-V-Server einen Eintrag für \\<App-V-Server FQDN> in "Gespeicherte Benutzernamen und Kennwörter" hinzu und geben Sie die Anmeldeinformationen des Benutzers ein, der Administratorberechtigungen auf dem App-V-Server hat. Weitere Informationen finden Sie unter <http://support.microsoft.com/kb/306541>.
  - Wenn die Anwendungserkennung fehlschlägt, überprüfen Sie Folgendes:
    1. Der Studio-Administrator ist ein Administrator auf dem App-V-Verwaltungsserver.
    2. Der App-V-Verwaltungsserver wird ausgeführt. Prüfen Sie dies, indem Sie den IIS-Manager öffnen. Der Server sollte sich im Zustand "Gestartet" und "Ausgeführt" befinden.
    3. PowerShell Remoting ist auf den App-V-Servern aktiviert. Wenn es nicht aktiviert ist, folgen Sie den Anweisungen unter <http://technet.microsoft.com/en-us/magazine/ff700227.aspx>.
    4. Pakete haben entsprechende Sicherheitsberechtigungen, sodass der Studio-Administrator Zugriff hat.
  - Wenn die App-V-Anwendungen nicht starten, überprüfen Sie Folgendes:
    1. Der Veröffentlichungsserver wird ausgeführt. Prüfen Sie dies, indem Sie den IIS-Manager öffnen. Der Server sollte sich im Zustand "Gestartet" und "Ausgeführt" befinden.
    2. App-V-Pakete haben entsprechende Sicherheitsberechtigungen, sodass Benutzer Zugriff haben.
    3. Auf dem VDA:
      - Stellen Sie sicher, dass Temp auf den richtigen Speicherort verweist und dass genügend Speicherplatz im Verzeichnis Temp verfügbar ist.
      - Stellen Sie sicher, dass der App-V-Client installiert ist, und zwar mindestens Version 5.0.
      - Stellen Sie sicher, dass Sie Administratorberechtigungen haben und Get-AppVClientConfiguration ausführen. Stellen Sie sicher, dass EnablePackageScripts auf "1" gesetzt ist. Wenn es nicht auf "1" gesetzt ist, führen Sie Set-AppVClientConfiguration -EnablePackageScripts \$true aus.  
Citrix empfiehlt, dass Sie diesen Schritt beim Erstellen eines Masterimages ausführen, sodass alle VDAs, die vom Masterimage erstellt werden, die richtige Konfiguration haben.
      - Navigieren Sie im Registrierungs-Editor (regedit) zu HKEY\_LOCAL\_MACHINE\SOFTWARE\Policies\Citrix\AppV. Stellen Sie sicher, dass der Schlüssel "AppVServers" den folgenden Wert hat: AppVManagementServer+metadata;PublishingServer (zum Beispiel http://xmas-demo-appv.blrstm.com+0+0+1+1+1+0+1;http://xmas-demo-appv.blrstm.com8082).
      - Stellen Sie sicher, dass CtxAppVCOMAdmin über Administratorrechte verfügt. Bei der Installation von VDA wird normalerweise CtxAppVCOMAdmin erstellt und der lokalen Administratorgruppe auf der VDA-Maschine hinzugefügt. Abhängig von der Active Directory-Richtlinie verliert der Benutzer u. U. die administrative Zuordnung.  
Führen Sie compmgmt.msc aus und navigieren Sie zu den Benutzern "Lokale Benutzer und Gruppen". Wenn CtxAppVCOMAdmin kein Administrator ist, bearbeiten Sie die Gruppenrichtlinie oder wenden Sie sich an den Administrator, damit das Benutzerkonto die administrative Zuordnung behält.
    4. Auf dem Masterimage, auf dem der App-V-Client installiert ist, muss "PowerShell ExecutionPolicy" auf RemoteSigned eingestellt sein, da das von Microsoft bereitgestellte App-V-Clientmodul nicht signiert ist und PowerShell mit "ExecutionPolicy" nicht signierte Skripts und Cmdlets ausführen kann. Stellen Sie "ExecutionPolicy" mit einer der folgenden Methoden ein:
      - Melden Sie sich als Administrator an und geben Sie das folgende PowerShell-Cmdlet ein: Set-ExecutionPolicy RemoteSigned.
      - Navigieren Sie in den Gruppenrichtlinieneinstellungen zu "Computerkonfiguration > Richtlinien > Administrative Vorlagen > Windows-Komponenten > Windows PowerShell > Skriptausführung aktivieren".
    5. Überprüfen Sie die Veröffentlichungsserver:
      - Führen Sie Get-AppVPublishingServer \* aus, damit die Liste der Veröffentlichungsserver angezeigt wird.
      - Überprüfen Sie, ob UserRefreshOnLogon auf "False" festgelegt ist. Sonst schlägt der Start der ersten App-V-Anwendung normalerweise fehl.
      - Führen Sie mit Administratorprivilegien Set-AppVPublishingServer aus und stellen Sie "UserRefreshOnLogon" auf "False" ein.
- Wenn Sie mit diesen Schritten die Probleme nicht beheben können, aktivieren und prüfen Sie die Protokolle.

So aktivieren Sie Studio-Protokolle

1. Erstellen Sie den Ordner C:\CtxAppVLogs.
2. Gehen Sie zu C:\Programme\Citrix\StudioAppVIntegration\Snapshot\Citrix.AppVAdmin.V1 und öffnen Sie CtxAppVCommon.dll.config als Administrator in einem Texteditor, z. B. Editor. Heben Sie die Auskommentierung der folgenden Zeile auf:

So aktivieren Sie VDA-Protokolle

1. Erstellen Sie den Ordner C:\CtxAppVLogs.
2. Gehen Sie zu C:\Programme\Citrix\Virtual Desktop Agent und öffnen Sie CtxAppVCommon.dll.config als Administrator in einem Texteditor, z. B. Editor. Heben Sie die Auskommentierung der folgenden Zeile auf:
3. Heben Sie die Auskommentierung der folgenden Zeile auf und stellen Sie den Wert auf "1" ein, wie im folgenden Beispiel:

Alle mit der Konfiguration zusammenhängenden Protokolle befinden sich unter C:\CtxAppVLogs. Die Anwendungsstartprotokolle finden Sie unter:

- XenDesktop 7.1 und höher und XenApp 7.5 und höher: %LOCALAPPDATA%\Citrix\CtxAppVLogs.
- XenDesktop 7.0: %LocalAppData%\temp\CtxAppVLogs\

LOCALAPPDATA wird in den lokalen Ordner des angemeldeten Benutzers aufgelöst. Prüfen Sie den lokalen Ordner des Benutzers, für den der Anwendungsstart fehlschlägt.

4. Starten Sie als Administrator den Brokerdienst oder die VDA-Maschine neu, damit die Protokollierung gestartet wird.

# Lokaler App-Zugriff und URL-Umleitung

May 10, 2016

Durch lokalen App-Zugriff werden lokal installierte Windows-Anwendungen problemlos in eine gehostete Desktopumgebung integriert, ohne dass ein Wechsel zwischen Computern nötig ist. Lokaler App-Zugriff ermöglicht Folgendes:

- Direkter Zugriff von virtuellen Desktops auf Anwendungen, die lokal auf einem Laptop, PC oder einem anderen Gerät installiert sind
- Bereitstellung einer flexiblen Anwendungsbereitstellungslösung Wenn Benutzer lokale Anwendungen haben, die Sie nicht virtualisieren können oder die IT nicht verwaltet, verhalten sich diese Anwendungen weiterhin so, als ob sie auf einem virtuellen Desktop installiert wären.
- Eliminieren Sie Doppelkopplanz bei separat vom virtuellen Desktop gehosteten Anwendungen, indem Sie eine Verknüpfung mit der veröffentlichten Anwendung auf das Windows-Gerät des Benutzers platzieren.
- Unter anderem können die folgenden Anwendungen verwendet werden:
  - Videokonferenzsoftware, z. B. GoToMeeting.
  - Spezial- oder Nischenanwendungen, die noch nicht virtualisiert sind.
  - Anwendungen und Peripheriegeräte, die andernfalls große Datenmengen von einem Benutzergerät zum Server und zurück zum Benutzergerät senden würden, wie DVD-Brenner und TV-Tuner.

In XenApp und XenDesktop verwenden gehostete Desktopsitzungen die URL-Umleitung zum Starten von lokalen App-Zugriff-Anwendungen. Durch URL-Umleitung wird die Anwendung unter mehr als einer URL-Adresse bereitgestellt. Durch Auswählen eingebetteter Links in einem Browser in einer Desktopsitzung wird ein lokaler Browser gestartet (basierend auf der URL-Sperrliste des Browsers). Wenn Sie auf eine URL klicken, die nicht auf der Sperrliste steht, wird die URL erneut in der Sitzung geöffnet.

Die URL-Umleitung funktioniert nur in Desktopsitzungen und nicht in Anwendungssitzungen. Für Anwendungssitzungen können Sie nur Host-zu-Client-Inhaltsumleitung verwenden, wobei es sich um eine Art von Server-Dateitypzuordnung handelt. Diese FTA leitet bestimmte Protokolle an den Client um, z. B. HTTP, HTTPS, RTSP oder MMS. Wenn Sie beispielsweise nur eingebettete Links mit HTTP öffnen, werden die Links direkt in der Clientanwendung geöffnet. Für diese Art der Umleitung wird keine URL-Sperrliste oder URL-Positivliste unterstützt.

Wenn der lokale App-Zugriff aktiviert ist, werden URLs, die Benutzern als Links von lokal ausgeführten Anwendungen oder von Benutzern gehosteten Anwendungen bzw. als Verknüpfungen auf dem Desktop angezeigt werden, auf eine der folgenden Arten umgeleitet:

- Umleitung vom Computer des Benutzers zum gehosteten Desktop
- Umleitung vom XenApp- bzw. XenDesktop-Server auf den Computer des Benutzers
- Wiedergabe in der Umgebung, in der sie gestartet werden (keine Umleitung)

Zur Angabe des Pfads für die Inhaltsumleitung von bestimmten Websites konfigurieren Sie die URL-Positivliste und die URL-Sperrliste auf dem Virtual Delivery Agent. Diese Listen enthalten mehrteilige Registrierungsschlüssel, die die Richtlinieneinstellungen für die URL-Umleitung festlegen; weitere Informationen hierzu finden Sie unter "Einstellungen der Richtlinie 'Lokaler App-Zugriff'".

Mit den folgenden Ausnahmen können URLs auf dem VDA wiedergegeben werden:

- Regions-/Gebietsschemainformationen: Websites, die Gebietsschemainformationen benötigen, wie msn.com oder news.google.com (je nach Region wird eine bestimmte Seite geöffnet). Wenn VDA beispielsweise von einem Datenzentrum in Großbritannien bereitgestellt wird und der Client eine Verbindung aus Indien herstellt, würde der

Benutzer erwarten, dass die Website in.msn.com erscheint, es wird jedoch uk.msn.com angezeigt.

- Multimedia-Inhalt: Websites mit Rich-Media-Inhalten, die auf dem Clientgerät wiedergegeben werden, ermöglichen eine gewohnte Benutzererfahrung und das Einsparen von Bandbreite während die Funktionalität auch in Netzwerken mit hoher Latenz gewährleistet ist. Die Flash-Umleitung wird durch die Umleitung von Sites mit anderen Medientypen wie Silverlight ergänzt. Somit ist die Umgebung sehr sicher. Die vom Administrator genehmigten URLs werden auf dem Client ausgeführt, während die restlichen URLs an VDA weitergeleitet werden.

Neben der URL-Umleitung können Sie auch FTA-Umleitung (File Type Association = Dateitypzuordnung) verwenden. FTA startet lokale Anwendungen, wenn Dateien in einer Sitzung geöffnet werden sollen. Wenn die lokale Anwendung gestartet wird, muss sie Zugriff auf die Datei haben, um sie zu öffnen. Daher können Sie mit lokalen Anwendungen nur Dateien öffnen, die sich auf Netzwerkfreigaben oder auf Clientlaufwerken (mit Clientlaufwerkzuordnung) befinden. Wenn beispielsweise der PDF-Reader eine lokale Anwendung ist und eine PDF-Datei geöffnet werden soll, wird zum Öffnen der Datei der lokale PDF-Reader verwendet. Da die lokale Anwendung direkt auf die Datei zugreifen kann, erfolgt keine Netzwerkübertragung über ICA zum Öffnen der Datei.

Lokaler App-Zugriff wird auf den gültigen Betriebssystemen für VDAs für Windows-Serverbetriebssysteme und VDAs für Windows-Desktopbetriebssysteme unterstützt und erfordert mindestens Citrix Receiver für Windows Version 4.1. Die folgenden Browser werden unterstützt:

- Internet Explorer 8, 9, 10 und 11
- Firefox 3.5 bis 21.0
- Chrome 10

Beachten Sie die folgenden Punkte und Einschränkungen, wenn Sie lokalen App-Zugriff und URL-Umleitung verwenden.

- Lokaler App-Zugriff ist für virtuelle Desktops im Vollbildmodus unter Einbeziehung aller Monitore gedacht:
  - Die Benutzererfahrung kann beeinträchtigt werden, wenn lokaler App-Zugriff auf einem virtuellen Desktop verwendet wird, der im Fenstermodus ausgeführt wird oder der nicht auf allen Monitoren ausgeführt wird.
  - Bei der Verwendung mehrerer Monitore ist der maximierte Monitor der Standarddesktop für alle Anwendungen, die in der Sitzung gestartet werden, selbst wenn nachfolgende Anwendungen normalerweise auf einem anderen Monitor starten würden.
  - Das Feature unterstützt einen VDA; es ist keine Integration mit mehreren gleichzeitigen VDAs möglich.
- Einige Anwendungen können sich unerwartet verhalten und Benutzer beeinträchtigen:
  - Benutzer können die Laufwerksbuchstaben verwechseln, z. B. das lokale C:-Laufwerk mit dem virtuellen C:-Desktoplaufwerk.
  - Auf virtuellen Desktops verfügbare Drucker sind nicht für die lokalen Anwendungen verfügbar.
  - Anwendungen, die erweiterte Berechtigungen erfordern, können nicht als clientgehostete Anwendungen gestartet werden.
  - Keine spezielle Behandlung von Anwendungen mit einer Instanz (z. B. Windows Media Player).
  - Lokale Anwendungen werden mit dem Windows-Design des lokalen Computers angezeigt.
  - Vollbildanwendungen werden nicht unterstützt. Dies schließt Anwendungen ein, die im Vollbildmodus geöffnet werden, z. B. PowerPoint-Bildschirmpräsentationen oder Fotoanzeigen, die den gesamten Desktop ausfüllen.
  - Lokaler App-Zugriff kopiert die Eigenschaften der lokalen Anwendung (z. B. die Verknüpfungen auf dem Clientdesktop und im Startmenü) auf dem VDA. Es werden jedoch keine anderen Eigenschaften, wie Tastenkombinationen und schreibgeschützte Attribute, kopiert.
  - Anwendungen, die die Reihenfolge der überlappenden Fenster anpassen, können unvorhersehbare Ergebnisse verursachen. Beispielsweise könnten einige Fenster ausgeblendet werden.
  - Verknüpfungen, einschließlich Arbeitsplatz, Papierkorb, Systemsteuerung, Netzlaufwerkverknüpfungen und

Ordnerverknüpfungen werden nicht unterstützt.

- Die folgenden Dateitypen und Dateien werden nicht unterstützt: benutzerdefinierte Dateitypen, Dateien ohne zugeordnete Programme, ZIP-Dateien und ausgeblendete Dateien.
- Taskleistengruppierung wird nicht für gemischte 32-Bit und 64-Bit clientgehostete Anwendungen und VDA-Anwendungen unterstützt, z. B. die Gruppierung von 32-Bit-Versionen lokaler Anwendungen mit 64-Bit-VDA-Anwendungen.
- Anwendungen können nicht über COM gestartet werden. Beispiel: Wenn Sie auf ein eingebettetes Office-Dokument in einer Office-Anwendung klicken, wird der Prozessstart nicht erkannt und die Integration der lokalen Anwendung schlägt fehl.
- Die URL-Umleitung unterstützt nur explizite URLs, d. h. solche, die in der Adressleiste des Browsers angezeigt werden oder mit der browserinternen Suchfunktion gefunden wurden (je nach Browser).
- Die URL-Umleitung funktioniert nur in Desktopsitzungen und nicht in Anwendungssitzungen.
- Benutzer haben keine Berechtigung, im lokalen Desktopordner in einer VDA-Sitzung neue Dateien zu erstellen.
- Mehrere Instanzen einer lokal ausgeführten Anwendung verhalten sich entsprechend den Taskleisteneinstellungen für den virtuellen Desktop. Verknüpfungen zu lokal ausgeführten Anwendungen werden jedoch nicht mit ausgeführten Instanzen dieser Anwendungen gruppiert. Sie werden auch nicht mit ausgeführten Instanzen von gehosteten Anwendungen oder mit an gehosteten Anwendungen angehefteten Verknüpfungen gruppiert. Benutzer können nur Fenster von lokal ausgeführten Anwendungen von der Taskleiste aus schließen. Zwar können Benutzer die Fenster von lokalen Anwendungen in der Desktop-Taskleiste und im Startmenü anheften, jedoch starten die Anwendungen bei Verwendung dieser Verknüpfungen möglicherweise nicht konsistent.

Bei der Interaktion zwischen lokaler App-Zugriff und Windows tritt u. a. das folgende Verhalten auf.

- Verhalten von Verknüpfungen in Windows 8 und Windows Server 2012
  - Windows Store-Apps, die auf dem Client installiert sind, werden nicht als Teil der Verknüpfungen von lokalem App-Zugriff aufgelistet.
  - Bild- und Videodateien werden normalerweise standardmäßig mit Windows Store-Apps geöffnet. Lokaler App-Zugriff listet die Windows Store-Apps jedoch auf und öffnet Verknüpfungen mit Desktopanwendungen.
- Lokale Programme
  - In Windows 7 ist der Ordner im Startmenü verfügbar.
  - In Windows 8 ist der Ordner "Local Programs" nur verfügbar, wenn der Benutzer Alle Apps als Kategorie auf der Startseite auswählt. Nicht alle Unterordner werden in Local Programs angezeigt.
- Windows 8-Grafikfunktionen für Anwendungen
  - Desktopanwendungen sind auf den Desktopbereich beschränkt und werden von der Startseite bzw. Anwendungen im Windows 8-Stil vollständig abgedeckt.
  - Mit lokalem App-Zugriff verwendete Anwendungen verhalten sich jedoch bei der Verwendung von mehreren Monitoren nicht wie Desktopanwendungen. Bei der Verwendung mehrerer Monitore werden die Startseite und der Desktop auf unterschiedlichen Monitoren angezeigt.
- Windows 8 und lokaler App-Zugriff mit URL-Umleitung
  - Da bei Windows 8 Internet Explorer keine Add-Ons aktiviert sind, müssen Sie den Desktop-Internet Explorer zum Aktivieren von URL-Umleitung verwenden.
  - In Windows Server 2012 werden Add-Ons von Internet Explorer standardmäßig deaktiviert. Um URL-Umleitung zu implementieren, deaktivieren Sie die verstärkte Sicherheitskonfiguration für Internet Explorer. Setzen Sie die Internet Explorer-Optionen zurück und starten Sie das Programm neu, um sicherzustellen, dass Add-Ons für Standardbenutzer aktiviert sind.



# Konfigurieren von lokalem App-Zugriff und URL-Umleitung

May 10, 2016

So verwenden Sie den lokalen App-Zugriff und die URL-Umleitung für Citrix Receiver:

- Installieren Sie Receiver auf dem lokalen Clientcomputer. Sie können beide Features während der Installation von Receiver aktivieren. Alternativ können Sie die Vorlage für den lokalen App-Zugriff mit dem Gruppenrichtlinien-Editor aktivieren.
- Legen Sie die Richtlinieneinstellung für Lokalen App-Zugriff zulassen auf Aktiviert fest. Sie können zudem URL-Positivlisten- und Sperrlisten-Richtlinieneinstellungen für die URL-Umleitung konfigurieren. Weitere Informationen finden Sie unter [Einstellungen der Richtlinie "Lokaler App-Zugriff"](#).

## Aktivieren des lokalen App-Zugriffs und der URL-Umleitung bei der Installation von Receiver

So aktivieren Sie den lokalen App-Zugriff und die URL-Umleitung für alle lokalen Anwendungen:

1. Legen Sie die Richtlinieneinstellung für Lokalen App-Zugriff zulassen auf Aktiviert fest. Mit dieser Einstellung überlässt der VDA dem Client die Entscheidung darüber, ob vom Administrator veröffentlichte Anwendungen und Verknüpfungen für den lokalen App-Zugriff in der Sitzung aktiviert sind. (Wenn die Einstellung deaktiviert ist, sind sowohl vom Administrator veröffentlichte Anwendungen als auch Verknüpfungen für den lokalen App-Zugriff für den VDA deaktiviert.) Diese Richtlinie gilt für den gesamten Computer als auch für die URL-Umleitungsrichtlinie.
2. Aktivieren Sie den lokalen App-Zugriff und die URL-Umleitung für alle Benutzer eines Computers, wenn Sie Citrix Receiver installieren. Bei dieser Aktion werden die für URL-Umleitung erforderlichen Browser-Add-Ons registriert. Führen Sie an der Eingabeaufforderung den jeweiligen Befehl zum Installieren von Receiver mit der folgenden Option aus:  
`CitrixReceiver.exe /ALLOW_CLIENTHOSTEDAPPSURL=1`  
`CitrixReceiverWeb.exe /ALLOW_CLIENTHOSTEDAPPSURL=1`

## Aktivieren der Vorlage für den lokalen App-Zugriff mit dem Gruppenrichtlinien-Editor

1. Führen Sie `gpedit.msc` aus.
2. Wählen Sie den Knoten Computerkonfiguration. Klicken Sie mit der rechten Maustaste auf Administrative Vorlagen und wählen Sie Vorlagen hinzufügen/entfernen > Hinzufügen aus.
3. Fügen Sie die Vorlage `icaclient.adm` im Receiver-Konfigurationsordner (normalerweise unter `C:\Programme (x86)\Citrix\Online Plugin\Configuration`) hinzu. (Nach dem Hinzufügen der Vorlage `icaclient.adm` zur Computerkonfiguration steht diese auch in der Benutzerkonfiguration zur Verfügung.)
4. Erweitern Sie Administrative Vorlagen > Klassische administrative Vorlage (ADM) > Citrix Komponenten > Citrix Receiver > Benutzerauthentifizierung.
5. Wählen Sie Einstellungen für lokalen App-Zugriff aus.
6. Wählen Sie Aktiviert und anschließend URL-Umleitung zulassen aus. Registrieren Sie für die URL-Umleitung Browser-Add-Ons über die Befehlszeile wie unten erläutert.

## Ermöglichen von Zugriff ausschließlich auf veröffentlichte Anwendungen

So ermöglichen Sie Zugriff nur auf veröffentlichte Anwendungen:

1. Führen Sie auf dem Server, auf dem der Delivery Controller installiert ist, `regedit.exe` aus.
  1. Navigieren Sie zu `HKLM\Software\Wow6432Node\Citrix\DesktopStudio`.
  2. Fügen Sie den `REG_DWORD`-Eintrag `ClientHostedAppsEnabled` mit dem Wert 1 hinzu. (Durch den Wert 0 wird der lokale App-Zugriff deaktiviert.)

2. Starten Sie den Delivery Controller-Server neu und starten Sie dann Studio neu.
3. Veröffentlichen Sie lokale App-Zugriff-Anwendungen.
  1. Wählen Sie im Studio-Navigationsbereich Bereitstellungsgruppen und dann die Registerkarte "Anwendungen" aus.
  2. Wählen Sie im Aktionsbereich Anwendung mit lokalem Zugriff erstellen.
  3. Wählen Sie die Desktopbereitstellungsgruppe.
  4. Geben Sie den vollständigen Pfad der ausführbaren Datei der Anwendung auf dem lokalen Computer des Benutzers ein.
  5. Geben Sie an, ob die Verknüpfung mit der lokalen Anwendung auf dem virtuellen Desktop im Startmenü, auf dem Desktop oder beiden angezeigt wird.
  6. Akzeptieren Sie die Standardwerte auf der Seite Name und überprüfen Sie die Einstellungen.
4. Aktivieren Sie den lokalen App-Zugriff und die URL-Umleitung für alle Benutzer eines Computers, wenn Sie Citrix Receiver installieren. Bei dieser Aktion werden die für URL-Umleitung erforderlichen Browser-Add-Ons registriert. Führen Sie an der Eingabeaufforderung den Befehl zum Installieren von Receiver mit der folgenden Option aus:  
 CitrixReceiver.exe /ALLOW\_CLIENTHOSTEDAPPSURL=1  
 CitrixReceiverWeb.exe /ALLOW\_CLIENTHOSTEDAPPSURL=1
5. Legen Sie die Richtlinieneinstellung für Lokalen App-Zugriff zulassen auf "Aktiviert" fest. Mit dieser Einstellung überlässt der VDA dem Client die Entscheidung darüber, ob vom Administrator veröffentlichte Anwendungen und Verknüpfungen für den lokalen App-Zugriff in der Sitzung aktiviert sind. (Wenn die Einstellung deaktiviert ist, sind sowohl vom Administrator veröffentlichte Anwendungen als auch Verknüpfungen für den lokalen App-Zugriff für den VDA deaktiviert.)

## Registrieren von Browser-Add-Ons

Hinweis: Die für URL-Umleitung erforderlichen Browser-Add-Ons werden automatisch registriert, wenn Sie Receiver über die Befehlszeile mit folgender Option installieren: /ALLOW\_CLIENTHOSTEDAPPSURL=1.

Sie können ein Add-On oder alle mit den folgenden Befehlen registrieren und die Registrierung aufheben:

- Registrieren von Add-Ons auf einem Clientgerät: <client-installation-folder>\redirector.exe /reg<browser>
- Registrierung von Add-Ons auf einem Clientgerät aufheben: <client-installation-folder>\redirector.exe /unreg<browser>
- Registrieren von Add-Ons auf einem VDA: <VDAinstallation-folder>\VDARedirector.exe /reg<browser>
- Registrierung von Add-Ons auf einem VDA aufheben: <VDAinstallation-folder>\VDARedirector.exe /unreg<browser>

<browser> kann "IE", "FF", "Chrome" oder "All" sein.

Beispiel: Mit dem folgenden Befehl werden Internet Explorer-Add-Ons auf einem Gerät mit Receiver registriert.

C:\Programme\Citrix\ICA Client\redirector.exe/regIE

Mit dem folgenden Befehl werden alle Add-Ons auf einem VDA für Windows-Serverbetriebssysteme registriert.

C:\Programme (x86)\Citrix\System32\VDARedirector.exe /regAll

## URL-Abfang in Browsern

| Beschreibung                                                                                                                                                                                                                                                                                   | Konfiguration                                                                                                                                                                                                                                                                      |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Standardmäßig wird die eingegebene URL von Internet Explorer umgeleitet. Wenn die URL nicht in der Sperrliste enthalten ist, aber vom Browser oder der Website an eine andere URL-Adresse umgeleitet wird, wird die endgültige URL nicht umgeleitet, selbst wenn sie auf der Sperrliste steht. | Zum richtigen Funktionieren der URL-Umleitung müssen Sie bei entsprechender Aufforderung durch den Browser das Add-On aktivieren. Wenn die mit Internetoptionen verbundenen Add-Ons bzw. die angeforderten Add-Ons deaktiviert sind, funktioniert die URL-Umleitung nicht richtig. |
| Firefox-Add-Ons leiten URLs immer um.                                                                                                                                                                                                                                                          | Wenn ein Add-On installiert wurde, bietet Firefox auf einer                                                                                                                                                                                                                        |

|                                                                                                                  |                                                                                                                                                                                                                                                                                                                                          |
|------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Beschreibung</b>                                                                                              | <b>Konfiguration</b>                                                                                                                                                                                                                                                                                                                     |
|                                                                                                                  | neuen Registerkarte die Möglichkeit, die Add-On-Installation zuzulassen oder zu verhindern. Sie müssen das Add-On zulassen, damit das Feature funktioniert.                                                                                                                                                                              |
| Chrome-Add-Ons leiten die endgültige URL stets um, wenn es sich um geleitete und nicht eingegebene URLs handelt. | Die Erweiterungen wurden extern installiert. Wenn Sie die Erweiterung deaktivieren, funktioniert die URL-Umleitung in Google Chrome nicht. Wenn die URL-Umleitung im Inkognito-Modus erforderlich ist, lassen Sie durch Auswählen dieser Option in den Browsereinstellungen zu, dass die Erweiterung im Inkognito-Modus ausgeführt wird. |

## Konfigurieren des Verhaltens von lokalen Anwendungen bei der Abmeldung und Trennung

1. Führen Sie auf dem gehosteten Desktop gpedit.msc aus.
  1. Navigieren Sie zu HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\Client Hosted Apps\Policies\Session State.  
Bei 64-Bit-Systemen navigieren Sie zu HKEY\_LOCAL\_MACHINE\SOFTWARE\wow6432node\Citrix\Client Hosted Apps\Policies\Session State.
  2. Fügen Sie den REG\_DWORD-Eintrag "Terminate" mit einem der folgenden Werte hinzu:
    - 1: Lokale Anwendungen werden weiterhin ausgeführt, wenn sich ein Benutzer abmeldet oder die Verbindung zum virtuellen Desktop trennt. Bei der Wiederverbindung werden lokale Anwendungen wieder integriert, wenn sie in der lokalen Umgebung verfügbar sind.
    - 3: Lokale Anwendungen werden geschlossen, wenn sich ein Benutzer abmeldet oder die Verbindung zum virtuellen Desktop trennt.

# Server-VDI

Nov 02, 2016

Verwenden Sie das Server-VDI-Feature (Virtual Desktop Infrastructure), um einen Desktop von einem Serverbetriebssystem einem einzelnen Benutzer bereitzustellen.

- Enterprise-Administratoren können Serverbetriebssysteme als VDI-Desktops bereitstellen. Dies ist für Benutzer, z. B. Techniker und Designer, nützlich.
- Dienstanbieter können Desktops von der Cloud anbieten. Diese Desktops entsprechen dem Microsoft Services Provider License Agreement (SPLA).

Mit der Citrix Richtlinieneinstellung "Enhanced Desktop Experience" können Sie das Serverbetriebssystem wie ein Windows 7-Betriebssystem aussehen lassen.

Die folgenden Features können nicht mit der Server-VDI verwendet werden:

- Personal vDisks
- HDX 3D Pro
- Gehostete Anwendungen
- Lokaler App-Zugriff
- Direkte (nicht vermittelte) Desktopverbindungen
- Remote-PC-Zugriff

Damit die Server-VDI mit TWAIN-Geräten wie Scannern funktioniert, muss das Windows Server Desktop Experience-Feature installiert sein. Diese ist ein optionales Feature in Windows Server 2012, das Sie in folgendem Pfad finden: Verwaltung > Server-Manager > Features > Features hinzufügen > Desktop Experience.

Server-VDI wird auf den gleichen Serverbetriebssystemen wie der VDA für Windows Serverbetriebssysteme unterstützt.

1. Bereiten Sie den Windows-Server auf die Installation vor: Stellen Sie sicher, dass die Rollendienste "Remotedesktopdienste" nicht installiert sind, und dass Benutzer auf eine Sitzung eingeschränkt sind:
  - Stellen Sie mit dem Windows-Server-Manager sicher, dass die Rollendienste für Remotedesktopdienste nicht installiert sind. Wenn sie installiert sind, entfernen Sie die Dienste.
  - Stellen Sie sicher, dass die Eigenschaft "Nur eine Sitzung pro Benutzer zulassen" aktiviert ist.
    - Unter Windows Server 2008 R2 greifen Sie auf diese Eigenschaft über Verwaltung > Remotedesktopdienste > Remotedesktop-Sitzungshostkonfiguration zu. Im Abschnitt "Einstellungen bearbeiten > Allgemein" muss die Einstellung Nur eine Sitzung pro Benutzer zulassen auf Ja festgelegt sein.
    - Unter Windows Server 2012 bearbeiten Sie die Registrierung und stellen Sie die Terminalservereinstellung ein. Legen Sie für den Registrierungsschlüssel HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\TerminalServer den Eintrag DWORD fSingleSessionPerUser auf 1 fest.
2. Installieren Sie für Windows Server 2008 R2 Microsoft .NET Framework 3.5 SP1 auf dem Server, bevor Sie den VDA installieren.
3. Installieren Sie einen VDA über die Befehlszeilenschnittstelle mit den Optionen "/quiet" und "/servvdi" auf einem unterstützten Server oder einem Servermasterimage. (In der Standardeinstellung blockiert das Installationsprogramm den VDA für Windows-Desktopbetriebssysteme auf einem Serverbetriebssystem. Mit der Befehlszeile überschreiben Sie dieses Verhalten.)  
XenDesktopVdaSetup.exe /quiet /servvdi

Sie können den oder die Delivery Controller bei der VDA-Installation über die Befehlszeile mit der Option **/controller** festlegen.

Öffnen Sie mit der Option **/enable\_hdx\_ports** Ports in der Firewall, wenn die Firewall nicht manuell konfiguriert wird.

Fügen Sie die Option **/masterimage** hinzu, wenn Sie den VDA auf einem Image installieren, und verwenden Sie MCS zum Erstellen von Server-VMs von diesem Image.

Schließen Sie keine Optionen für Features ein, die nicht mit Server-VDI unterstützt werden, u. a. **/baseimage**, **/enable\_hdx\_3d\_pro** oder **/xa\_server\_location**.

4. Erstellen Sie einen Maschinenkatalog für Server-VDI.

1. Klicken Sie auf der Seite Betriebssystem auf Windows-Desktopbetriebssystem.
2. Geben Sie auf der Seite Zusammenfassung einen Maschinenkatalognamen und eine Beschreibung für Administratoren an, die ausdrücklich Server-VDI angeben. Dies ist die einzige Angabe in Studio, dass der Katalog Server-VDI unterstützt.

Wenn Sie eine Suche in Studio durchführen, wird der erstellte Server-VDI-Katalog auf der Registerkarte "Desktopbetriebssystemmaschinen" angezeigt, obwohl der VDA auf einem Server installiert wurde.

5. Erstellen Sie eine Bereitstellungsgruppe und weisen Sie den Server-VDI-Katalog zu, den Sie im letzten Schritt erstellt haben.

Wenn Sie während der Installation des VDA keinen Delivery Controller festgelegt haben, geben sie ihn später über eine Citrix RichtlinienEinstellung, Active Directory oder durch Bearbeiten der Registrierungswerte des VDA-Computers an.

# Entfernen von Komponenten

May 10, 2016

Zum Entfernen von Komponenten empfiehlt Citrix die Verwendung der Windows-Funktion zum Entfernen oder Ändern von Programmen. Alternativ können Sie Komponenten über die Befehlszeile oder mit einem auf dem Installationsmedium enthaltenen Skript entfernen.

Beim Entfernen von Komponenten werden keine Voraussetzungen entfernt und keine Firewall-Einstellungen geändert. Wenn Sie einen Controller entfernen, werden die SQL-Serversoftware und die Datenbanken nicht entfernt.

Bevor Sie einen Controller entfernen, müssen Sie ihn aus der Site entfernen. Vor dem Entfernen von Studio oder Director empfiehlt Citrix das Schließen dieser Anwendungen.

Wenn Sie einen Controller von einer früheren Bereitstellung mit dem Web Interface aktualisiert haben, müssen Sie zuerst die Webinterface-Komponente separat entfernen. Das Webinterface kann nicht mit dem Installationsprogramm entfernt werden.

So entfernen Sie Komponenten mit der Windows-Funktion zum Entfernen oder Ändern von Programmen

Gehen Sie mit der Windows-Funktion zum Entfernen oder Ändern von Programmen wie folgt vor:

- Zum Entfernen eines Controllers, von Studio, Director, eines Lizenzservers oder von StoreFront wählen Sie Citrix XenApp oder Citrix XenDesktop, klicken Sie mit der rechten Maustaste und wählen Sie Deinstallieren. Das Installationsprogramm wird gestartet und Sie können die zu entfernenden Komponenten markieren.  
Alternativ können Sie StoreFront entfernen, indem Sie mit der rechten Maustaste auf Citrix StoreFront klicken und dann Deinstallieren auswählen.
- Klicken Sie zum Entfernen eines VDAs auf Citrix Virtual Delivery Agent, klicken Sie mit der rechten Maustaste und wählen Sie Deinstallieren. Das Installationsprogramm wird gestartet und Sie können die zu entfernenden Komponenten markieren.
- Zum Entfernen des universellen Druckservers wählen Sie Citrix Universeller Druckserver, klicken Sie mit der rechten Maustaste und wählen Sie Deinstallieren.

So entfernen Sie Kernkomponenten über die Befehlszeile

Führen Sie auf dem Installationsmedium im Setupverzeichnis `\x64\XenDesktop` den Befehl `XenDesktopServerSetup.exe` aus.

- Zum Entfernen einer oder mehrerer Komponenten verwenden Sie die Optionen `/remove` und `/components`.
- Zum Entfernen aller Komponenten verwenden Sie die Option `/removeall`.

Informationen zu Befehl und Parametern finden Sie unter [Installieren an der Befehlszeile](#).

Mit dem folgenden Befehl wird beispielsweise Studio entfernt:

```
\x64\XenDesktop Setup\XenDesktopServerSetup.exe /remove /components studio
```

So entfernen Sie einen VDA über die Befehlszeile

Führen Sie auf dem Installationsmedium im Setupverzeichnis `\x64\XenDesktop` den Befehl `XenDesktopServerSetup.exe` aus.

- Zum Entfernen einer oder mehrerer Komponenten verwenden Sie die Optionen `/remove` und `/components`.
- Zum Entfernen aller Komponenten verwenden Sie die Option `/removeall`.

Informationen zu Befehl und Parametern finden Sie unter [Installieren an der Befehlszeile](#).

Mit dem folgenden Befehl werden beispielsweise der VDA und Receiver entfernt:

```
\x64\XenDesktop Setup\XenDesktopVdaSetup.exe /removeall
```

Informationen zum Entfernen von VDAs mit einem Skript in Active Directory finden Sie unter [Installieren oder Entfernen von Virtual Delivery Agents mit Skripts](#).

# Upgrades und Migration

May 10, 2016

## Upgrade

Bei Upgrades werden Bereitstellungen auf die neuesten Komponentenversionen aktualisiert, ohne dass neue Maschinen oder Sites erstellt werden müssen. Ein solches Upgrade wird als direktes Upgrade bezeichnet. Sie können ein Upgrade folgender Versionen durchführen:

- Von XenDesktop 5 (oder einer höheren Version) auf XenDesktop 7.6
- Von XenApp 7.5 auf XenApp 7.6

Sie können auch einen XenApp 6.5-Workerserver auf einen XenApp 7.6-VDA für Windows-Serverbetriebssysteme aktualisieren. Dies ist eine zusätzliche Aktivität bei der Migration von XenApp 6.5.

So aktualisieren Sie eine XenDesktop 5-Farm (oder höher) oder eine XenApp 7.5-Site

1. Führen Sie das Installationsprogramm auf den Maschinen aus, auf denen die Kernkomponenten und VDAs installiert sind. Die Software ermittelt, ob ein Upgrade zur Verfügung steht und installiert die aktuelle Version.
2. Verwenden Sie das aktualisierte Studio zum Aktualisieren der Datenbank und der Site.

Weitere Informationen finden Sie unter [Upgrade einer Bereitstellung](#).

Weitere Informationen zur Installation von Hotfixes für Controller finden Sie im Knowledge Center-Artikel [CTX201988](#).

So aktualisieren Sie einen XenApp 6.5-Workerserver auf einen XenApp 7.6-VDA

1. Führen Sie das Produktinstallationsprogramm auf dem XenApp 6.5-Workerserver aus. Die Software entfernt den Server aus der XenApp 6.5-Farm, dann wird die XenApp 6.5-Software entfernt und ein 7.6-VDA für Windows-Serverbetriebssysteme installiert.
2. Fügen Sie den Server nach dem Upgrade den Maschinenkatalogen und Bereitstellungsgruppen der 7.6-Site hinzu.

Weitere Informationen finden Sie unter [Upgrade eines XenApp 6.5-Workers auf einen neuen VDA für Windows-Serverbetriebssysteme](#).

## Migration

Durch Migration werden Daten von einer früheren Bereitstellung in die aktuelle Version verschoben. Sie können sowohl eine XenApp 6.5- als auch eine XenDesktop 4-Bereitstellung migrieren. Eine Migration umfasst die Installation von 7.6-Komponenten und das Erstellen einer neuen Site, das Exportieren der Daten aus der älteren Farm und dann das Importieren der Daten in die neue Site.

So migrieren Sie von XenApp 6.5

1. Installieren Sie die Kernkomponenten und erstellen Sie eine neue XenApp-Site.
2. Exportieren Sie Richtlinien- und/oder Farmdaten vom XenApp 6.5-Controller mit PowerShell-Cmdlets in XML-Dateien. Sie können den Inhalt der XML-Dateien bearbeiten, um die zu importierenden Informationen anzupassen.
3. Importieren Sie in der neuen 7.6-Site die Richtlinien- und/oder Anwendungsdaten mit PowerShell-Cmdlets und den XML-Dateien in die neue Site.
4. Schließen Sie die nach der Migration erforderlichen Schritte für die neue Site ab.

Weitere Informationen finden Sie unter [Migrieren von XenApp 6.x](#)

So migrieren Sie von XenDesktop 4



1. Installieren Sie die Kernkomponenten und erstellen Sie eine neue XenDesktop-Site.
2. Exportieren Sie mit dem Exportbefehl die Farmdaten von der XenDesktop 4-Farm in eine XML-Datei. Sie können den Inhalt der XML-Dateien bearbeiten, um die zu importierenden Informationen anzupassen.
3. Importieren Sie in der neuen 7.6-Site die Farmdaten mit dem Importbefehl und der XML-Datei in die neue Site.
4. Schließen Sie die nach der Migration erforderlichen Schritte für die neue Site ab.

Weitere Informationen finden Sie unter [Migrieren von XenDesktop 4](#).

# Upgrade einer Bereitstellung

Nov 02, 2016

Sie können bestimmte Bereitstellungen aktualisieren, ohne zunächst neue Maschinen oder Sites erstellen zu müssen. Ein solches Upgrade wird als direktes Upgrade bezeichnet. Sie können ein Upgrade folgender Versionen durchführen:

- Von XenDesktop 5 (oder einer höheren Version) auf die neueste freigegebene ("aktuelle") XenDesktop-Version
- Von XenApp 7.5 (oder einer höheren Version) auf die neueste freigegebene ("aktuelle") XenApp-Version

Sie können mit dem XenApp 7.6-Installationsprogramm auch einen XenApp 6.5-Workerserver auf einen XenApp 7.6-VDA für Windows-Serverbetriebssysteme aktualisieren. Dies ist eine zusätzliche Aktivität bei der Migration von XenApp 6.5. Weitere Informationen finden Sie unter [Upgrade eines XenApp 6.5-Workers auf einen neuen VDA für Windows-Serverbetriebssysteme](#).

Zum Starten eines Upgrades führen Sie das Installationsprogramm in der neuen Version aus, um zuvor installierte Kernkomponenten (Delivery Controller, Citrix Studio, Citrix Director, Citrix Lizenzserver) sowie Virtual Delivery Agents zu aktualisieren. Das Installationsprogramm ermittelt, welche Komponenten ein Upgrade erfordern und startet dann das Upgrade auf Ihren Befehl hin. Nach dem Upgrade der Komponenten verwenden Sie die neu aktualisierte Version von Studio zum Durchführen des Upgrades der Site-Datenbank und der Site.

Im vorliegenden Dokument bezieht sich Produkt, sofern nicht anders angegeben, auf XenApp 7.x oder XenDesktop 7.x.

In folgenden Situationen ist kein Upgrade möglich:

- Bei einer Early Release- oder Technology Preview-Version
- Bei einer XenApp-Version vor Version 7.5 (außer Sie ersetzen XenApp 6.5-Software auf einem Server mit einem aktuellen VDA für Serverbetriebssysteme, siehe [Migrieren von XenApp 6.x](#))
- Bei einer XenDesktop-Version vor Version 5.6, siehe [Migrieren von XenDesktop 4](#).
- Von XenApp auf XenDesktop oder von XenDesktop auf XenApp

Welche Versionen der Produktkomponenten können aktualisiert werden?

Mit dem Produktinstallationsprogramm und Studio können Sie folgende Komponenten aktualisieren:

- Delivery Controller 5 oder höher
- VDA 5.0 SP1 oder höher
  - Im Gegensatz zu früheren VDA-Releases müssen Sie für das Upgrade der VDAs das Produktinstallationsprogramm verwenden, MSI-Dateien können nicht verwendet werden.
  - Wenn das Installationsprogramm auf der Maschine Receiver für Windows erkennt (Receiver.exe), erfolgt ein Upgrade auf die Receiver-Version auf dem Installationsmedium des Produkts.
  - Wenn das Installationsprogramm Receiver für Windows Enterprise (CitrixReceiverEnterprise.exe) auf einer Maschine erkennt, wird das Programm auf Receiver für Windows Enterprise 3.4 aktualisiert
- Director 1 oder höher
- Datenbank: Es erfolgen ein Upgrade des Schemas und eine Migration der Daten der Sitedatenbank (sowie der Konfigurationsprotokollierungsdatenbank und der Überwachungsdatenbank, sofern eine frühere 7.x-Version aktualisiert wird).
- Personal vDisk

Aktualisieren Sie, falls erforderlich, anhand der Informationen in der Feature-/Produktdokumentation die folgenden Elemente:

- Provisioning Services (für XenApp 7.x und XenDesktop 7.x empfiehlt Citrix, dass Sie die aktuelle freigegebene Version

verwenden; die unterstützte Mindestversion ist Provisioning Services 7.0).

- Aktualisieren Sie den Provisioning Services-Server mit dem parallelen Serverupgrade und die Clients mit der vDisk-Versionsverwaltung.
- Provisioning Services 7.x unterstützt nicht das Erstellen von neuen Desktops mit Versionen von XenDesktop 5. Obwohl vorhandene Desktops weiterhin funktionieren, können Sie daher mit Provisioning Services 7.x erst dann neue Desktops erstellen, wenn Sie für XenDesktop ein Upgrade durchgeführt haben. Wenn Sie eine heterogene Umgebung mit XenDesktop 5.6 und XenDesktop 7.x-Sites planen, führen Sie daher kein Upgrade von Provisioning Services auf Version 7 durch.
- Microsoft System Center Virtual Machine Manager SCVMM. Das aktuelle Produkt unterstützt SCVMM 2012 und SCVMM 2012 SP1. XenDesktop 5.x unterstützt frühere Versionen. Verwenden Sie zur Minimierung von Ausfallzeiten die folgende Upgradereihenfolge:
  1. Wenn Sie Controller mit Versionen vor XenDesktop 5.6 FP1 haben, aktualisieren Sie diese auf XenDesktop 5.6 FP1 (siehe XenDesktop-Dokumentation für Version 5.6 FP1).
  2. Aktualisieren Sie die SCVMM-Server auf SCVMM 2012 (Anweisungen finden Sie in der Dokumentation von Microsoft).
  3. Aktualisieren Sie die XenDesktop-Komponenten auf die aktuelle Version.
  4. Falls gewünscht, können Sie ein Upgrade des SCVMM-Servers auf SCVMM 2012 SP1 durchführen.
- StoreFront:

## Anforderungen, Einschränkungen und Vorbereitung

|   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|---|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| • |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|   | Sie müssen die grafische Benutzeroberfläche oder die Befehlszeilenschnittstelle des Installationsprogramms für das Upgrade der Kernkomponenten und VDAs verwenden. Sie können keine Daten von einer früheren Version importieren oder migrieren.                                                                                                                                                                                                                                                                          |
|   | <p>Wenn Sie Komponenten auf die neue Version aktualisieren, andere Komponenten (auf anderen Maschinen) jedoch nicht, wird von Studio eine Erinnerung ausgegeben.</p> <p>Beispiel: Wenn ein Upgrade neue Versionen von Controller und Studio enthält und Sie nur den Controller aktualisieren (jedoch das Installationsprogramm auf der Maschine, auf der Studio installiert ist, nicht ausführen) lässt Studio eine Verwaltung der Site erst dann wieder zu, wenn Sie auch für Studio ein Upgrade durchgeführt haben.</p> |
|   | <p>Stellen Sie vor dem Upgrade des Citrix Lizenzservers sicher, dass Ihr Subscription Advantage-Datum für die neue Produktversion gültig ist.</p> <p>Wenn Sie ein Upgrade einer früheren 7.x-Produktversion durchführen, muss das Datum mindestens 2013.0522 sein.</p>                                                                                                                                                                                                                                                    |
|   | Sie können kein Upgrade von XenDesktop Express Edition durchführen. Beschaffen und installieren Sie eine Lizenz für eine derzeit unterstützte Edition und führen Sie anschließend ein Upgrade durch.                                                                                                                                                                                                                                                                                                                      |
|   | Vor Beginn einer Upgradeaktivität sichern Sie die Datenbank (siehe <a href="#">CTX135207</a> ), damit Sie sie im Falle eines Problems nach dem Datenbankupgrade wieder herstellen können.                                                                                                                                                                                                                                                                                                                                 |
|   | Optional können Sie Vorlagen sichern und eventuell verwendete Hypervisors aktualisieren.                                                                                                                                                                                                                                                                                                                                                                                                                                  |

|   |                                                                                                                                                                          |
|---|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| • | Wenn Sie Sites einer früheren Version neben Sites der aktuellen Version beibehalten müssen, lesen Sie den Abschnitt <a href="#">Hinweise zu heterogenen Umgebungen</a> . |
|   | Wenn Sie VDAs unter Windows XP oder Windows Vista installiert haben, lesen Sie den Abschnitt <a href="#">VDAs auf Maschinen mit Windows XP oder Windows Vista</a> .      |
|   | Wenn Sie nicht alle VDAs auf die aktuelle Version aktualisieren möchten, lesen Sie <a href="#">Unterstützung gemischter VDAs</a> .                                       |
|   | Enthält Ihre Bereitstellung Webinterface, empfiehlt Citrix die Verwendung von StoreFront.                                                                                |
|   | Auf den Maschinen, auf denen Sie die Produktkomponenten aktualisieren, müssen Sie sowohl Domänenbenutzer als auch lokaler Administrator sein.                            |
|   | Beim Upgrade legen Sie nicht das Produkt (XenApp oder XenDesktop) fest, das bei der ersten Installation festgelegt wurde.                                                |
|   | Lesen Sie den Abschnitt zur Upgradereihenfolge, damit Sie mögliche Ausfälle einplanen und das Risiko entschärfen können.                                                 |

## Hinweise zu heterogenen Umgebungen

Wenn Ihre Umgebung Sites/Farmen mit mehreren Produktversionen enthält (heterogene Umgebung), empfiehlt Citrix die Verwendung von StoreFront zum Aggregieren von Anwendungen und Desktops aus den unterschiedlichen Produktversionen (Beispiel: Sie haben eine XenDesktop 7.1-Site und eine XenDesktop 7.5-Site). Weitere Informationen finden Sie in der Dokumentation zu StoreFront.

- Im Allgemeinen werden von der aktuellen Studio- bzw. Director-Version nur aktuelle Sites verwaltet bzw. überwacht. (Diese Version von Director kann zwar XenDesktop 5.x-VDAs überwachen, jedoch sind einige Daten, u. a. die Anmeldedauer, für diese VDAs nicht verfügbar.) Sie können beispielsweise keine XenDesktop 7.1-Site mit Studio 7.6 verwalten. Analog dazu können Sie eine XenDesktop 7.6-Site nicht mit Studio 7.1 verwalten.
- Sie können aktuelle VDAs in Bereitstellungen mit älteren Controllerversionen verwenden. Allerdings sind in diesem Fall neue Features des aktuellen Releases möglicherweise nicht verfügbar. Weitere Informationen finden Sie unter *— Hinweise zu gemischten VDAs*.
- Sites mit Controllern der Version 5.x und VDAs der Version 7.x sollten in diesem Zustand nur vorübergehend verbleiben. Im Idealfall sollten Sie das Upgrade aller Komponenten so schnell wie möglich durchführen.
- Verwenden Sie in einer heterogenen Umgebung weiterhin Studio und Director für das jeweilige Release. Die verschiedenen Versionen müssen jedoch auf separaten Maschinen installiert sein.
- Wenn Sie XenDesktop 5.6 und XenDesktop 7.x-Sites parallel ausführen und für beide Provisioning Services verwenden möchten, stellen Sie entweder eine neue Version von Provisioning Services für die Verwendung mit der 7.x-Site bereit oder aktualisieren Sie die aktuelle Version von Provisioning Services. In diesem Fall können Sie jedoch keine neuen Arbeitslasten in der XenDesktop 5.6-Site mehr bereitstellen.
- Führen Sie ein Upgrade einer eigenständigen Studio-Version erst dann durch, wenn Sie zur Verwendung der neuen Version bereit sind.

## VDAs auf Maschinen mit Windows XP oder Windows Vista

Sie können kein Upgrade von VDAs auf Maschinen mit Windows XP oder Windows Vista auf eine 7.x-Version durchführen. Sie müssen VDA 5.6 FP1 mit bestimmten Hotfixes verwenden. Anweisungen hierfür finden Sie unter [CTX140941](#). Obwohl ältere VDAs in einer 7.x-Site funktionieren, können zahlreiche Features nicht verwendet werden. Beispiele:

- Features in Studio, die eine neuere VDA-Version erfordern
- Konfiguration von App-V-Anwendungen über Studio
- Konfiguration von Receiver StoreFront-Adressen über Studio
- Automatische Unterstützung der Microsoft Windows-KMS-Lizenzierung bei Verwendung von Maschinenerstellungsdiensten Weitere Informationen finden Sie unter [CTX128580](#).
- Informationen in Director:
  - Anmeldezeiten und Anmeldeende-Ereignisse, die sich auf die Anmeldungsdauer auswirken, in den Ansichten "Dashboard", "Trends" und "Benutzerdetails"
  - Eine Aufschlüsselung der Anmeldedauer für HDX-Verbindungs- und Authentifizierungszeiten sowie Zeitdauerangaben für Profil- und GPO-Ladevorgänge, Anmeldeskript und die Herstellung interaktiver Sitzungen
  - Einige Kategorien der Fehlerraten für Maschinen und Verbindungen
  - Aktivitätsmanager in der Ansicht "Helpdesk" und "Benutzerdetails"

Citrix empfiehlt ein Reimaging der Windows XP- und Windows Vista-Maschinen unter einer unterstützten Betriebssystemversion und die anschließende Installation der aktuellen VDA-Version.

## VDAs auf Maschinen mit Windows 8.x oder Windows 7

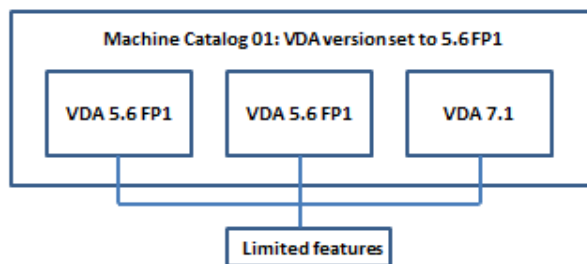
Für ein Upgrade von VDAs auf Maschinen mit Windows 8.x oder Windows 7 bis Windows 10 empfiehlt Citrix ein Reimaging der Maschinen gefolgt von der Installation des VDAs für Windows 10 unter Verwendung des eigenständigen VDA-Installationspakets, das im Lieferumfang von XenApp und XenDesktop 7.6 FP3 enthalten ist. Ist ein Reimaging nicht möglich, deinstallieren Sie den VDA vor dem Upgrade des Betriebssystems, da er ansonsten einen nicht unterstützten Zustand annimmt.

## Unterstützung gemischter VDAs

Für ein Upgrade des Produkts auf eine neuere Version empfiehlt Citrix, dass Sie alle Kernkomponenten und VDAs aktualisieren, damit Sie alle neuen und verbesserten Features der Edition verwenden können. Beispiel: Zur Verwendung der Features Sitzungsvorabstart, Sitzungsfortbestehen und nicht authentifizierte Benutzer des Releases 7.6 müssen die VDAs mindestens in Version 7.6 vorliegen.

In einigen Umgebungen ist ein Upgrade aller VDAs auf die aktuelle Version möglicherweise nicht möglich. In diesem Fall können Sie beim Erstellen eines Maschinenkatalogs die auf den Maschinen installierte VDA-Version angeben. Standardmäßig ist für diese Einstellung die neueste empfohlene VDA-Version festgelegt. Sie müssen eine Änderung dieser Einstellung nur dann in Betracht ziehen, wenn der Maschinenkatalog Maschinen mit früheren VDA-Versionen enthält. Das Vorhandensein mehrerer VDA-Versionen im selben Maschinenkatalog kann jedoch unerwünschte Auswirkungen haben.

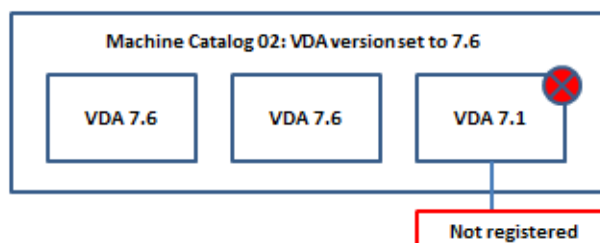
Wie oben erwähnt muss in Bereitstellungen mit Windows XP- und Windows Vista-Systemen eine frühere VDA-Version verwendet werden. Für den Maschinenkatalog, der die entsprechenden Maschinen enthält, muss VDA Version 5.6 FP1 festgelegt werden. Die VDAs können sich erfolgreich beim Controller registrieren, doch die Maschinen können viele der neuen Features der 7.x-Versionen (einschließlich StoreFront) nicht verwenden. Das Gleiche gilt für Maschinen mit VDAs der Version 7.x, die Sie einem solchen Katalog hinzufügen. Dies wird in der folgenden Abbildung dargestellt.



Wenn Sie wie im o. g. Fall ältere VDAs weiterhin verwenden müssen, fassen Sie diese in einem eigenen Maschinenkatalog zusammen.

Wenn ein Maschinenkatalog mit der standardmäßig empfohlenen VDA-Versionseinstellung erstellt wird und auf Maschinen in dem Katalog eine frühere VDA-Version installiert ist, können sich diese Maschinen nicht beim Controller registrieren und funktionieren nicht.

Beispiel (aktuelle VDA-Version sei 7.6): Sie erstellen einen Maschinenkatalog mit der VDA-StandardEinstellung: "7.6 (für die neuesten Features empfohlen)". Sie fügen dem Katalog drei Maschinen hinzu, zwei mit VDA 7.6 und eine mit VDA 7.1.

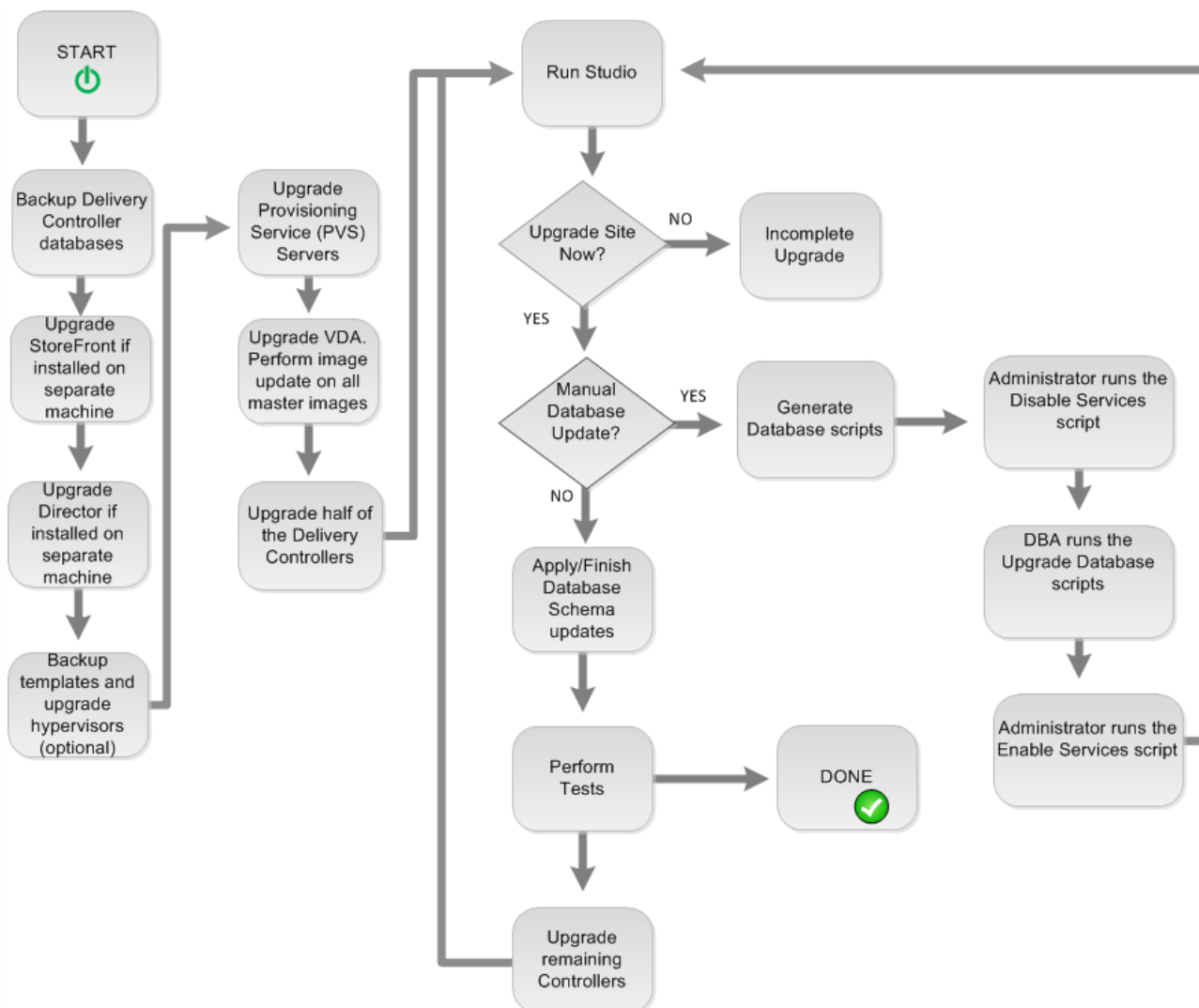


Die Maschine mit VDA 7.1 kann sich in diesem Fall nicht beim Controller registrieren. Wenn Sie kein Upgrade dieses VDAs durchführen können, können Sie einen separaten Maschinenkatalog mit einer VDA-Einstellung von Version 7.0 oder höher erstellen und diesem die Maschine hinzufügen. Die Maschine kann dann zwar die neuen Features der Version 7.6 nicht nutzen, sie kann sich jedoch beim Controller registrieren.

## Aktualisierungsreihenfolge

Wenn Komponenten auf unterschiedlichen Maschinen installiert sind, führen Sie das Installationsprogramm auf jeder dieser Maschinen durch.

Die Reihenfolge für das Upgrade ist nachfolgend aufgeführt, Erläuterungen folgen.



Zum Ausführen der grafischen Oberfläche des Installationsprogramms melden Sie sich bei der Maschine an und legen Sie anschließend das Installationsmedium ein oder stellen Sie das ISO-Laufwerk für das neue Release bereit. Doppelklicken Sie auf AutoSelect. Anweisungen zur Verwendung der Befehlszeilenschnittstelle finden Sie unter

— *Installieren über die Befehlszeile*

1. Wenn mehrere Kernkomponenten (z. B. Controller, Studio und Lizenzserver) auf dem gleichen Server installiert sind und für mehrere dieser Komponenten eine neue Version verfügbar ist, werden alle beim Ausführen des Installationsprogramms auf dem Server aktualisiert. Wenn Kernkomponenten auf anderen Maschinen als dem Controller installiert sind, führen Sie das Installationsprogramm auf diesen Maschinen aus (empfohlene Reihenfolge: Lizenzserver, StoreFront und dann Director).
2. Aktualisieren Sie die Provisioning Services-Server und -Clients (Anweisungen hierzu finden Sie in der Dokumentation zu Provisioning Services).
3. Führen Sie das Produktinstallationsprogramm auf Maschinen mit VDAs aus. Sie können zwar ein Upgrade der VDAs vor oder nach dem Upgrade des Controllers durchführen, Citrix empfiehlt jedoch ein Upgrade vor dem des Controllers, damit neue Features anschließend schnell aktiviert werden können.  
Wenn Sie ein Upgrade für VDAs von einer früheren 7.x-Version auf physischen Maschinen (einschließlich Remote-PC-Zugriff-Maschinen) durchführen, verwenden Sie die Befehlszeilenschnittstelle mit dem folgenden Parameter: /EXCLUDE "Personal vDisk","Machine Identity Service". Beispiel:  
`C:\x64\XenDesktop Setup\XenDesktopVdaSetup.exe /EXCLUDE "Personal vDisk","Machine Identity Service"`
4. Führen Sie das Produktinstallationsprogramm auf der Hälfte der Controller aus. (Damit werden auch alle anderen

Kernkomponenten auf diesen Servern aktualisiert.) Wenn Ihre Site beispielsweise vier Controller enthält, führen Sie das Installationsprogramm auf zwei Controllern aus.

- Dadurch dass die Hälfte der Controller aktiv bleibt, können Benutzer auf die Site zugreifen. Die VDAs können sich bei den anderen Controllern registrieren. Zeitweise wird die Site möglicherweise mit reduzierter Kapazität ausgeführt, da weniger Controller verfügbar sind. Durch das Upgrade wird nur für das Einrichten neuer Clientverbindungen während der letzten Datenbankaktualisierungsschritte eine kurze Unterbrechung verursacht. Die aktualisierten Controller können Anforderungen erst verarbeiten, wenn die gesamte Site aktualisiert wurde.
  - Wenn die Site nur einen Controller hat, ist sie während des Upgrades nicht funktionsfähig.
5. Ist Studio auf einer anderen Maschine installiert als einer der im vorherigen Schritt aktualisierten Controller, führen Sie das Installationsprogramm auf der Maschine aus, auf der Studio installiert ist.
  6. Führen Sie über die neu aktualisierte Studio-Version ein Upgrade der Sitedatenbank aus. Einzelheiten finden Sie unter [Aktualisieren der Datenbank und Site](#).
  7. Wählen Sie in der neu aktualisierten Studio-Version im Navigationsbereich Citrix Studio site-name aus. Wählen Sie die Registerkarte "Häufige Aufgaben". Wählen Sie Upgrade der übrigen Delivery Controller durchführen.
  8. Nach Abschluss des Upgrades und Bestätigen von dessen Durchführung schließen Sie Studio und öffnen Sie es erneut.
  9. Wählen Sie auf der Seite "Häufige Aufgaben" im Abschnitt "Sitekonfiguration" die Option Registrierung durchführen aus. Durch das Registrieren der Controller werden diese für die Site verfügbar.
  10. Nach Auswahl von Fertig stellen im Anschluss an das Upgrade können Sie sich optional für die Teilnahme beim Programm zur Verbesserung der Benutzerfreundlichkeit von Citrix (Customer Experience Improvement Program, CEIP) registrieren, durch das anonyme Informationen zu Ihrer Bereitstellung gesammelt werden. Diese Informationen werden Verbesserung von Qualität, Zuverlässigkeit und Leistung des Produkts verwendet.
  11. Nach dem Upgrade von Komponenten, Datenbank und Site führen Sie mit Studio folgende Arbeitsgänge durch:
    - Testen Sie die neu aktualisierte Site. Wählen Sie in Studio im Navigationsbereich Citrix Studio site-name aus. Wählen Sie die Registerkarte "Häufige Aufgaben" und dann Site testen. Diese Tests werden automatisch nach dem Upgrade der Datenbank ausgeführt, Sie können sie jedoch jederzeit wiederholen.
    - Aktualisieren Sie alle Masterimages, die den aktualisierten VDA verwenden.
    - Aktualisieren Sie Maschinenkataloge und Bereitstellungsgruppen.

## Aktualisieren Sie die Datenbank und die Site.

Nach dem Upgrade der Kernkomponenten und VDAs verwenden Sie die neu aktualisierte Studio-Version für ein automatisches oder manuelles Upgrade von Datenbank und Site.

- Für ein automatisches Datenbankupgrade müssen die Berechtigungen des Studio-Benutzers die Berechtigung zum Aktualisieren des SQL Server-Datenbankschemas umfassen (z. B. Datenbankrolle "db\_securityadmin" oder "db\_owner").
- Hat der Benutzer diese Berechtigungen nicht, werden bei einem manuellen Datenbankupgrade Skripts generiert. Der Studio-Benutzer führt einige dieser Skripts über Studio aus, andere werden vom Datenbankadministrator mit einem Tool wie SQL Server Management Studio ausgeführt. Bei der manuellen Ausführung von SQL-Skripts sollten sie mit dem SQLCMD-Hilfsprogramm oder mit SQL Management Studio im SQLCMD-Modus ausgeführt werden. Andernfalls können Fehler aufgrund von Ungenauigkeiten auftreten.

Wichtig: Citrix empfiehlt dringend, die Datenbank vor dem Upgrade zu sichern (siehe [CTX135207](#)).

Während des Datenbankupgrades sind die Produktdienste deaktiviert. Während dieser Zeit können Controller keine neuen Verbindungen für die Site verhandeln. Planen Sie daher sorgfältig.

Nach dem Upgrade der Datenbank und der Aktivierung der Produktdienste testet Studio Umgebung und Konfiguration und



generiert einen HTML-Bericht. Wenn Probleme identifiziert werden, können Sie die Datenbank aus dem Backup wiederherstellen. Wenn die Probleme beseitigt sind, können Sie die Datenbank erneut aktualisieren.

**Automatisches Upgrade von Datenbank und Site** Starten Sie die aktualisierte Studio-Version. Sobald Sie das automatische Upgrade der Site gestartet und bestätigt haben, dass Sie bereit sind, beginnt das Upgrade von Datenbank und Site.

**Manuelles Upgrade von Datenbank und Site** Bei diesem Vorgang werden Skripts generiert.

1. Starten Sie die aktualisierte Studio-Version. Sobald Sie das manuelle Upgrade der Site ausgewählt haben, prüft der Assistent die Kompatibilität des Lizenzservers und fordert eine Bestätigung an. Wenn Sie bestätigt haben, dass Sie die Datenbank gesichert haben, erstellt der Assistent Skripts und eine Checkliste der Upgradeschritte und zeigt diese an.
2. Führen Sie die folgenden Skripts in der angegebenen Reihenfolge aus:

| Skript                     | Beschreibung                                                                                                                                                                                                                                                                                    |
|----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DisableServices.ps1        | PowerShell-Skript, das vom Studio-Benutzer auf einem Controller ausgeführt werden muss und die Produktdienste deaktiviert                                                                                                                                                                       |
| UpgradeSiteDatabase.sql    | SQL-Skript, das vom Datenbankadministrator mit einem Tool wie SQL Server Management Studio auf dem Server mit der Sitedatenbank ausgeführt werden muss                                                                                                                                          |
| UpgradeMonitorDatabase.sql | SQL-Skript, das vom Datenbankadministrator mit einem Tool wie SQL Server Management Studio auf dem Server mit der Überwachungsdatenbank ausgeführt werden muss                                                                                                                                  |
| UpgradeLoggingDatabase.sql | SQL-Skript, das vom Datenbankadministrator mit einem Tool wie SQL Server Management Studio auf dem Server mit der Konfigurationsprotokollierungsdatenbank ausgeführt werden muss Führen Sie dieses Skript nur aus, wenn diese Datenbank geändert wird (z. B. nach dem Anwenden eines Hotfixes). |
| EnableServices.ps1         | PowerShell-Skript, das vom Studio-Benutzer auf einem Controller ausgeführt werden muss und die Produktdienste aktiviert                                                                                                                                                                         |

3. Wenn Sie alle angezeigten Checklistenaufgaben erledigt haben, wählen Sie Schließen Sie das Upgrade ab und kehren Sie zu 'Häufige Aufgaben' zurück.

# Upgrade eines XenApp 6.5-Workers auf einen neuen VDA für Windows-Serverbetriebssysteme

May 10, 2016

Wenn Sie das XenApp 7.6-Installationsprogramm auf einem XenApp 6.5-Workerserver ausführen, werden folgende Aktionen durchgeführt:

- Der Server wird aus der XenApp 6.5-Farm entfernt (für diese Aufgabe wird automatisch die Befehlszeile des XenApp 6.5-Installationsprogramms aufgerufen)
- Die XenApp 6.5-Software wird entfernt
- Ein neuer (XenApp 7.6 oder ein höheres unterstütztes Release) VDA für Windows-Serverbetriebssysteme wird installiert

Wenn Sie die grafische Benutzeroberfläche des Installationsprogramms verwenden, werden Sie durch denselben Assistenten geführt wie bei der Installation von VDAs für Windows-Serverbetriebssysteme in der neuen XenApp-Site. Auch die Befehlszeile verwendet die gleichen Befehle und Parameter, die Sie für die Installation anderer VDAs verwenden.

Durch die Installation der XenApp 7.6-Kernkomponenten und der anderen VDAs sind Sie wahrscheinlich bereits mit dem Installationsprogramm vertraut. Informationen zu den Vorbereitungen finden Sie unter [VDA-Installationsvorbereitung](#). Starten Sie dann das Installationsprogramm ([Installieren über die grafische Oberfläche](#)) oder führen Sie den Befehl auf dem XenApp 6.5-Workerserver aus ([Installieren von VDA über die Befehlszeile](#)).

Gut zu wissen:

- Dieses Upgrade ist auf XenApp 6.5-Servern gültig, die im Servermodus "nur Sitzungsmodus" konfiguriert wurden (auch Nur-Sitzungsserver oder Workerserver genannt).
- Das Deinstallieren von XenApp 6.5 erfordert mehrere Serverneustarts. Wenn Sie die Befehlszeile verwenden, können Sie mit der Option /NOREBOOT diese automatische Aktion unterdrücken. Sie müssen den Server jedoch neu starten, damit die Deinstallation und die nachfolgende Neuinstallation ausgeführt werden können.
- Wenn während der XenApp-Deinstallation ein Fehler auftritt, prüfen Sie das in der Fehlermeldung angegebene Deinstallationsfehlerprotokoll. Deinstallationsprotokolldateien befinden sich im Ordner "%TEMP%\Citrix\XenDesktop\Installation\XenApp 6.5 Uninstall Log Files\".
- Nach dem Upgrade der XenApp 6.5-Workerserver erstellen Sie mit Studio in der neuen XenApp-Site Maschinenkataloge (oder bearbeiten Sie vorhandene Kataloge) für die aktualisierten Worker.
- Wenn Sie Richtlinien- und Anwendungseinstellungen von einem XenApp 6.5-Controllerserver migriert haben (siehe [Migrieren von XenApp 6.x](#)), weisen Sie die Bereitstellungsgruppen mit den migrierten veröffentlichten Anwendungen dem Maschinenkatalog zu, in dem diese Anwendungen in XenApp 6.5 gehostet wurden.

## Problembehandlung

Symptome: Fehler beim Entfernen der XenApp 6.5-Software. Das Deinstallationsprotokoll enthält die Meldung: "Error 25703. An error occurred while plugging XML into Internet Information Server. Setup cannot copy files to your IIS Scripts directory. Please make sure that your IIS installation is correct."

- Ursache: Das Problem tritt auf Systemen auf, wenn (1) Sie während der XenApp 6.5-Erstinstallation angegeben haben, dass der Citrix XML-Dienst (CtxHttp.exe) keinen Port gemeinsam mit IIS verwenden soll, und (2) .NETFramework 3.5.1 installiert ist.
- Lösung:
  1. Entfernen Sie die Webserver (IIS)-Rolle mit dem Windows-Assistenten zum Entfernen von Rollen. (Sie können die Webserver (IIS)-Rolle später wieder installieren.)

2. Starten Sie den Server neu.
3. Deinstallieren Sie Folgendes mit der Windows-Funktion zum Installieren und Deinstallieren:
  1. Citrix XenApp 6.5
  2. Microsoft Visual C++ 2005 Redistributable (x64), Version 8.0.56336
4. Starten Sie den Server neu.
5. Installieren Sie den VDA für Windows-Serverbetriebssysteme mit dem XenApp 7.6-Installationsprogramm.

# Migrieren von XenApp 6.x

May 10, 2016

Wichtig: Lesen Sie den gesamten Artikel, bevor Sie mit der Migration beginnen.

Das Tool für die Migration von XenApp 6.x (das Migrationstool) ist eine Sammlung von PowerShell-Skripts und Cmdlets, die Richtliniendaten und Farmdaten für XenApp 6.x (6.0 und 6.5) migrieren. Dazu führen Sie auf dem XenApp 6.x-Controllerserver Export-Cmdlets aus, die die Daten in XML-Dateien zusammenfassen. Anschließend führen Sie vom XenApp 7.6-Controller aus die Import-Cmdlets aus, die mit den beim Export gesammelten Daten Objekte erstellen.

Eine Videoübersicht über das Migrationstool ist [hier](#) verfügbar.

Unten ist die Abfolge des Migrationsvorgangs zusammengefasst. Details werden später erläutert.

1. Auf einem XenApp 6.0- oder 6.5-Controller:
  1. Importieren Sie die PowerShell-Exportmodule.
  2. Exportieren Sie mit den Export-Cmdlets die Richtlinien- und/oder Farmdaten in XML-Dateien.
2. Kopieren Sie die XML-Dateien (und den Ordner mit den Symbolen, wenn sie für den Export nicht in die XML-Dateien eingebettet werden) auf den XenApp 7.6-Controller.
3. Auf dem XenApp 7.6-Controller:
  1. Importieren Sie die PowerShell-Importmodule.
  2. Importieren Sie mit den Import-Cmdlets die Richtlinien- und/oder Farmdaten (Anwendungen), wobei Sie die XML-Dateien als Eingabe verwenden.
4. Führen Sie die nach der Migration erforderlichen Schritte aus.

Vor der eigentlichen Migration können Sie die XenApp 6.x-Einstellungen exportieren und eine Exportvorschau in der XenApp 7.6-Site ausführen. Die Vorschau lässt mögliche Schwachstellen erkennen, damit Sie die Probleme vor der eigentlichen Migration beheben können. Bei einer Vorschau kann sich beispielsweise herausstellen, dass eine Anwendung mit dem gleichen Namen bereits in der neuen XenApp 7.6-Site vorhanden ist. Sie können die bei der Vorschau erstellten Protokolldateien bei der Migration als Leitfaden verwenden.

Sofern nicht anders angegeben, bezieht sich der Begriff 6.x auf XenApp 6.0 oder 6.5.

## Neue Features in diesem Release

Dieses Dezember 2014-Release (Version 20141125) enthält die folgenden Updates:

- Wenn bei der Verwendung des Migrationstools für eine XenApp 6.x-Farm Probleme auftreten, veröffentlichen Sie einen Bericht im Supportforum <http://discussions.citrix.com/forum/1411-xenapp-7x/>, sodass Citrix eine Untersuchung durchführen und das Tool verbessern kann.
- Neues Paketformat: Die Datei XAMigration.zip enthält jetzt zwei separate Pakete: ReadIMA.zip und ImportFMA.zip. Zum Exportieren von einem XenApp 6.x-Server benötigen Sie nur die ReadIMA.zip. Zum Importieren auf einen XenApp 7.6.x-Server benötigen Sie nur die ImportFMA.zip.
- Das Cmdlet "Export-XAFarm" unterstützt einen neuen Parameter (EmbedIconData), durch den das Kopieren der Symboldaten in verschiedene Dateien nicht mehr nötig ist.
- Das Cmdlet "Import-XAFarm" unterstützt drei neue Parameter:
  - MatchServer: Zum Importieren von Anwendungen von Servern, deren Namen mit einem Ausdruck übereinstimmen
  - NotMatchServer: Zum Importieren von Anwendungen von Servern, deren Namen nicht mit einem Ausdruck übereinstimmen
  - IncludeDisabledApps: Zum Importieren von deaktivierten Anwendungen
- Vorab gestartete Anwendungen werden nicht importiert.
- Das Cmdlet "Export-Policy" ist für XenDesktop 7.x.

## Migrationstoolpaket

Das Migrationstool ist auf der Citrix [Downloadsite](#) unter XenApp 7.6 verfügbar. Die Datei XAMigration.zip enthält zwei separate, unabhängige Pakete:

- ReadIMA.zip: enthält die Dateien zum Exportieren von Daten aus der XenApp 6.x-Farm sowie freigegebene Module.

| Modul bzw. Datei  | Beschreibung                                                                         |
|-------------------|--------------------------------------------------------------------------------------|
| ExportPolicy.psm1 | PowerShell-Skriptmodul zum Exportieren von XenApp 6.x-Richtlinien in eine XML-Datei. |

| Modul bzw. Datei  | Beschreibung                                                                               |
|-------------------|--------------------------------------------------------------------------------------------|
| ExportXAFarm.psm1 | PowerShell-Skriptmodul zum Exportieren von XenApp 6.x-Farmeinstellungen in eine XML-Datei. |
| ExportPolicy.psd1 | PowerShell-Manifestdatei für Skriptmodul ExportPolicy.psm1                                 |
| ExportXAFarm.psd1 | PowerShell-Manifestdatei für Skriptmodul ExportXAFarm.psm1                                 |
| LogUtilities.psm1 | Freigegebenes PowerShell-Skriptmodul mit Protokollierungsfunktionen                        |
| XmlUtilities.psd1 | PowerShell-Manifestdatei für Skriptmodul XmlUtilities.psm1                                 |
| XmlUtilities.psm1 | Freigegebenes PowerShell-Skriptmodul mit XML-Funktionen                                    |

- ImportFMA.zip: enthält die Dateien zum Importieren von Daten aus der XenApp 7.6-Farm sowie freigegebene Module.

| Modul bzw. Datei  | Beschreibung                                                            |
|-------------------|-------------------------------------------------------------------------|
| ImportPolicy.psm1 | PowerShell-Skriptmodul zum Importieren von Richtlinien nach XenApp 7.6. |
| ImportXAFarm.psm1 | PowerShell-Skriptmodul zum Importieren von Richtlinien nach XenApp 7.6. |
| ImportPolicy.psd1 | PowerShell-Manifestdatei für Skriptmodul ImportPolicy.psm1              |
| ImportXAFarm.psd1 | PowerShell-Manifestdatei für Skriptmodul ImportXAFarm.psm1              |
| PolicyData.xsd    | XML-Schema für Richtliniendaten.                                        |
| XAFarmData.xsd    | XML-Schema für XenApp-Farmdaten.                                        |
| LogUtilities.psm1 | Freigegebenes PowerShell-Skriptmodul mit Protokollierungsfunktionen     |
| XmlUtilities.psd1 | PowerShell-Manifestdatei für Skriptmodul XmlUtilities.psm1              |
| XmlUtilities.psm1 | Freigegebenes PowerShell-Skriptmodul mit XML-Funktionen                 |

## Einschränkungen

- Nicht alle Richtlinieneinstellungen werden importiert; siehe [Nicht importierte Richtlinieneinstellungen](#). Einstellungen, die nicht unterstützt werden, werden ignoriert und in der Protokolldatei angegeben.
- Zwar werden alle Anwendungsdetails während des Exportvorgangs in der XML-Ausgabedatei gesammelt, aber nur auf Servern installierte Anwendungen werden in die XenApp 7.6-Site importiert. Veröffentlichte Desktops, Inhalte und die meisten gestreamten Anwendungen werden nicht unterstützt (Informationen zu Ausnahmen finden Sie unter [Schrittweise Anleitungen: Importieren von Daten](#)) im Abschnitt zu den Import-XAFarm-Cmdlet-Parametern.
- Anwendungsserver werden nicht importiert.
- Viele Anwendungseigenschaften werden nicht importiert wegen der Unterschiede zwischen der XenApp 6.x Independent Management Architecture (IMA) und der XenApp 7.6 FlexCast Management Architecture (FMA). Weitere Informationen hierzu finden Sie unter [Zuordnung von Anwendungseigenschaften](#).
- Während des Imports wird eine Bereitstellungsgruppe erstellt. Weitere Informationen zum Filtern des importierten Inhalts mit Parametern finden Sie unter [Erweiterte Verwendung](#).
- Nur Citrix Richtlinieneinstellungen, die mit der AppCenter-Verwaltungskonsolle erstellt wurden, werden importiert. Citrix Richtlinieneinstellungen, die mit Windows Gruppenrichtlinienobjekten erstellt wurden, werden nicht importiert.
- Die Migrationsskripts sind nur für die Migrationen von XenApp 6.x auf XenApp 7.6 vorgesehen.
- Mehr als fünffach verschachtelte Ordner werden von Studio nicht unterstützt und werden nicht importiert. Wenn die Ordnerstruktur Ihrer Anwendung Ordner mit mehr als fünf Ebenen von Unterordnern enthält, reduzieren Sie vor dem Importieren die Anzahl der verschachtelten Ordner.

## Sicherheitsüberlegungen

Die durch die Exportskripts erstellten XML-Dateien können vertrauliche Informationen über Ihre Umgebung und Organisation enthalten, z. B. Benutzernamen, Servernamen und andere XenApp-Farm-, Anwendungs- und Richtlinienkonfigurationsdaten. Speichern und verwenden Sie diese Dateien in einer sicheren Umgebung.

Prüfen Sie die XML-Dateien sorgfältig, bevor Sie sie als Eingabe für den Import von Richtlinien und Anwendungen verwenden, um sicherzustellen, dass sie keine unbefugten Änderungen enthalten.

Richtlinienobjektzuweisungen (bisher "Richtlinienfilter") steuern die Anwendung von Richtlinien. Nach dem Importieren von Richtlinien prüfen Sie die Objektzuweisungen für jede Richtlinie sorgfältig, um sicherzustellen, dass durch den Import keine Sicherheitsrisiken entstanden sind. Nach dem Import können auf die Richtlinie verschiedene Gruppen von Benutzern, IP-Adressen oder Clientnamen angewendet werden. Die Einstellungen zum Zulassen und Verweigern haben möglicherweise nach dem Import eine andere Bedeutung.

## Protokollierung und Fehlerbehandlung

Die Skripts sorgen für umfangreiche Protokollierung, wobei die Ausführung aller Cmdlets, informative Meldungen, die Ergebnisse der Cmdlet-Ausführung sowie Warnungen und Fehler aufgezeichnet werden.

- Die Verwendung der Citrix PowerShell-Cmdlets wird größtenteils protokolliert. Alle PowerShell-Cmdlets in den Importskripts, die neue Siteobjekte erstellen, werden protokolliert.
- Der Skriptausführungsverlauf wird protokolliert, einschließlich der Objekte, die verarbeitet werden.
- Große Aktionen, die sich auf den Flusstatus auswirken, werden protokolliert, einschließlich über die Befehlszeile geleitete Flüsse.
- Alle Meldungen, die auf der Konsole gedruckt werden, einschließlich Warnungen und Fehler werden protokolliert.
- Jede Zeile wird mit einem Zeitstempel versehen, der auf die Millisekunde genau ist.

Citrix empfiehlt, dass Sie beim Ausführen der Export- und Import-Cmdlets jeweils eine Protokolldatei angeben.

Wenn Sie keinen Protokolldateinamen angeben, wird die Protokolldatei im Basisordner des aktuellen Benutzers (in der PowerShell-Variablen \$HOME angegeben) gespeichert. Wenn dieser Ordner nicht vorhanden ist, wird die Protokolldatei im aktuellen Ausführungsordner des Skripts gespeichert. Der Standardname der Protokolldatei ist "XFarmYYYYMMDDHHmmSS-xxxxxx", wobei die letzten sechs Ziffern eine zufällige Zahl sind.

Standardmäßig werden die gesamten Fortschrittsinformationen angezeigt. Um die Anzeige zu unterdrücken, legen Sie den NoDetails-Parameter in den Export- und Import-Cmdlets fest.

Bei einem Fehler wird die Ausführung eines Skripts im Allgemeinen angehalten. Wenn der Fehler behoben ist, können Sie das Cmdlet noch einmal ausführen.

Bedingungen, die nicht als Fehler gelten, werden protokolliert und oft als Warnung gemeldet, während die Skriptausführung fortgesetzt wird. Beispielsweise werden nicht unterstützte Anwendungstypen als Warnung gemeldet und nicht importiert. Anwendungen, die bereits in der XenApp 7.6-Site vorhanden sind, werden nicht importiert. Richtlinieneinstellungen, die in XenApp 7.6 veraltet sind, werden nicht importiert.

Die Migrationsskripts verwenden viele PowerShell-Cmdlets und nicht alle möglichen Fehler werden protokolliert. Zusätzliche Protokollierungsfunktionen sind mit den PowerShell-Protokollierungsfeatures verfügbar. Beispielsweise wird alles, was auf dem Bildschirm gedruckt wird, in PowerShell-Aufzeichnungen protokolliert. Weitere Informationen finden Sie in der Hilfe zu den Cmdlets "Start-Transcript" und "Stop-Transcript".

## Anforderungen, Vorbereitungen und Best Practices

Wichtig: Lesen Sie den gesamten Artikel, bevor Sie mit der Migration beginnen.

Sie sollten die grundlegenden PowerShell-Konzepte der Ausführungsrichtlinie, Module, Cmdlets und Skripts verstehen. Obwohl umfangreiche Erfahrung in der Erstellung von Skripts nicht erforderlich ist, sollten Sie die ausgeführten Cmdlets verstehen. Mit dem Cmdlet "Get-Help" können Sie sich die Hilfe zu jedem Migrations-Cmdlet ansehen, bevor Sie es ausführen. Beispiel:

Get-Help -full Import-XAFarm

Geben Sie eine Protokolldatei in der Befehlszeile an und überprüfen Sie die Protokolldatei jedes Mal, nachdem Sie ein Cmdlet ausgeführt haben. Wenn ein Skript fehlschlägt, identifizieren Sie den Fehler mit der Protokolldatei und beheben Sie ihn. Führen Sie dann das Cmdlet noch einmal aus.

### Gut zu wissen:

- Zur Vereinfachung der Bereitstellung von Anwendungen während der Ausführung beider Bereitstellungen (vorhandene XenApp 6.x-Farm und neue XenApp 7.6-Site) können Sie beide Bereitstellungen in StoreFront oder dem Webinterface aggregieren. Weitere Informationen zu Ihrem StoreFront- oder Webinterface-Release finden Sie in der Dokumentation in den eDocs (Verwalten > Store erstellen).
- Für die Handhabung der Anwendungssymboldaten gibt es zwei Möglichkeiten:
  - Wenn Sie den Parameter "EmbedIconData" im Cmdlet "Export-XAFarm" angeben, werden exportierte Anwendungssymboldaten in der XML-Ausgabedatei eingebettet.
  - Wenn Sie den Parameter "EmbedIconData" im Cmdlet "Export-XAFarm" nicht angeben, werden exportierte Anwendungssymboldaten in einem Ordner gespeichert. Der Name des Ordners wird durch Anfügen der Zeichenfolge "-icons" an den Basisnamen der XML-Ausgabedatei erstellt. Wenn der Parameter "XmlOutputFile" beispielsweise "FarmData.xml" ist, wird der Ordner "FarmData-icons" zum Speichern der Anwendungssymbole erstellt. Die Symboldateien in diesem Ordner sind TXT-Dateien, die nach den Browsernamen der veröffentlichten Anwendungen benannt sind (die Dateien sind zwar TXT-Dateien, die gespeicherten Daten sind jedoch verschlüsselte binäre Symboldaten, die vom Importskript zum erneuten Erstellen des Anwendungssymbols gelesen werden können). Wenn der Symbolordner während des Importvorgangs nicht im selben Verzeichnis gefunden wird wie die

XML-Importdatei, werden allgemeine Symbole für die importierten Anwendungen verwendet.

- Die Namen der Skriptmodule, Manifestdateien, freigegebenen Module und Cmdlets sind ähnlich. Verwenden Sie die Tabulatortaste vorsichtig, damit es nicht zu Fehlern kommt. Beispiel: Export-XAFarm ist ein Cmdlet. ExportXAFarm.psd1 und ExportXAFarm.psm1 sind Dateien, die nicht ausgeführt werden können.
- In den nachfolgenden schrittweisen Anleitungen sind die meisten Parameterwerte für <string> mit Anführungszeichen versehen. Diese sind optional für Zeichenfolgen, die nur aus einem Wort bestehen.

#### Für den Export des XenApp 6.x-Servers gilt Folgendes:

- Der Export muss auf einem XenApp 6.x-Server ausgeführt werden, der mit dem Servermodus Controller- und Sitzungshostmodus (üblicherweise Controller genannt) konfiguriert wurde.
- Zum Ausführen der Export-Cmdlets müssen Sie XenApp-Administrator mit der Berechtigung zum Lesen von Objekten sein. Sie müssen auch über die erforderlichen Windows-Berechtigungen zum Ausführen von PowerShell-Skripts verfügen. Schrittweise Anleitungen finden Sie unten.
- Stellen Sie sicher, dass die XenApp 6.x-Farm funktionsfähig ist, bevor Sie mit dem Export beginnen. Sichern Sie die Farmdatenbank. Überprüfen Sie die Integrität der Farm mit dem Dienstprogramm "Citrix IMA Helper" ([CTX133983](#)): Führen Sie von der Registerkarte für den IMA Datastore aus einen Master Check aus (und lösen Sie alle ungültigen Einträge mit der Option "DSCheck" auf). Durch das Reparieren von Problemen vor der Migration werden Fehler beim Export vermieden. Wenn ein Server beispielsweise nicht richtig aus der Farm entfernt wird, bleiben seine Daten möglicherweise in der Datenbank vorhanden, was zu Fehlern bei den Cmdlets im Exportskript führen kann (z. B. Get-XAServer -ZoneName). Wenn die Cmdlets fehlschlagen, schlägt das Skript fehl.
- Sie können die Export-Cmdlets auf einer funktionierenden Farm ausführen, die aktive Benutzerverbindungen hat. Die Exportskripts lesen nur die statische Farmkonfiguration und die Richtliniendaten.

#### Für den Import auf den XenApp 7.6-Server gilt Folgendes:

- Sie können Daten in XenApp 7.6-Bereitstellungen (und höhere unterstützte Versionen) importieren. Sie müssen einen XenApp 7.6-Controller und Studio installieren und eine Site erstellen, bevor Sie die aus der XenApp 6.x-Farm exportierten Daten importieren. VDAs sind zwar zum Importieren von Einstellungen nicht erforderlich, sie gestatten jedoch das Verfügbarmachen von Anwendungsdateitypen.
- Zum Ausführen der Import-Cmdlets müssen Sie XenApp-Administrator mit der Berechtigung zum Lesen und Erstellen von Objekten sein. Ein Volladministrator hat diese Berechtigungen. Sie müssen auch über die erforderlichen Windows-Berechtigungen zum Ausführen von PowerShell-Skripts verfügen. Schrittweise Anleitungen finden Sie unten.
- Während eines Imports dürfen keine anderen Benutzerverbindungen aktiv sein. Die Importskripts erstellen viele neue Objekte und wenn andere Benutzer gleichzeitig Änderungen an der Konfiguration vornehmen, können Unterbrechungen auftreten.

Sie können Daten exportieren und dann den Parameter "-Preview " für das Import-Cmdlet verwenden, um eine Vorschau des Imports zu sehen, ohne dass tatsächliche Importvorgänge stattfinden. In den Protokollen wird genau angegeben, was während eines tatsächlichen Importvorgangs passieren würde. Wenn Fehler auftreten, können Sie diese beheben, bevor Sie einen tatsächlichen Import durchführen.

#### Schrittweise Anleitungen: Exportieren von Daten

Ein Video des Exports ist [hier](#) verfügbar.

Führen Sie die folgenden Schritte aus, um Daten aus einem XenApp 6.x-Controller in XML-Dateien zu exportieren.

1. Laden Sie das Paket mit dem Migrationstool (XAMigration.zip) von der Citrix Downloadsite herunter. Speichern Sie es der Einfachheit halber in einer Netzwerkfreigabe, damit von der XenApp 6.x-Farm und der XenApp 7.6-Site darauf zugegriffen werden kann. Entzippen Sie XAMigration.zip in der Netzwerkfreigabe. Sie sollten nun zwei ZIP-Dateien haben: ReadIMA.zip und ImportFMA.zip.
2. Melden Sie sich am XenApp 6.x-Controller als XenApp-Administrator mit mindestens Lesezugriff und Windows-Berechtigung zum Ausführen von PowerShell-Skripts an.
3. Kopieren Sie die Datei ReadIMA.zip von der Netzwerkfreigabe auf den XenApp 6.x-Controller. Entzippen und extrahieren Sie ReadIMA.zip auf dem Controller in einen Ordner (z. B.: C:\XAMigration).
4. Öffnen Sie eine PowerShell-Konsole und legen Sie das aktuelle Verzeichnis als Skriptspeicherort fest. Beispiel:  
`cd C:\XAMigration`
5. Überprüfen Sie die Skriptausführungsrichtlinie durch Ausführen von Get-ExecutionPolicy.
6. Legen Sie die Skriptausführungsrichtlinie mindestens auf "RemoteSigned" fest, damit die Skripts ausgeführt werden können. Beispiel:  
`Set-ExecutionPolicy RemoteSigned`
7. Importieren Sie die Moduldefinitionsdateien "ExportPolicy.psd1" und "ExportXAFarm.psd1":  
`Import-Module .\ExportPolicy.psd1`  
`Import-Module .\ExportXAFarm.psd1`

#### Gut zu wissen:

- Wenn Sie nur Richtliniendaten exportieren möchten, können Sie nur die Moduldefinitionsdatei "ExportPolicy.psd1" importieren. Genauso gilt, wenn Sie nur Farmdaten exportieren möchten, importieren Sie nur "ExportXAFarm.psd1".
  - Beim Importieren der Moduldefinitionsdateien werden auch die erforderlichen PowerShell-Snap-Ins hinzugefügt.
  - Importieren Sie nicht die Skriptdateien mit der Erweiterung .psm1.
8. Führen Sie zum Exportieren von Richtliniendaten das Cmdlet Export-Policy aus.

| Parameter                      | Beschreibung                                                                                                                                                                                                                                                                                                                                                 |
|--------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| - XmlOutputFile "<string>.xml" | Name der XML-Ausgabedatei. Diese Datei enthält die exportierten Daten. Sie muss die Erweiterung .xml haben. Die Datei darf nicht vorhanden sein, aber wenn Sie den Pfad angeben, muss der übergeordnete Pfad vorhanden sein.<br><br>Standardwert: Keiner. Dieser Parameter ist erforderlich.                                                                 |
| -LogFile "<string>"            | Name der Protokolldatei. Eine Erweiterung ist optional. Die Datei wird erstellt, wenn sie nicht vorhanden ist. Wenn die Datei vorhanden ist und der Parameter "NoClobber" ebenfalls angegeben ist, wird ein Fehler generiert, sonst wird der Inhalt der Datei überschrieben.<br><br>Standardwert: siehe <a href="#">Protokollierung und Fehlerbehandlung</a> |
| -NoLog                         | Keine Protokollausgabe erstellen. Dieser Parameter überschreibt den Parameter "LogFile", wenn er ebenfalls angegeben ist.<br><br>Standardwert: False; Protokollausgabe wird erstellt                                                                                                                                                                         |
| -NoClobber                     | Vorhandene Protokolldatei, die im Parameter "LogFile" angegeben wurde, nicht überschreiben. Wenn die Protokolldatei nicht vorhanden ist, hat dieser Parameter keine Auswirkung.<br><br>Standardwert: False; eine vorhandene Protokolldatei wird überschrieben                                                                                                |
| -NoDetails                     | Keine ausführlichen Berichte zur Skriptausführung an die Konsole senden.<br><br>Standardwert: False; ausführliche Berichte werden an der Konsole gesendet                                                                                                                                                                                                    |
| - SuppressLogo                 | Meldung "XenApp 6.x to XenApp/XenDesktop 7.6 Migration Tool Version #yyyyMMdd-hhmm#" nicht auf Konsole drucken. Diese Meldung, in der die Skriptversion angegeben wird, kann bei der Problembehandlung hilfreich sein, daher empfiehlt Citrix, diesen Parameter wegzulassen.<br><br>Standardwert: False; die Meldung wird auf der Konsole gedruckt           |

Beispiel: Das folgende Cmdlet exportiert Richtlinieninformationen in die XML-Datei "MyPolicies.xml". Der Vorgang wird in der Datei "MyPolicies.log" protokolliert.

Export-Policy -XmlOutputFile ".\MyPolicies.XML" -LogFile ".\MyPolicies.Log"

9. Führen Sie zum Exportieren von Farmdaten das Cmdlet Export-XAFarm aus und geben Sie dabei eine Protokolldatei und eine XML-Datei an.

| Parameter                     | Beschreibung                                                                                                                                                                                                                                                                                                                                                 |
|-------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -XmlOutputFile "<string>.xml" | Name der XML-Ausgabedatei. Diese Datei enthält die exportierten Daten. Sie muss die Erweiterung .xml haben. Die Datei darf nicht vorhanden sein, aber wenn Sie den Pfad angeben, muss der übergeordnete Pfad vorhanden sein.<br><br>Standardwert: Keiner. Dieser Parameter ist erforderlich.                                                                 |
| -LogFile "<string>"           | Name der Protokolldatei. Eine Erweiterung ist optional. Die Datei wird erstellt, wenn sie nicht vorhanden ist. Wenn die Datei vorhanden ist und der Parameter "NoClobber" ebenfalls angegeben ist, wird ein Fehler generiert, sonst wird der Inhalt der Datei überschrieben.<br><br>Standardwert: siehe <a href="#">Protokollierung und Fehlerbehandlung</a> |
| -NoLog                        | Keine Protokollausgabe erstellen. Dieser Parameter überschreibt den Parameter "LogFile", wenn er ebenfalls angegeben ist.<br><br>Standardwert: False; Protokollausgabe wird erstellt                                                                                                                                                                         |
| -NoClobber                    | Vorhandene Protokolldatei, die im Parameter "LogFile" angegeben wurde, nicht überschreiben. Wenn die Protokolldatei nicht vorhanden ist, hat dieser Parameter keine Auswirkung.<br><br>Standardwert: False; eine vorhandene Protokolldatei wird überschrieben                                                                                                |
| -NoDetails                    | Keine ausführlichen Berichte zur Skriptausführung an die Konsole senden.                                                                                                                                                                                                                                                                                     |



| Parameter              | Beschreibung                                                                                                                                                                                                                                                                                                                                                                                                           |
|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -SuppressLogo          | Meldung "XenApp 6.x to XenApp/XenDesktop 7.6 Migration Tool Version #yyyyMMdd-hhmm#" nicht auf Konsole drucken. Diese Meldung, in der die Skriptversion angegeben wird, kann bei der Problembehandlung hilfreich sein, daher empfiehlt Citrix, diesen Parameter wegzulassen.<br><br>Standardwert: False; die Meldung wird auf der Konsole gedruckt                                                                     |
| -IgnoreAdmins          | Administratorinformationen nicht exportieren. Informationen zur Verwendung finden Sie unter <a href="#">Erweiterte Verwendung</a> .<br><br>Standardwert: False; Administratorinformationen werden exportiert                                                                                                                                                                                                           |
| -IgnoreApps            | Anwendungsinformationen nicht exportieren. Informationen zur Verwendung finden Sie unter <a href="#">Erweiterte Verwendung</a> .<br><br>Standardwert: False; Anwendungsinformationen werden exportiert                                                                                                                                                                                                                 |
| -IgnoreServers         | Serverinformationen nicht exportieren.<br><br>Standardwert: False; Serverinformationen werden exportiert                                                                                                                                                                                                                                                                                                               |
| -IgnoreZones           | Zoneninformationen nicht exportieren.<br><br>Standard: False; Zoneninformationen werden exportiert                                                                                                                                                                                                                                                                                                                     |
| -IgnoreOthers          | Daten wie Folgende nicht exportieren: Konfigurationsprotokollierung, Lastauswertungsprogramme, Lastausgleichsrichtlinien, Druckertreiber und Workergruppen.<br><br>Standardwert: False; andere Informationen werden exportiert<br><br>Hinweis: Der Schalter -IgnoreOthers ermöglicht das Durchführen eines Exports, wenn ein Fehler vorliegt, der keine Auswirkungen auf die exportierten oder importierten Daten hat. |
| -AppLimit<br><integer> | Anzahl der Anwendungen, die exportiert werden. Informationen zur Verwendung finden Sie unter <a href="#">Erweiterte Verwendung</a> .<br><br>Standardwert: Alle Anwendungen werden exportiert                                                                                                                                                                                                                           |
| -EmbedIconData         | Anwendungssymboldateien in die gleiche XML-Datei einbetten wie die anderen Objekte.<br><br>Standard: Symbole werden separat gespeichert. Weitere Informationen finden Sie unter <a href="#">Anforderungen, Vorbereitungen und Best Practices</a> .                                                                                                                                                                     |
| -SkipApps<br><integer> | Anzahl der Anwendungen, die übersprungen werden. Informationen zur Verwendung finden Sie unter <a href="#">Erweiterte Verwendung</a> .<br><br>Standardwert: Keine Anwendungen werden übersprungen                                                                                                                                                                                                                      |

Beispiel: Das folgende Cmdlet exportiert Farminformationen in die XML-Datei "MyFarm.xml". Der Vorgang wird in der Datei "MyFarm.log" protokolliert. Ein Ordner mit dem Namen "MyFarm-icons" wird zum Speichern der Datendateien für die Anwendungssymbole erstellt. Dieser Ordner ist am gleichen Speicherort wie "MyFarm.xml".

```
Export-XAFarm -XmlOutputFile ".\MyFarm.XML" -LogFile ".\MyFarm.Log"
```

Nachdem die Ausführung der Exportskripts abgeschlossen ist, enthalten die in den Befehlszeilen angegebenen XML-Dateien die Richtliniendaten und die XenApp-Farmdaten. Die Anwendungssymboldateien enthalten die Symboldateien und die Protokolldatei gibt an, was sich beim Export ereignet hat.

#### Schrittweise Anleitungen: Importieren von Daten

Ein Video des Imports ist [hier](#) verfügbar.

Sie können eine Importvorschau ausführen (indem Sie das Cmdlet "Import-Policy" oder "Import-XAFarm" mit dem Preview-Parameter ausführen) und die Protokolldateien überprüfen, bevor Sie einen tatsächlichen Import ausführen.

Führen Sie die folgenden Schritte aus, um Daten mit den beim Export erstellten XML-Dateien in eine XenApp 7.6-Site zu importieren.

1. Melden Sie sich als Administrator mit Lese- und Schreibrechten und Windows-Berechtigung zum Ausführen von PowerShell-Skripts am XenApp 7.6-Controller an.
2. Wenn Sie das Paket mit dem Migrationstool (XAMigration) noch nicht in der Netzwerkfreigabe entzippt haben, führen Sie den Vorgang nun aus. Kopieren Sie die Datei ImportFMA.zip von der Netzwerkfreigabe auf den XenApp 7.6-Controller. Entzippen und extrahieren Sie ImportFMA.zip auf dem Controller in einen Ordner (z. B.: C:\XAMigration).
3. Kopieren Sie die XML-Dateien (die während des Exports erstellten Ausgabedateien) vom XenApp 6.x-Controller in den Speicherort auf dem XenApp 7.6-Controller, wo Sie die Dateien aus ImportFMA.zip extrahiert haben.  
Wenn Sie die Anwendungssymboldaten beim Ausführen des Cmdlets "Export-XAFarm" nicht in die XML-Ausgabedatei eingebettet haben, kopieren Sie den Ordner mit den Symboldaten in den gleichen Speicherort auf dem XenApp 7.6-Controller, in den Sie die XML-Ausgabedateien kopiert haben und in dem sich die extrahierten Dateien aus ImportFMA.zip befinden.
4. Öffnen Sie eine PowerShell-Konsole und legen Sie das aktuelle Verzeichnis als Skriptspeicherort fest.  
cd C:\XAMigration
5. Überprüfen Sie die Skriptausführungsrichtlinie durch Ausführen von Get-ExecutionPolicy.
6. Legen Sie die Skriptausführungsrichtlinie mindestens auf "RemoteSigned" fest, damit die Skripts ausgeführt werden können. Beispiel:  
Set-ExecutionPolicy RemoteSigned
7. Importieren Sie die PowerShell-Moduldefinitionsdateien "ImportPolicy.psd1" und "ImportXAFarm.psd1":  
Import-Module .\ImportPolicy.psd1  
  
Import-Module .\ImportXAFarm.psd1

#### Gut zu wissen:

- Wenn Sie nur Richtliniendaten importieren möchten, können Sie nur die Moduldefinitionsdatei "ImportPolicy.psd1" importieren. Genauso gilt, wenn Sie nur Farmdaten importieren möchten, importieren Sie nur "ImportXAFarm.psd1".
  - Beim Importieren der Moduldefinitionsdateien werden auch die erforderlichen PowerShell-Snap-Ins hinzugefügt.
  - Importieren Sie nicht die Skriptdateien mit der Erweiterung .psm1.
8. Wenn Sie Richtliniendaten importieren, führen Sie das Cmdlet Import-Policy aus. Geben Sie dabei die XML-Datei mit den exportierten Richtliniendaten an.

| Parameter                       | Beschreibung                                                                                                                                                                                                                                                                 |
|---------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -XmlInputFile<br>"<string>.xml" | Name der XML-Eingabedatei. Diese Datei enthält Daten, die mit dem Cmdlet "Export-Policy" gesammelt wurden. Sie muss die Erweiterung .xml haben.<br><br>Standardwert: Keiner. Dieser Parameter ist erforderlich.                                                              |
| -XsdFile "<br><string>"         | Name der XSD-Datei. Mit dieser Datei überprüfen die Importskripts die Syntax der XML-Eingabedatei. Informationen zur Verwendung finden Sie unter <a href="#">Erweiterte Verwendung</a> .<br><br>Standardwert: PolicyData.XSD                                                 |
| -LogFile "<br><string>"         | Name der Protokolldatei. Wenn Sie Exportprotokolldateien auf diesen Server kopiert haben, sollten Sie einen anderen Namen für die Protokolldatei des Import-Cmdlets verwenden.<br><br>Standardwert: siehe <a href="#">Protokollierung und Fehlerbehandlung</a>               |
| -NoLog                          | Keine Protokollausgabe erstellen. Dieser Parameter überschreibt den Parameter "LogFile", wenn er ebenfalls angegeben ist.<br><br>Standardwert: False; Protokollausgabe wird erstellt                                                                                         |
| -NoClobber                      | Vorhandene Protokolldatei, die im Parameter "LogFile" angegeben wurde, nicht überschreiben. Wenn die Protokolldatei nicht vorhanden ist, hat dieser Parameter keine Auswirkung.<br><br>Standardwert: False; eine vorhandene Protokolldatei wird überschrieben                |
| -NoDetails                      | Keine ausführlichen Berichte zur Skriptausführung an die Konsole senden.<br><br>Standardwert: False; ausführliche Berichte werden an der Konsole gesendet                                                                                                                    |
| -<br>SuppressLogo               | Meldung "XenApp 6.x to XenApp/XenDesktop 7.6 Migration Tool Version #yyyyMMdd-hhmm#" nicht auf Konsole drucken. Diese Meldung, in der die Skriptversion angegeben wird, kann bei der Problembehandlung hilfreich sein, daher empfiehlt Citrix, diesen Parameter wegzulassen. |

| Parameter | Beschreibung                                                                                                                                                                                                                                                                                                                                                                                              |
|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -Preview  | <p>Führen Sie eine Importvorschau aus: Daten werden aus der XML-Eingabedatei gelesen, aber es werden keine Objekte in die Site importiert. In der Protokolldatei und Konsole wird protokolliert, was während der Importvorschau vorgegangen ist. Eine Vorschau zeigt Administratoren, was während eines echten Imports passieren würde.</p> <p>Standardwert: False; ein echter Import wird ausgeführt</p> |

Beispiel: Mit dem folgenden Cmdlet werden Richtliniendaten aus der XML-Datei "MyPolicies.xml" importiert. Der Vorgang wird in der Datei "MyPolicies.log" protokolliert.

Import-Policy -XmlInputFile ".\MyPolicies.XML" -LogFile ".\MyPolicies.Log"

9. Wenn Sie Anwendungen importieren, führen Sie das Cmdlet Import-XAFarm aus. Geben Sie dabei eine Protokolldatei und die XML-Datei mit den exportierten Farmdaten an.

| Parameter                     | Beschreibung                                                                                                                                                                                                                                                                                                                                                                                              |
|-------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -XmlInputFile "<string>.xml"  | <p>Name der XML-Eingabedatei. Diese Datei enthält Daten, die mit dem Cmdlet "Export-XAFarm" gesammelt wurden. Sie muss die Erweiterung .xml haben.</p> <p>Standardwert: Keiner. Dieser Parameter ist erforderlich.</p>                                                                                                                                                                                    |
| -XsdFile "<string>"           | <p>Name der XSD-Datei. Mit dieser Datei überprüfen die Importskripts die Syntax der XML-Eingabedatei. Informationen zur Verwendung finden Sie unter <a href="#">Erweiterte Verwendung</a>.</p> <p>Standardwert: XAFarmData.XSD</p>                                                                                                                                                                        |
| -LogFile "<string>"           | <p>Name der Protokolldatei. Wenn Sie Exportprotokolldateien auf diesen Server kopiert haben, sollten Sie einen anderen Namen für die Protokolldatei des Import-Cmdlets verwenden.</p> <p>Standardwert: siehe <a href="#">Protokollierung und Fehlerbehandlung</a></p>                                                                                                                                     |
| -NoLog                        | <p>Keine Protokollausgabe erstellen. Dieser Parameter überschreibt den Parameter "LogFile", wenn er ebenfalls angegeben ist.</p> <p>Standardwert: False; Protokollausgabe wird erstellt</p>                                                                                                                                                                                                               |
| -NoClobber                    | <p>Vorhandene Protokolldatei, die im Parameter "LogFile" angegeben wurde, nicht überschreiben. Wenn die Protokolldatei nicht vorhanden ist, hat dieser Parameter keine Auswirkung.</p> <p>Standardwert: False; eine vorhandene Protokolldatei wird überschrieben</p>                                                                                                                                      |
| -NoDetails                    | <p>Keine ausführlichen Berichte zur Skriptausführung an die Konsole senden.</p> <p>Standardwert: False; ausführliche Berichte werden an der Konsole gesendet</p>                                                                                                                                                                                                                                          |
| -SuppressLogo                 | <p>Meldung "XenApp 6.x to XenApp/XenDesktop 7.6 Migration Tool Version #yyyyMMdd-hhmm#" nicht auf Konsole drucken. Diese Meldung, in der die Skriptversion angegeben wird, kann bei der Problembehandlung hilfreich sein, daher empfiehlt Citrix, diesen Parameter wegzulassen.</p> <p>Standardwert: False; die Meldung wird auf der Konsole gedruckt</p>                                                 |
| -Preview                      | <p>Führen Sie eine Importvorschau aus: Daten werden aus der XML-Eingabedatei gelesen, aber es werden keine Objekte in die Site importiert. In der Protokolldatei und Konsole wird protokolliert, was während der Importvorschau vorgegangen ist. Eine Vorschau zeigt Administratoren, was während eines echten Imports passieren würde.</p> <p>Standardwert: False; ein echter Import wird ausgeführt</p> |
| -DeliveryGroupName "<string>" | <p>Bereitstellungsgruppenname für alle importierten Anwendungen. Informationen zur Verwendung finden Sie unter <a href="#">Erweiterte Verwendung</a>.</p>                                                                                                                                                                                                                                                 |

| Parameter                              | Standardwert: "<xenapp-farm-name> - Delivery Group"<br>Beschreibung                                                                                                                                                                                                       |
|----------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -MatchFolder "<string>"                | Import von Anwendungen in Ordnern, deren Namen mit der Zeichenfolge (String) übereinstimmen. Informationen zur Verwendung finden Sie unter <a href="#">Erweiterte Verwendung</a> .<br><br>Standardwert: Keine Übereinstimmung                                             |
| -NotMatchFolder "<br><string>"         | Import von Anwendungen in Ordnern, deren Namen mit der Zeichenfolge (String) nicht übereinstimmen. Informationen zur Verwendung finden Sie unter <a href="#">Erweiterte Verwendung</a> .<br><br>Standardwert: Keine Übereinstimmung                                       |
| -MatchServer "<string>"                | Import von Anwendungen auf Servern, deren Namen mit der Zeichenfolge (String) übereinstimmen. Informationen zur Verwendung finden Sie unter <a href="#">Erweiterte Verwendung</a> .                                                                                       |
| -NotMatchServer "<br><string>"         | Import von Anwendungen auf Servern, deren Namen nicht mit der Zeichenfolge (String) übereinstimmen. Informationen zur Verwendung finden Sie unter <a href="#">Erweiterte Verwendung</a> .<br><br>Standardwert: Keine Übereinstimmung                                      |
| -MatchWorkerGroup "<br><string>"       | Import von Anwendungen, die für Workergruppen veröffentlicht wurden und deren Namen mit der Zeichenfolge (String) übereinstimmen. Informationen zur Verwendung finden Sie unter <a href="#">Erweiterte Verwendung</a> .<br><br>Standardwert: Keine Übereinstimmung        |
| -<br>NotMatchWorkerGroup<br>"<string>" | Import von Anwendungen, die für Workergruppen veröffentlicht wurden und deren Namen nicht mit der Zeichenfolge (String) übereinstimmen. Informationen zur Verwendung finden Sie unter <a href="#">Erweiterte Verwendung</a> .<br><br>Standardwert: Keine Übereinstimmung  |
| -MatchAccount "<br><string>"           | Import von Anwendungen, die für Benutzerkonten veröffentlicht wurden und deren Namen mit der Zeichenfolge (String) übereinstimmen. Informationen zur Verwendung finden Sie unter <a href="#">Erweiterte Verwendung</a> .<br><br>Standardwert: Keine Übereinstimmung       |
| -NotMatchAccount "<br><string>"        | Import von Anwendungen, die für Benutzerkonten veröffentlicht wurden und deren Namen nicht mit der Zeichenfolge (String) übereinstimmen. Informationen zur Verwendung finden Sie unter <a href="#">Erweiterte Verwendung</a> .<br><br>Standardwert: Keine Übereinstimmung |
| -IncludeStreamedApps                   | Import von Anwendungen des Typs "StreamedToClientOrServerInstalled". (Keine anderen gestreamten Anwendungen werden importiert.)<br><br>Standard: Gestreamte Anwendungen werden nicht importiert                                                                           |
| -IncludeDisabledApps                   | Import von Anwendungen, die als deaktiviert markiert sind.<br><br>Standard: Deaktivierte Anwendungen werden nicht importiert                                                                                                                                              |

Beispiel: Das folgende Cmdlet importiert Anwendungen aus der XML-Datei "MyFarm.xml". Der Vorgang wird in der Datei "MyFarm.log" protokolliert.

Import-XAFarm -XmlInputFile ".\MyFarm.XML" -LogFile ".\MyFarm.Log"

10. Führen Sie nach dem Abschluss des Imports die nach der Migration erforderlichen Aufgaben durch.

#### Aufgaben nach der Migration

Nach dem erfolgreichen Import von XenApp 6.x-Richtlinien und Farmeinstellungen in eine XenApp 7.6-Site stellen Sie mit den folgenden Richtlinien sicher, dass die Daten richtig importiert wurden.

- **Richtlinien und Richtlinieneinstellungen**

Das Importieren von Richtlinien ist im Prinzip ein Kopiervorgang mit Ausnahme von veralteten Einstellungen und Richtlinien, die nicht importiert werden. Mit der Prüfung nach der Migration werden die beiden Seiten verglichen.

1. In der Protokolldatei werden alle importierten und ignorierten Richtlinien und Einstellungen aufgeführt. Überprüfen Sie zuerst die Protokolldatei und identifizieren Sie die Einstellungen und Richtlinien, die nicht importiert wurden.
2. Vergleichen Sie die XenApp 6.x-Richtlinien mit den nach XenApp 7.6 importierten Richtlinien. Die Werte der Einstellungen sollten gleich bleiben (außer bei veralteten Richtlinieneinstellungen, siehe nächster Schritt).
  - Bei einer kleinen Anzahl von Richtlinien können Sie einen visuellen Vergleich der Richtlinien im XenApp 6.x AppCenter und in XenApp 7.6 Studio durchführen.
  - Bei einer großen Anzahl von Richtlinien ist ein visueller Vergleich u. U. nicht möglich. Verwenden Sie in solchen Fällen das Export-Cmdlet (Export-Policy), um die XenApp 7.6-Richtlinien in eine andere XML-Datei zu exportieren. Vergleichen Sie dann mit einem Textvergleichsprogramm (z. B. Windiff) die Daten der Datei mit den Daten in der XML-Datei, die zum Richtlinienexport aus XenApp 6.x verwendet wurde.
3. Der Abschnitt **Nicht importierte Richtlinieneinstellungen** enthält Informationen dazu, was sich beim Import geändert haben könnte. Wenn eine XenApp 6.x-Richtlinie nur veralteten Einstellungen enthält, wird die gesamte Richtlinie nicht importiert. Beispiel: Wenn eine XenApp 6.x-Richtlinie nur HMR-Testeinstellungen enthält, wird die Richtlinie vollständig ignoriert, da es keine entsprechende Einstellung in XenApp 7.6 gibt. Einige XenApp 6.x-Richtlinieneinstellungen werden nicht mehr unterstützt, aber vergleichbare Funktionen wurden in XenApp 7.6 implementiert. In XenApp 7.6 können Sie beispielsweise einen Neustartzeitplan für Serverbetriebssystemmaschinen konfigurieren, indem Sie eine Bereitstellungsgruppe bearbeiten. Diese Funktionalität wurde zuvor über Richtlinieneinstellungen implementiert.
4. Lesen Sie sich noch einmal durch, wie Filter in einer XenApp 7.6-Site im Gegensatz zu XenApp 6.x angewendet werden. Durch die wesentlichen Unterschiede zwischen der XenApp 6.x-Farm und der XenApp 7.6-Site kann sich die Wirkung von Filtern ändern.

#### • Filter

Überprüfen Sie sorgfältig die Filter für die einzelnen Richtlinien. Damit sie in XenApp 7.6 weiterhin genauso funktionieren wie in XenApp 6.x, sind u. U. Änderungen erforderlich.

| Filter               | Überlegungen                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Zugriffssteuerung    | Die Zugriffssteuerung sollte die gleichen Werte wie die ursprünglichen XenApp 6.x-Filter enthalten und ohne Änderungen funktionieren.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| Citrix CloudBridge   | Ein einfacher Boolescher Wert, der ohne Änderungen funktionieren sollte.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| Client-IP-Adresse    | Listet Client-IP-Adressbereiche auf. Jeder Bereich ist entweder zugelassen oder verweigert. Das Importskript behält die Werte bei, aber Änderungen können erforderlich sein, wenn sich andere Clients mit den XenApp 7.6-VDA-Maschinen verbinden.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Clientname           | Ähnlich wie beim Client-IP-Adressenfilter behält das Importskript die Werte bei. Es können jedoch Änderungen erforderlich sein, wenn sich andere Clients mit den XenApp 7.6-VDA-Maschinen verbinden.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| Organisationseinheit | <p>Die Werte werden beibehalten, wenn die Organisationseinheiten beim Import aufgelöst werden können. Überprüfen Sie diesen Filter sorgfältig, besonders wenn die XenApp 6.x- und XenApp 7.6-Maschinen in unterschiedlichen Domänen sind. Wenn Sie die Filterwerte nicht richtig konfigurieren, wird die Richtlinie möglicherweise auf einen falschen Satz Organisationseinheiten angewendet.</p> <p>Die Organisationseinheiten werden nur durch Namen dargestellt, daher ist es möglich, dass eine Organisationseinheit zu einer Organisationseinheit aufgelöst wird, die andere Mitglieder enthält als die Organisationseinheit in der XenApp 6.x-Domäne. Selbst wenn einige Werte des Organisationseinheitsfilters beibehalten werden, prüfen Sie die Werte sorgfältig.</p> |
| Benutzer oder Gruppe | <p>Die Werte werden beibehalten, wenn die Konten beim Import aufgelöst werden können.</p> <p>Ähnlich wie Organisationseinheiten werden die Konten nur nach Namen aufgelöst. Wenn die XenApp 7.6-Site eine Domäne mit den gleichen Domänen- und Benutzernamen enthält, wobei es sich aber tatsächlich um zwei verschiedene Domänen und Benutzer handelt, entsprechen die aufgelösten Konten möglicherweise nicht den Domänenbenutzern in XenApp 6.x. Wenn Sie die Filterwerte nicht richtig überprüfen und ändern, kann es zur falschen Anwendung von Richtlinien kommen.</p>                                                                                                                                                                                                   |
| Workergruppe         | <p>Workergruppen werden in XenApp 7.6 nicht unterstützt. Verwenden Sie die Bereitstellungsgruppe, den Bereitstellungsgruppentyp und die Tagfilter, die in XenApp 7.6 unterstützt werden (nicht in XenApp 6.x).</p> <ul style="list-style-type: none"> <li>• Bereitstellungsgruppe: Ermöglicht die Anwendung von Richtlinien basierend auf Bereitstellungsgruppen. Jeder Filtereintrag gibt eine Bereitstellungsgruppe an und kann zugelassen oder verweigert werden.</li> <li>• Bereitstellungsgruppentyp: Ermöglicht die Anwendung von Richtlinien basierend auf Bereitstellungsgruppentypen. Jeder Filter</li> </ul>                                                                                                                                                         |

|               |                                                                                                                                                                                                                                                                                                                              |
|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Filter</b> | <p>gibt einen Bereitstellungsgruppentyp an und kann zugelassen oder verweigert werden.</p> <p><b>Überlegungen</b></p> <ul style="list-style-type: none"> <li>• Tag: Gibt die Richtlinienanwendung basierend auf Tags an, die für VDA-Maschinen erstellt wurden. Jedes Tag kann zugelassen oder verweigert werden.</li> </ul> |
|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Filter, die Domänenbenutzeränderungen umfassen, müssen besonders sorgfältig überprüft werden, wenn die XenApp 6.x-Farm in einer anderen Domäne ist als die XenApp 7.6-Site. Da das Importskript nur die Zeichenfolgen von Domänen- und Benutzernamen verwendet, um Benutzer in der neuen Domäne aufzulösen, werden einige Konten möglicherweise aufgelöst und andere nicht. Obwohl nicht sehr wahrscheinlich ist, dass verschiedene Domänen und Benutzer den gleichen Namen haben, sollten Sie unbedingt die Filter sorgfältig überprüfen, um sicherzustellen, dass sie korrekte Werte enthalten.

## • Anwendungen

Die Skripts zum Import von Anwendungen importieren nicht nur Anwendungen, sondern sie erstellen auch Objekte, z. B. Bereitstellungsgruppen. Wenn der Anwendungsimport mehrere Durchläufe umfasst, können sich die Originalhierarchien der Anwendungsordner erheblich ändern.

1. Lesen Sie als Erstes die Migrationsprotokolldateien, die Informationen dazu enthalten, welche Anwendungen importiert oder ignoriert wurden und welche Cmdlets zum Erstellen der Anwendungen verwendet wurden.
2. Für jede Anwendung gilt Folgendes:
  - Sehen Sie sich das Protokoll an und prüfen Sie, ob die grundlegenden Eigenschaften beim Importieren beibehalten wurden. Bestimmen Sie mit den Informationen unter [Zuordnung von Anwendungseigenschaften](#), welche Eigenschaften ohne Änderungen importiert, nicht importiert oder mit den XenApp 6.x-Anwendungsdaten initialisiert wurden.
  - Überprüfen Sie die Benutzerliste. Das Importskript importiert automatisch die explizite Liste der Benutzer in die Liste "Sichtbarkeit beschränken" der Anwendung in XenApp 7.6. Stellen Sie sicher, dass die Liste unverändert ist.
3. Anwendungsserver werden nicht importiert. Dies bedeutet, dass auf keine der importierten Anwendungen zugegriffen werden kann. Den Bereitstellungsgruppen, die diese Anwendungen enthalten, müssen Maschinenkataloge mit den Maschinen zugewiesen werden, auf denen die ausführbaren Images der veröffentlichten Anwendungen sind. Für jede Anwendung gilt Folgendes:
  - Stellen Sie sicher, dass der Name der ausführbaren Datei und das Arbeitsverzeichnis auf eine ausführbare Datei verweisen, die auf den Maschinen vorhanden ist, die der Bereitstellungsgruppe (über die Maschinenkataloge) zugewiesen sind.
  - Überprüfen Sie einen Befehlszeilenparameter (dies kann ein beliebiges Objekt sein, z. B. Dateiname, Umgebungsvariable oder der Name einer ausführbaren Datei). Stellen Sie sicher, dass der Parameter für alle Maschinen in den Maschinenkatalogen, die der Bereitstellungsgruppe zugewiesen sind, gültig ist.

## • Protokolldateien

Die Protokolldateien sind die wichtigsten Referenzressourcen beim Import und Export. Aus diesem Grund werden bestehende Protokolldateien standardmäßig nicht überschrieben und Standardprotokolldateien haben eindeutige Namen.

Im Abschnitt "Protokollierung und Fehlerbehandlung" wurde bereits erwähnt, dass die Ausgabe und die Protokolldatei, die Sie erhalten, wenn Sie die verfügbaren zusätzlichen Protokollierungsfunktionen für die PowerShell-Cmdlets "Start-Transcript" und "Stop-Transcript" verwenden (sie protokollieren alles, was in die Konsole eingegeben und gedruckt wird), eine vollständige Referenz der Import- und Exportaktivitäten bieten.

Mit den Zeitstempeln in den Protokolldateien können Sie bestimmte Probleme diagnostizieren. Wenn beispielsweise ein Export oder Import sehr lange gedauert hat, können Sie feststellen, ob eine fehlerhafte Datenbankverbindung oder das Auflösen von Benutzerkonten viel Zeit in Anspruch genommen haben.

Aus den in den Protokolldateien aufgezeichneten Befehlen lässt sich auch ermitteln, wie manche Objekte gelesen oder erstellt werden. Beispielsweise werden zum Erstellen einer Bereitstellungsgruppe mehrere Befehle ausgeführt, und zwar nicht nur, um das Bereitstellungsgruppenobjekt selbst zu erstellen, sondern auch andere Objekte, wie die Zugriffsrichtlinienregeln, mit denen Anwendungsobjekte Bereitstellungsgruppen zugewiesen werden.

Mit der Protokolldatei kann auch ein fehlgeschlagener Export oder Import diagnostiziert werden. Normalerweise ist in den letzten Zeilen der Protokolldatei angegeben, was den Fehler verursacht hat. Die Fehlermeldung ist ebenfalls in der Protokolldatei gespeichert. Mit der Protokolldatei und der XML-Datei zusammen können Sie bestimmen, welches Objekt an dem Fehler beteiligt war.

Nach der Überprüfung und dem Test der Migration haben Sie folgende Möglichkeiten:

1. Upgrade der XenApp 6.5-Workerserver auf aktuelle Virtual Delivery Agents (VDAs) durch Ausführen des Installers für 7.6 auf den Servern. Der Installer entfernt die XenApp 6.5-Software und installiert dann automatisch einen aktuellen VDA. Anweisungen finden Sie unter [Upgrade eines XenApp 6.5-Workers auf einen neuen VDA für Windows-Serverbetriebssysteme](#).  
Bei XenApp 6.0-Workerservern müssen Sie die XenApp 6.0-Software manuell vom Server deinstallieren. Danach können Sie mit dem Installer für 7.6 den aktuellen VDA installieren. Sie können mit dem Installer für 7.6 nicht automatisch die XenApp 6.0-Software entfernen.
2. Erstellen von Maschinenkatalogen (oder Bearbeiten von vorhandenen Katalogen) für die aktualisierten Worker in der neuen XenApp-Site mit Studio.
3. Hinzufügen der aktualisierten Maschinen aus dem Maschinenkatalog zu den Bereitstellungsgruppen, die die auf den VDAs für Windows-Serverbetriebssysteme installierten Anwendungen enthalten.

## Erweiterte Verwendung

Standardmäßig exportiert das Cmdlet "Export-Policy" alle Richtliniendaten in eine XML-Datei. Parallel dazu exportiert das Cmdlet "Export-XAFarm" alle Farmdaten in eine XML-Datei. Sie können mit Befehlszeilenparametern genauer steuern, was importiert und exportiert wird.

- **Teilweiser Export von Anwendungen:** Wenn Sie eine große Anzahl von Anwendungen haben und steuern möchten, wie viele in die XML-Datei exportiert werden, verwenden Sie die folgenden Parameter:
  - **AppLimit:** Gibt die Anzahl der zu exportierenden Anwendungen an.
  - **SkipApps:** Gibt die Anzahl der zu überspringenden Anwendungen an, bevor Anwendungen exportiert werden.
 Sie können beide Parameter verwenden, um große Mengen von Anwendungen in praktischen Segmenten zu exportieren. Beispiel: Wenn Sie das erste Mal "Export-XAFarm" ausführen, möchten Sie nur die ersten 200 Anwendungen exportieren und geben daher den Wert im Parameter "AppLimit" an.  
 Export-XAFarm -XmlOutputFile "Apps1-200.xml" -AppLimit "200"  
 Beim nächsten Mal, wenn Sie "Export-XAFarm" ausführen, sollen die nächsten 100 Anwendungen exportiert werden. Sie verwenden den Parameter "SkipApps", um die bereits exportierten Anwendungen (die ersten 200) zu ignorieren, und exportieren mit dem Parameter "AppLimit" die nächsten 100 Anwendungen.  
 Export-XAFarm -XmlOutputFile "Apps201-300.xml" -AppLimit "100" -SkipApps "200"
- **Bestimmte Objekte nicht exportieren:** Einige Objekte brauchen nicht exportiert zu werden und können ignoriert werden. Dazu zählen besonders Objekte, die nicht importiert werden. Weitere Informationen hierzu finden Sie unter [Nicht importierte Richtlinienereinstellungen](#) und [Zuordnung von Anwendungseigenschaften](#). Mit den folgenden Parametern können Sie den Export unnötiger Objekte verhindern:
  - **IgnoreAdmins:** Administratorobjekte werden nicht exportiert
  - **IgnoreServers:** Serverobjekte werden nicht exportiert
  - **IgnoreZones:** Zonenobjekte werden nicht exportiert
  - **IgnoreOthers:** Konfigurationsprotokollierungs-, Lastauswertungsprogramm-, Lastausgleichsrichtlinien-, Druckertreiber- und Workergruppenobjekte werden nicht exportiert
  - **IgnoreApps:** Anwendungen werden nicht exportiert. Hiermit können Sie andere Daten in eine XML-Ausgabedatei exportieren und dann den Export erneut ausführen, um die Anwendungen in eine andere XML-Ausgabedatei zu exportieren.
 Sie können mit diesen Parametern auch Probleme umgehen, die zum Fehlschlagen des Exports führen können. Wenn Sie beispielsweise einen fehlerhaften Server in einer Zone haben, schlägt der Export der Zone möglicherweise fehl. Wenn Sie den Parameter "IgnoreZones" verwenden, wird der Exportvorgang mit anderen Objekten fortgesetzt.
- **Bereitstellungsgruppennamen:** Wenn nicht alle Anwendungen in einer Bereitstellungsgruppe platziert werden sollen (z. B. weil verschiedene Benutzergruppen auf sie zugreifen und sie auf verschiedenen Servern veröffentlicht werden), führen Sie "Import-XAFarm" mehrmals aus und geben Sie dabei jedes Mal unterschiedliche Anwendungen und eine andere Bereitstellungsgruppe an. Sie können Anwendungen nach der Migration zwar mit PowerShell-Cmdlets von einer Bereitstellungsgruppe in eine andere verschieben, jedoch kann das Verschieben von Anwendungen durch selektives Importieren in eindeutige Bereitstellungsgruppen reduziert oder eliminiert werden.
  1. Verwenden Sie den Parameter "DeliveryGroupName" mit dem Cmdlet "Import-XAFarm". Das Skript erstellt die angegebene Bereitstellungsgruppe, wenn sie nicht vorhanden ist.
  2. Verwenden Sie die folgenden Parameter mit regulären Ausdrücken, um die Anwendungen, die in die Bereitstellungsgruppe importiert werden sollen, basierend auf Ordner, Workergruppe, Benutzerkontonamen und/oder Servernamen zu filtern. Es empfiehlt sich, den regulären Ausdruck in einzelne oder doppelte Anführungszeichen zu setzen. Weitere Informationen über reguläre Ausdrücke finden Sie unter [http://msdn.microsoft.com/en-us/library/hs600312\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/hs600312(v=vs.110).aspx).
    - **MatchWorkerGroup** und **NotMatchWorkerGroup:** Zum Beispiel bei Anwendungen, die auf Workergruppen veröffentlicht wurden, importiert das folgende Cmdlet Anwendungen in der Workergruppe "Productivity Apps" in eine XenApp 7.6-Bereitstellungsgruppe mit demselben Namen:  
 Import-XAFarm -XmlInputFile XAFarm.xml -LogFile XAFarmImport.log -MatchWorkerGroup 'Productivity Apps' -DeliveryGroupName 'Productivity Apps'
    - **MatchFolder** und **NotMatchFolder:** Zum Beispiel bei Anwendungen, die in Anwendungsordnern organisiert sind, importiert das folgende Cmdlet Anwendungen im Ordner "Productivity Apps" in eine XenApp 7.6-Bereitstellungsgruppe mit dem gleichen Namen.  
 Import-XAFarm -XmlInputFile XAFarm.xml -LogFile XAFarmImport.log -MatchFolder 'Productivity Apps' -DeliveryGroupName 'Productivity Apps'  
 Beispielsweise importiert das folgende Cmdlet Anwendungen in Ordnern, deren Name "MS Office Apps" enthält, in die Standardbereitstellungsgruppe.  
 Import-XAFarm -XmlInputFile .\TheFarmApps.XML -MatchFolder ".\*MS Office Apps.\*"
    - **MatchAccount** und **NotMatchAccount:** Zum Beispiel bei Anwendungen, die für Active Directory-Benutzer oder -Benutzergruppen veröffentlicht wurden, importiert das folgende Cmdlet Anwendungen, die für die Benutzergruppe "Finance Group" veröffentlicht wurden, in eine XenApp 7.6-Bereitstellungsgruppe mit dem Namen "Finance"  
 Import-XAFarm -XmlInputFile XAFarm.xml -LogFile XAFarmImport.log -MatchAccount 'DOMAIN\Finance Group' -DeliveryGroupName 'Finance'
    - **MatchServer** und **NotMatchServer:** Zum Beispiel bei Anwendungen, die auf Servern organisiert sind, importiert das folgende Cmdlet Anwendungen von Servern, deren Name nicht "Current" ist, in eine XenApp-Bereitstellungsgruppe mit dem Namen "Legacy."  
 Import-XAFarm -XmlInputFile XAFarm.xml -LogFile XAFarmImport.log -NotMatchServer 'Current' -DeliveryGroupName 'Legacy'
- **Anpassung:** PowerShell-Programmierer können eigene Tools erstellen. Sie können z. B. das Exportskript als Bestandstool verwenden und damit die Änderungen in einer XenApp 6.x-Farm verfolgen. Sie können auch die XSD-Dateien ändern oder Ihre eigenen XSD-Dateien erstellen, um zusätzliche Daten oder Daten in unterschiedlichen Formaten in den XML-Dateien zu speichern. Sie können eine nicht standardmäßige XSD-Datei mit jedem der Import-Cmdlets angeben.  
 Hinweis: Obwohl Sie Skriptdateien für bestimmte oder höhere Migrationsanforderungen ändern können, ist der Support auf unveränderte Skripts beschränkt. Der technische Support von Citrix empfiehlt, die Skripts in den Originalzustand zurückzusetzen, um bei Bedarf erwartetes Verhalten ermitteln und Support bereitstellen zu können.

Problembehandlung



- Wenn Sie PowerShell Version 2.0 verwenden und das PowerShell-Anbieter-Snap-In für Citrix Gruppenrichtlinien oder das Citrix Common Commands Snap-In mit dem Cmdlet Add-PSSnapIn hinzugefügt haben, wird möglicherweise die folgende Fehlermeldung angezeigt, wenn Sie Cmdlets zum Exportieren oder Importieren ausführen: "Objekt verweist nicht auf eine Instanz eines Objekts". Dieser Fehler wirkt sich nicht auf die Skriptausführung aus und kann bedenkenlos ignoriert werden.
- Vermeiden Sie es, das PowerShell-Anbieter-Snap-In für Citrix Gruppenrichtlinien in der gleichen Konsolensitzung hinzuzufügen oder zu entfernen, in der Sie die Export- und Importskriptmodule verwenden, da diese Skriptmodule das Snap-In automatisch hinzufügen. Wenn Sie das Snap-In separat hinzufügen oder entfernen, wird u. U. einer der folgenden Fehler angezeigt:
  - "Ein Laufwerk mit dem Namen 'LocalGpo' ist bereits vorhanden." Dieser Fehler tritt auf, wenn das Snap-In zweimal hinzugefügt wird. Beim Laden versucht das Snap-In, das Laufwerk "LocalGpo" bereitzustellen und meldet dann den Fehler.
  - "Es wurde kein Parameter gefunden, der dem Parameternamen 'Controller' entspricht." Dieser Fehler wird angezeigt, wenn das Snap-In nicht hinzugefügt wurde und das Skript versucht, das Laufwerk bereitzustellen. Das Skript erkennt nicht, dass das Snap-In entfernt wurde. Schließen Sie die Konsole und starten Sie eine neue Sitzung. Importieren Sie in der neuen Sitzung die Skriptmodule. Fügen Sie das Snap-In nicht separat hinzu und entfernen Sie es auch nicht separat.
- Wenn Sie beim Importieren der Module mit der rechten Maustaste auf eine PSD1-Datei klicken und dann Öffnen oder Mit PowerShell öffnen wählen, wird das Fenster der PowerShell-Konsole in schneller Abfolge geöffnet und geschlossen, bis Sie den Vorgang anhalten. Sie vermeiden diesen Fehler, indem Sie den vollständigen Namen des PowerShell-Skriptmoduls direkt im PowerShell-Konsolenfenster eingeben (z. B. Import-Module .\ExportPolicy.psd1).
- Wenn beim Ausführen eines Exports oder Imports ein Berechtigungsfehler angezeigt wird, stellen Sie sicher, dass Sie ein XenApp-Administrator mit der Berechtigung zum Lesen von Objekten (beim Export) oder zum Lesen und Erstellen von Objekten (beim Import) sind. Sie müssen auch über die erforderlichen Berechtigungen zum Ausführen von Windows-PowerShell-Skripts verfügen.
- Wenn ein Export fehlschlägt, prüfen Sie, ob die XenApp 6.x-Farm funktionsfähig ist, indem Sie die Dienstprogramme DSMaint und DSCheck auf dem XenApp 6.x Controller-Server ausführen.
- Wenn Sie mit den Import-Cmdlets eine Importvorschau ausführen und später bei der tatsächlichen Migration nichts importiert wird, prüfen Sie, ob Sie den Parameter "Preview" aus den Import-Cmdlets entfernt haben.

## Nicht importierte Richtlinieneinstellungen

Die folgenden Computer- und Benutzerrichtlinieneinstellungen werden nicht importiert, da sie nicht mehr unterstützt werden. Ungefilterte Richtlinien werden nie importiert. Die Features und Komponenten, die diese Einstellungen unterstützen, wurden entweder durch neue Technologien/Komponenten ersetzt oder sind aufgrund von Änderungen an Architektur oder Plattform nicht relevant.

### Nicht importierte Computerrichtlinieneinstellungen

- Verbindungszugriffssteuerungen
- CPU-Managementserverstufe
- DNS-Adressauflösung
- Farmname
- Vollständiges Symbolcaching
- Systemüberwachung, Systemüberwachungstests
- Hostname des Lizenzservers, den Lizenzserverport
- Limit für Benutzersitzungen, Limits für Administratorsitzungen
- Lastauswertungsprogrammname
- Protokollieren von Anmeldelimitereignissen
- Maximaler Prozentsatz der Server mit Anmeldesteuerung
- Speicheroptimierung, Speicheroptimierung - Anwendungsausschlussliste, Speicheroptimierung - Intervall, Speicheroptimierung - Tag des Monats, Speicheroptimierung - Wochentag, Speicheroptimierung - Zeit
- Offlineanwendungsklient vertrauen, Ereignisprotokollierung für Offlineanwendungen, Offlineanwendungslizenzzeitraum, Offlineanwendungsbenutzer
- Eingabeaufforderung für Kennwort
- Benutzerdefinierte Neustartwarnung, Text für benutzerdefinierte Neustartwarnung, Anmeldungen vor Neustart deaktivieren (Zeit), Neustarthäufigkeit, Willkürliches Neustartintervall, Neustartbeginn, Neustartzeit, Neustartwarnungsintervall, Neustartwarnung - Startzeit, Neustartwarnung an Benutzer, Geplante Neustarts
- Spiegeln von Sitzungen \*
- XML-Anfragen vertrauen (Konfiguration in StoreFront)
- Virtuelle IP - Adapteradressenfilterung, Virtuelle IP - Liste kompatibler Programme, Virtuelle IP - Erweiterte Kompatibilität, Virtuelle IP - Adapteradressenprogrammliste
- Arbeitslastname
- XenApp-Produktedition, XenApp-Produktmodell
- Port für XML-Dienst

\* Ersetzt durch Windows-Remoteunterstützung

### Nicht importierte Benutzerrichtlinieneinstellungen

- Client-COM-Ports automatisch verbinden, Client-LPT-Ports automatisch verbinden
- Client-COM-Portumleitung, Client-LPT-Portumleitung



- Clientdruckernamen
- Limit für gleichzeitige Anmeldungen
- Eingaben in gespiegelten Verbindungen \*
- Trennentimerintervall - Fortbestehen, Beendentimerintervall - Fortbestehen
- Spiegelungsversuche protokollieren \*
- Benutzer bei ausstehenden Spiegelungsverbindungen benachrichtigen \*
- PreLaunch-Trennentimerintervall, PreLaunch-Beendentimerintervall
- Sitzungspriorität
- Single Sign-On, Zentraler Speicher für Single Sign-On
- Benutzer, die andere Benutzer spiegeln können; Benutzer, die andere Benutzer nicht spiegeln können \*

\* Ersetzt durch Windows-Remoteunterstützung

## Nicht importierte Anwendungstypen

Die folgenden Anwendungstypen werden nicht importiert:

- Serverdesktops
- Inhalt
- Gestreamte Anwendungen (App-V ist die neue Methode für das Streaming von Anwendungen)

## Zuordnung von Anwendungseigenschaften

Das Importskript für Farmdaten importiert nur Anwendungen. Die folgenden Anwendungseigenschaften werden ohne Änderungen importiert.

| IMA-Eigenschaft       | FMA-Eigenschaft          |
|-----------------------|--------------------------|
| AddToClientDesktop    | ShortcutAddedToDesktop   |
| AddToClientStartMenu  | ShortcutAddedToStartMenu |
| ClientFolder          | ClientFolder             |
| CommandLineExecutable | CommandLineExecutable    |
| CpuPriorityLevel      | CpuPriorityLevel         |
| Beschreibung          | Beschreibung             |
| DisplayName           | PublishedName            |
| Aktiviert             | Aktiviert                |
| StartMenuFolder       | StartMenuFolder          |
| WaitOnPrinterCreation | WaitForPrinterCreation   |
| WorkingDirectory      | WorkingDirectory         |
| FolderPath            | AdminFolderName          |

Hinweis: IMA und FMA haben unterschiedliche Beschränkungen bei der Länge der Ordernamen. In IMA ist die Länge der Ordernamen auf 256 Zeichen beschränkt, in FMA auf 64 Zeichen. Anwendungen, deren Ordnerpfad einen Ordernamen mit mehr als 64 Zeichen enthält, werden beim Import übersprungen. Das Limit gilt nur für die Ordernamen im Ordnerpfad. Der gesamte Ordnerpfad kann länger sein als die aufgeführten Limits. Damit Anwendungen beim Importieren nicht übersprungen werden, empfiehlt es sich, die Länge der Anwendungsordernamen zu prüfen und bei Bedarf vor dem Export zu kürzen. Die folgenden Anwendungseigenschaften sind standardmäßig initialisiert oder nicht initialisiert oder auf die in den XenApp 6.x-Daten bereitgestellten Werte festgelegt:

| FMA-Eigenschaft | Wert |
|-----------------|------|
|-----------------|------|

| Name<br>IMA-Eigenschaft       | Wert                                                                                                                                                                               |
|-------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                               | Initialisiert auf den vollständigen Pfadnamen, der die IMA-Eigenschaften "FolderPath" und "DisplayName" enthält, aber die voranstehende Zeichenfolge "Applications\" wurde gekürzt |
| ApplicationType               | HostedOnDesktop                                                                                                                                                                    |
| CommandLineArguments          | Initialisiert mit den XenApp 6.x-Befehlszeilenargumenten                                                                                                                           |
| IconFromClient                | Nicht initialisiert, Standardwert = false                                                                                                                                          |
| IconUid                       | Initialisiert auf ein Symbolobjekt, das mit XenApp 6.x-Symboldaten erstellt wurde                                                                                                  |
| SecureCmdLineArgumentsEnabled | Nicht initialisiert, Standardwert = true                                                                                                                                           |
| UserFilterEnabled             | Nicht initialisiert, Standardwert = false                                                                                                                                          |
| UUID                          | Schreibgeschützt, vom Controller zugewiesen                                                                                                                                        |
| Sichtbar                      | Nicht initialisiert, Standardwert = true                                                                                                                                           |

Die folgenden Anwendungseigenschaften werden teilweise migriert:

| IMA-Eigenschaft | Anmerkungen                                                                                                                                                                                                                                                                                                                                                         |
|-----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| FileTypes       | Nur in der neuen XenApp-Site existierende Dateitypen werden migriert. Dateitypen, die in der neuen Site nicht existieren, werden ignoriert. Dateitypen werden erst importiert, wenn die Dateitypen in der neuen Site aktualisiert wurden.                                                                                                                           |
| IconData        | Neue Symbolobjekte werden erstellt, wenn die Symboldaten für die exportierten Anwendungen angegeben wurden.                                                                                                                                                                                                                                                         |
| Accounts        | Die Benutzerkonten einer Anwendung werden zwischen der Benutzerliste für die Bereitstellungsgruppe und der Anwendung aufgeteilt. Explizite Benutzer werden zur Initialisierung der Benutzerliste für die Anwendung verwendet. Zudem wird der Benutzerliste für die Bereitstellungsgruppe das Konto "Domänenbenutzer" für die Domäne der Benutzerkonten hinzugefügt. |

Die folgenden XenApp 6.x-Eigenschaften werden nicht importiert:

| IMA-Eigenschaft                        | Anmerkungen                                              |
|----------------------------------------|----------------------------------------------------------|
| ApplicationType                        | Wird ignoriert.                                          |
| HideWhenDisabled                       | Wird ignoriert.                                          |
| AccessSessionConditions                | Ersetzt durch Bereitstellungsgruppenzugriffsrichtlinien. |
| AccessSessionConditionsEnabled         | Ersetzt durch Bereitstellungsgruppenzugriffsrichtlinien. |
| ConnectionsThroughAccessGatewayAllowed | Ersetzt durch Bereitstellungsgruppenzugriffsrichtlinien. |
| OtherConnectionsAllowed                | Ersetzt durch Bereitstellungsgruppenzugriffsrichtlinien. |
| AlternateProfiles                      | FMA unterstützt keine gestreamten Anwendungen.           |
| OfflineAccessAllowed                   | FMA unterstützt keine gestreamten Anwendungen.           |
| ProfileLocation                        | FMA unterstützt keine gestreamten Anwendungen.           |

| IMA-Eigenschaft                      | Anmerkungen                                                                                                  |
|--------------------------------------|--------------------------------------------------------------------------------------------------------------|
| FileProgramArguments                 | FMA unterstützt keine gestreamten Anwendungen.                                                               |
| ProfileProgramName                   | FMA unterstützt keine gestreamten Anwendungen.                                                               |
| RunAsLeastPrivilegedUser             | FMA unterstützt keine gestreamten Anwendungen.                                                               |
| AnonymousConnectionsAllowed          | FMA verwendet eine andere Technologie für die Unterstützung nicht authentifizierter (anonymer) Verbindungen. |
| ApplicationId, SequenceNumber        | IMA-eigene Daten.                                                                                            |
| AudioType                            | FMA unterstützt keine erweiterten Clientverbindungsoptionen.                                                 |
| EncryptionLevel                      | SecureICA ist in Bereitstellungsgruppen aktiviert/deaktiviert.                                               |
| EncryptionRequired                   | SecureICA ist in Bereitstellungsgruppen aktiviert/deaktiviert.                                               |
| SslConnectionEnabled                 | FMA verwendet eine andere SSL-Implementierung.                                                               |
| ContentAddress                       | FMA unterstützt keinen veröffentlichten Inhalt.                                                              |
| ColorDepth                           | FMA unterstützt keine erweiterten Fensterformen.                                                             |
| MaximizedOnStartup                   | FMA unterstützt keine erweiterten Fensterformen.                                                             |
| TitleBarHidden                       | FMA unterstützt keine erweiterten Fensterformen.                                                             |
| WindowsType                          | FMA unterstützt keine erweiterten Fensterformen.                                                             |
| InstanceLimit                        | FMA unterstützt keine Anwendungslimits.                                                                      |
| MultipleInstancesPerUserAllowed      | FMA unterstützt keine Anwendungslimits.                                                                      |
| LoadBalancingApplicationCheckEnabled | FMA verwendet eine andere Technologie für den Lastausgleich.                                                 |
| PreLaunch                            | FMA verwendet eine andere Technologie für den Sitzungsvorabstart.                                            |
| CachingOption                        | FMA verwendet eine andere Technologie für den Sitzungsvorabstart.                                            |
| ServerNames                          | FMA verwendet eine andere Technologie.                                                                       |
| WorkerGroupNames                     | FMA unterstützt keine Workergruppen.                                                                         |

# Migrieren von XenDesktop 4

May 10, 2016

Sie können die Daten und Einstellungen von einer XenDesktop 4-Farm in eine XenDesktop 7.x-Site mit dem Migrationstool übertragen. Dieses ist auf dem Installationsmedium im Ordner "Support > Tools > MigrationTool". Das Tool enthält Folgendes:

- Das Exporttool XdExport für den Export der XenDesktop 4-Farmdaten in eine XML-Datei (Standardname: XdSettings.xml). Das XML-Dateischema ist in der Datei XdFarmxsd.
- Das Importtool XdImport zum Importieren der Daten unter Ausführen des PowerShell-Skripts Import-XdSettings.ps1.

Das Migrationstool kann nur verwendet werden, wenn beide Bereitstellungen dieselbe Hypervisorversion (beispielsweise XenServer 6.2) und eine Active Directory-Umgebung haben.

Sie können mit diesem Tool nicht XenApp oder XenDesktop 4 nach XenApp migrieren.

Tipp: Sie können ein Upgrade von XenDesktop 5 (oder höheren XenDesktop-Versionen) auf die aktuelle XenDesktop-Version durchführen (siehe [Upgrade einer Bereitstellung](#)).

## Einschränkungen

Nicht alle Daten und Einstellungen werden exportiert. Die folgenden Konfigurationselemente werden nicht migriert. Sie werden zwar exportiert, aber nicht importiert:

- Administratoren
- Einstellungen für delegierte Administration
- Desktopgruppenordner
- Lizenzierungskonfiguration
- Registrierungsschlüssel

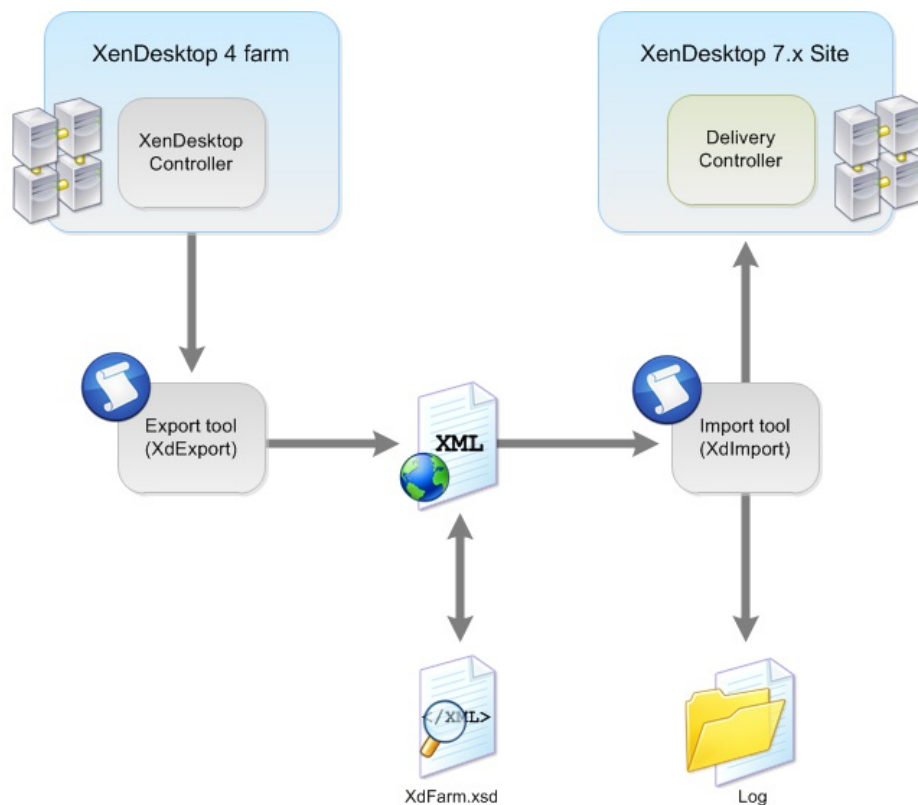
Folgende Anwendungsfälle werden bei der Migration nicht direkt unterstützt:

- Zusammenführen von Richtlinieneinstellungen oder Desktopgruppen- oder Hostingeinstellungen
- Zusammenführen von privaten Desktops in zufällige Bereitstellungsgruppen
- Anpassen vorhandener Komponenteneinstellungen mit Migrationstools

Weitere Informationen finden Sie unter [Bei der Migration eingeschlossene und ausgeschlossene Elemente](#).

## Migrationsschritte

In der folgenden Abbildung wird der Migrationsvorgang zusammengefasst.



Der Migrationsprozess besteht aus folgenden Schritten:

1. Aktivieren Sie in der Studio-Konsole auf dem XenDesktop 4-Controller den Wartungsmodus für alle Maschinen, die Sie exportieren möchten.
2. Exportieren Sie Daten und Einstellungen von der XenDesktop 4-Farm mit XdExport in eine XML-Datei (siehe [Exportieren aus einer XenDesktop 4-Farm](#)).
3. Bearbeiten Sie die XML-Datei so, dass sie nur die Daten und Einstellungen enthält, die Sie in Ihre XenDesktop-Site importieren möchten (siehe [Bearbeiten der XML-Datei des Migrationstools](#)).
4. Importieren Sie die Daten und Einstellungen aus der XML-Datei mit XdImport in Ihre neue XenDesktop-Site (siehe [Importieren von XenDesktop 4-Daten](#)).
5. Wiederholen Sie die Schritte 3 und 4, um weitere Änderungen vorzunehmen. Nach dem Vornehmen von Änderungen können Sie zusätzliche Desktops in vorhandene Bereitstellungsgruppen importieren. Verwenden Sie hierzu den Mergedesktops-Parameter beim Import.
6. Führen Sie die nach der Migration erforderlichen Schritte durch (siehe [Aufgaben nach der Migration](#)).

## Vorbereitung der Migration

Führen Sie vor der Migration die folgenden Schritte durch:

- Informieren Sie sich darüber, welche Daten exportiert und importiert werden können und welche Konsequenzen dies für Ihre Bereitstellung hat. Siehe [Bei der Migration eingeschlossene und ausgeschlossene Elemente](#).
- Citrix empfiehlt, dass Sie manuell eine Sicherungskopie der Sitedatenbank erstellen, damit Sie sie bei Problemen wiederherstellen können.
- Installieren Sie die XenDesktop 7.x-Komponenten und erstellen Sie eine Site einschließlich Datenbank.
- Für die Migration von XenDesktop 4 müssen alle VDAs auf der XenDesktop 5.x-Stufe sein, damit sie mit XenDesktop 4- und XenDesktop 7-Controllern kompatibel sind. Wenn in der Controllerinfrastruktur XenDesktop 7.x vollständig ausgeführt wird, können Windows 7-VDAs auf XenDesktop 7.x aktualisiert werden. Weitere Details finden Sie unter [Migrationsbeispiele](#).

# Exportieren aus einer XenDesktop 4-Farm

May 10, 2016

Das Exporttool "XdExport" extrahiert Daten aus einer einzelnen XenDesktop 4-Farm und generiert eine XML-Datei aus Darstellungen der Datenwerte.

Das Schema der XML-Datei wird in der Datei XdFarm.xsd bereitgestellt, die im Download des Importtools XdExport.zip und XdImport.zip enthalten ist.

Führen Sie XdExport auf einer Maschine aus, die in der Farm, aus der Daten exportiert werden, als XenDesktop 4-Controller fungiert. Auf dieser Maschine muss das XenDesktop 4-PowerShell SDK installiert sein. Sie müssen über die folgenden Berechtigungen für den Export der Daten verfügen:

- Die Benutzeridentität eines mindestens lesezugriffsberechtigten Citrix Administrators der Farm
- Berechtigung zum Lesen der Registrierung

Obwohl dies nicht empfohlen wird, können Sie das Tool ausführen, während der XenDesktop Controller aktiv verwendet wird (z. B. wenn Benutzer bei VDAs angemeldet sind).

Citrix empfiehlt dringend Folgendes:

- Update des XenDesktop 4-Controllers, auf dem das Tool ausgeführt wird, mit öffentlichen Hotfixes
- Keine Konfigurationsänderungen an der Site während der Export ausgeführt wird (z. B. Entfernen von Desktopgruppen)

1. Laden Sie XdExport.zip herunter und extrahieren Sie die Dateien auf den XenDesktop 4-Controller.
2. Führen Sie an einer Eingabeaufforderung XdExport.exe mit folgenden optionalen Parametern aus:

| Parameter           | Beschreibung                                                                                                                                                                                                                                                                                 |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -Verbose            | Generiert Nachrichten, die detaillierte Fortschrittsinformationen liefern.                                                                                                                                                                                                                   |
| -FilePath<br><path> | Der Speicherort der XML-Datei, in die die Farmdaten exportiert werden. Standard = .\XdSettings.xml                                                                                                                                                                                           |
| -Overwrite          | Überschreibt jede in -FilePath angegebene Datei. Wenn Sie diesen Parameter nicht angeben und eine Ausgabedatei bereits vorhanden ist, schlägt das Tool folgender Fehlermeldung fehl: "Fehler: Datei bereits vorhanden. Geben Sie -Overwrite an, um das Überschreiben der Datei zuzulassen. " |
| -? oder -<br>help   | Zeigt eine Beschreibung der Parameter an. Beim Beenden werden keine Daten exportiert.                                                                                                                                                                                                        |

3. Bei erfolgreicher Ausführung des Tools wird die Meldung Fertig angezeigt. Die Datei XdSettings.xml befindet sich an dem vom Parameter FilePath angegebenen Speicherort. Schlägt das Tool fehl, wird eine Fehlermeldung angezeigt.

# Bearbeiten der XML-Datei des Migrationstools

May 10, 2016

Vor dem Importieren von Daten in eine XenDesktop 7.x-Site bearbeiten und prüfen Sie den Inhalt der vom Exporttool (XdExport) erstellten XML-Datei, besonders, wenn Sie die Migration in mehreren Phasen durchführen und beim Import einige Benutzer, Bereitstellungsgruppen und Richtlinien anderen vorgezogen haben.

Sie können den Inhalt der Datei mit einem beliebigen Texteditor anzeigen oder ändern. Sie können auch einen speziellen XML-Editor wie Microsoft XML Notepad verwenden.

Bestimmte Elemente des XML-Inhalts müssen in der XML-Datei vorhanden sein, damit sie vom Importtool (XdImport) verarbeitet wird.

Das benötigte XML-Schema ist in der Datei XdFarm.xsd definiert, die im Download des Migrationstools enthalten ist. Beim Arbeiten mit dieser Datei gilt Folgendes:

- In der Datei gibt das Attribut "minOccurs" mit einem Wert von 1 oder höher an, dass bestimmte Elemente vorhanden sein müssen, wenn das übergeordnete Element vorhanden ist.
- Wenn die an das Importtool übergebene XML-Datei ungültig ist, stoppt das Tool und eine Fehlermeldung wird angezeigt, mit deren Hilfe Sie in der XML-Datei nach dem Problem suchen können.

## Importieren einer Untergruppe von Desktops oder Bereitstellungsgruppen

Zum Importieren nur einer Untergruppe von Desktopgruppen und Desktops müssen Sie den Inhalt des Elements DesktopGroups bearbeiten. Das Element DesktopGroups kann viele DesktopGroup-Elemente enthalten und in jedem DesktopGroup-Element befindet sich ein Desktops-Element, das zahlreiche Desktop-Elemente enthalten kann.

Löschen Sie das DesktopGroups-Element nicht. Sie können aber alle DesktopGroup-Elemente löschen und es leer lassen. Ebenso muss in jedem DesktopGroup-Element das Desktops-Element vorhanden sein. Es muss aber keine Desktop-Elemente enthalten.

Löschen Sie Desktop- oder DesktopGroup-Elemente, um zu verhindern, dass bestimmte einzelne Maschinen oder ganze Bereitstellungsgruppen importiert werden. Die XML-Datei enthält beispielsweise:

```
... ..
```

Bei diesem Beispiel wird durch die Änderungen verhindert, dass die Gruppe "Group1" importiert wird. Nur Machine3 aus der Gruppe "Group2" wird importiert:

```
... ..
```

## Verwalten von Bereitstellungsgruppen mit doppelten Namen

In XenDesktop 4 können Desktopgruppen in Ordnern organisiert werden, Desktopgruppen mit dem gleichen Namen können in verschiedenen Ordnern erscheinen und der interne Name einer Desktopgruppe ist der Name, der den Benutzern angezeigt wird. In diesem Release hingegen dürfen Bereitstellungsgruppen nicht in Ordnern platziert werden und jede Bereitstellungsgruppe muss einen eindeutigen internen Namen haben, wobei der Name, der den Benutzern angezeigt wird, sich vom internen Namen unterscheiden kann. Aufgrund dieser Unterschiede müssen Sie möglicherweise Desktopgruppen umbenennen.

In Ihrer XenDesktop 4-Farm können Sie beispielsweise zwei verschiedene Desktopgruppen haben, die zwei verschiedenen Benutzern mit dem Namen "Mein Desktop" angezeigt wird. Dies kann durch Desktopgruppenordner ermöglicht werden. Wenn diese Bereitstellungsgruppen in der XenDesktop 7.x-Site getrennt bleiben sollen, müssen Sie die Namen der Desktopgruppen in der XML-Datei in eindeutige Namen ändern.

Wenn eine Bereitstellungsgruppe in der XenDesktop 7.x-Site über denselben Namen verfügt wie die zu importierende Desktopgruppe und die Bereitstellungsgruppen in der XenDesktop 7.x-Site getrennt bleiben sollen, müssen Sie den Namen der XenDesktop 4-Desktopgruppe in der XML-Datei ändern, damit der Name in der Site eindeutig bleibt. Wenn die zu importierende Desktopgruppe tatsächlich der XenDesktop 7.x-Bereitstellungsgruppe entspricht und die Maschinen in der XML-Datei mit der vorhandenen Desktopgruppe zusammengeführt werden sollen, müssen Sie die Desktopgruppe nicht umbenennen. Geben Sie stattdessen den Parameter -MergeDesktops im Importtool an. Enthält die XML-Datei beispielsweise:

... \Sales ... \Finance

Entfernen Sie die doppelten Namen wie folgt:

... \Sales ... \Finance

### Verwalten von Richtlinienimports

Sie können einzelne Richtlinien aus der XML-Datei löschen und eindeutige Namen festlegen, um doppelte Richtlinienamen zu vermeiden. Das Zusammenführen von Richtlinien wird nicht unterstützt.

- Beim Import von Richtlinien werden entweder alle Richtlinien erfolgreich importiert oder es werden keine Richtlinien importiert, wenn an irgendeiner Stelle ein Fehler auftritt.
- Der Import zahlreicher Richtlinien mit umfangreichen Einstellungen kann mehrere Stunden dauern.
- Wenn Sie Richtlinien in Batches importieren, können deren ursprüngliche Prioritäten beeinträchtigt werden. Wenn Sie Richtlinien importieren, werden die relativen Prioritäten der importierten Richtlinien beibehalten. Sie erhalten jedoch eine höhere Priorität als bereits in der Site vorhandene Richtlinien. Wenn beispielsweise vier Richtlinien mit den Prioritäten 1 bis 4 in zwei Batches importiert werden sollen, sollten Sie die Richtlinien mit den Prioritäten 3 und 4 zuerst importieren, da der zweiten Gruppe automatisch eine höhere Priorität eingeräumt wird.

Zum ausschließlichen Importieren einer Untergruppe der Richtlinien in die XenDesktop 7.x-Site bearbeiten Sie den Inhalt des Policies-Elements. Das Policies-Element kann zahlreiche Policy-Elemente enthalten. Sie dürfen das Policies-Element nicht löschen. Sie können aber alle Policy-Elemente löschen und es leer lassen. Löschen Sie alle Policy-Elemente, um den Import bestimmter XenDesktop 4-Farmrichtlinien zu vermeiden. Enthält die XML-Datei beispielsweise:

... ...

Wenn Sie den Import von XenDesktop 4-Richtlinien verhindern möchten, um mögliche Konflikte mit bereits in der XenDesktop 7.x-Site konfigurierten Richtlinien zu vermeiden, bearbeiten Sie die Datei, indem Sie einzelne Policy-Elemente wie folgt entfernen:

Alternativ können Sie die Datei so bearbeiten, dass die Richtlinie wie folgt mit einem anderen Namen importiert wird:

... ...



# Importieren von XenDesktop 4-Daten

May 10, 2016

Das Importtool "XdImport" liest Einstellungen aus XenDesktop 4, die in der vom Exporttool "XdExport" erzeugten XML-Datei enthalten sind, und wendet diese Einstellungen auf eine vorhandene XenDesktop 7.x-Site an. Das Importtool verwendet das PowerShell-Skript Import-XdSettings.ps1.

Um nur eine Teilmenge der exportierten Daten anzuwenden, bearbeiten Sie die XML-Datei, bevor Sie das Importtool ausführen. Sie können beispielsweise Desktopgruppen und Richtlinien entfernen, die in der XenDesktop 7.x-Bereitstellung nicht benötigt werden. Das Importtool wird erfolgreich ausgeführt, wenn Sie ganze Elemente leer lassen. Sie können beispielsweise alle Desktopgruppen problemlos löschen. Das Tool überprüft vor dem Import der Daten immer die XML-Datei.

Führen Sie XdImport auf jeder Maschine aus, auf der alle XenDesktop 7.x-SDKs installiert sind. Sie müssen Volladministrator für XenDesktop sein, um das Tool auszuführen.

Stellen Sie vor dem Importieren sicher, dass Sie eine XenDesktop 7.x-Site einschließlich Datenbank eingerichtet haben. Citrix empfiehlt, den Import in XenDesktop 7.x durchzuführen, bevor Benutzertests oder eine allgemeine Sitekonfiguration durchgeführt werden. Führen Sie Konfigurationen nur zusammen, wenn die Site nicht verwendet wird.

1. Erstellen Sie eine XenDesktop 7.x-Site.
2. Laden Sie die Datei XdImport.zip herunter und extrahieren Sie die Dateien auf die Maschine, auf der das Tool ausgeführt werden soll.
3. Führen Sie in einer PowerShell-Sitzung Import-XdSettings.ps1 mit folgenden Parametern aus:

| Parameter                         | Beschreibung                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|-----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| - HypervisorConnectionCredentials | (erforderlich). Eine PowerShell-Hashtabelle, in der PSCredential-Instanzen Hypervisoradressen als Voraussetzung für die Erstellung von Hypervisorverbindungen<br>Geben Sie die Anmeldeinformationen für den Hypervisor ein, mit dem die XenDesktop 4-Farm eine Verbindung herstellt.<br><br>Für einen einzelnen Hypervisor können Sie das Argument folgendermaßen erstellen:<br><br>\$credential = Get-Credential \$mappings = @{"http://" = \$credential} .\Import-XdSettings.ps1 -FilePath .\XdSettings.xml -HypervisorConnectionCredentials \$mappings<br>Die in der Hashtabelle angegebene Adresse muss mit der Adresse in der XML-Datei genau übereinstimmen.<br><br>Beispiel: Erstellen Sie bei Verwendung eines XenServer- und eines VMware-Hypervisors folgendes Argument:<br><br>\$Xencredential = Get-Credential \$VMWcredential = Get-Credential \$mappings = @{"http://" = \$Xencredential;"http://SDK" = \$VMWcredential} .\Import-XdSettings.ps1 -FilePath .\XdSettings.xml -HypervisorConnectionCredentials \$mappings |
| -FilePath <path>                  | (Der Wert für ist erforderlich.) Der Speicherort der XML-Datei, aus der die Farmdaten importiert werden                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| -AdminAddress                     | Name des Controllers in der XenDesktop 7.x-Site Standard = localhost                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| -MergeDesktops                    | Fügt in der XML-Datei definierte Desktops Bereitstellungsgruppen in der XenDesktop 7.x-Site hinzu, deren Namen mit Gruppen in der XML-Datei übereinstimmen hinzugefügt.<br><br>Wird dieser Parameter nicht angegeben, wird vorhandenen Bereitstellungsgruppen in der XenDesktop 7.x-Site kein Inhalt hinzugefügt.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| -SkipMachinePolicy                | Das Skript erstellt keine Maschinenrichtlinie mit Einstellungen auf Siteebene. Wenn Sie diesen Parameter nicht angeben und die Maschinenrichtlinie für die Site                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| -WhatIf                           | Führt einen Testlauf durch, um festzustellen, was an der XenDesktop 7.x-Site geändert und was ihr hinzugefügt werden würde. Wenn dieser Parameter angegeben wird jedoch nicht geändert.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| -LogFilePath <path>               | Der vollständige Pfad der Protokolldatei. Das Protokoll enthält Text, der alle auf der XenDesktop 7.x-Site durchgeführten Schreibvorgänge beschreibt. Standard                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| -? oder -help                     | Zeigt eine Beschreibung von Parametern an. Beim Beenden werden keine Daten importiert.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |

Wenn die XML-Datei Richtliniendaten enthält, werden alle Richtlinien erfolgreich importiert. Tritt jedoch an irgendeiner Stelle ein Fehler auf, werden keine Richtliniendaten importiert. Der Import zahlreicher Richtlinien mit umfangreichen Einstellungen kann mehrere Stunden dauern.

Nach Abschluss des Skripts wird die Meldung Fertig angezeigt. Nach erfolgreichem Import der Daten aus der XML-Datei können Sie weitere Import- und Exportvorgänge durchführen. Wenn Sie jedoch alle relevanten Daten importiert haben, können Sie die Aufgaben nach der Migration durchführen.

# Aufgaben nach der Migration

May 10, 2016

Nach dem erfolgreichen Importieren von Daten aus einer XenDesktop 4-Farm in eine XenDesktop 7.x-Site müssen Sie folgende Aufgaben durchführen, bevor Sie die neue Site als Produktionsumgebung nutzen können:

- Aktualisieren Sie die Virtual Delivery Agents (VDAs). Obwohl es nicht erforderlich ist, empfiehlt Citrix, dass Sie zunächst ein Upgrade der VDAs und dann erst ein Upgrade der Controller, von Studio oder von Director durchführen.
  - Windows Vista und Windows XP: Upgrade auf XenDesktop 5.6 Feature Pack 1 Virtual Desktop Agent
  - Windows 7: Upgrade auf XenDesktop 7.x Virtual Delivery Agent.
- Erstellen Sie die Administratoren, die für die XenDesktop 7.x-Site benötigt werden.
- Citrix empfiehlt, ein Update der Benutzergeräte auf die aktuelle Version von Citrix Receiver durchzuführen, um von Hotfixes zu profitieren und Unterstützung für die aktuellen Funktionen zu erhalten.
- Ändern Sie die importierten Desktops so, dass die registrierungsbasierte Controller-Discovery verwendet wird, und verweisen Sie sie auf die XenDesktop 7.x-Controller mit einer der folgenden Methoden:
  - Entfernen Sie manuell den nicht benötigten Organisationseinheits-GUID-Eintrag aus der Registrierung und fügen Sie einen Registrierungseintrag "ListOfDDCs" hinzu.
  - Richten Sie eine Maschinenrichtlinie ein, um die Liste der Controller auf die Desktops mit der Active Directory-Richtlinie GPMC.msc zu verteilen. Sie können diese Einstellung nicht mit Studio konfigurieren.

Registrierungsbasierte Controller-Discovery ist die Standardeinstellung für XenDesktop 7.x, Active Directory-basierte Discovery steht aber weiterhin zur Verfügung.

- Optional können Sie die folgenden Registrierungsschlüsseleinstellungen implementieren, die im Abschnitt zu den bewährten Methoden für die registrierungsbasierte Registrierung in XenDesktop unter [CTX133384](#) beschrieben werden:
  - HeartbeatPeriodMS
  - PrepareSessionConnectionTimeoutSec
  - MaxWorkers
  - DisableActiveSessionReconnect
  - ControllersGroupGuid

Wenn Sie dies unterlassen, werden die XenDesktop 7.x-Standardeinstellungen für diese Schlüssel verwendet.

- Deaktivieren Sie den Wartungsmodus für die importierten Maschinen, sofern diese vor dem Generieren der XML-Datei in XenDesktop 4 im Wartungsmodus waren.
- Überprüfen Sie die XenDesktop 7.x-Einstellungen, um sicherzustellen, dass sie korrekt sind, insbesondere dann, wenn Sie die XML-Datei PortICAConfig unter XenDesktop 4 geändert haben.
- Überprüfen Sie alle Migrationskomponenten, um sicherzustellen, dass die Migration erfolgreich abgeschlossen wurde.

# Migrationsbeispiele

May 10, 2016

## Beispiel 1: Migrieren einer großen XenDesktop 4-Farm in eine XenDesktop 7-Site

In diesem Beispiel wird eine XenDesktop 4-Farm verwendet. Die XenDesktop 4-Farm hat 50 Desktopgruppen, wobei jede Gruppe durchschnittlich 100 Desktops hat. Die XenDesktop 4-Desktops werden über Provisioning Services (PVS) bereitgestellt und die Maschinen werden auf VMware ESX-Hypervisors ausgeführt. Auf allen VMs ist die XenDesktop 4-Version des VDAs installiert.

### Migrationsschritte

1. Aktualisieren Sie alle XenDesktop 4-VDAs auf die XenDesktop 5.6 Feature Pack 1-VDA-Software. Dadurch können die VDAs sowohl beim XenDesktop 4-Controller als auch beim XenDesktop 7-Delivery Controller registriert werden.
  - Informationen zu Windows 7-VDAs finden Sie unter [Durchführen eines Upgrades von Virtual Desktop Agent auf einer VM oder einem Blade-Computer](#).
  - Informationen zu VDAs unter Windows XP und Windows Vista finden Sie unter [Virtual Desktop Agents unter Windows XP oder Windows Vista](#).
2. Stellen Sie sicher, dass sich alle Benutzer von der XenDesktop 4-Farm abgemeldet haben.
3. Stellen Sie sicher, dass alle Maschinen in den Wartungsmodus gesetzt wurden.
4. Führen Sie das Exporttool (XdExport) auf der XenDesktop 4-Farm aus.
5. Installieren Sie XenDesktop 7-Komponenten.
  1. Erstellen Sie mit Studio eine Site im vollständigen Produktionsmodus.
  2. Ist Provisioning Services Teil der Bereitstellung, aktualisieren Sie den Provisioning Services-Server und die Provisioning Services-Agents.
  3. Aktualisieren Sie den Lizenzserver und die zugehörigen Lizenzen.
6. Entzippen Sie das Importtool (XdImport) in einem lokalen Verzeichnis auf dem XenDesktop 7-Controller.
7. Kopieren Sie die in Schritt 4 mit dem Exporttool erstellte XML-Datei (XdSettings.xml) in das lokale Verzeichnis.
8. Starten Sie in der PowerShell-Konsole des Studio-Stammknotens der XenDesktop 7-Site eine PowerShell-Sitzung.
9. Führen Sie das Importtool (XdImport) aus, sodass die Anmeldeinformationen der zugeordneten Hypervisors und der Pfad der XML-Datei übergeben werden.
10. Erstellen Sie manuell Administratoreinstellungen über den Administratorknoten in Studio neu. Informationen hierzu finden Sie unter [Delegierte Administration](#).
11. Ändern Sie die importierten Desktops so, dass sie die registrierungsbasierte Controller-Discovery verwenden, und verweisen Sie sie auf den neuen XenDesktop 7-Controller.
12. Für VDAs unter Windows 7 empfiehlt Citrix ein Upgrade zur Verwendung von XenDesktop 7-VDAs für Windows-Desktopbetriebssysteme, damit alle neuen Features genutzt werden können.

Nach dem Upgrade von VDAs auf XenDesktop 7 für Maschinen in einem Katalog oder einer Bereitstellungsgruppe aktualisieren Sie den Katalog (siehe [Verwalten von Maschinenkatalogen](#)) und die Bereitstellungsgruppen (siehe [Verwalten von Einstellungen in Bereitstellungsgruppen](#)).
13. Deaktivieren Sie den Wartungsmodus für die Bereitstellungsgruppen.
14. Konfigurieren Sie StoreFront, um die zuvor vom Webinterface bereitgestellten Desktops zur Verfügung zu stellen. Weitere Informationen finden Sie in der StoreFront-Dokumentation.

## Beispiel 2: Export einer XenDesktop 4-Farm mit teilweisem Import in eine XenDesktop 7.1-Site

In diesem Beispiel erfolgt die Migration in mehreren Schritten, wobei bei jedem Schritt ein Teil der verbleibenden Desktops

migriert wird. Derzeit wird eine XenDesktop 4-Farm verwendet und eine XenDesktop 7.1-Site wurde bereits erstellt und ist im Einsatz. Die XenDesktop 4-Farm hat 50 Desktopgruppen, wobei jede Gruppe durchschnittlich 100 Desktops hat. Die XenDesktop 4-Desktops werden über Provisioning Services bereitgestellt und die Maschinen werden auf Citrix XenServer-Hypervisors ausgeführt. Auf allen VMs ist die XenDesktop 4-Version des VDAs installiert.

### **Migrationsschritte**

1. Führen Sie das Exporttool (XdExport) auf der XenDesktop 4-Farm aus.
  1. Entzippen Sie das Exporttool (XdExport) auf einem Desktop Delivery Controller in der Farm.
  2. Führen Sie das Exporttool als Citrix Administrator ohne Parameter aus.
2. Kopieren und bearbeiten Sie die resultierende XML-Datei, sodass sie nur die Gruppen und Desktops enthält, die Sie migrieren möchten.
3. Vergewissern Sie sich, dass in der XenDesktop 4-Farm sich alle Benutzer von den zu migrierenden Desktops abgemeldet haben und aktivieren Sie den Wartungsmodus für alle Desktops, die migriert werden sollen.
4. Entzippen Sie das Importtool (XdImport) in einem lokalen Verzeichnis auf dem XenDesktop 7.1-Delivery-Controller.
5. Kopieren Sie die bearbeitete XML-Datei in das lokale Verzeichnis.
6. Starten Sie in der PowerShell-Konsole des Studio-Stammknotens der XenDesktop 7.1-Site eine PowerShell-Sitzung.
7. Führen Sie das Importtool (XdImport) aus, sodass die Anmeldeinformationen der zugeordneten Hypervisors und der Pfad der XML-Datei übergeben werden.
8. Erstellen Sie manuell Administratoreinstellungen über den Administratorknoten in Studio neu. Informationen hierzu finden Sie unter [Delegierte Administration](#).
9. Ändern Sie die importierten Desktops so, dass sie die registrierungsbasierte Controller-Discovery verwenden, und verweisen Sie sie auf den neuen XenDesktop 7.1-Controller.
10. Aktualisieren Sie alle VDAs auf die entsprechende VDA-Software:
  - Windows 7-VDAs:
    - Führen Sie ein Upgrade auf XenDesktop 7-VDAs durch, wie unter [Durchführen eines Upgrades von Virtual Desktop Agent auf einer VM oder einem Blade-Computer](#) beschrieben.
    - Nach dem Upgrade der gesamten VDA-Software auf XenDesktop 7 für Maschinen in einem Katalog oder einer Bereitstellungsgruppe aktualisieren Sie den Katalog (siehe [Verwalten von Maschinenkatalogen](#)) und die Bereitstellungsgruppen (siehe [Verwalten von Einstellungen in Bereitstellungsgruppen](#)).
  - Aktualisieren Sie VDAs, die unter Windows XP oder Windows Vista ausgeführt werden auf XenDesktop 5.6 FP1 (siehe [Virtual Desktop Agents unter Windows XP oder Windows Vista](#)).
11. Deaktivieren Sie den Wartungsmodus für die Bereitstellungsgruppen.
12. Konfigurieren Sie StoreFront, um die zuvor vom Webinterface bereitgestellten Desktops zur Verfügung zu stellen. Weitere Informationen finden Sie in der StoreFront-Dokumentation.

# Bei der Migration eingeschlossene und ausgeschlossene Elemente

May 10, 2016

## Eingeschlossene Elemente

In der folgenden, wenn auch nicht vollständigen, Tabelle wird beschrieben, was während der Migration zu diesem Release mit den wichtigsten Daten geschieht. Sofern nicht anders angegeben, werden die jeweiligen Datentypen importiert.

| Datentyp             | Hinweise                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Desktopgruppen       | <p>Desktopgruppen werden in diesem Release zu Bereitstellungsgruppen. Symbole von Desktopgruppen werden nicht exportiert.</p> <p>SecureIcaRequired ist auf "True" gesetzt, wenn DefaultEncryptionLevel in XenDesktop 4 nicht auf "Basic" festgelegt ist.</p> <p>Wenn eine Desktopgruppe in der XenDesktop 4-Farm den gleichen Namen wie eine Bereitstellungsgruppe in der XenDesktop 7.x-Site hat, können Sie Desktops der XenDesktop 4-Gruppe einer Bereitstellungsgruppe mit demselben Namen in der Zielsite hinzufügen.</p> <p>Geben Sie hierfür den Parameter MergeDesktops an, wenn Sie das Importtool ausführen. Die Einstellungen der XenDesktop 7.x-Bereitstellungsgruppe werden nicht mit den Einstellungen der XenDesktop 4-Gruppe überschrieben. Wenn dieser Parameter nicht angegeben wird und eine Gruppe mit demselben Namen wie eine in der XML-Datei definierte Gruppe existiert, zeigt das Tool einen Fehler an und wird angehalten, bevor Daten importiert werden.</p> |
| Desktops             | <p>Sie können keine privaten Desktops zu zufälligen Bereitstellungsgruppen hinzufügen. Zufällige Desktops können einer statischen Bereitstellungsgruppe nicht hinzugefügt werden.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| Maschinen            | <p>Maschinen werden in vier Maschinenkataloge importiert. Die folgenden Maschinenkataloge werden vom Importtool in der XenDesktop 7.x-Site automatisch erstellt:</p> <ul style="list-style-type: none"><li>• Importierte bestehende zufällige (für gepoolte VMs)</li><li>• Importierte bestehende statische (für zugewiesene VMs)</li><li>• Importierte physische zufällige (für gepoolte oder Blade-PC)</li><li>• Importierte physische statische (für private oder Blade-PC)</li></ul> <p>Alle folgenden Importe von Maschinen verwenden dieselben vier Maschinenkataloge.</p>                                                                                                                                                                                                                                                                                                                                                                                                         |
| Poolverwaltungspools | <p>Umfasst aus mehreren Pools bestehende Pools und Einstellungen für betriebsbereite Pools, einschließlich Zeitplan.</p> <ul style="list-style-type: none"><li>• PeakBufferSizePercent ist standardmäßig auf 10 % gesetzt.</li><li>• OffPeakBufferSizePercent ist standardmäßig auf 10 % gesetzt.</li><li>• Alle nicht ausgewählten Tage in der Einstellung Business days in XenDesktop 4 werden in dieses Release als Teil des Wochenend-Energieschemas importiert.</li><li>• HostingXD4-Aktionszeiten werden auf die nächste volle Minute gerundet.</li><li>• Startzeiten werden auf die nächste volle Stunde gerundet.</li><li>• Endzeiten werden auf die nächste volle Stunde gerundet.</li></ul>                                                                                                                                                                                                                                                                                    |
| Farmeinstellungen    | <p>Die folgenden Farmeinstellungen werden als Maschinenrichtlinie importiert:</p> <ul style="list-style-type: none"><li>• IcaKeepAlive</li><li>• AutoClientReconnect</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |

| Datentyp                   |  SessionReliability<br><b>Hinweise</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                            | Die Einstellung zum Aktivieren von Flash Player wird nicht importiert.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Richtlinien                | <p>Bestimmte Richtliniendaten werden importiert. Filter, Einstellungen und Drucker werden als Benutzerrichtlinien importiert. Weitere Informationen zum Export und Import von Benutzerrichtlinien finden Sie in der anderen Tabelle in diesem Dokument.</p> <ul style="list-style-type: none"> <li>• Neue Regeln für Zugriffsrichtlinien werden basierend auf XenDesktop 4-Gruppeneinstellungen erstellt.</li> <li>• Beim Import von Richtlinien wird deren relative Prioritätsreihenfolge beibehalten. Sie werden jedoch immer mit einer höheren Priorität als alle auf der XenDesktop 7.x-Site vorhandenen Richtlinien hinzugefügt.</li> <li>• Das Zusammenführen von Richtlinien wird nicht unterstützt.</li> </ul> <p>Es gibt keine Möglichkeit, Richtlinien in Active Directory zu importieren. Sie werden immer in der Site gespeichert.</p> |
| Benutzerzuweisungen        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| Hypervisorereinstellungen  | <p>Dieser Parameter ist für das XdImport-Tool erforderlich.</p> <p>Hypervisoradressen werden exportiert, nicht aber die Anmeldeinformationen, die für den Zugriff auf diese Hypervisoren erforderlich sind. Zum Erstellen von Hypervisorverbindungen in der XenDesktop 7.x-Site müssen Sie die Adressen aus der XML-Datei extrahieren und eine PowerShell-Hashtabelle anlegen, in der die Adressen den entsprechenden Anmeldeinformationen zugeordnet sind. Geben Sie dann diese Tabelle im Parameter HypervisorConnectionCredentials des Importtools an. Weitere Informationen finden Sie unter <a href="#">Importieren von XenDesktop 4-Daten</a>.</p> <p>Das Zusammenführen oder Aktualisieren von Hypervisorereinstellungen für vorhandene Desktopgruppen und Hypervisorverbindungen wird nicht unterstützt.</p>                               |
| Administratoren            | (Wird nicht importiert.) Es werden keine Administratordaten importiert, einschließlich Daten über delegierte Administratoren. Sie müssen neue Administratoren für die XenDesktop 7.x-Site erstellen.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| Lizenzierungskonfiguration | (Wird nicht importiert.) Enthält Informationen wie Lizenzservername und Edition. Lizenzdateien werden nicht exportiert.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Desktopgruppenordner       | (Wird nicht importiert.) Dieses Release unterstützt keine Desktopgruppenordner. Bei doppelten Desktopgruppennamen (weil unterschiedliche Ordner in der XenDesktop 4-Farm Gruppen mit identischen Namen enthielten) wird das Importtool angehalten, wenn Sie die Namen in der XML-Datei nicht bearbeiten.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| Registrierungsschlüssel    | (Wird nicht importiert.) Weitere Informationen zur Implementierung von Registrierungsschlüsseln finden Sie unter <a href="#">Aufgaben nach der Migration</a> .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |

## Benutzerrichtliniendaten

In der folgenden Tabelle wird beschrieben, wie Benutzerrichtliniendaten exportiert und importiert werden.

| Kategorie und Einstellung in XenDesktop 4  | XML-Datei            | Kategorie und Einstellung in XenDesktop 7.x |
|--------------------------------------------|----------------------|---------------------------------------------|
| Bandbreite\Visuelle Effekte\Sitzungslimits | ClientOEMVCBandwidth | Nicht importiert                            |
| OEM-Virtuelle Kanäle                       |                      |                                             |

| Kategorie und Einstellung in XenDesktop 4                                                  | Kategorie und Einstellung in XenDesktop 7.x | Kategorie und Einstellung in XenDesktop 7.x                                                                                                                                                                                                                                                               |
|--------------------------------------------------------------------------------------------|---------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OEM-Virtuelle Kanäle deaktivieren                                                          | DoNotUseClientLocalTime                     | Nicht importiert                                                                                                                                                                                                                                                                                          |
| Benutzerarbeitsbereich\Zeitzone<br>Lokale Zeit des Clients nicht verwenden                 | ClientSecurityRequirement                   | Nicht importiert                                                                                                                                                                                                                                                                                          |
| Sicherheit\Verschlüsselung<br>SecureICA-Verschlüsselung                                    | LossyCompression settings                   | ICA\Visuelle Anzeige\Festbilder<br><br>Grad der verlustreichen Komprimierung<br><br>Schwellenwert für verlustreiche Komprimierung<br><br>Heavyweight-Komprimierung<br><br>ICA\Visuelle Anzeige\Bewegtbilder<br><br>Grad der progressiven Komprimierung<br><br>Schwellenwert für progressive Komprimierung |
| Bandbreite\SpeedScreen<br>Beschleunigung der Bildanzeige durch verlustreiche Komprimierung | TurnOffWallpaper                            | ICA\Desktopbenutzeroberfläche<br>Desktophintergrund                                                                                                                                                                                                                                                       |
| Bandbreite\Visuelle Effekte<br>Desktophintergrund deaktivieren                             | TurnOffMenuWindowAnimation                  | ICA\Desktopbenutzeroberfläche<br>Menüanimation                                                                                                                                                                                                                                                            |
| Bandbreite\Visuelle Effekte<br>Menüanimation                                               | DoNotShowWindowContentsWhileDragging        | ICA\Desktopbenutzeroberfläche<br>Fensterinhalt beim Verschieben anzeigen                                                                                                                                                                                                                                  |
| Bandbreite\Visuelle Effekte<br>Fensterinhalt beim Verschieben nicht anzeigen               | ClientAudioBandwidth__AllowedBandWidth      | ICA\Bandbreite<br>Bandbreitenlimit für die Audioumleitung                                                                                                                                                                                                                                                 |
| Bandbreite\Visuelle Effekte\Sitzungslimits<br>Audio                                        | ClientClipboardBandwidth__AllowedBandWidth  | ICA\Bandbreite<br>Bandbreitenlimit für Zwischenablagenumleitung                                                                                                                                                                                                                                           |
| Bandbreite\Visuelle Effekte\Sitzungslimits<br>Zwischenablage                               |                                             |                                                                                                                                                                                                                                                                                                           |

|                                                                      |                                                   |                                                             |
|----------------------------------------------------------------------|---------------------------------------------------|-------------------------------------------------------------|
| Bandbreite\Visuelle Effekte\Sitzungslimits<br>COM-Ports              | ClientComBandwidth__AllowedBandWidth<br>XML-Daten | Die COM-Portumleitung ist in<br>XenDesktop 7.x veraltet.    |
| Bandbreite\Visuelle Effekte\Sitzungslimits<br>Laufwerke              | ClientDriveBandwidth__AllowedBandWidth            | ICA\Bandbreite<br>Bandbreitenlimit für<br>Dateiumleitung    |
| Bandbreite\Visuelle Effekte\Sitzungslimits<br>LPT-Ports              | ClientLptBandwidth__AllowedBandWidth              | Die LPT-Portumleitung ist in<br>XenDesktop 7.x veraltet.    |
| Bandbreite\Visuelle Effekte\Sitzungslimits<br>Sitzung insgesamt      | OverallBandwidth__AllowedBandWidth                | ICA\Bandbreite<br>Bandbreitenlimit für Sitzung<br>insgesamt |
| Bandbreite\Visuelle Effekte\Sitzungslimits<br>Drucker                | LimitPrinterBandWidth__AllowedBandWidth           | ICA\Bandbreite<br>Bandbreitenlimit für<br>Druckerumleitung  |
| Clientgeräte\Ressourcen\Audio<br>Mikrofone                           | ClientAudioMicrophone__TurnOn                     | ICA\Audio<br>Clientmikrofonumleitung                        |
| Clientgeräte\Ressourcen\Audio<br>Tonqualität                         | ClientAudioQuality__Quality                       | ICA\Audio<br>Audioqualität                                  |
| Clientgeräte\Ressourcen\Audio<br>Lautsprecher deaktivieren           | DisableClientAudioMapping                         | ICA\Audio<br>Clientaudioumleitung                           |
| Clientgeräte\Ressourcen\Laufwerke<br>Verbindung (Connection)         | ConnectClientDriveAtLogon__TurnOn                 | ICA\Dateiumleitung<br>Laufwerke automatisch<br>verbinden    |
| Clientgeräte\Ressourcen\Laufwerke<br>Diskettenlaufwerke deaktivieren | DisableClientDriveMapping__DisableFloppyDrive     | ICA\Dateiumleitung<br>Clientdiskettenlaufwerke              |
| Clientgeräte\Ressourcen\Laufwerke<br>Festplatten deaktivieren        | DisableClientDriveMapping__DisableHardDrive       | ICA\Dateiumleitung<br>Lokale<br>Clientfestplattenlaufwerke  |
| Clientgeräte\Ressourcen\Laufwerke<br>CD-ROM-Laufwerke deaktivieren   | DisableClientDriveMapping__DisableCdrom           | ICA\Dateiumleitung<br>Optische Clientlaufwerke              |



| Clientgeräte\Ressourcen\Laufwerke<br><b>Kategorie und Einstellung in XenDesktop 4</b> | DisableClientDriveMapping__DisableRemote<br><b>XML-Daten</b> | ICA\Datei umleitung<br><b>Kategorie und Einstellung in XenDesktop 7.x</b> |
|---------------------------------------------------------------------------------------|--------------------------------------------------------------|---------------------------------------------------------------------------|
| Remotelaufwerke deaktivieren                                                          |                                                              | Clientnetzlaufwerke                                                       |
| Clientgeräte\Ressourcen\Laufwerke<br>USB-Laufwerke deaktivieren                       | DisableClientDriveMapping__DisableUSB                        | ICA\Datei umleitung<br>Clientwechsellaufwerke                             |
| Clientgeräte\Ressourcen\Laufwerke\Optimierung<br>Asynchrones Schreiben                | CDMAsyncWrites                                               | ICA\Datei umleitung<br>Asynchrones Schreiben verwenden                    |
| Clientgeräte\Ressourcen\Sonstiges<br>Zwischenablagezuordnung deaktivieren             | DisableClientClipboardMapping                                | ICA<br>Clientzwischenablagenumleitung                                     |
| Clientgeräte\Ressourcen\Ports<br>COM-Ports deaktivieren                               | DisableClientCOMPortMapping                                  | Die COM-Portumleitung ist in XenDesktop 7.x veraltet.                     |
| Clientgeräte\Ressourcen\Ports<br>LPT-Ports deaktivieren                               | DisableClientLPTPortMapping                                  | Die LPT-Portumleitung ist in XenDesktop 7.x veraltet.                     |
| Clientgeräte\Ressourcen\USB<br>USB                                                    | RemoteUSBDevices__DisableRemoteUSBDevices                    | ICA\USB-Geräte<br>Client-USB-Geräteumleitung                              |
| Drucken\Clientdrucker<br>Automatische Erstellung                                      | ConnectClientPrinterAtLogon__Flag                            | ICA\Drucken\Clientdrucker<br>Clientdrucker automatisch erstellen          |
| Drucken\Clientdrucker<br>Legacy-Clientdrucker                                         | LegacyClientPrinters__TurnOn                                 | ICA\Drucken\Clientdrucker<br>Clientdruckernamen                           |
| Drucken\Clientdrucker<br>Speicherung von Druckereigenschaften                         | ModifiedPrinterProperties__WriteMethod                       | ICA\Drucken\Clientdrucker<br>Speicherung von Druckereigenschaften         |
| Drucken\Clientdrucker<br>Weiterleiten von Druckaufträgen                              | ClientPrintingForNetworkPrinter__TurnOn                      | ICA\Drucken\Clientdrucker<br>Direkte Verbindungen zu Druckservern         |
| Drucken\Clientdrucker<br>Clientdruckerzuordnung deaktivieren                          | DisableClientPrinterMapping                                  | ICA\Drucken<br>Clientdruckerumleitung                                     |

|                                                                                                                |                                                                                  |                                                                                                                                    |
|----------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------|
| Drucken\Treiber<br>Kategorie und Einstellung in XenDesktop 4<br>Native Druckertreiber automatisch installieren | PrintDriverAutoInstall__TurnOn<br>XML-Datei                                      | ICA\Drucken\Treiber<br>Kategorie und Einstellung in XenDesktop 7.x<br>Automatische Installation von mitgelieferten Druckertreibern |
| Drucken\Treiber<br>Universeller Treiber                                                                        | ClientPrintDriverToUse                                                           | ICA\Drucken\Treiber<br>Verwendung universeller Druckertreiber                                                                      |
| Drucken\Sitzungsdrucker<br>Sitzungsdrucker                                                                     | NetworkPrinters                                                                  | ICA\Drucken<br>Sitzungsdrucker                                                                                                     |
| Drucken\Sitzungsdrucker<br>Standarddrucker des Clients wählen                                                  | DefaultToMainClientPrinter__NetworkDefault<br>DefaultToMainClientPrinter__TurnOn | ICA\Drucken<br>Standarddrucker                                                                                                     |

## Ausgeschlossene Elemente

Nicht alle XenDesktop 4-Komponenten werden in diesem Release unterstützt. Die folgenden Elemente werden nicht migriert:

- **Virtual Delivery Agent:** Bevor Sie mit einem XenDesktop 7.x-Delivery Controller virtuelle Desktops von XenDesktop 4 verwalten können, müssen Sie ein Upgrade der VDAs auf mindestens XenDesktop 5.x durchführen. Weitere Informationen zum Upgrade der VDAs finden Sie unter [Aufgaben nach der Migration](#).
- **Controller:** Sie müssen neue Controller-Server bereitstellen. Sie können einen XenDesktop 4-Controller nicht auf eine XenDesktop 7.x-Site aktualisieren. XenDesktop 7.x-Sites können keiner XenDesktop 4-Farm und XenDesktop 4-Controller können keiner XenDesktop 7.x-Site beitreten. Darüber hinaus gelten für jede Version andere Serveranforderungen. XenDesktop 4 erfordert Windows Server 2003 und XenDesktop 7.x neuere Windows Server-Versionen.
- **Webinterface:** Citrix empfiehlt die Verwendung von StoreFront mit XenDesktop 7.x Informationen zu Installation und Einrichtung finden Sie in der StoreFront-Dokumentation. Wenn das XenDesktop-Installationsprogramm das Webinterface erkennt, wird StoreFront zwar installiert, das Webinterface jedoch nicht entfernt.
- **Konfiguration der Organisationseinheiten in Active Directory:** Die Freigabe einer Organisationseinheit zwischen zwei Farmen oder zwei Sites oder einer Farm und einer Site wird nicht unterstützt. Wenn Sie die neue Site mit der Active Directory-basierten Controller-Discovery statt der standardmäßigen registrierungsbasierten Controller-Discovery konfigurieren, müssen Sie eine neue Organisationseinheit hierfür erstellen.
- **XML-Datei PortICAConfig:** Wenn Sie die Standardeinstellungen für diese Datei geändert haben, müssen Sie diese Einstellungen für die neue Site unter Umständen über Gruppenrichtlinienobjekte konfigurieren.
- **Über XenDesktop 4 Service Pack 1 bereitgestellte Einstellungen für die Konfigurationsprotokollierung:**
  - **Daten von Provisioning Services:**
  - **Anwendungen:**
  - **Liste der Controller:**
  - **NetScaler Gateway:**
  - **Drosselungseinstellungen für das Ereignisprotokoll:**

# Sicherheit

May 10, 2016



## Sicherheit mit Citrix XenApp und XenDesktop: Erste Schritte

XenApp und XenDesktop bieten eine auf Sicherheit ausgelegte Lösung, mit der Sie Ihre Umgebung Ihren Sicherheitsanforderungen anpassen können.

Bei mobilen Mitarbeitern steht die IT-Abteilung dem Sicherheitsrisiko durch verlorene oder gestohlene Daten gegenüber. Durch Hosten von Anwendungen und Desktops trennen XenApp und XenDesktop sicher vertrauliche Daten und geistiges Eigentum von Endpunktgeräten, da alle Daten in einem Datacenter gespeichert werden. Wenn Richtlinien für das Zulassen von Datenübertragungen aktiviert sind, werden alle Daten verschlüsselt.

Die XenDesktop- und XenApp-Datacenter vereinfachen auch die Reaktion auf Vorfälle mit einem zentralisierten Überwachungs- und Verwaltungsdienst. Mit Director überwachen und analysieren IT-Mitarbeiter Daten, auf die im Netzwerk zugegriffen wird, und mit Studio kann die IT-Abteilung im Datacenter Patches anwenden und Systemanfälligkeiten verhindern statt Probleme lokal auf jedem Endbenutzergerät zu beheben.

XenApp und XenDesktop vereinfachen auch Audits und die Einhaltung der Richtlinienreue, da Untersuchende mit einer zentralisierten Überwachungsliste ermitteln können, wer auf welche Anwendungen und Daten zugegriffen hat. Director sammelt durch Zugriff auf die Konfigurationsprotokollierung und die OData-API Verlaufsdaten über Updates des Systems und der Benutzerdatennutzung.

Mit der delegierten Administration richten Sie Administratorrollen ein, um den Zugriff auf XenDesktop und XenApp auf granulärer Ebene zu steuern. Dies ermöglicht Flexibilität in Ihrer Organisation, um bestimmten Administratoren vollständigen Zugriff auf Aufgaben, Vorgänge und Geltungsbereiche zugeben, während der Zugriff anderer Administratoren beschränkt ist.

Mit XenApp und XenDesktop wenden Administratoren Richtlinien auf verschiedenen Netzwerkebenen an, von der lokalen Ebene bis zur Organisationseinheitsebene, und steuern damit Benutzer granulär. Diese Steuerung der Richtlinien legt fest, ob ein Benutzer, ein Gerät oder eine Gruppe von Benutzern und Geräten eine Verbindung herstellen, Kopieren bzw. Einfügen oder lokale Laufwerke zuordnen können. Dies verringert Sicherheitsbedenken bei Zeitpersonal von Drittanbietern. Administratoren können auch Desktop Lock verwenden, sodass Benutzer nur den virtuellen Desktop verwenden können und der Zugriff auf das lokale Betriebssystem des Endbenutzergeräts verhindert wird.

Administratoren können die Sicherheit in XenApp oder XenDesktop erhöhen und die Site so konfigurieren, dass sie das SSL-Sicherheitsprotokoll (Secure Sockets Layer) des Controllers oder zwischen Endbenutzern und VDAs verwendet. Das TLS-Sicherheitsprotokoll (Transport Layer Security) kann auch für eine Site aktiviert werden, um die Serverauthentifizierung, die Verschlüsselung des Datenstroms und die Prüfung der Nachrichtenintegrität für eine TCP/IP-Verbindungen bereitzustellen.

XenApp und XenDesktop unterstützen auch die mehrstufige Authentifizierung für Windows oder eine bestimmte Anwendung. Mit der mehrstufigen Authentifizierung können auch alle Ressourcen, die von XenApp und XenDesktop bereitgestellt werden, verwaltet werden. Diese Methoden sind u. a.:

- Tokens
- Smartcards
- RADIUS
- Kerberos

- Biometrie

XenDesktop kann mit vielen Sicherheitslösungen von Drittanbietern integriert werden, von der Identitätsverwaltung bis zu Antivirensoftware. Eine Liste der unterstützten Produkte finden Sie unter <http://www.citrix.com/ready>.

Bestimmte Releases von XenApp und XenDesktop sind für Common Criteria zertifiziert. Eine Liste dieser Standards finden Sie unter <http://www.commoncriteriaportal.org/cc/>.

#### Verwandter Inhalt

- [Best Practices und Überlegungen zur Sicherheit](#)
- [Delegierte Administration](#)
- [Smartcards](#)
- [SSL](#)
- [Desktop Lock](#)

# Best Practices und Überlegungen zur Sicherheit

Nov 02, 2016

In diesem Dokument wird Folgendes beschrieben:

- Allgemeine bewährte Methoden zur Gewährleistung der Sicherheit beim Verwenden dieses Releases sowie sicherheitsrelevante Unterschiede zwischen diesem Release und einer konventionellen Computerumgebung
- Verwalten von Benutzerkonten
- Verwalten von Benutzerprivilegien
- Verwalten von Anmelderechten
- Konfigurieren von Benutzerrechten
- Konfigurieren von Dienstinstellungen
- Bereitstellungsszenarios und ihre Auswirkungen auf die Sicherheit
- Sicherheitsüberlegungen für Remote-PC-Zugriff

Möglicherweise muss Ihre Organisation bestimmte Sicherheitsstandards einhalten, um den gesetzlichen Anforderungen zu genügen. In diesem Dokument wird dieses Thema nicht behandelt, da sich Sicherheitsstandards mit der Zeit ändern. Aktuelle Informationen über Sicherheitsstandards und Citrix Produkte finden Sie unter <http://www.citrix.com/security/>.

## Bewährte Methoden zur Gewährleistung der Sicherheit

Halten Sie stets alle Computer in der Umgebung mit Sicherheitspatches auf dem neuesten Stand. Ein Vorteil besteht darin, dass Thin Clients als Terminals verwendet werden können. Das erleichtert diese Aufgabe.

Schützen Sie alle Maschinen in der Umgebung mit Antivirensoftware.

Schützen Sie alle Maschinen in der Umgebung mit Umkreisfirewalls, u. a. bei Bedarf auch an Grenzen von Enklaven.

Wenn Sie eine konventionelle Umgebung zu diesem Release migrieren, müssen Sie ggf. eine vorhandene Umkreisfirewall neu positionieren oder neue Umkreisfirewalls hinzufügen. Beispiel: Zwischen einem konventionellen Client und einem Datenbankserver im Datenzentrum befindet sich eine Umkreisfirewall. Bei diesem Release muss diese Umkreisfirewall aber so platziert werden, dass sich der virtuelle Desktop und das Benutzergerät auf der einen Seite befinden und die Datenbankserver und Controller im Datacenter auf der anderen Seite. Sie sollten deshalb erwägen, im Datacenter einen Netzbereich für die von XenDesktop verwendeten Datenbankserver und Controller zu erstellen. Sie sollten zudem in Betracht ziehen, einen Schutz zwischen dem Benutzergerät und dem virtuellen Desktop zu installieren.

Alle Maschinen in der Umgebung müssen durch eine persönliche Firewall geschützt werden. Wenn Sie Kernkomponenten und Virtual Delivery Agents (VDAs) installieren, können Sie die erforderlichen Ports für Komponenten und Features so einrichten, dass sie automatisch geöffnet werden, sobald der Windows-Firewalldienst erkannt wird (auch wenn die Firewall nicht aktiviert ist). Sie können die Firewallports auch manuell konfigurieren. Wenn Sie eine andere Firewall verwenden, muss diese manuell konfiguriert werden.

Hinweis: Da die TCP-Ports 1494 und 2598 für ICA und CGP verwendet werden, sind sie normalerweise an der Firewall geöffnet, damit Benutzer außerhalb des Rechenzentrums auf sie zugreifen können. Citrix empfiehlt, dass diese Ports nicht für etwas Anderes verwendet werden, damit administrative Benutzeroberflächen nicht versehentlich gefährdet werden. Die Ports 1494 und 2598 sind bei der Internet Assigned Number Authority (siehe <http://www.iana.org/>) offiziell registriert. Die gesamte Netzwerkkommunikation sollte Ihren Sicherheitsrichtlinien gemäß angemessen gesichert und verschlüsselt werden. Sie können die gesamte Kommunikation zwischen Microsoft Windows-Computern mit IPSec sichern. Weitere Informationen hierzu finden Sie in der Dokumentation zum Betriebssystem. Die Kommunikation zwischen Benutzergeräten

und Desktops ist außerdem mit Citrix SecureICA gesichert, das in der Standardeinstellung 128-Bit-Verschlüsselung verwendet. Sie können beim Erstellen oder Aktualisieren einer Zuweisung SecureICA konfigurieren (siehe [Ändern von Grundeinstellungen](#)).

## Verwalten von Benutzerkonten

Wenn Sie bei der Installation eines Virtual Delivery Agents (VDA) die Option zum Installieren von App-V-Veröffentlichungskomponenten wählen oder diese Komponenten später installieren, wird dem VDA das lokale Administratorkonto CtxAppVCOMAdmin hinzugefügt. Wenn Sie die App-V-Veröffentlichung verwenden, ändern Sie dieses Konto nicht. Wenn Sie die App-V-Veröffentlichung nicht benötigen, aktivieren Sie das Feature bei der Installation nicht. Wenn Sie sich nach der Installation der App-V-Veröffentlichung gegen eine Verwendung des Features entscheiden, können Sie das Konto deaktivieren oder löschen.

### Verwalten von Benutzerprivilegien

Geben Sie Benutzern nur die Rechte, die sie benötigen. Microsoft Windows-Privilegien können weiterhin in der üblichen Weise auf Desktops angewendet werden: Konfigurieren Sie Privilegien mit "Zuweisung von Benutzerrechten" und Gruppenmitgliedschaften mit einer Gruppenrichtlinie. Der Vorteil dieses Release besteht darin, dass einem Benutzer Administratorrechte für einen Desktop eingeräumt werden können, ohne ihm auch die physische Kontrolle über den Computer, auf dem der Desktop gespeichert ist, zu gewähren.

Beachten Sie beim Planen von Desktopprivilegien Folgendes:

- Standardmäßig wird nicht berechtigten Benutzern beim Herstellen einer Verbindung mit einem Desktop die Zeitzone des Systems, auf dem der Desktop ausgeführt wird, statt der Zeitzone ihres eigenen Benutzergerätes angezeigt. Weitere Informationen dazu, wie Sie Benutzern erlauben, ihre Ortszeit beim Verwenden von Desktops anzuzeigen, finden Sie unter [Ändern von Grundeinstellungen](#).
- Ein Benutzer mit Administratorrechten auf einem Desktop hat Vollzugriff auf diesen Desktop. Wenn ein Desktop ein gepoolter Desktop und kein dedizierter Desktop ist, muss dem Benutzer von allen anderen Benutzern dieses Desktops, einschließlich zukünftiger Benutzer, vertraut werden. Alle Benutzer des Desktops müssen sich des potenziellen permanenten Risikos für ihre Datensicherheit bewusst sein, die diese Situation mit sich bringt. Diese Überlegung trifft nicht auf dedizierte Desktops zu, die nur einen einzelnen Benutzer haben. Dieser Benutzer sollte kein Administrator auf einem anderen Desktop sein.
- Ein Benutzer mit Administratorrechten auf einem Desktop kann auf diesem Desktop generell Software installieren, einschließlich potenziell schädlicher Software. Zudem kann der Benutzer u. U. den Datenverkehr in allen mit dem Desktop verbundenen Netzwerken überwachen und steuern.

## Verwalten von Anmelderechten

Anmelderechte sind für Benutzerkonten und Computerkonten erforderlich. Wie Microsoft Windows-Privilegien werden Anmelderechte weiterhin in der üblichen Weise auf Desktops angewendet: Konfigurieren Sie Anmelderechte mit "Zuweisung von Benutzerrechten" und Gruppenmitgliedschaften mit einer Gruppenrichtlinie.

Es gibt folgende Windows-Anmelderechte: Lokal anmelden, Anmelden über Remotedesktopdienste, über das Netzwerk ("Auf diesen Computer vom Netzwerk aus zugreifen"), Anmelden als Stapelverarbeitungsauftrag und Anmelden als Dienst.

Erteilen Sie Computerkonten nur die Anmelderechte, die diese benötigen. Die Berechtigung "Auf diesen Computer vom

Netzwerk aus zugreifen" ist erforderlich:

- Auf VDAs für die Computerkonten der Delivery Controller
- Auf Delivery Controllern für die Computerkonten der VDAs. Siehe hierzu den Artikel [Auf Organisationseinheiten von Active Directory-basierte Controller-Discovery](#).
- Auf StoreFront-Servern für die Computerkonten der anderen Server in der gleichen StoreFront-Servergruppe

Erteilen Sie Benutzerkonten nur die Anmelderechte, die diese benötigen.

Laut Microsoft wird der Gruppe Remotedesktopbenutzer standardmäßig das Anmelderecht "Anmelden über Remotedesktopdienste" gewährt (außer für Domänencontroller).

Die Sicherheitsrichtlinie Ihres Unternehmens legt möglicherweise explizit fest, dass diese Gruppe aus dem Anmelderecht entfernt werden sollte. Erwägen Sie folgenden Ansatz:

- Der Virtual Delivery Agent (VDA) für Serverbetriebssysteme verwendet Microsoft-Remotedesktopdienste. Sie können die Gruppe der Remotedesktopbenutzer als eine eingeschränkte Gruppe konfigurieren und die Gruppenmitgliedschaft durch Active Directory-Gruppenrichtlinien steuern. Weitere Informationen finden Sie in der Dokumentation von Microsoft.
- Für andere XenApp- und XenDesktop-Komponenten, einschließlich dem VDA für Desktopbetriebssysteme, ist die Gruppe der Remotedesktopbenutzer nicht erforderlich. Für diese Komponenten benötigt die Gruppe der Remotedesktopbenutzer das Recht "Anmelden über Remotedesktopdienste" also nicht und Sie können es entfernen. Beachten Sie außerdem Folgendes:
  - Wenn Sie diese Computer mit Remotedesktopdienste verwalten, stellen Sie sicher, dass alle Administratoren Mitglieder der Administratorgruppe sind.
  - Wenn Sie diese Computer nicht mit Remotedesktopdienste verwalten, könnten Sie Remotedesktopdienste auf diesen Computern deaktivieren.

Es ist zwar möglich, dem Anmelderecht "Anmelden über Remotedesktopdienste verweigern" Benutzer und Gruppen hinzuzufügen, jedoch wird von der Verwendung von verweigernden Rechten allgemein abgeraten. Weitere Informationen finden Sie in der Dokumentation von Microsoft.

## Konfigurieren von Benutzerrechten

Bei der Installation des Delivery Controllers werden die folgenden Windows-Dienste erstellt:

- Citrix AD-Identitätsdienst (NT SERVICE\CitrixADIdentityService): Verwaltet Microsoft Active Directory-Computerkonten für VMs.
- Citrix Analytics (NT SERVICE\CitrixAnalytics): Sammelt Sitekonfigurations- und Nutzungsinformationen zur Verwendung von Citrix, wenn das Sammeln vom Siteadministrator genehmigt wurde. Diese Informationen werden dann an Citrix gesendet, damit das Produkt verbessert werden kann.
- Citrix App-Bibliothek (NT SERVICE\CitrixAppLibrary): Unterstützt die Verwaltung und das Provisioning von AppDisks, AppDNA-Integration und die Verwaltung von App-V.
- Citrix Brokerdienst (NT SERVICE\CitrixBrokerService): Wählt die virtuellen Desktops oder Anwendungen aus, die den Benutzern zur Verfügung stehen.
- Citrix Konfigurationsprotokollierungsdienst (NT SERVICE\CitrixConfigurationLogging): Erfasst alle Konfigurationsänderungen und andere Zustandsänderungen, die von den Administratoren an der Site vorgenommen werden.
- Citrix Konfigurationsdienst (NT SERVICE\CitrixConfigurationService): Repository der Site für freigegebene

Konfigurationen.

- Citrix Dienst für die delegierte Administration (NT SERVICE\CitrixDelegatedAdmin): Verwaltet die Berechtigungen, die Administratoren gewährt werden.
- Citrix Umgebungstestdienst (NT SERVICE\CitrixEnvTest): Verwaltet Selbsttests der anderen Delivery Controller-Dienste.
- Citrix Hostdienst (NT SERVICE\CitrixHostService): Speichert Informationen zu den Hypervisorinfrastrukturen, die in einer XenApp- oder XenDesktop-Bereitstellung verwendet werden, und bietet der Konsole die Funktionalität zum Enumerieren von Ressourcen in einem Hypervisorpool.
- Citrix Maschinenerstellungsdienste (NT SERVICE\CitrixMachineCreationService): Orchestriert das Erstellen von Desktop-VMs.
- Citrix Überwachungsdienst (NT SERVICE\CitrixMonitor): Sammelt Metrik für XenApp oder XenDesktop, speichert historische Informationen und bietet eine Abfrageschnittstelle für Problembehandlungs- und Berichterstattungstools.
- Citrix StoreFront-Dienst (NT SERVICE\CitrixStorefront): Unterstützt die Verwaltung von StoreFront. (Der Dienst selbst gehört nicht zur StoreFront-Komponente.)
- Citrix StoreFront-Dienst für die privilegierte Administration (NT SERVICE\CitrixPrivilegedService): Unterstützt privilegierte Verwaltungsvorgänge von StoreFront. (Der Dienst selbst gehört nicht zur StoreFront-Komponente.)

Bei der Installation des Delivery Controllers werden zudem die folgenden Windows-Dienste erstellt: Diese werden auch erstellt, wenn sie mit anderen Citrix Komponenten installiert werden:

- Citrix Diagnostic Facility COM-Server (NT SERVICE\CdfSvc): Unterstützt das Sammeln von Diagnoseinformationen für den Citrix Support.
- Citrix Telemetriedienst (NT SERVICE\CitrixTelemetryService): Sammelt Diagnoseinformationen zur Analyse durch Citrix. Die Analyseergebnisse und Empfehlungen können von Administratoren angezeigt werden, um die Diagnose von Problemen mit der Site zu erleichtern.

Abgesehen vom Citrix StoreFront-Dienst für die privilegierte Administration werden diesen Diensten die Anmeldeberechtigung "Anmelden als Dienst" und die Privilegien "Anpassen von Speicherkontingenten für einen Prozess", "Generieren von Sicherheitsüberwachungen" und "Ersetzen eines Tokens auf Prozessebene" zugewiesen. Sie brauchen die Benutzerrechte nicht zu ändern. Diese Privilegien werden vom Delivery Controller nicht verwendet und werden automatisch deaktiviert.

## Konfigurieren von Diensteinstellungen

Mit Ausnahme des Citrix StoreFront-Diensts für die privilegierte Administration und des Citrix Telemetriediensts melden sich die oben im Abschnitt "Konfigurieren von Benutzerrechten" aufgeführten Windows-Dienste des Delivery Controllers als NETWORK SERVICE an. Ändern Sie diese Diensteinstellungen nicht.

Der Citrix StoreFront-Dienst für die privilegierte Administration meldet sich als lokales System an (NT AUTHORITY\SYSTEM). Dies ist für StoreFront-Vorgänge des Delivery Controllers erforderlich, die normalerweise nicht für Dienste verfügbar sind (einschließlich das Erstellen von Microsoft IIS-Sites). Ändern Sie die Diensteinstellungen nicht.

Der Citrix Telemetriedienst meldet sich als seine eigene dienstspezifische Identität an.

Sie können den Citrix Telemetriedienst deaktivieren. Abgesehen von diesem Dienst und Diensten, die bereits deaktiviert sind, deaktivieren Sie keine der anderen Windows-Dienste für Delivery Controller.

Auswirkungen von Bereitstellungsszenarios auf die Sicherheit



Ihre Benutzerumgebung kann aus Benutzergeräten bestehen, die von Ihrer Organisation nicht verwaltet werden und dem Vollzugriff der jeweiligen Benutzer unterliegen. Sie kann aber auch aus Benutzergeräten bestehen, die von Ihrer Organisation verwaltet werden. Die Sicherheitsüberlegungen für diese beiden Umgebungen sind generell unterschiedlich.

- **Verwaltete Benutzergeräte** unterliegen einer administrativen Steuerung. Sie werden entweder von Ihnen gesteuert oder von einer anderen Organisation, der Sie vertrauen. Sie können Benutzergeräte konfigurieren und Benutzern direkt bereitstellen. Alternativ können Sie Terminals bereitstellen, auf denen ein einzelner Desktop im Vollbildmodus ausgeführt wird. Sie sollten die oben beschriebenen allgemeinen bewährten Sicherheitsmethoden für alle verwalteten Benutzergeräte anwenden. Dieses Release bietet den Vorteil, dass nur ganz wenig Software auf einem Benutzergerät erforderlich ist.

Ein verwaltetes Benutzergerät kann für die Verwendung im Vollbildmodus oder im Fenstermodus eingerichtet werden.

- Wenn ein Benutzergerät für die Verwendung im Vollbildmodus konfiguriert ist, können Benutzer sich mit dem üblichen Bildschirm "Anmelden an Windows" anmelden. Dieselben Anmeldeinformationen des Benutzers werden dann zum automatischen Anmelden für dieses Release verwendet.
- Wenn Benutzergeräte so konfiguriert sind, dass Benutzern ihre Desktops in einem Fenster angezeigt werden, melden sich die Benutzer zunächst beim Benutzergerät an. Anschließend melden sie sich über die in diesem Release bereitgestellte Website bei diesem Release an.
- **Nicht verwaltete Benutzergeräte:** Wenn Benutzergeräte nicht von einer vertrauenswürdigen Organisation verwaltet werden, kann nicht von einer administrativen Steuerung ausgegangen werden. Beispiel: Sie erlauben Benutzern, sich ihre eigenen Geräte zu besorgen und sie zu konfigurieren, doch die Benutzer halten sich u. U. nicht an die oben beschriebenen generellen optimalen Sicherheitsverfahren. Dieses Release hat den Vorteil, nicht verwalteten Benutzergeräten Desktops sicher bereitstellen zu können. Diese Geräte sollten jedoch einen grundlegenden Antivirenschutz haben, um Keylogger und ähnliche Angriffe auf Benutzereingaben abzuwehren.
- **Überlegungen zur Datenspeicherung:** Bei diesem Release können Sie verhindern, dass Benutzer Daten auf Benutzergeräten speichern, die sie selbst physisch steuern können. Sie müssen dennoch bedenken, welche Auswirkungen es haben kann, wenn Benutzer Daten auf Desktops speichern. Im Allgemeinen sollten Benutzer keine Daten auf Desktops speichern. Daten sollten an einem Ort gespeichert werden, an dem sie entsprechend geschützt werden können, wie z. B. auf Dateiservern, Datenbankservern oder in anderen Repositories.

Möglicherweise enthält Ihre Desktopumgebung verschiedene Desktoptypen, wie gepoolte und dedizierte Desktops.

- Benutzer sollten zu keiner Zeit Daten auf Desktops speichern, die für andere Benutzer freigegeben sind, wie z. B. gepoolte Desktops.
- Wenn Benutzer Daten auf dedizierten Desktops speichern, sollten diese Daten entfernt werden, wenn der Desktop zu einem späteren Zeitpunkt anderen Benutzern zugänglich gemacht wird.
- **Umgebungen mit mehreren Versionen** Umgebungen mit mehreren Versionen sind während einiger Upgrades unvermeidbar. Folgen Sie bewährten Methoden und minimieren Sie die Zeitdauer, während der unterschiedliche Versionen von Citrix Komponenten koexistieren.

In Umgebungen mit mehreren Versionen wird beispielsweise die Sicherheitsrichtlinie nicht gleichförmig durchgesetzt.

**Hinweis:** Dies ist typisch für andere Softwareprodukte. Bei Verwendung einer älteren Version von Active Directory wird die Gruppenrichtlinie bei neueren Windows-Versionen nur teilweise durchgesetzt.

Nachfolgend wird eine spezifische Citrix Umgebung mit mehreren Versionen beschrieben, bei der ein Sicherheitsproblem auftreten kann. Wenn Citrix Receiver 1.7 zum Herstellen einer Verbindung mit einem virtuellen Desktop verwendet wird, auf dem der Virtual Delivery Agent in XenApp und XenDesktop 7.6 Feature Pack 2 ausgeführt werden, ist die Richtlinie

"Dateiübertragungen zwischen Desktop und Client zulassen" für die Site aktiviert, kann jedoch nicht von einem Delivery Controller deaktiviert werden, auf dem XenApp und XenDesktop 7.1 ausgeführt werden. Die Richtlinie, die erst in der neueren Version des Produkts hinzugefügt wurde, wird nicht erkannt. Die Richtlinie ermöglicht Benutzern das Hochladen und Herunterladen von Dateien zum/vom virtuellen Desktop und repräsentiert damit ein Sicherheitsproblem. Zur Problemumgehung aktualisieren Sie den Delivery Controller bzw. eine eigenständige Instanz von Studio auf Version 7.6 Feature Pack 2 und deaktivieren Sie die Richtlinie dann mit der Gruppenrichtlinie. Alternativ verwenden Sie die lokale Richtlinie auf allen betroffenen virtuellen Desktops.

## Remote-PC-Zugriff

Mit Remote-PC-Zugriff werden die folgenden Sicherheitsfeatures implementiert:

- Die Verwendung von Smartcards wird unterstützt.
- Bei Verbindung einer Remotesitzung wird der Monitor des Büro-PCs leer angezeigt.
- Remote-PC-Zugriff leitet alle Tastatur- und Mauseingaben in die Remotesitzung um, ausgenommen Strg + Alt + Entf, USB-aktivierte Smartcards und biometrische Geräte.
- SmoothRoaming wird nur für einen einzelnen Benutzer unterstützt.
- Wenn ein Benutzer über eine Remotesitzung mit einem Büro-PC verbunden ist, kann nur dieser Benutzer den lokalen Zugriff auf den Büro-PC wiederaufnehmen. Zum Wiederaufnehmen des lokalen Zugriffs muss der Benutzer Strg-Alt-Entf auf dem lokalen PC drücken und sich dann mit denselben Anmeldeinformationen wie für die Remotesitzung anmelden. Er kann zudem auch über eine Smartcard oder biometrische Geräte wieder lokal zugreifen, wenn das System die entsprechende Anmeldeinformationsanbieter-Integration besitzt.

Das Standardverhalten kann über die schnelle Benutzerumschaltung über Gruppenrichtlinienobjekte oder durch Bearbeiten der Registrierung außer Kraft gesetzt werden.

- Standardmäßig unterstützt Remote-PC-Zugriff die automatische Zuweisung von mehreren Benutzern zu einem VDA. Unter XenDesktop 5.6 Feature Pack 1 konnten Administratoren dieses Verhalten mit dem PowerShell-Skript RemotePCAccess.ps1 außer Kraft setzen. Dieses Release verwendet einen Registrierungseintrag, mit dem mehrere automatische Remote-PC-Zuweisungen zugelassen oder abgelehnt werden; diese Einstellung gilt für die gesamte Site. Achtung: Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

Beschränken der automatischen Zuweisung auf einen einzelnen Benutzer:

1. Legen Sie den folgenden Registrierungseintrag auf jedem Controller in der Site fest:

HKEY\_LOCAL\_MACHINE\Software\Citrix\DesktopServer

Name: AllowMultipleRemotePCAssignments

Typ: REG\_DWORD

Wert: 0 = Zuweisung mehrerer Benutzer deaktivieren, 1 = (Standard) Zuweisung mehrerer Benutzer aktivieren.

2. Liegen bereits Benutzerzuweisungen vor, entfernen Sie diese mit SDK-Befehlen, damit der VDA anschließend für eine einzelne automatische Zuweisung zur Verfügung steht.
  1. Entfernen Sie alle zugewiesenen Benutzer aus dem VDA: `$machine.AssociatedUserNames | %{ Remove-BrokerUser-Name $_ -Machine $machine }`
  2. Entfernen Sie den VDA aus der Bereitstellungsgruppe: `$machine | Remove-BrokerMachine -DesktopGroup $desktopGroup`

3. Starten Sie den physischen Büro-PC neu.

# Delegierte Administration

May 10, 2016

Das Modell der delegierten Administration bietet Flexibilität bei der Delegierung der Administratoraktivitäten mit Rollen und der objektbasierten Steuerung. Die delegierte Administration ist für Bereitstellungen aller Größen geeignet und ermöglicht es Ihnen, mit zunehmender Komplexität der Bereitstellung die Berechtigungsgranularität zu erhöhen. Bei der delegierten Administration werden drei Konzepte eingesetzt: Administratoren, Rollen und Geltungsbereiche.

- **Administratoren:** Administratoren sind Einzelpersonen oder Personengruppen, die durch ein Active Directory-Konto identifiziert werden. Jeder Administrator ist mit mindestens einem Paar aus Rolle und Geltungsbereich verknüpft.
- **Rollen:** Eine Rolle repräsentiert eine Stellenfunktion und ist mit definierten Berechtigungen ausgestattet. Beispiel: Die Rolle "Bereitstellungsgruppenadministrator" verfügt über Berechtigungen wie etwa "Bereitstellungsgruppe erstellen" und "Desktop aus Bereitstellungsgruppe entfernen". Ein Administrator kann mehrere Rollen für eine Site haben, d. h. eine Person kann sowohl Bereitstellungsgruppenadministrator als auch Maschinenkatalogadministrator sein. Rollen können integriert oder benutzerdefiniert sein.

Integrierte Rollen:

| Rolle                               | Berechtigungen                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Volladministrator                   | Kann alle Aufgaben und Vorgänge ausführen. Ein Volladministrator wird immer mit dem Geltungsbereich "Alle" kombiniert.                                                                                                                                                                                                                                                                                                                                                                                                      |
| Lesezugriffadministrator            | Kann alle Objekte in den angegebenen Geltungsbereichen sowie globale Informationen anzeigen, aber nicht ändern. Beispiel: Ein Lesezugriffadministrator mit Geltungsbereich = London kann alle globalen Objekte (z. B. Konfigurationsprotokollierung) und alle London-bezogenen Geltungsbereichsobjekte (z. B. London-Bereitstellungsgruppen) sehen. Dieser Administrator kann jedoch nicht die Objekte im Geltungsbereich "New York" sehen (sofern die Geltungsbereiche "London" und "New York" einander nicht überlappen). |
| Helpdeskadministrator               | Kann Bereitstellungsgruppen anzeigen und die diesen Gruppen zugeordneten Sitzungen und Maschinen verwalten. Kann den Maschinenkatalog und die Hostinformationen der überwachten Bereitstellungsgruppen sehen sowie Sitzungsverwaltungs- und Energieverwaltungsvorgänge für die Maschinen in diesen Bereitstellungsgruppen durchführen.                                                                                                                                                                                      |
| Maschinenkatalogadministrator       | Kann Maschinenkataloge erstellen und verwalten sowie darin Maschinen bereitstellen. Kann Maschinenkataloge aus der Virtualisierungsinfrastruktur, Provisioning Services und von physischen Maschinen anlegen. Mit dieser Rolle können Basisimages verwaltet und Software installiert werden, aber den Benutzern können keine Anwendungen oder Desktops zugewiesen werden.                                                                                                                                                   |
| Bereitstellungsgruppenadministrator | Kann Anwendungen, Desktops und Maschinen bereitstellen sowie die mit ihnen verbundenen Sitzungen verwalten. Kann zudem Anwendungs- und Desktopkonfigurationen wie Richtlinien und die Energieverwaltungseinstellungen verwalten.                                                                                                                                                                                                                                                                                            |
| Hostadministrator                   | Kann Hostverbindungen und ihnen zugeordnete Ressourceneinstellungen verwalten. Kann keine Maschinen, Anwendungen oder Desktops für Benutzer bereitstellen.                                                                                                                                                                                                                                                                                                                                                                  |

Bei bestimmten Produkteditionen können Sie benutzerdefinierte Rollen erstellen, um sie den Anforderungen Ihrer Organisation anzupassen und die Berechtigungen entsprechend delegieren. Sie können benutzerdefinierte Rollen dazu verwenden, Berechtigungen in der Granularität einer Aktion oder Aufgabe in einer Konsole zuzuteilen.

- **Geltungsbereiche:** Ein Geltungsbereich steht für eine Sammlung von Objekten. Geltungsbereiche werden verwendet, um die Objekte in einer für Ihre Organisation angemessenen Weise zu gruppieren (z. B. die Bereitstellungsgruppen der Vertriebsabteilung). Objekte können in mehreren Geltungsbereichen vertreten sein, d. h. Objekte können durch einen oder mehrere Geltungsbereiche bezeichnet sein. Der einzige integrierte Geltungsbereich "Alle" enthält alle Objekte. Die Volladministratorrolle bildet immer ein Paar mit dem Geltungsbereich "Alle".

## Beispiel

Firma XYZ entscheidet sich zum Verwalten von Anwendungen und Desktops basierend auf ihrer Abteilungsstruktur (Buchhaltung, Vertrieb und Lager) und ihren Desktopbetriebssystemen (Windows 7 oder Windows 8). Der Administrator erstellt fünf Geltungsbereiche und erfasst jede Bereitstellungsgruppe in zwei Geltungsbereichen: einem Geltungsbereich für die Abteilung, in der sie verwendet werden und einem Geltungsbereich für das verwendete Betriebssystem.

Die folgenden Administratoren wurden erstellt:

| Administrator | Rollen                   | Geltungsbereiche                                                 |
|---------------|--------------------------|------------------------------------------------------------------|
| domain/fred   | Volladministrator        | Alle (Volladministratorrolle wird immer mit "Alle" ausgestattet) |
| domain/rob    | Lesezugriffadministrator | Alle                                                             |
| domain/heidi  | Lesezugriffadministrator | Alle                                                             |
|               | Helpdeskadministrator    | Vertrieb                                                         |

| Administrator         | Rollen                                                               | Geltungsbereiche |
|-----------------------|----------------------------------------------------------------------|------------------|
| domain/warehouseadmin | Helpdeskadministrator                                                | Lager            |
| domain/peter          | Bereitstellungsgruppenadministrator<br>Maschinenkatalogadministrator | Win7             |

- Fred ist Volladministrator und kann alle Elemente im System anzeigen, bearbeiten und löschen.
- Rob kann alle Objekte der Site anzeigen jedoch nicht bearbeiten oder löschen.
- Heidi kann alle Objekte anzeigen und Helpdeskaufgaben an Bereitstellungsgruppen des Geltungsbereichs "Vertrieb" durchführen. Somit kann sie die diesen Gruppen zugeordneten Sitzungen und Maschinen verwalten; sie kann allerdings keine Änderungen an der Bereitstellungsgruppe durchführen, wie Hinzufügen oder Entfernen von Maschinen.
- Jedes Mitglied der Active Directory-Sicherheitsgruppe "warehouseadmin" kann Helpdeskaufgaben für Maschinen des Geltungsbereichs "Lager" ausführen.
- Peter ist Spezialist für Windows 7 und kann alle Windows 7-Maschinenkataloge verwalten und Windows 7-Anwendungen, -Desktops und -Maschinen bereitstellen, unabhängig davon, in welchem Abteilungsgeltungsbereich sie sich befinden. Der Administrator erwägt, Peter zum Volladministrator für den Geltungsbereich "Win7" zu machen; entscheidet sich jedoch dagegen, da ein Volladministrator ebenfalls über vollständige Administratorrechte für alle Objekte verfügt, die nicht in einen Geltungsbereich fallen, z. B. "Site" und "Administrator".

So verwenden Sie die delegierte Administration

Im Allgemeinen hängt die Anzahl der Administratoren und die Granularität der Berechtigungen von der Größe und Komplexität der Bereitstellung ab.

- In kleinen Bereitstellungen oder Testbereitstellungen übernehmen ein oder wenige Administratoren alle Aufgaben und es findet keine Delegation statt. Erstellen Sie in diesem Fall einen einzelnen Administrator mit der integrierten Rolle "Volladministrator", die den Geltungsbereich "Alle" hat.
- In größeren Bereitstellungen mit mehr Maschinen, Anwendungen und Desktops ist mehr Delegation erforderlich. Mehrere Administratoren haben möglicherweise bestimmte funktionale Zuständigkeiten (Rollen). Beispiel: Es gibt zwei Volladministratoren, andere sind Helpdeskadministratoren. Weiterhin werden von einem Administrator ggf. nur bestimmte Objektgruppen (Geltungsbereiche) wie Maschinenkataloge verwaltet. Erstellen Sie in diesem Fall neue Geltungsbereiche und Administratoren mit einer der integrierten Rollen und den entsprechenden Geltungsbereichen.
- Noch größere Bereitstellungen erfordern möglicherweise weitere (oder differenziertere) Geltungsbereiche sowie andere Administratoren mit ungewöhnlichen Rollen. Bearbeiten oder erstellen Sie in diesem Fall weitere Geltungsbereiche, erstellen Sie benutzerdefinierte Rollen und erstellen Sie jeden Administrator mit einer integrierten oder benutzerdefinierten Rolle sowie vorhandenen und neuen Geltungsbereichen.

Für mehr Flexibilität und zur Vereinfachung der Konfiguration können Sie neue Geltungsbereiche erstellen, wenn Sie einen Administrator erstellen. Sie können auch beim Erstellen oder Bearbeiten von Maschinenkatalogen oder Verbindungen Geltungsbereiche festlegen.

### Erstellen und Verwalten von Administratoren

Beim Erstellen einer Site als lokaler Administrator wird dieses Benutzerkonto automatisch zum Volladministrator mit Vollzugriff auf alle Objekte. Nachdem die Site erstellt wurde, verfügen lokale Administratoren über keine besonderen Rechte.

Die Volladministratorrolle hat immer den Geltungsbereich "Alle"; dies kann nicht geändert werden.

Standardmäßig wird ein Administrator aktiviert. Beim Erstellen des neuen Administrators kann das Deaktivieren eines Administrators erforderlich sein, die betroffene Person übernimmt jedoch erst zu einem späteren Zeitpunkt Verwaltungsaufgaben. Bei vorhandenen aktivierten Administratoren kann es vorkommen, dass Sie einige deaktivieren müssen, während Sie Objekte/Geltungsbereiche neu strukturieren und sie dann wieder aktivieren, wenn Sie die Aktualisierung der Konfiguration abgeschlossen haben. Der Volladministrator kann nicht deaktiviert werden, wenn dies dazu führen würde, dass kein aktivierter Volladministrator mehr vorhanden ist. Das Kontrollkästchen zum Aktivieren/Deaktivieren steht zur Verfügung, wenn Sie einen Administrator erstellen, kopieren oder bearbeiten.

Wenn Sie ein Rollen-/Geltungsbereichspaar beim Kopieren, Bearbeiten oder Löschen eines Administrators löschen, wird nur die Beziehung zwischen Rolle und Geltungsbereich für diesen Administrator gelöscht, jedoch nicht die Rolle oder der Geltungsbereich selbst. Außerdem hat dies keine Auswirkungen auf andere Administratoren, die mit diesem Rollen-/Geltungsbereichspaar konfiguriert sind.

Klicken Sie zum Verwalten von Administratoren im Studio-Navigationsbereich auf Konfiguration > Administratoren und dann im mittleren Bereich oben auf die Registerkarte Administratoren.

- Zum Erstellen eines Administrators klicken Sie im Aktionsbereich auf Administrator erstellen. Geben Sie den Namen eines Benutzerkontos ein oder navigieren zu einem Benutzerkonto, wählen oder erstellen Sie einen Geltungsbereich und wählen Sie eine Rolle. Der neue Administrator ist standardmäßig aktiviert. Sie können dies ändern.
- Zum Kopieren eines Administrators wählen Sie diesen im mittleren Bereich aus und klicken Sie im Aktionsbereich auf Administrator kopieren. Geben Sie den Namen des Benutzerkontos ein oder navigieren zu dem Benutzerkonto. Sie können die Rollen-/Geltungsbereichspaare auswählen und dann bearbeiten oder löschen und neue hinzufügen. Der neue Administrator ist standardmäßig aktiviert. Sie können dies ändern.
- Zum Bearbeiten eines Administrators wählen Sie diesen im mittleren Bereich aus und klicken Sie im Aktionsbereich auf Administrator bearbeiten. Sie können die Rollen-/Geltungsbereichspaare bearbeiten oder löschen und neue hinzufügen.
- Zum Löschen eines Administrators wählen Sie diesen im mittleren Bereich aus und klicken Sie im Aktionsbereich auf Administrator löschen. Der Volladministrator kann nicht gelöscht werden, wenn dies dazu führen würde, dass kein aktivierter Volladministrator mehr vorhanden ist.

### Erstellen und Verwalten von Rollen

Rollenamen können maximal 64 Unicode-Zeichen enthalten. Die folgenden Zeichen sind nicht zulässig: \ (umgekehrter Schrägstrich), / (Schrägstrich), ; (Semikolon), :

(Doppelpunkt), # (Nummernzeichen), , (Komma), \* (Sternchen), ? (Fragezeichen), = (Gleichheitszeichen), < (Pfeil nach links), > (Pfeil nach rechts), | (senkrechter Strich), [ ] (eckige Klammern), () (runde Klammern), " (Anführungszeichen) und ' (Apostroph). Beschreibungen können bis zu 256 Unicode-Zeichen enthalten.

Integrierte Rollen können nicht bearbeitet oder gelöscht werden. Benutzerdefinierte Rollen können nicht gelöscht werden, wenn sie von einem Administrator verwendet werden.

Hinweis: Nur bestimmte Produkteditionen unterstützen benutzerdefinierte Rollen. Editionen, die keine benutzerdefinierten Rollen unterstützen, verfügen nicht über verwandte Einträge im Aktionsbereich.

Klicken Sie zum Verwalten von Rollen im Studio-Navigationsbereich auf Konfiguration > Administratoren und dann im unteren mittleren Bereich auf die Registerkarte Rollen.

- Zum Anzeigen von Rollendetails wählen Sie die Rolle im mittleren Bereich aus. Im unteren Teil des mittleren Bereichs werden die Objekttypen und die zugehörigen Berechtigungen für die Rolle angezeigt. Klicken Sie auf die Registerkarte Administratoren im unteren Bereich, um eine Liste der Administratoren anzuzeigen, die derzeit diese Rolle haben.
- Zum Erstellen einer benutzerdefinierten Rolle klicken Sie im Aktionsbereich auf Rolle erstellen. Geben Sie einen Namen und eine Beschreibung ein. Wählen Sie die Objekttypen und Berechtigungen aus.
- Zum Kopieren einer Rolle wählen Sie diese im mittleren Bereich aus und klicken Sie im Aktionsbereich auf Rolle kopieren. Ändern Sie den Namen und die Beschreibung sowie die Objekttypen und Berechtigungen nach Bedarf.
- Zum Bearbeiten einer Rolle wählen Sie diese im mittleren Bereich aus und klicken Sie im Aktionsbereich auf Rolle bearbeiten. Ändern Sie den Namen und die Beschreibung sowie die Objekttypen und Berechtigungen nach Bedarf.
- Zum Löschen einer Rolle wählen Sie diese im mittleren Bereich aus und klicken Sie im Aktionsbereich auf Rolle löschen. Bestätigen Sie die Löschung.

## Erstellen und Verwalten von Geltungsbereichen

Beim Erstellen einer Site steht nur der Geltungsbereich "Alle" zur Verfügung. Dieser kann nicht gelöscht werden.

Sie können Geltungsbereiche mit der unten aufgeführten Vorgehensweise erstellen. Sie können auch die Geltungsbereiche erstellen, wenn Sie einen Administrator erstellen. Jeder Administrator muss mindestens einem Rollen-/Geltungsbereichspaar zugeordnet werden. Beim Erstellen oder Bearbeiten von Desktops, Maschinenkatalogen, Anwendungen oder Hosts können Sie diese einem bestehenden Geltungsbereich hinzufügen. Wenn Sie sie keinem Geltungsbereich hinzufügen, bleiben sie Teil des Geltungsbereichs "Alle".

Die Geltungsbereichszuordnung ist bei der Erstellung von Sites und für Objekte der delegierten Administration (Geltungsbereiche und Rollen) nicht möglich. Objekte, die nicht zugeordnet werden können, gehören zum Geltungsbereich "Alle". (Volladministratoren ist immer der Geltungsbereich "Alle" zugeordnet.) Maschinen, Energieaktionen, Desktops und Sitzungen werden nicht direkt einem Geltungsbereich zugeordnet. Administratoren können Berechtigungen für diese Objekte über die zugehörigen Maschinenkataloge oder Bereitstellungsgruppen zugewiesen werden.

Geltungsbereichsnamen können maximal 64 Unicode-Zeichen enthalten. Die folgenden Zeichen sind nicht zulässig: \ (umgekehrter Schrägstrich), / (Schrägstrich), ; (Semikolon), : (Doppelpunkt), # (Nummernzeichen), , (Komma), \* (Sternchen), ? (Fragezeichen), = (Gleichheitszeichen), < (Pfeil nach links), > (Pfeil nach rechts), | (senkrechter Strich), [ ] (eckige Klammern), () (runde Klammern), " (Anführungszeichen) und ' (Apostroph). Beschreibungen können bis zu 256 Unicode-Zeichen enthalten.

Wenn Sie einen Geltungsbereich kopieren oder bearbeiten, dürfen Sie nicht vergessen, dass Objekte, die aus dem Geltungsbereich entfernt werden, für den Administrator ggf. nicht mehr zugänglich sind. Ist der bearbeitete Geltungsbereich mit einer oder mehreren Rollen gepaart, stellen Sie sicher, dass durch Änderungen an dem Bereich kein Rollen-/Geltungsbereichspaar unbrauchbar wird.

Klicken Sie zum Verwalten von Geltungsbereichen im Studio-Navigationsbereich auf Konfiguration > Administratoren und dann im mittleren Bereich oben auf die Registerkarte Geltungsbereiche.

- Zum Erstellen eines Geltungsbereichs klicken Sie im Aktionsbereich auf Geltungsbereich erstellen. Geben Sie einen Namen und eine Beschreibung ein. Zum Einschließen aller Objekte eines bestimmten Typs (z. B. Bereitstellungsgruppen), wählen Sie den Objekttyp aus. Zum Einschließen bestimmter Objekte erweitern Sie den Typ und wählen Sie die einzelnen Objekte (z. B. einzelne Bereitstellungsgruppen des Vertriebs) aus.
- Zum Kopieren eines Geltungsbereichs wählen Sie diesen im mittleren Bereich aus und klicken Sie im Aktionsbereich auf Geltungsbereich kopieren. Geben Sie einen Namen und eine Beschreibung ein. Ändern Sie die Objekttypen und Berechtigungen nach Bedarf.
- Zum Bearbeiten eines Geltungsbereichs wählen Sie diesen im mittleren Bereich aus und klicken Sie im Aktionsbereich auf Geltungsbereich bearbeiten. Ändern Sie den Namen und die Beschreibung sowie die Objekttypen und Objekte nach Bedarf.
- Zum Löschen eines Geltungsbereichs wählen Sie diesen im mittleren Bereich aus und klicken Sie im Aktionsbereich auf Geltungsbereich löschen. Bestätigen Sie die Löschung.

## Erstellen von Berichten

Sie können zwei Arten delegierter Administrationsberichte erstellen:

- Einen HTML-Bericht, der die Rollen-/Geltungsbereichspaare, die einem Administrator zugeordnet sind, sowie die einzelnen Berechtigungen für jeden Objekttyp (z. B. Bereitstellungsgruppen, und Maschinenkataloge) enthält. Sie generieren diesen Bericht in Studio.  
Zum Erstellen dieses Berichts klicken Sie im Navigationsbereich auf Konfiguration > Administratoren. Wählen Sie im mittleren Bereich einen Administrator aus, und klicken Sie dann im Aktionsbereich auf Bericht erstellen.

Sie können diesen Bericht auch beim Erstellen, Kopieren oder Bearbeiten eines Administrators anfordern.

- HTML- oder CSV Bericht, in dem alle integrierten benutzerdefinierten Rollen und Berechtigungen zugeordnet sind. Sie generieren diesen Bericht durch Ausführen des PowerShell-Skripts "OutputPermissionMapping.ps1".  
Um dieses Skript auszuführen, müssen Sie ein Volladministrator, ein Lesezugriffadministrator oder ein benutzerdefinierter Administrator mit der Berechtigung zum Lesen von Rollen sein. Das Skript befindet sich in: Programme\Citrix\DelegatedAdmin\SnapiN\Citrix.DelegatedAdmin.Admin.V1\Scripts\.

Syntax:

OutputPermissionMapping.ps1 [-Help] [-Csv] [-Path <string>] [-AdminAddress <string>] [-Show] [<CommonParameters>]

| Parameter     | Beschreibung                                                                                                                                                                                       |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -Help         | Zeigt Skripthilfe an.                                                                                                                                                                              |
| -Csv          | Gibt CSV-Ausgabe an. Standard = HTML                                                                                                                                                               |
| -Path         | Zielspeicherort für die Ausgabe. Standard = stdout                                                                                                                                                 |
| -AdminAddress | IP-Adresse oder Hostname des Delivery Controllers, mit dem eine Verbindung hergestellt wird. Standard = localhost                                                                                  |
| -Show         | Gilt nur, wenn der Parameter "-Path" ebenfalls angegeben wird. Wenn die Ausgabe in eine Datei geschrieben wird, wird sie mit -Show in einem geeigneten Programm, z. B. einem Webbrowser, geöffnet. |
|               | Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer und OutVariable. Weitere Informationen finden Sie in der Dokumentation von Microsoft.                        |

Mit dem Befehl im folgenden Beispiel wird eine HTML-Tabelle in eine Datei namens Roles.html geschrieben und die Tabelle in einem Webbrowser geöffnet.

& "\$env:ProgramFiles\Citrix\DelegatedAdmin\SnapIn\Citrix.DelegatedAdmin.Admin.V1\Scripts\OutputPermissionMapping.ps1" -Path Roles.html -Show

Mit dem Befehl im folgenden Beispiel wird eine CSV-Tabelle in eine Datei namens Roles.csv geschrieben. Die Tabelle wird nicht angezeigt.

& "\$env:ProgramFiles\Citrix\DelegatedAdmin\SnapIn\Citrix.DelegatedAdmin.Admin.V1\Scripts\OutputPermissionMapping.ps1" -CSV -Path Roles.csv

An einer Windows-Eingabeaufforderung wird der Befehl aus dem vorherigen Beispiel folgendermaßen eingegeben:

powershell -command "& '%ProgramFiles%\Citrix\DelegatedAdmin\SnapIn\Citrix.DelegatedAdmin.Admin.V1\Scripts\OutputPermissionMapping.ps1' -CSV -Path Roles.csv"

# Smartcards

Nov 02, 2016

Smartcards und ähnliche Technologien werden im Rahmen der in diesem Abschnitt beschriebenen Richtlinien unterstützt. Für die Verwendung von Smartcards mit XenApp oder XenDesktop gelten folgende Richtlinien:

- Machen Sie sich mit den Sicherheitsrichtlinien Ihrer Organisation für die Verwendung von Smartcards vertraut. Mit diesen Richtlinien wird z. B. festgelegt, wie Smartcards ausgegeben werden und wie diese von Benutzern gesichert werden sollten. Einige Aspekte dieser Richtlinien müssen ggf. in einer XenApp- bzw. XenDesktop-Umgebung neu bewertet werden.
- Legen Sie fest, welche Benutzergerätetypen, Betriebssysteme und veröffentlichten Anwendungen mit Smartcards verwendet werden dürfen.
- Machen Sie sich mit der Smartcard-Technologie und der Hardware und Software des von Ihnen gewählten Smartcardanbieters vertraut.
- Sie sollten wissen, wie Sie digitale Zertifikate in einer verteilten Umgebung bereitstellen.

## Smartcardtypen

Smartcards für Unternehmen und Kunden haben die gleiche Größe, elektrischen Verbindungen und passen in die gleichen Smartcardleser.

Smartcards für die Verwendung in Unternehmen enthalten digitale Zertifikate. Solche Smartcards unterstützen die Windows-Anmeldung und können auch in Kombination mit Anwendungen für die digitale Signierung und Verschlüsselung von Dokumenten und E-Mail verwendet werden. XenApp- und XenDesktop unterstützen eine derartige Verwendung.

Smartcards für Kunden enthalten anstelle eines digitalen Zertifikats einen gemeinsamen geheimen Schlüssel. Mit solchen Smartcards ist ggf. eine Bezahlung möglich (z. B. Kreditkarte mit Chip und PIN/Unterschrift). Sie unterstützen keine Windows-Anmeldung oder typische Windows-Anwendungen. Zur Verwendung solcher Smartcards sind spezielle Windows-Anwendungen und eine geeignete Softwareinfrastruktur (z. B. eine Verbindung mit einem Zahlssystemnetzwerk) erforderlich. Informationen zur Unterstützung solcher Spezialanwendungen unter XenApp bzw. XenDesktop erhalten Sie bei Ihrem Citrix Repräsentanten.

Für Unternehmenssmartcards gibt es entsprechende kompatible Technologien, die ähnlich funktionieren.

- Ein smartcardäquivalentes USB-Token stellt eine direkte Verbindung mit einem USB-Anschluss her. Diese USB-Token sind normalerweise so groß wie ein USB-Stick, aber sie können auch so klein wie die SIM-Karte eines Mobiltelefons sein. Sie sind eine Kombination aus einer Smartcard und einem USB-Smartcardleser.
- Virtuelle Smartcards mit Windows Trusted Platform Module (TPM) erscheinen als Smartcard. Solche virtuellen Smartcards werden für Windows 8 und Windows 10 unter Citrix Receiver ab Version 4.3 unterstützt.
  - XenApp- und XenDesktop-Versionen vor Version 7.6 FP3 unterstützen keine virtuellen Smartcards.
  - Weitere Informationen zu virtuellen Smartcards finden Sie unter [Virtual Smart Card Overview](#).

**Hinweis:** Der Begriff "virtuelle Smartcard" wird auch für ein digitales Zertifikat verwendet, das auf dem Computer des Benutzers gespeichert wird. Diese digitalen Zertifikate sind nicht unbedingt gleichbedeutend mit Smartcards.

Die Smartcard-Unterstützung in XenApp und XenDesktop basiert auf dem PC/SC-Standard (Personal Computer/Smart Card) von Microsoft. Als Mindestanforderung müssen Smartcards und Smartcardleser vom zugrunde liegenden Windows-



Betriebssystem unterstützt werden und vom Microsoft Windows Hardware Quality Labs (WHQL) für die Verwendung auf Computern mit einem qualifizierenden Windows-Betriebssystem zugelassen sein. Weitere Informationen zur Hardware-PC/SC-Kompatibilität finden Sie in der Microsoft-Dokumentation. Weitere Benutzergeräte können PS/SC-konform sein. Weitere Informationen finden Sie unter <http://www.citrix.com/ready/>.

Normalerweise wird für jede Smartcard bzw. ähnliche Geräte ein eigener Gerätetreiber benötigt. Entsprechen Smartcards jedoch einem Standard wie NIST PIV (Personal Identity Verification), kann evtl. ein Treiber für mehrere Smartcardtypen verwendet werden. Der Gerätetreiber muss auf dem Benutzergerät und dem Virtual Delivery Agent installiert werden. Der Gerätetreiber ist häufig im Smartcard-Middlewarepaket eines Citrix Partners enthalten, welches zudem erweiterte Features bietet. Der Gerätetreiber wird u. U. auch als Kryptografiedienstanbieter (CSP), Schlüsselspeicheranbieter (KSP) oder Minitreiber bezeichnet.

Die folgenden Kombinationen aus Smartcard und Middleware für Windows-Systeme wurden von Citrix als repräsentatives Beispiel ihres Typs getestet. Es können jedoch auch andere Smartcards und Middleware verwendet werden. Weitere Informationen über Citrix-kompatible Smartcards und Middleware finden Sie unter <http://www.citrix.com/ready>.

| <b>Middleware</b>                     | <b>Geeignete Karten</b>    |
|---------------------------------------|----------------------------|
| ActivClient 7.0 (DoD-Modus aktiviert) | DoD CAC-Karte              |
| ActivClient 7.0 im Modus PIV          | NIST PIV-Karte             |
| Microsoft-Minitreiber                 | NIST PIV-Karte             |
| GemAlto-Minitreiber für .NET-Karte    | GemAlto .NET v2+           |
| nativer Microsoft-Treiber             | virtuelle Smartcards (TPM) |

Informationen zur Verwendung von Smartcards mit anderen Gerätetypen finden Sie in der Citrix Receiver-Dokumentation für das jeweilige Gerät. Weitere Informationen zur Verwendung von PIV mit XenDesktop finden Sie unter [Configuring Citrix XenDesktop 7.6 and NetScaler Gateway 10.5 with PIV SmartCard Authentication](#).

Informationen zur Verwendung von Smartcards mit anderen Gerätetypen finden Sie in der Citrix Receiver-Dokumentation für das jeweilige Gerät.

## Remote-PC-Zugriff

Smartcards werden nur für den Remotezugriff auf physische Büro-PCs mit Windows 10, Windows 8 oder Windows 7 unterstützt; Smartcards werden nicht für Büro-PCs mit Windows XP unterstützt.

Die folgenden Smartcards wurden mit Remote-PC-Zugriff getestet:

| <b>Middleware</b>        | <b>Geeignete Karten</b> |
|--------------------------|-------------------------|
| Gemalto .NET-Minitreiber | Gemalto .NET v2+        |

|                               |                      |
|-------------------------------|----------------------|
| ActivIdentity ActivClient 6.2 | NIST PIV             |
| ActivIdentity ActivClient 6.2 | CAC                  |
| Microsoft-Minitreiber         | NIST PIV             |
| nativer Microsoft-Treiber     | Virtuelle Smartcards |

## Smartcardleser

Ein Smartcardleser kann im Benutzergerät eingebaut sein oder an dieses angeschlossen werden (normalerweise über USB oder Bluetooth). Kontaktkartenleser, die dem USB-Protokoll CCID (Chip Card Interface Device) entsprechen, werden unterstützt. Diese enthalten einen Schlitz, in den die Smartcard eingeführt wird. In der DK-Norm (Deutsche Kreditwirtschaft) sind vier Kontaktkartenleserklassen festgelegt.

- Smartcardleser der Klasse 1 sind die häufigsten Geräte und haben normalerweise nur einen Steckplatz. Smartcardleser der Klasse 1 werden in der Regel durch einen CCID-Standardgerätetreiber unterstützt, der mit dem Betriebssystem geliefert wurde.
- Smartcardleser der Klasse 2 enthalten eine sichere Tastatur, auf die über das Benutzergerät nicht zugegriffen werden kann. Smartcardleser der Klasse 2 können in eine Tastatur mit integrierter sicherer Tastatur integriert werden. Wenn Sie Smartcardleser der Klasse 2 verwenden, wenden Sie sich an einen Citrix Mitarbeiter, da u. U. ein spezifischer Gerätetreiber erforderlich ist, damit die sichere Tastatur funktioniert.
- Smartcardleser der Klasse 3 haben ein sicheres Display. Diese Smartcardleser werden nicht unterstützt.
- Smartcardleser der Klasse 4 haben ein sicheres Übertragungsmodul. Diese Smartcardleser werden nicht unterstützt.

**Hinweis:** Die Klasse der Smartcardleser hat nichts mit der USB-Gerätekategorie zu tun.

Smartcardleser müssen mit einem entsprechenden Gerätetreiber auf dem Benutzergerät installiert sein.

## Benutzererfahrung

Smartcardunterstützung ist in XenApp und XenDesktop durch einen virtuellen ICA/HDX-Smartcardkanal integriert, der standardmäßig aktiviert ist.

**Wichtig:** Verwenden Sie für Smartcardleser keine generische USB-Umleitung. Diese ist für Smartcardleser standardmäßig deaktiviert und wird bei Aktivierung nicht unterstützt.

Mehrere Smartcards und mehrere Leser können an dem gleichen Benutzergerät verwendet werden, wenn jedoch Passthrough-Authentifizierung verwendet wird, kann nur eine Smartcard eingesteckt werden, wenn der Benutzer einen virtuellen Desktop oder eine virtuelle Anwendung startet. Wenn eine Smartcard innerhalb einer Anwendung verwendet wird (z. B. zur digitalen Signierung oder für Verschlüsselungsfunktionen), werden Sie möglicherweise mehrmals zum Einlegen einer Smartcard oder zur Eingabe einer PIN-Nummer aufgefordert. Dieser Fall kann eintreten, wenn eine oder mehrere

Smartcards gleichzeitig eingelegt wurden.

- Wenn Benutzer zum Einlegen einer Smartcard aufgefordert werden und diese sich bereits im Leser befindet, sollten sie auf "Abbrechen" klicken.
- Wenn Benutzer zur Eingabe der PIN-Nummer aufgefordert werden, sollten sie diese erneut eingeben.

Wenn Sie gehostete Anwendungen unter Windows Server 2008 oder 2008 R2 und mit Smartcards verwenden, die den Microsoft-Kryptografiedienstanbieter für Basissmartcards benötigen, werden im Fall der Ausführung einer Smartcard-Transaktion alle anderen Benutzer, die eine Smartcard bei der Anmeldung verwendet haben, blockiert. Einzelheiten und einen Hotfix zur Behebung dieses Problems finden Sie unter <http://support.microsoft.com/kb/949538>.

Sie können PINs mit einem Kartenverwaltungsprogramm oder einem Herstellerdienstprogramm zurücksetzen.

## Führen Sie vor dem Bereitstellen von Smartcards folgende Schritte aus:

- Installieren Sie für den Smartcardleser einen Gerätetreiber auf dem Benutzergerät. Viele Smartcardleser können mit dem von Microsoft bereitgestellten CCID-Gerätetreiber benutzt werden.
- Beziehen Sie einen Gerätetreiber und Kryptografiedienstanbietersoftware (CSP) vom Smartcard-Hersteller und installieren Sie beides auf Benutzergeräten und auf virtuellen Desktops. Der Treiber und die CSP-Software müssen mit XenApp bzw. XenDesktop kompatibel sein (Informationen zur Kompatibilität enthält die Dokumentation). Für virtuelle Desktops mit Smartcards, die das Minitreibermodell unterstützen und verwenden, werden die Smartcard-Minitreiber automatisch heruntergeladen. Die Treiber können auch über <http://catalog.update.microsoft.com> oder den Hersteller bezogen werden. Wird PKCS#11-Middleware benötigt, wenden Sie sich an den Smartcardhersteller.
- **Wichtig:** Citrix empfiehlt, dass Sie die Treiber und CSP-Software vor der Installation von Citrix Software auf einem physischen Computer installieren und testen.
- Fügen Sie die Citrix Receiver für Web-URL der Liste der vertrauenswürdigen Sites für Benutzer hinzu, die mit Smartcards im Internet Explorer unter Windows 10 arbeiten. In Windows 10 wird Internet Explorer für vertrauenswürdige Sites nicht standardmäßig im geschützten Modus ausgeführt.
- Stellen Sie sicher, dass die Public Key-Infrastruktur entsprechend konfiguriert ist. Hierzu gehört, dass die Zertifikat-zu-Konto-Zuordnung richtig für die Active Directory-Umgebung konfiguriert ist, und dass die Validierung des Benutzerzertifikats ausgeführt werden kann.
- Stellen Sie sicher, dass die Bereitstellung die Systemanforderungen der anderen Citrix Komponenten erfüllt, die mit Smartcards verwendet werden, u. a. Citrix Receiver und StoreFront.
- Stellen Sie sicher, dass auf die folgenden Server in der Site Zugriff besteht:
  - Active Directory-Domänencontroller für das Benutzerkonto mit zugeordnetem Anmeldezertifikat auf der Smartcard
  - Delivery Controller
  - Citrix StoreFront
  - Citrix NetScaler Gateway/Citrix Access Gateway 10.x
  - VDA
  - (Optional für Remotezugriff): Microsoft Exchange Server

## Aktivieren der Smartcard-Verwendung

**Schritt 1:** Geben Sie die Smartcards an die Benutzer aus und berücksichtigen Sie dabei die Kartenausstellungsrichtlinie.

**Schritt 2:** (optional): Richten Sie Smartcards ein, damit Benutzer Remote-PC-Zugriff verwenden können.

**Schritt 3:** Installieren Sie ggf. den Delivery Controller und StoreFront und konfigurieren Sie beides für Smartcard-Remoting.

**Schritt 4:** Aktivieren Sie StoreFront für die Verwendung von Smartcards. Einzelheiten finden Sie unter "Konfigurieren der Smartcardauthentifizierung" in der StoreFront-Dokumentation.

**Schritt 5:** Aktivieren Sie NetScaler Gateway/Access Gateway für die Verwendung von Smartcards. Einzelheiten finden Sie unter "Configuring Authentication and Authorization und Configuring Smart Card Access with the Web Interface" in der NetScaler-Dokumentation.

**Schritt 6:** Aktivieren Sie VDAs für die Verwendung mit Smartcard.

- Stellen Sie sicher, dass die erforderlichen Anwendungen und Updates auf dem VDA installiert wurden.
- Installieren Sie die Middleware.
- Richten Sie Smartcard-Remoting ein, damit die Kommunikation von Smartcarddaten zwischen Citrix Receiver auf einem Benutzergerät und einer virtuellen Desktopsitzung möglich ist.

**Schritt 7:** Aktivieren Sie Benutzergeräte (einschließlich der Maschinen innerhalb und außerhalb von Domänen) für die Verwendung von Smartcards. Einzelheiten finden Sie unter "Konfigurieren der Smartcardauthentifizierung" in der StoreFront-Dokumentation.

- Importieren Sie das Zertifizierungsstellen-Stammzertifikat und das Zertifikat der ausstellenden Zertifizierungsstelle in den Schlüsselspeicher des Geräts.
- Installieren Sie die Smartcard-Middleware des Herstellers.
- Installieren und konfigurieren Sie Citrix Receiver für Windows. Importieren Sie icaclient.adm mit der Gruppenrichtlinien-Verwaltungskonsole und aktivieren Sie die Smartcard-Authentifizierung.

**Schritt 8:** Testen Sie die Bereitstellung. Stellen Sie sicher, dass die Bereitstellung richtig konfiguriert ist, indem Sie den virtuellen Desktop mit der Smartcard eines Testbenutzers starten. Testen Sie alle möglichen Zugriffsmechanismen (beispielsweise Zugriff auf den Desktop über Internet Explorer und Citrix Receiver).

# Smartcardbereitstellungen

May 10, 2016

Die folgenden Typen von Smartcardbereitstellungen werden von dieser Produktversion und von gemischten Umgebungen, die diese Version enthalten, unterstützt. Weitere Konfigurationen funktionieren eventuell, werden aber nicht unterstützt.

| Typ                                                                                                       | Verbindung mit StoreFront            |
|-----------------------------------------------------------------------------------------------------------|--------------------------------------|
| Lokale in Domänen eingebundene Computer                                                                   | Direkte Verbindung                   |
| Remotezugriff von in Domänen eingebundenen Computern                                                      | Verbindung über NetScaler Gateway    |
| Nicht in Domänen eingebundene Computer                                                                    | Direkte Verbindung                   |
| Remotezugriff von nicht in Domänen eingebundenen Computern                                                | Verbindung über NetScaler Gateway    |
| Nicht in Domänen eingebundene Computer und Thin Clients mit Zugriff auf die Desktopgerätesite             | Verbindung über Desktopgerätesites   |
| In Domänen eingebundene Computer und Thin Clients mit Zugriff auf StoreFront über die XenApp Services-URL | Verbindung über XenApp Services-URLs |

Die Bereitstellungstypen werden durch die Merkmale des Benutzergeräts definiert, mit dem der Smartcardleser verbunden ist:

- In Domäne eingebundenes Gerät oder nicht in Domäne eingebundenes Gerät
- Art der Verbindung zwischen Gerät und StoreFront
- Zur Anzeige der virtuellen Desktops und Anwendungen verwendete Software

Darüber hinaus können smartcardfähige Anwendungen wie Microsoft Word oder Microsoft Excel in diesen Bereitstellungen verwendet werden. In diesen Anwendungen können Benutzer Dokumente digital signieren und verschlüsseln.

## Bimodale Authentifizierung

Soweit in der jeweiligen Bereitstellung möglich, unterstützt Receiver die bimodale Authentifizierung, d. h. der Benutzer hat die Wahl, sich mit einer Smartcard oder mit dem Benutzernamen und Kennwort anzumelden. Dies ist nützlich, wenn die Smartcard nicht verwendet werden kann (z. B. sie wurde vom Benutzer zu Hause vergessen oder das Zertifikat ist abgelaufen).

Da Benutzer nicht domänengebundener Geräte sich direkt an Receiver für Windows anmelden, können Sie für diese Benutzer ein Fallback auf die explizite Authentifizierung aktivieren. Wenn Sie die bimodale Authentifizierung konfigurieren, müssen sich Benutzer zuerst mit den Smartcards und PINs anmelden; sie können aber die explizite Authentifizierung auswählen, wenn sie Probleme mit den Smartcards haben.

Wenn Sie NetScaler Gateway bereitstellen, melden sich Benutzer an den Geräten an und werden von Receiver für Windows zur Authentifizierung bei NetScaler Gateway aufgefordert. Dies gilt sowohl für in Domänen eingebundene Geräte als auch

für Geräte, die nicht in Domänen eingebunden sind. Benutzer können sich bei NetScaler Gateway mit Smartcard und PIN oder mit expliziten Anmeldeinformationen anmelden. Sie können dann Benutzern die bimodale Authentifizierung für Anmeldungen an NetScaler Gateway bereitstellen. Konfigurieren Sie die Passthrough-Authentifizierung von NetScaler Gateway an StoreFront und delegieren Sie die Validierung der Anmeldeinformationen für Smartcardbenutzer an NetScaler Gateway, sodass Benutzer automatisch bei StoreFront authentifiziert werden.

## Überlegungen zu mehreren Active Directory-Gesamtstrukturen

In einer Citrix Umgebung werden Smartcards in einer einzelnen Gesamtstruktur unterstützt. Strukturübergreifende Smartcard-Anmeldungen erfordern eine direkte bidirektionale Gesamtstruktur-Vertrauensstellung für alle Benutzerkonten. Komplexere Mehrfachstruktur-Bereitstellungen mit Smartcards (d. h. Vertrauensstellungen sind nur unidirektional oder sonstiger Art) werden nicht unterstützt.

Sie können Smartcards in einer Citrix Umgebung mit Remotedesktops verwenden. Dieses Feature kann lokal installiert werden (auf dem Benutzergerät, mit dem die Smartcard verbunden ist) oder remote (auf dem Remotedesktop, mit dem das Benutzergerät verbunden wird).

## Richtlinie zum Entfernen der Smartcard

Die Richtlinie zum Entfernen der Smartcard legt fest, was passiert, wenn die Smartcard während einer Sitzung entfernt wird. Die Richtlinie zum Entfernen der Smartcard wird im Windows-Betriebssystem konfiguriert und verarbeitet.

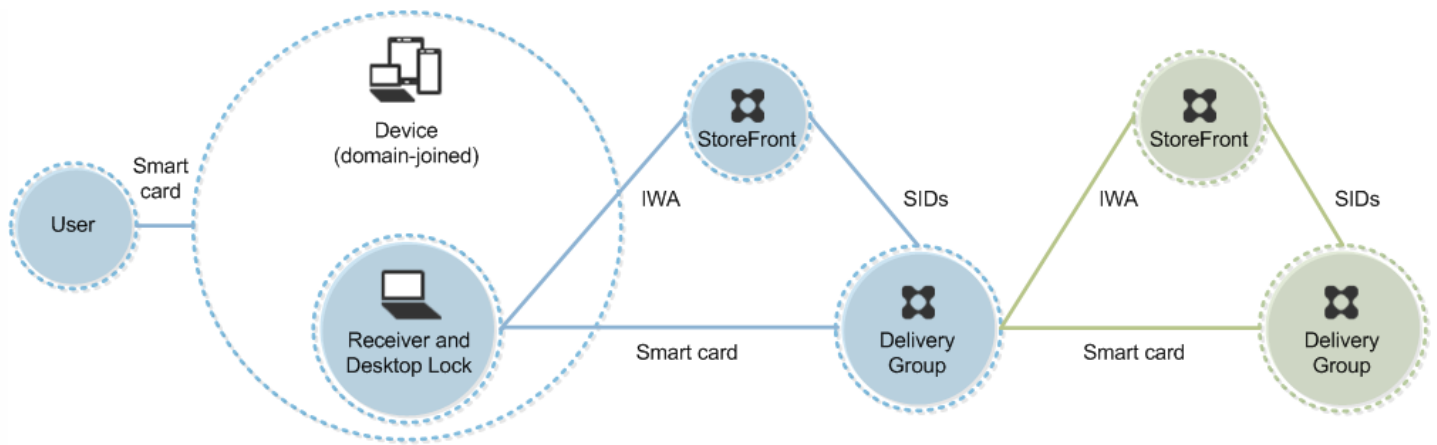
| Richtlinieneinstellung                 | Desktop-Verhalten                                                                                                                                                                                         |
|----------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Keine Aktion                           | Keine Aktion.                                                                                                                                                                                             |
| Arbeitsstation sperren                 | Die Desktopsitzung wird getrennt und der virtuelle Desktop gesperrt.                                                                                                                                      |
| Abmeldung erzwingen                    | Der Benutzer wird zur Abmeldung gezwungen. Wenn die Netzwerkverbindung unterbrochen ist und diese Einstellung aktiviert wird, wird die Sitzung möglicherweise abgemeldet und der Benutzer verliert Daten. |
| Trennen bei einer Remotedienstesitzung | Die Sitzung wird getrennt und der virtuelle Desktop gesperrt.                                                                                                                                             |

## Überprüfen der Zertifikatssperrlisten

Wenn die Überprüfung von Zertifikatssperrlisten aktiviert ist und ein Benutzer führt eine Smartcard mit einem ungültigen Zertifikat in einen Smartcardleser ein, kann der Benutzer nicht authentifiziert werden oder nicht auf den mit dem Zertifikat verbundenen Desktop oder die Anwendung zugreifen. Bei einem ungültigen Zertifikat für die E-Mail-Entschlüsselung bleibt die E-Mail beispielsweise verschlüsselt. Wenn andere Zertifikate auf der Smartcard, z. B. solche, die für die Authentifizierung verwendet werden, noch gültig sind, bleiben diese Funktionen weiterhin aktiv.

## Bereitstellungsbeispiel: in Domänen eingebundene Computer

Diese Bereitstellung bezieht sich auf in Domänen eingebundene Benutzergeräte mit Desktop Viewer und Direktverbindung mit StoreFront.

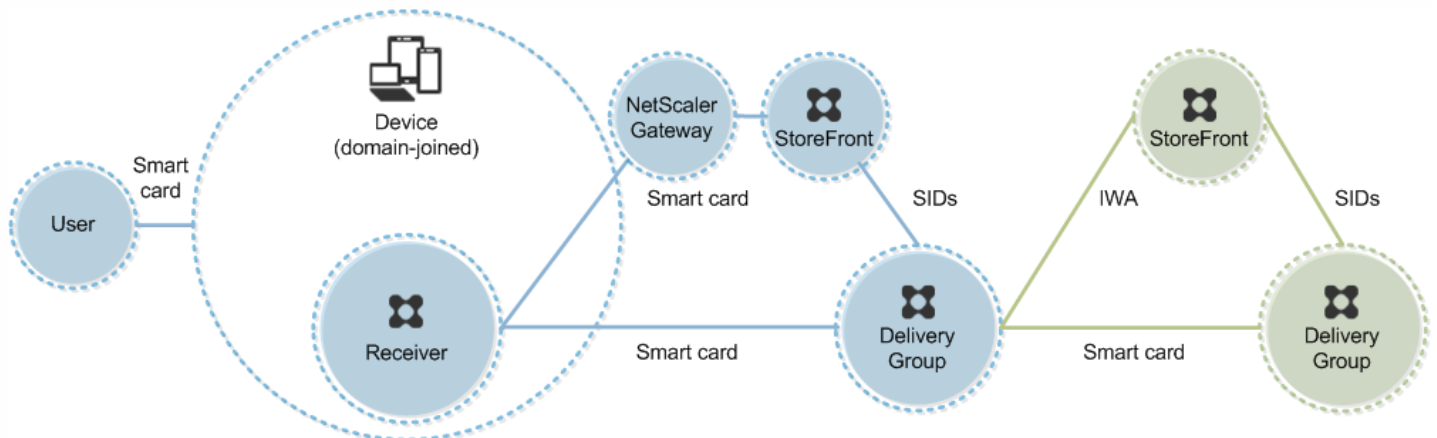


Zum Anmelden beim Gerät benötigt der Benutzer eine Smartcard und eine PIN. Der Benutzer wird dann durch Receiver beim StoreFront-Server mittels integrierter Windows-Authentifizierung (IWA) authentifiziert. StoreFront übergibt die Sicherheits-IDs (SIDs) an XenApp bzw. XenDesktop. Wenn der Benutzer einen virtuellen Desktop oder eine Anwendung startet, wird er nicht aufgefordert, die PIN neu einzugeben, da in Receiver das Feature "Single Sign-On" konfiguriert ist.

Diese Bereitstellung kann um einen zweiten StoreFront-Server und einen Server mit gehosteten Anwendungen auf ein Double-Hop-System erweitert werden. Ein Receiver auf dem virtuellen Desktop übernimmt die Authentifizierung beim zweiten StoreFront-Server. Für diese zweite Verbindung kann eine beliebige Authentifizierungsmethode verwendet werden. Die für den ersten Hop dargestellte Konfiguration kann im zweiten Hop wiederverwendet oder nur für den zweiten Hop verwendet werden.

### Bereitstellungsbeispiel: Remotezugriff von in Domänen eingebundenen Computern

Diese Bereitstellung bezieht sich auf in Domänen eingebundene Benutzergeräte mit Desktop Viewer und Verbindung mit StoreFront über NetScaler Gateway/Access Gateway.



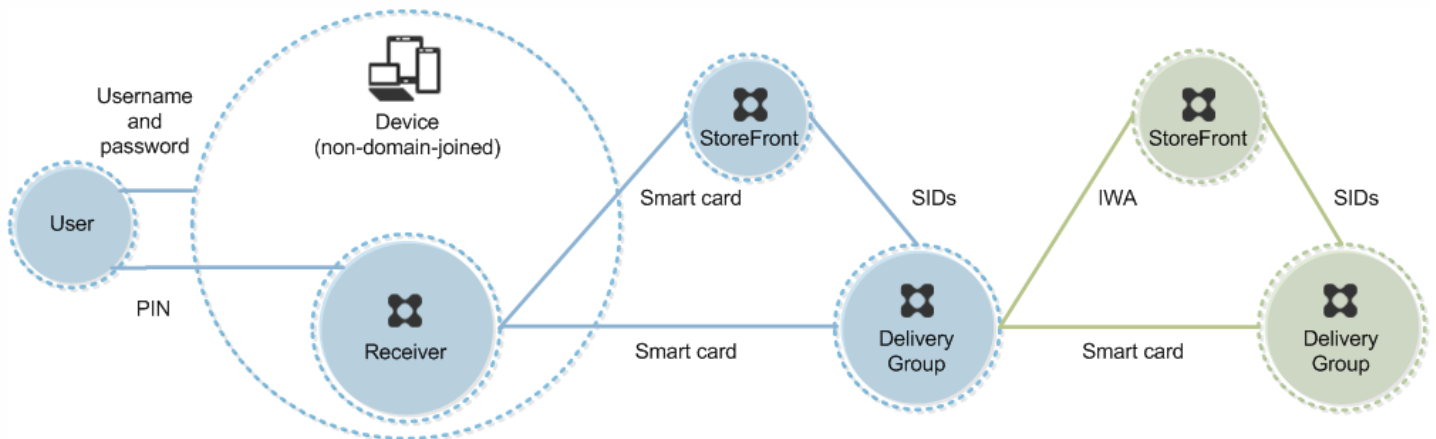
Der Benutzer meldet sich mit Smartcard und PIN beim Gerät und anschließend erneut bei NetScaler Gateway oder Access Gateway an. Die zweite Anmeldung kann entweder mit Smartcard und PIN oder einem Benutzernamen und einem Kennwort erfolgen, da Receiver in dieser Bereitstellung eine bimodale Authentifizierung zulässt.

Der Benutzer wird automatisch bei StoreFront angemeldet; StoreFront übergibt die Sicherheits-IDs (SIDs) an XenApp bzw. XenDesktop. Wenn der Benutzer einen virtuellen Desktop oder eine Anwendung startet, wird er nicht aufgefordert, die PIN neu einzugeben, da in Receiver das Feature "Single Sign-On" konfiguriert ist.

Diese Bereitstellung kann um einen zweiten StoreFront-Server und einen Server mit gehosteten Anwendungen auf ein Double-Hop-System erweitert werden. Ein Receiver auf dem virtuellen Desktop übernimmt die Authentifizierung beim zweiten StoreFront-Server. Für diese zweite Verbindung kann eine beliebige Authentifizierungsmethode verwendet werden. Die für den ersten Hop dargestellte Konfiguration kann im zweiten Hop wiederverwendet oder nur für den zweiten Hop verwendet werden.

### Bereitstellungsbeispiel: nicht in Domänen eingebundene Computer

Diese Bereitstellung bezieht sich auf nicht in Domänen eingebundene Benutzergeräte mit Desktop Viewer und Direktverbindung mit StoreFront.



Ein Benutzer meldet sich beim Gerät an. Normalerweise muss er seinen Benutzernamen und das Kennwort eingeben, aber da das Gerät nicht Mitglied einer Domäne ist, sind die Anmeldeinformationen für diese Anmeldung optional. Da bimodale Authentifizierung in dieser Bereitstellung möglich ist, fordert Receiver den Benutzer auf, sich entweder mit Smartcard und PIN oder mit Benutzernamen und Kennwort anzumelden. Receiver authentifiziert dann bei StoreFront.

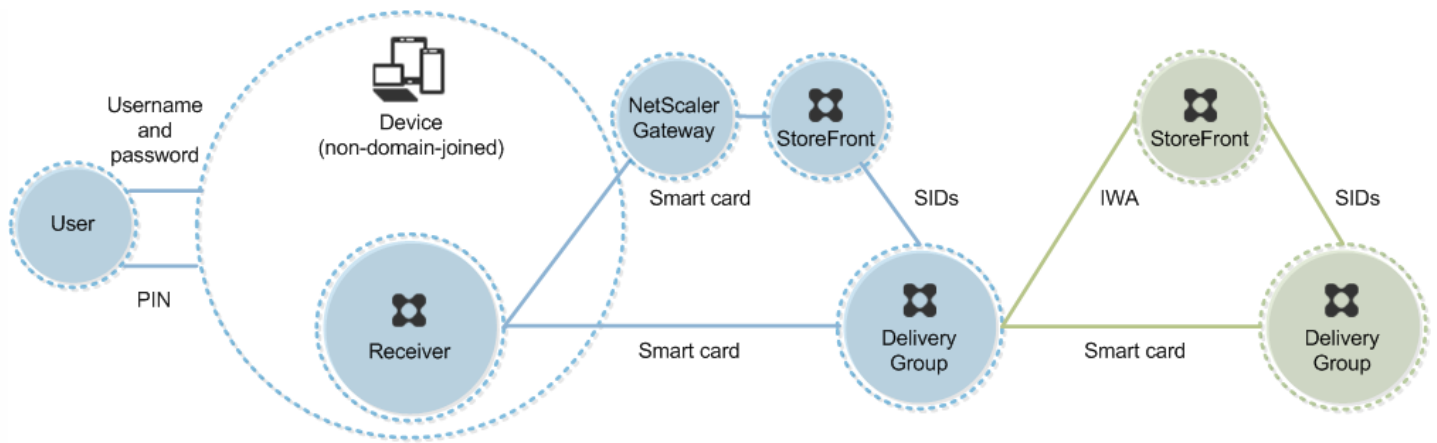
StoreFront übergibt die Sicherheits-IDs (SIDs) an XenApp bzw. XenDesktop. Wenn der Benutzer einen virtuellen Desktop oder eine Anwendung startet, wird er aufgefordert, die PIN neu einzugeben, da Single Sign-On in dieser Bereitstellung nicht verfügbar ist.

Diese Bereitstellung kann um einen zweiten StoreFront-Server und einen Server mit gehosteten Anwendungen auf ein Double-Hop-System erweitert werden. Ein Receiver auf dem virtuellen Desktop übernimmt die Authentifizierung beim zweiten StoreFront-Server. Für diese zweite Verbindung kann eine beliebige Authentifizierungsmethode verwendet werden. Die für den ersten Hop dargestellte Konfiguration kann im zweiten Hop wiederverwendet oder nur für den zweiten Hop verwendet werden.

### Bereitstellungsbeispiel: Remotezugriff von nicht in Domänen eingebundenen Computern

Diese Bereitstellung bezieht sich auf nicht in Domänen eingebundene Benutzergeräte mit Desktop Viewer und Direktverbindung mit StoreFront.





Ein Benutzer meldet sich beim Gerät an. Normalerweise muss er seinen Benutzernamen und das Kennwort eingeben, aber da das Gerät nicht Mitglied einer Domäne ist, sind die Anmeldeinformationen für diese Anmeldung optional. Da bimodale Authentifizierung in dieser Bereitstellung möglich ist, fordert Receiver den Benutzer auf, sich entweder mit Smartcard und PIN oder mit Benutzernamen und Kennwort anzumelden. Receiver authentifiziert dann bei StoreFront.

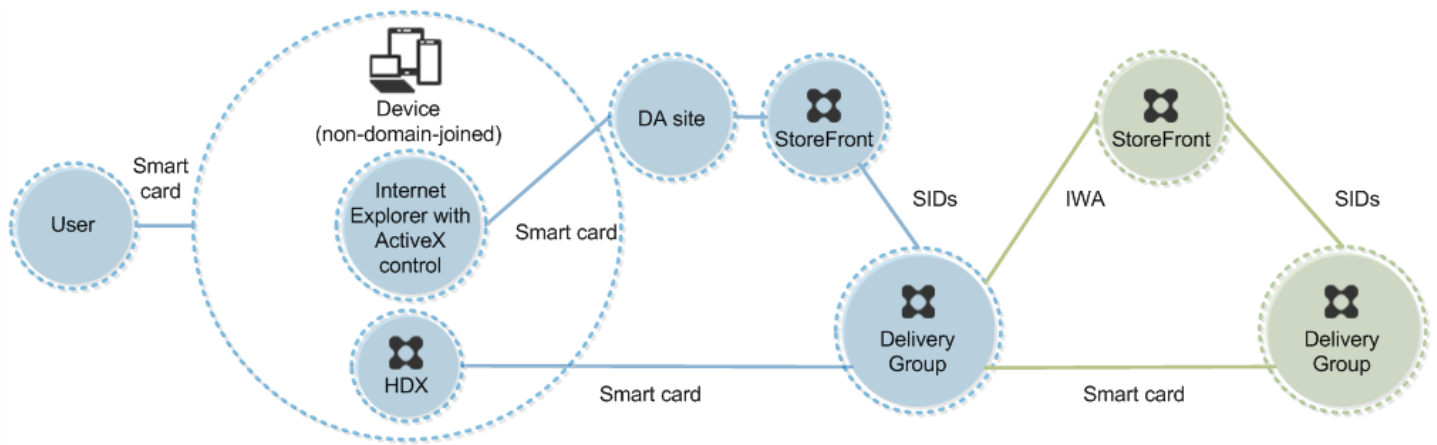
StoreFront übergibt die Sicherheits-IDs (SIDs) an XenApp bzw. XenDesktop. Wenn der Benutzer einen virtuellen Desktop oder eine Anwendung startet, wird er aufgefordert, die PIN neu einzugeben, da Single Sign-On in dieser Bereitstellung nicht verfügbar ist.

Diese Bereitstellung kann um einen zweiten StoreFront-Server und einen Server mit gehosteten Anwendungen auf ein Double-Hop-System erweitert werden. Ein Receiver auf dem virtuellen Desktop übernimmt die Authentifizierung beim zweiten StoreFront-Server. Für diese zweite Verbindung kann eine beliebige Authentifizierungsmethode verwendet werden. Die für den ersten Hop dargestellte Konfiguration kann im zweiten Hop wiederverwendet oder nur für den zweiten Hop verwendet werden.

**Bereitstellungsbeispiel: nicht in Domänen eingebundene Computer und Thin Clients mit Zugriff auf die Desktopgerätesite**

Diese Bereitstellung bezieht sich auf nicht in Domänen eingebundene Benutzergeräte, auf denen möglicherweise Desktop Lock ausgeführt wird und die mit StoreFront über Desktopgerätesites verbunden werden.

Desktop Lock ist eine eigenständige Komponente, die mit XenApp, XenDesktop und VDI-in-a-Box auf den Markt gebracht wurde. Das Programm ist eine Alternative zu Desktop Viewer und wird hauptsächlich für umfunktionierte Windows-Computer und Thin Clients verwendet. Desktop Lock ersetzt die Windows-Shell und Task-Manager bei diesen Benutzergeräten, wodurch der Benutzerzugriff auf die zugrunde liegenden Geräte verhindert wird. Mit Desktop Lock können Benutzer auf Desktops von Windows-Servermaschinen und Windows-Desktopmaschinen zugreifen. Die Installation von Desktop Lock ist optional.



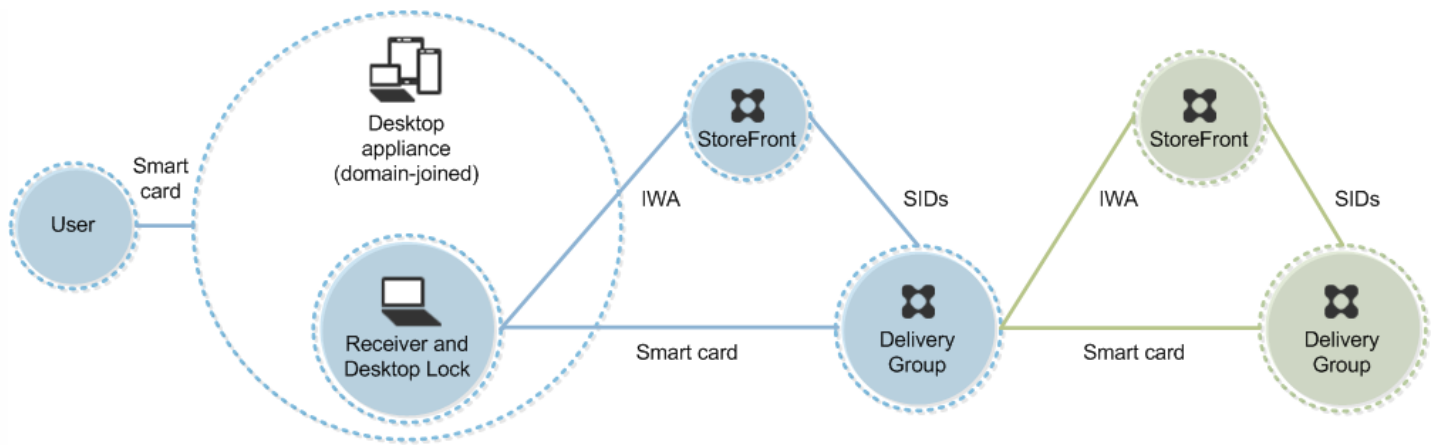
Zum Anmelden beim Gerät benötigt der Benutzer eine Smartcard. Wenn Desktop Lock auf dem Gerät ausgeführt wird, wird das Gerät so konfiguriert, dass eine Desktopgerätesite über Internet Explorer im Kioskmodus gestartet wird. Der Benutzer wird durch ein ActiveX-Steuerelement der Site aufgefordert, seine PIN einzugeben, die dann an StoreFront gesendet wird. StoreFront übergibt die Sicherheits-IDs (SIDs) an XenApp bzw. XenDesktop. Der erste verfügbare Desktop in der alphabetischen Liste einer zugewiesenen Desktopgruppe wird gestartet.

Diese Bereitstellung kann um einen zweiten StoreFront-Server und einen Server mit gehosteten Anwendungen auf ein Double-Hop-System erweitert werden. Ein Receiver auf dem virtuellen Desktop übernimmt die Authentifizierung beim zweiten StoreFront-Server. Für diese zweite Verbindung kann eine beliebige Authentifizierungsmethode verwendet werden. Die für den ersten Hop dargestellte Konfiguration kann im zweiten Hop wiederverwendet oder nur für den zweiten Hop verwendet werden.

**Bereitstellungsbeispiel:** in Domänen eingebundene Computer und Thin Clients mit Zugriff auf StoreFront über die XenApp Services-URL

Diese Bereitstellung bezieht sich auf in Domänen eingebundene Benutzergeräte, auf denen Desktop Lock ausgeführt wird und die mit StoreFront über XenApp Services-URLs verbunden werden.

Desktop Lock ist eine eigenständige Komponente, die mit XenApp, XenDesktop und VDI-in-a-Box auf den Markt gebracht wurde. Das Programm ist eine Alternative zu Desktop Viewer und wird hauptsächlich für umfunktionierte Windows-Computer und Thin Clients verwendet. Desktop Lock ersetzt die Windows-Shell und Task-Manager bei diesen Benutzergeräten, wodurch der Benutzerzugriff auf die zugrunde liegenden Geräte verhindert wird. Mit Desktop Lock können Benutzer auf Desktops von Windows-Servermaschinen und Windows-Desktopmaschinen zugreifen. Die Installation von Desktop Lock ist optional.



Zum Anmelden beim Gerät benötigt der Benutzer eine Smartcard und eine PIN. Wenn Desktop Lock auf dem Gerät ausgeführt wird, wird der Benutzer beim Storefront-Server über die integrierte Windows-Authentifizierung (IWA) authentifiziert. StoreFront übergibt die Sicherheits-IDs (SIDs) an XenApp bzw. XenDesktop. Wenn der Benutzer einen virtuellen Desktop startet, wird er nicht aufgefordert, die PIN neu einzugeben, da in Receiver Single Sign-On konfiguriert ist.

Diese Bereitstellung kann um einen zweiten StoreFront-Server und einen Server mit gehosteten Anwendungen auf ein Double-Hop-System erweitert werden. Ein Receiver auf dem virtuellen Desktop übernimmt die Authentifizierung beim zweiten StoreFront-Server. Für diese zweite Verbindung kann eine beliebige Authentifizierungsmethode verwendet werden. Die für den ersten Hop dargestellte Konfiguration kann im zweiten Hop wiederverwendet oder nur für den zweiten Hop verwendet werden.

# Passthrough-Authentifizierung und Single Sign-On mit Smartcards

May 10, 2016

## Passthrough-Authentifizierung

Die Passthrough-Authentifizierung mit Smartcards bei virtuellen Desktops wird auf Benutzergeräten unterstützt, auf denen Windows 10, Windows 8 und Windows 7 SP1, Enterprise und Professional Editionen ausgeführt werden.

Die Passthrough-Authentifizierung mit Smartcards bei gehosteten Anwendungen wird auf Servern unterstützt, auf denen Windows Server 2008 und Windows Server 2012 ausgeführt werden.

Wenn Sie die Passthrough-Authentifizierung mit Smartcards für gehostete Anwendungen verwenden, stellen Sie sicher, dass Sie für Passthrough mit Smartcard als Authentifizierungsmethode für die Site die Verwendung von Kerberos aktivieren.

Hinweis: Die Verfügbarkeit der Passthrough-Authentifizierung mit Smartcards hängt von vielen Faktoren ab, u. a.:

- Sicherheitsrichtlinien für die Passthrough-Authentifizierung der jeweiligen Organisation
- Typ und Konfiguration der Middleware
- Typen der Smartcardleser
- Richtlinie für das Zwischenspeichern von Middleware-PINs

Die Passthrough-Authentifizierung mit Smartcards wird in Citrix StoreFront konfiguriert. Weitere Informationen finden Sie unter

— *Konfigurieren des Authentifizierungsdiensts*  
in der StoreFront-Dokumentation.

## Single Sign-On

Single Sign-On ist ein Citrix Feature, mit dem die Passthrough-Authentifizierung in Starts von virtuellen Desktops und Anwendungen implementiert wird. Sie können dieses Feature bei Smartcardbereitstellungen verwenden, die in Domänen eingebunden und direkt mit StoreFront verbunden sind, sowie bei in Domänen eingebundenen und über NetScaler mit StoreFront verbundenen Bereitstellungen. So müssen Benutzer ihre PIN weniger häufig eingeben. Zur die Verwendung von Single Sign-On in diesen Bereitstellungstypen bearbeiten Sie die folgenden Parameter in der Datei default.ica, die sich auf dem StoreFront-Server befindet:

- In Domänen eingebundene, direkt mit StoreFront verbundene Smartcardbereitstellungen: Einstellung für DisableCtrlAltDel auf Off
- In Domänen eingebundene, über NetScaler mit StoreFront verbundene Smartcardbereitstellungen: Einstellung für UseLocalUserAndPassword auf On

Weitere Anweisungen zum Einrichten dieser Parameter finden Sie in der Dokumentation für StoreFront oder NetScaler Gateway.

Die Verfügbarkeit der Single Sign-On-Funktion hängt von vielen Faktoren ab, u. a.:

- Sicherheitsrichtlinien für Single Sign-On der jeweiligen Organisation
- Typ und Konfiguration der Middleware
- Typen der Smartcardleser
- Richtlinie für das Zwischenspeichern von Middleware-PINs

Hinweis: Wenn Benutzer sich beim Virtual Delivery Agent (VDA) mit einer Maschine anmelden, an die ein Smartcardleser angeschlossen ist, wird möglicherweise eine Windows-Kachel angezeigt, die die letzte erfolgreiche Authentifizierungsmethode repräsentiert, z. B. Smartcard oder Kennwort. Daher wird bei aktiviertem Single Sign-On ggf. eine entsprechende Kachel angezeigt. Zum Anmelden müssen Benutzer Benutzer wechseln auswählen, um eine andere Kachel auszuwählen, da die Single Sign-On-Kachel nicht funktioniert.

# SSL

Jul 13, 2016

Das Konfigurieren einer XenApp- oder XenDesktop-Site zur Verwendung des SSL-Sicherheitsprotokolls (Secure Sockets Layer) umfasst folgende Schritte:

- Rufen Sie ein Serverzertifikat ab und installieren und registrieren Sie es auf allen Delivery Controllern. Konfigurieren Sie zudem einen Port mit dem SSL-Zertifikat. Einzelheiten finden Sie unter [Installieren von SSL-Serverzertifikaten auf Controllern](#).

Sie können die Ports ändern, die der Controller zum Abhören von HTTP- und HTTPS-Datenverkehr verwendet.

- Aktivieren Sie SSL-Verbindungen zwischen Benutzern und Virtual Delivery Agents (VDAs) mit den folgenden Schritten:
  - Konfigurieren Sie SSL auf den Maschinen, auf denen die VDAs installiert sind. (Der Einfachheit halber werden nachfolgend Maschinen, auf denen VDAs installiert sind, als "VDAs" bezeichnet.) Sie können ein PowerShell-Skript von Citrix verwenden oder eine manuelle Konfiguration vornehmen. Allgemeine Informationen finden Sie unter [SSL-Einstellungen auf VDAs](#). Detaillierte Informationen finden Sie unter [Konfigurieren von SSL auf einem VDA mit dem PowerShell-Skript](#) und [Manuelle Konfiguration von SSL auf einem VDA](#).
  - Konfigurieren Sie SSL in den Bereitstellungsgruppen, die die VDAs enthalten, indem Sie eine Reihe von PowerShell-Cmdlets in Studio ausführen. Einzelheiten finden Sie unter [Konfigurieren von SSL auf Bereitstellungsgruppen](#).

Anforderungen und Überlegungen:

- Das Aktivieren von SSL-Verbindungen zwischen Benutzern und VDAs gilt nur für XenApp 7.6- und XenDesktop 7.6-Sites sowie für unterstützte höhere Releases.
- Konfigurieren Sie SSL in den Bereitstellungsgruppen und auf den VDAs nach der Installation von Komponenten sowie nach dem Erstellen von Sites, Maschinenkatalogen und Bereitstellungsgruppen.
- Zum Konfigurieren von SSL in den Bereitstellungsgruppen müssen Sie die Berechtigung zum Ändern der Zugriffsregeln für Controller haben; ein Volladministrator hat diese Berechtigung.
- Zum Konfigurieren von SSL auf den VDAs müssen Sie ein Windows-Administrator auf der Maschine sein, auf der der VDA installiert ist.
- Wenn Sie SSL auf VDAs konfigurieren möchten, für die ein Upgrade von einer früheren Version durchgeführt wurde, deinstallieren Sie die SSL-Relay-Software vor dem Upgrade von den Maschinen.
- Das PowerShell-Skript konfiguriert SSL auf statischen VDAs, jedoch nicht auf gepoolten VDAs, die von Maschinenerstellungsdienste oder Provisioning Services bereitgestellt wurden und deren Maschinenimage bei jedem Neustart zurückgesetzt wird.

Beachten Sie Folgendes beim Arbeiten in der Windows-Registrierung:

Achtung: Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

Weitere Informationen zur Aktivierung von SSL auf der Sitedatenbank finden Sie unter [CTX137556](#).

## Installieren von SSL-Serverzertifikaten auf Controllern

Für HTTPS wird SSL vom XML-Dienst über Serverzertifikate, nicht aber über Clientzertifikate unterstützt. Nachfolgend finden Sie Anleitungen zum Abrufen eines Zertifikats und zur Installation und Registrierung auf einem Controller sowie zum Konfigurieren eines Ports mit dem SSL-Zertifikat:

- Wenn IIS auf dem Controller installiert ist, folgen Sie den Anleitungen unter <https://technet.microsoft.com/en-us/library/cc771438%28v=ws.10%29.aspx>.
- Wenn der Controller nicht über IIS verfügt, können Sie das Zertifikat mit folgender Methode konfigurieren:
  1. Rufen Sie ein SSL-Serverzertifikat ab und installieren Sie es auf dem Controller, wie unter <http://blogs.technet.com/b/pki/archive/2009/08/05/how-to-create-a-web-server-ssl-certificate-manually.aspx> beschrieben. Weitere Informationen zum certreq-Tool finden Sie unter [http://technet.microsoft.com/en-us/library/cc736326\(Ws.10\).aspx](http://technet.microsoft.com/en-us/library/cc736326(Ws.10).aspx).  
Wenn Sie SSL auf VDAs mit dem PowerShell-Skript konfigurieren und dabei den Fingerabdruck des SSL-Zertifikats nicht angeben, stellen Sie sicher, dass das Zertifikat im Zertifikatspeicher unter "Lokaler Computer > Eigene Zertifikate > Zertifikate" gespeichert ist. Wenn mehrere Zertifikate in dem Speicherort sind, wird das zuerst gefundene verwendet.
  2. Konfigurieren Sie einen Port mit dem Zertifikat. Weitere Informationen finden Sie unter <http://msdn.microsoft.com/en-us/library/ms733791%28v=vs.110%29.aspx>.

## Ändern von HTTP- oder HTTPS-Ports

Der XML-Dienst hört standardmäßig Port 80 auf HTTP-Datenverkehr und Port 443 auf HTTPS-Datenverkehr ab. Zwar können auch andere Ports verwendet werden, jedoch wird der Controller dabei nicht vertrauenswürdigen Netzwerken ausgeliefert, und es entsteht ein Sicherheitsrisiko. Das Bereitstellen eines eigenständigen StoreFront-Servers ist dem Ändern der Standardwerte vorzuziehen. Zum Ändern der vom Controller verwendeten standardmäßigen HTTP- oder HTTPS-Ports führen Sie den folgenden Befehl in Studio aus: BrokerService.exe -WIPORT <http-port> -WISSLPORT <https-port>

<http-port> ist die Portnummer für HTTP-Datenverkehr und <https-port> ist die Portnummer für HTTPS-Datenverkehr.

Hinweis: Nachdem Sie einen Port geändert haben, zeigt Studio möglicherweise eine Meldung zu Lizenzkompatibilität und Upgrades an. Sie lösen das Problem, indem Sie Dienstanstalten mit den folgenden PowerShell-Cmdlets neu registrieren:

```
Get-ConfigRegisteredServiceInstance -ServiceType Broker -Binding XML_HTTPS | Unregister-ConfigRegisteredServiceInstance Get-BrokerServiceInstance | where Binding -eq "XML_HTTPS" | Regist
```

## Erzwingen von HTTPS-Datenverkehr

Wenn der XML-Dienst den HTTP-Datenverkehr ignorieren soll, legen Sie den folgenden Registrierungswert unter HKLM\Software\Citrix\DesktopServer\ auf dem Controller fest und starten Sie den Brokerdienst neu.

Um den HTTP-Datenverkehr zu ignorieren, legen Sie für XmlServicesEnableNonSsl 0 fest.

Es gibt einen entsprechenden Registrierungswert, damit der HTTPS-Datenverkehr ignoriert wird: XmlServicesEnableSsl. Stellen Sie sicher, dass er nicht auf 0 festgelegt ist.

## SSL-Einstellungen auf VDAs

Wenn Sie SSL auf VDAs konfigurieren, werden Berechtigungen auf dem installierten SSL-Zertifikat geändert. Der ICA-Dienst erhält Lesezugriff für den privaten Schlüssel des Zertifikats und informiert den ICA-Dienst über Folgendes:

- **Das für SSL zu verwendende Zertifikat im Zertifikatspeicher**
- **Die für SSL-Verbindungen zu verwendende TCP-Portnummer.**  
Die Windows-Firewall (wenn sie aktiviert ist) muss so konfiguriert sein, dass eingehende Verbindungen auf diesem TCP-Port zugelassen sind. Diese Konfiguration wird für Sie ausgeführt, wenn Sie das PowerShell-Skript verwenden.
- **Die zulässigen Versionen des SSL-Protokolls.**  
Die unterstützten SSL-Protokollversionen unterliegen einer Hierarchie (von der niedrigsten zur höchsten Version): SSL 3.0, TLS 1.0, TLS 1.1 und TLS 1.2. Wenn Sie die zulässige Mindestversion angeben, sind alle Protokollverbindungen, die diese Version oder eine höhere Version verwenden, zulässig.

Wenn Sie beispielsweise TLS 1.1 als Mindestversion angeben, werden auch TLS 1.1- und TLS 1.2-Protokollverbindungen zugelassen. Wenn Sie SSL 3.0 als Mindestversion angeben, sind Verbindungen

für alle unterstützten Versionen zulässig. Wenn Sie TLS 1.2 als Mindestversion angeben, werden nur TLS 1.2-Verbindungen zugelassen.

- Die zulässigen SSL-Verschlüsselungsverfahren.**  
 Eine Verschlüsselungssammlung ist eine Liste gebräuchlicher SSL-Verschlüsselungsverfahren. Wenn ein Client eine Verbindung herstellt und eine Liste unterstützter SSL-Verschlüsselungsverfahren übermittelt, ordnet der VDA ein Verschlüsselungsverfahren des Clients einem Verschlüsselungsverfahren in der konfigurierten Verschlüsselungssammlung zu und akzeptiert die Verbindung. Wenn der Client ein Verschlüsselungsverfahren übermittelt, das nicht in der Verschlüsselungssammlung des VDAs ist, lehnt der VDA die Verbindung ab.  
  
 Drei Verschlüsselungssammlungen werden unterstützt: GOV (Government), COM (Commercial) und ALL. Die Verschlüsselungsverfahren in diesen Verschlüsselungssammlungen hängen vom Windows FIPS-Modus ab. Weitere Informationen zum Windows FIPS-Modus finden Sie unter <http://support.microsoft.com/kb/811833>. In der folgenden Tabelle finden Sie die für die jeweilige Verschlüsselungssammlung unterstützten Verschlüsselungsverfahren.

| SSL-Verschlüsselungssammlung | GOV | COM | ALL | GOV | COM | ALL |
|------------------------------|-----|-----|-----|-----|-----|-----|
| FIPS-Modus                   | Aus | Aus | Aus | Ein | Ein | Ein |
| RSA_KEYX                     | x   | x   | x   | x   | x   | x   |
| RSA_SIGN                     | x   | x   | x   | x   | x   | x   |
| 3DES                         | x   |     | x   | x   |     | x   |
| RC4                          |     | x   | x   |     |     |     |
| MD5                          | x   | x   | x   |     |     |     |
| SHA                          | x   | x   | x   | x   | x   | x   |
| SHA_256                      | x   | x   | x   | x   | x   | x   |
| SHA_384                      | x   | x   | x   | x   | x   | x   |
| SHA_512                      | x   | x   | x   | x   | x   | x   |
| AES                          | x   | x   | x   | x   | x   | x   |

Eine Bereitstellungsgruppe darf nicht eine Mischung von VDAs mit und ohne konfiguriertem SSL enthalten. Wenn Sie SSL für eine Bereitstellungsgruppe konfigurieren, sollten Sie SSL bereits für alle VDAs in dieser Bereitstellungsgruppe konfiguriert haben.

#### Konfigurieren von SSL auf einem VDA mit dem PowerShell-Skript

Das Skript Enable-VdaSSL.ps1 aktiviert oder deaktiviert den SSL-Listener auf einem VDA. Dieses Skript ist im Ordner Support > Tools > SslSupport auf dem Installationsmedium verfügbar.

Wenn Sie SSL aktivieren, deaktiviert das Skript alle vorhandenen Windows-Firewallregeln für den angegebenen TCP-Port. Dann wird eine Regel hinzugefügt, die dem ICA-Dienst die Annahme eingehender Verbindungen nur auf dem SSL-TCP-Port gewährt. Außerdem werden die Windows-Firewallregeln für Folgendes deaktiviert:

- Citrix ICA (Standard: 1494)
- Citrix CGP (Standard: 2598)
- Citrix WebSocket (Standard: 8008)

Als Folge können Benutzer Verbindungen nur über SSL herstellen und nicht über unformatiertes ICA, CGP oder WebSocket.

Das Skript enthält die folgenden Syntax-Beschreibungen sowie zusätzliche Beispiele. Sie können diese Informationen mit einem Tool wie Notepad++ überprüfen.

Sie müssen einen der Parameter –Enable oder –Disable angeben, alle anderen Parameter sind optional.

#### Syntax

Enable-VdaSSL {-Enable | -Disable} [-SSLPort <port>] [-SSLMinVersion "<min-ssl-version>"] [-SSLCipherSuite "<suite>"] [-CertificateThumbPrint "<thumbprint>"]

| Parameter                          | Beschreibung                                                                                                                                                                      |
|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -Enable                            | Installiert und aktiviert den SSL-Listener auf dem VDA. Dieser Parameter oder der Parameter –Disable ist erforderlich.                                                            |
| -Disable                           | Deaktiviert den SSL-Listener auf dem VDA. Dieser Parameter oder der Parameter –Enable ist erforderlich. Wenn Sie diesen Parameter festlegen, sind keine anderen Parameter gültig. |
| –SSLPort <port>                    | SSL port: Standard: 443                                                                                                                                                           |
| -SSLMinVersion "<min-ssl-version>" | Mindestversion des SSL-Protokolls, zwischen Anführungszeichen. Gültige Werte: "SSL_3.0", "TLS_1.0", "TLS_1.1" und "TLS_1.2". Standard: "TLS_1.0"                                  |
| -SSLCipherSuite "<suite>"          | SSL-Verschlüsselungssammlung, zwischen Anführungszeichen. Gültige Werte: "GOV", "COM" und "ALL". Standard: "ALL"                                                                  |
| -                                  | Fingerabdruck des SSL-Zertifikats im Zertifikatspeicher, zwischen Anführungszeichen. Dieser Parameter wird normalerweise verwendet, wenn der Zertifikatspeicher über              |

| Parameter                              | Beschreibung                                                                                                                                                                                                                                      |
|----------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CertificateThumbPrint<br>"thumbprint"> | mehrere Zertifikate verfügt. Das Skript verwendet den Fingerabdruck, um das gewünschte Zertifikat auszuwählen. Standard: das erste verfügbare Zertifikat im Bereich "Lokaler Computer > Eigene Zertifikate > Zertifikate" des Zertifikatspeicher. |

## Beispiele

Das folgende Skript installiert und aktiviert den SSL-Listener. Für alle optionalen Parameter werden Standardwerte verwendet.

Enable-VdaSSL –Enable

Das folgende Skript installiert und aktiviert den TLS-Listener und gibt den TLS-Port 400 an sowie die Verschlüsselungssammlung GOV und als Mindestprotokollversion "SSL 1.2 SSL".

Enable-VdaSSL – Enable –SSLPort 400 'SSLMinVersion "TLS\_1.2" –SSLCipherSuite "GOV"

Das folgende Skript deaktiviert den SSL-Listener auf dem VDA.

Enable-VdaSSL –Disable

Manuelle Konfiguration von SSL auf einem VDA

Bei der manuellen Konfiguration von SSL auf einem VDA gewähren Sie dem privaten Schlüssel des SSL-Zertifikats allgemeinen Lesezugriff für den entsprechenden Dienst auf jedem VDA: NT SERVICE\PorticaService für einen VDA für Windows-Desktopbetriebssysteme oder NT SERVICE\TermService für einen VDA für Windows-Serverbetriebssysteme. Führen Sie auf der Maschine, auf der der VDA installiert ist, folgende Schritte aus:

1. Starten Sie die Microsoft Management Console (MMC): Start > Ausführen > mmc.exe.
2. Fügen Sie dem MMC das Zertifikat-Snap-In hinzu:
  1. Wählen Sie Datei > Snap-In hinzufügen/entfernen.
  2. Wählen Sie Zertifikate aus und klicken Sie auf Hinzufügen.
  3. Wählen Sie unter "Dieses Snap-In verwaltet die Zertifikate für:" die Option Computerkonto und klicken Sie dann auf Weiter.
  4. Wählen Sie unter "Wählen Sie den Computer aus, den dieses Snap-In verwalten soll" die Option Lokalen Computer und klicken Sie dann auf Fertig stellen.
3. Klicken Sie unter "Zertifikate (Lokaler Computer) > Eigene Zertifikate > Zertifikate" mit der rechten Maustaste auf das Zertifikat und wählen Sie Alle Aufgaben > Private Schlüssel verwalten.
4. In der Zugriffssteuerungsliste wird "Berechtigungen für <FriendlyName> private keys" angezeigt, wobei <FriendlyName> der Name des SSL-Zertifikats ist. Fügen Sie einen der folgenden Dienste hinzu und geben Sie ihm Lesezugriff:
  - Für einen VDA für Windows-Desktopbetriebssysteme "PORTICASERVICE"
  - Für einen VDA für Windows-Serverbetriebssysteme "TERMSERVICE"
5. Doppelklicken Sie auf das installierte SSL-Zertifikat. Wählen Sie im Dialogfeld des Zertifikats die Registerkarte "Details" aus und führen Sie einen Bildlauf nach unten durch. Klicken Sie auf Fingerabdruck.
6. Führen Sie regedit aus und navigieren Sie zu HKLM\SYSTEM\CurrentControlSet\Control\Terminal Server\Wds\icawd.
  1. Bearbeiten Sie den SSL-Fingerabdruckschlüssel und kopieren Sie den Fingerabdruckwert des SSL-Zertifikats in den binären Wert. Sie können unbekannte Elemente im Dialogfeld Binärwert bearbeiten ignorieren (z. B. "0000" und Sonderzeichen).
  2. Bearbeiten Sie den Schlüssel "SslEnabled" und ändern Sie den Wert für DWORD in "1". (Um SSL zu einem späteren Zeitpunkt zu deaktivieren, ändern Sie den Wert für DWORD in "0&".)
  3. Wenn Sie die Standardeinstellungen ändern möchten (optional), verwenden Sie Folgendes im gleichen Registrierungs Pfad:
    - SSLPort DWORD – SSL-Portnummer. Standard: 443.
    - SSLMinVersion DWORD – 1 = SSL 3.0, 2 = TLS 1.0, 3 = TLS 1.1, 4 = TLS 1.2. Standard: 2 (TLS 1.0).
    - SSLCipherSuite DWORD – 1 = GOV, 2 = COM, 3 = ALL. Standard: 3 (ALL).
7. Stellen Sie sicher, dass der SSL-TCP-Port in der Windows-Firewall geöffnet ist, wenn Sie nicht den Standardport 443 verwenden. (Wenn Sie die eingehende Regel für die Windows-Firewall erstellen, wählen Sie in den Eigenschaften die Optionen "Verbindung zulassen" und "Aktiviert" aus.)
8. Stellen Sie sicher, dass keine anderen Anwendungen oder Dienste (z. B. IIS) den SSL-TCP-Port verwenden.
9. Damit die Änderungen auf VDAs für Windows-Serverbetriebssysteme wirksam werden, starten Sie die Maschine neu. (Sie brauchen Maschinen mit VDAs für Windows-Desktopbetriebssysteme nicht neu starten.)

## Konfigurieren von SSL auf Bereitstellungsgruppen

Führen Sie diese Schritte für jede Bereitstellungsgruppe aus, die VDAs enthält, die Sie für SSL-Verbindungen konfiguriert haben.

1. Öffnen Sie in Studio die PowerShell-Konsole.
2. Führen Sie asnp Citrix.\* aus, um die Citrix Produkt-Cmdlets zu laden.
3. Führen Sie Get-BrokerAccessPolicyRule – DesktopGroupName '<delivery-group-name>' | Set-BrokerAccessPolicyRule –HdxSslEnabled \$true aus.  
<delivery-group-name> ist der Name der Bereitstellungsgruppe mit den VDAs.
4. Führen Sie Set-BrokerSite –DnsResolutionEnabled \$true aus.

## Problembehandlung

Wenn ein Verbindungsfehler auftritt, überprüfen Sie Systemereignisprotokoll des VDAs.

Wenn Sie Receiver für Windows verwenden und ein Verbindungsfehler auftritt (z. B. 1030), der auf einen SSL-Fehler hinweist, deaktivieren Sie Desktop Viewer und versuchen Sie erneut, die Verbindung herzustellen. Der Verbindungsfehler tritt zwar immer noch auf, aber u. U. wird eine Erklärung zu dem zugrunde liegenden SSL-Fehler angegeben (z. B. dass Sie beim Anfordern eines Zertifikats von der Zertifizierungsstelle die falsche Vorlage angegeben haben).

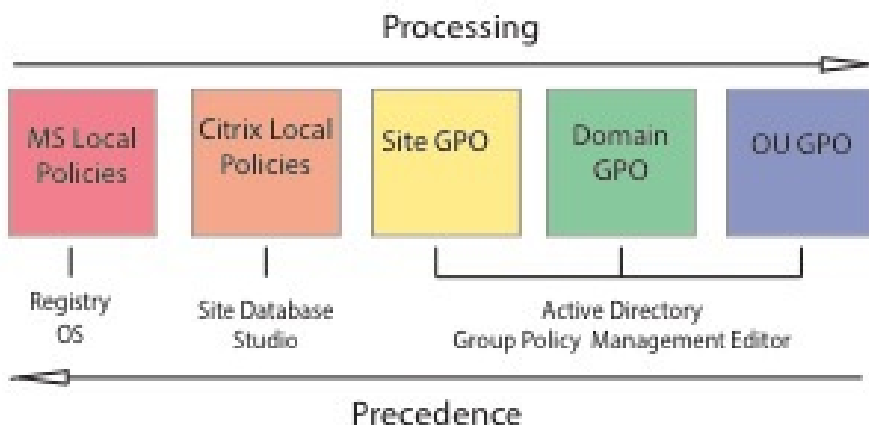


# Richtlinien

May 10, 2016

Richtlinien sind eine Sammlung von Einstellungen, die definieren, wie Sitzungen, Bandbreite und Sicherheit für eine Gruppe von Benutzern, Geräten oder Verbindungstypen verwaltet werden.

Richtlinieneinstellungen können auf physische und virtuelle Maschinen oder auf Benutzer angewendet werden. Sie können Einstellungen auf einzelne Benutzer auf lokaler Ebene oder auf Sicherheitsgruppen in Active Directory anwenden. Die Konfigurationen definieren bestimmten Kriterien und Regeln, und wenn Sie die Richtlinien nicht ausdrücklich zuweisen, gelten die Einstellungen für alle Verbindungen.



Sie können Richtlinien auf unterschiedliche Ebenen des Netzwerks zuweisen. Richtlinieneinstellungen, die auf der GPO-Ebene der Organisationseinheit zugewiesen werden, haben die höchste Priorität im Netzwerk. Richtlinien auf der Domänen-GPO-Ebene überschreiben Richtlinien auf der Ebene der Sitegruppenrichtlinienobjekte, die wiederum alle lokalen Richtlinien von Microsoft und Citrix überschreiben, die mit ihnen in Konflikt stehen.

Alle lokalen Citrix Richtlinien werden in der Citrix Studio-Konsole erstellt und verwaltet und in der Sitedatenbank gespeichert. Gruppenrichtlinien werden hingegen mit der Microsoft-Gruppenrichtlinien-Verwaltungskonsole (GPMC) erstellt und verwaltet und in Active Directory gespeichert. Lokale Microsoft-Richtlinien werden im Windows-Betriebssystem erstellt und in der Registrierung gespeichert.

Studio verwendet einen Modellierungsassistenten, mit dem Administratoren Konfigurationseinstellungen in Vorlagen und Richtlinien vergleichen können, um miteinander in Konflikt stehende und redundante Einstellungen zu eliminieren. Administratoren können Gruppenrichtlinienobjekte mit der Gruppenrichtlinien-Verwaltungskonsole festlegen, um Einstellungen zu konfigurieren und diese Einstellungen auf eine Zielgruppe von Benutzern auf unterschiedlichen Ebenen des Netzwerks anzuwenden.

Diese Gruppenrichtlinienobjekte werden in Active Directory gespeichert und die meisten IT-Mitarbeiter haben aus Sicherheitsgründen nur eingeschränkten Zugriff auf die Verwaltung dieser Einstellungen.

Einstellungen werden entsprechend ihrer Priorität und Bedingung zusammengefasst. Deaktivierte Einstellungen haben Vorrang vor aktivieren Einstellungen mit niedriger Priorität. Richtlinieneinstellungen, die nicht konfiguriert sind, werden ignoriert und setzen keine Einstellungen mit niedrigerer Priorität außer Kraft.

Lokale Richtlinien können auch mit Gruppenrichtlinien in Active Directory in Konflikt stehen. Abhängig von der Situation könnten sie einander überschreiben.

Alle Richtlinien werden in der folgenden Reihenfolge verarbeitet:

1. Der Endbenutzer meldet sich mit Domänenanmeldeinformationen an einer Maschine an.
2. Die Anmeldeinformationen werden an den Domänencontroller gesendet.
3. Active Directory wendet alle Richtlinien an (Endbenutzer, Endpunkt, Organisationseinheit und Domäne).
4. Der Endbenutzer meldet sich bei Receiver an und greift auf eine Anwendung oder einen Desktop zu.
5. Richtlinien von Citrix und Microsoft werden für den Endbenutzer und die Maschine, die die Ressource hostet, verarbeitet.
6. Active Directory bestimmt die Priorität für die Richtlinieneinstellungen und wendet sie auf die Registrierung des Endpunktgeräts und die Maschine an, auf der die Ressource gehostet wird.
7. Der Endbenutzer meldet sich von der Ressource ab. Citrix Richtlinien für Endbenutzer und Endpunktgerät sind nicht mehr aktiv.
8. Der Endbenutzer meldet sich vom Benutzergerät ab, das die GPO-Benutzerrichtlinien freigibt.
9. Der Endbenutzer schaltet das Gerät aus und die GPO-Maschinenrichtlinien werden freigegeben.

Beim Erstellen von Richtlinien für Benutzergruppen, Geräte und Maschinen haben einige Mitglieder u. U. unterschiedliche Anforderungen und benötigen Ausnahmen zu einigen Einstellungen. Ausnahmen werden durch Filter in Studio und in der Gruppenrichtlinien-Verwaltungskonsole erstellt und bestimmten, für wen oder was die Richtlinie gilt.

#### Verwandter Inhalt

- [Arbeiten mit Richtlinien](#)
- [Richtlinienvorlagen](#)
- [Erstellen von Richtlinien](#)
- [Vergleichen, Priorisieren, Modellieren und Problembehandlung für Richtlinien](#)
- [Standardrichtlinieneinstellungen](#)
- [Referenz für Richtlinieneinstellungen](#)

# Arbeiten mit Richtlinien

May 10, 2016

Durch das Konfigurieren von Citrix Richtlinien steuern Sie den Benutzerzugriff und die Sitzungsumgebung. Citrix Richtlinien sind die effizienteste Methode zum Steuern der Verbindungs-, Sicherheits- und Bandbreiteneinstellungen. Sie erstellen Richtlinien für bestimmte Benutzergruppen, Geräte oder Verbindungstypen. Jede Richtlinie kann mehrere Einstellungen enthalten.

## Tools zum Arbeiten mit Citrix Richtlinien

Sie können die folgenden Tools zum Arbeiten mit Citrix Richtlinien verwenden.

- **Studio:** Wenn Sie ein Citrix Administrator ohne Berechtigung zum Verwalten von Gruppenrichtlinien sind, verwenden Sie Studio, um Richtlinien für Ihre Site zu erstellen. Mit Studio erstellte Richtlinien werden in der Sitedatenbank gespeichert und Updates werden per Push auf den virtuellen Desktop übertragen, wenn der virtuelle Desktop beim Broker registriert wird oder ein Benutzer eine Verbindung mit dem virtuellen Desktop herstellt.
- **Editor für lokale Gruppenrichtlinien** (Snap-In der Microsoft Management Console): Wenn Sie in Ihrer Netzwerkumgebung Active Directory verwenden und Sie die Berechtigungen zur Verwaltung von Gruppenrichtlinien haben, können Sie den Editor für lokale Gruppenrichtlinien verwenden, um Richtlinien für Ihre Site zu erstellen. Die Einstellungen, die Sie konfigurieren, beeinträchtigen die Gruppenrichtlinienobjekte, die Sie in der Gruppenrichtlinien-Verwaltungskonsolle angeben.  
Wichtig: Sie müssen den Editor für lokale Gruppenrichtlinien zum Konfigurieren einiger Einstellungen verwenden, u. a. die Einstellungen zum Registrieren von VDAs bei einem Controller und die Einstellungen für Microsoft App-V Server.

## Reihenfolge und Priorität bei der Richtlinienverarbeitung

Gruppenrichtlinieneinstellungen (GPOs) werden in der folgenden Reihenfolge verarbeitet:

1. Lokale GPO
2. XenApp- bzw. XenDesktop-Site-GPO (in der Sitedatenbank gespeichert)
3. GPOs auf Siteebene
4. GPOs auf Domänenebene
5. Organisationseinheiten

Bei einem Konflikt können Richtlinieneinstellungen, die zuletzt verarbeitet werden, vorher verarbeitete überschreiben. Das heißt, dass Richtlinieneinstellungen die folgende Rangfolge haben:

1. Organisationseinheiten
2. GPOs auf Domänenebene
3. GPOs auf Siteebene
4. XenApp- bzw. XenDesktop-Site-GPO (in der Sitedatenbank gespeichert)
5. Lokale GPO

Beispiel: Ein Citrix Administrator erstellt eine Richtlinie (Richtlinie A) über Studio, mit der die Clientdateiumleitung für die Vertriebsmitarbeiter des Unternehmens aktiviert wird. Gleichzeitig erstellt ein anderer Administrator mit dem Gruppenrichtlinien-Editor eine Richtlinie (Richtlinie B), mit der die Clientdateiumleitung für die Vertriebsmitarbeiter deaktiviert wird. Wenn sich die Vertriebsmitarbeiter an den virtuellen Desktops anmelden, wird Richtlinie B angewendet und Richtlinie A ignoriert, da Richtlinie B auf der Domänenebene und Richtlinie A auf der Ebene der XenApp- bzw. XenDesktop-Site-GPOs verarbeitet wurde.

Beachten Sie jedoch, dass die Citrix Sitzungseinstellungen die gleichen Einstellungen in einer Active Directory-Richtlinie oder

einer Remotedesktop-Sitzungshostkonfiguration überschreiben, wenn ein Benutzer eine ICA- oder Remotedesktopprotokoll (RDP)-Sitzung startet. Zu diesen Einstellungen gehören solche, die mit typischen RDP-Clientverbindungseinstellungen zusammenhängen, wie Desktofhintergrund, Menüanimationen und das Anzeigeverhalten bei Drag & Drop.

Wenn Sie mehrere Richtlinien verwenden, können Sie Richtlinien, deren Einstellungen Konflikte verursachen, Prioritäten zuweisen. Weitere Informationen hierzu finden Sie unter [Vergleichen, Priorisieren, Modellieren und Problembehandlung für Richtlinien](#).

## Arbeitsablauf bei Citrix Richtlinien

Der Prozess für das Konfigurieren von Richtlinien ist:

1. Erstellen Sie die Richtlinie.
2. Konfigurieren Sie Richtlinieneinstellungen.
3. Weisen Sie die Richtlinie Benutzer- und Maschinenobjekten zu.
4. Weisen Sie der Richtlinie eine Priorität zu.
5. Prüfen Sie die effektive Richtlinie durch Ausführen des Gruppenrichtlinien-Modellierungsassistenten.

## Navigieren durch die Citrix Richtlinien und Einstellungen

Im Editor für lokale Gruppenrichtlinien werden Richtlinien und Einstellungen in zwei Hauptkategorien eingeteilt: Computerkonfiguration und Benutzerkonfiguration. Jede Kategorie hat einen Knoten für Citrix Richtlinien. Weitere Informationen zum Verwenden dieses Snap-Ins finden Sie in der Dokumentation von Microsoft.

In Studio sind die Richtlinieneinstellungen je nach Funktionalität bzw. Feature, für die bzw. das sie gelten, in Kategorien eingeteilt. Beispielsweise enthält der Bereich "Profilverwaltung" Richtlinieneinstellungen für die Profilverwaltung.

- Computereinstellungen (Richtlinieneinstellungen für Maschinen) definieren das Verhalten von virtuellen Desktops und werden beim Start eines virtuellen Desktops angewendet. Diese Einstellungen werden auch angewendet, wenn keine aktiven Benutzersitzungen auf dem virtuellen Desktop durchgeführt werden. Benutzerrichtlinieneinstellungen definieren die Benutzererfahrung bei Verbindungen über ICA. Benutzerrichtlinien werden angewendet, wenn ein Benutzer eine Verbindung über ICA herstellt oder erneut herstellt. Benutzerrichtlinien werden nicht angewendet, wenn ein Benutzer eine Verbindung über RDP herstellt oder sich direkt bei der Konsole anmeldet.

Sie greifen auf Richtlinien, Einstellungen oder Vorlagen zu, indem Sie im Navigationsbereich von Studio Richtlinien auswählen.

- Die Registerkarte **Richtlinien** listet alle Richtlinien auf. Wenn Sie eine Richtlinie auswählen, wird auf den Registerkarten rechts Folgendes angezeigt: Übersicht (Name, Priorität, Status: Aktiviert bzw. Deaktiviert, und Beschreibung), Einstellungen (Liste der konfigurierten Einstellungen) und Zugewiesen zu (Benutzer- und Maschinenobjekte, denen die Richtlinie momentan zugewiesen ist). Weitere Informationen finden Sie unter [Erstellen von Richtlinien](#).
- Auf der Registerkarte **Vorlagen** werden von Citrix bereitgestellte und benutzerdefinierte Vorlagen, die Sie erstellt haben, aufgelistet. Wenn Sie eine Vorlage auswählen, wird auf den Registerkarten rechts Folgendes angezeigt: Beschreibung (Zweck der Vorlage) und Einstellungen (Liste der konfigurierten Einstellungen). Weitere Informationen finden Sie unter [Richtlinienvorlagen](#).
- Mit der Registerkarte **Vergleich** können Sie die Einstellungen einer Richtlinie oder Vorlage mit denen in anderen Richtlinien oder Vorlagen vergleichen. Sie können beispielsweise Einstellungswerte prüfen, um sicherzustellen, dass optimale Verfahren eingehalten werden. Weitere Informationen finden Sie unter [Vergleichen, Priorisieren, Modellieren und Problembehandlung für Richtlinien](#).
- Auf der Registerkarte **Modellierung** können Sie Verbindungsszenarios mit Citrix Richtlinien simulieren. Weitere Informationen finden Sie unter [Vergleichen, Priorisieren, Modellieren und Problembehandlung für Richtlinien](#).

So suchen Sie nach einer Einstellung in einer Richtlinie oder Vorlage

1. Wählen Sie die Richtlinie oder Vorlage aus.
2. Wählen Sie im Aktionsbereich Richtlinie bearbeiten oder Vorlage bearbeiten.
3. Geben Sie auf der Seite Einstellungen den Namen der Einstellung ein.

Sie können die Suche verfeinern, indem Sie eine bestimmte Produktversion oder Kategorie (z. B. Bandbreite) auswählen oder indem Sie das Kontrollkästchen Nur ausgewählte anzeigen aktivieren. Außerdem können Sie nur die Einstellungen suchen, die der ausgewählten Richtlinie hinzugefügt wurden. Für eine ungefilterte Suche wählen Sie Alle Einstellungen.

- So suchen Sie nach einer Einstellung in einer Richtlinie
  1. Markieren Sie die Richtlinie.
  2. Geben Sie auf der Registerkarte Einstellungen den Namen der Einstellung ein.

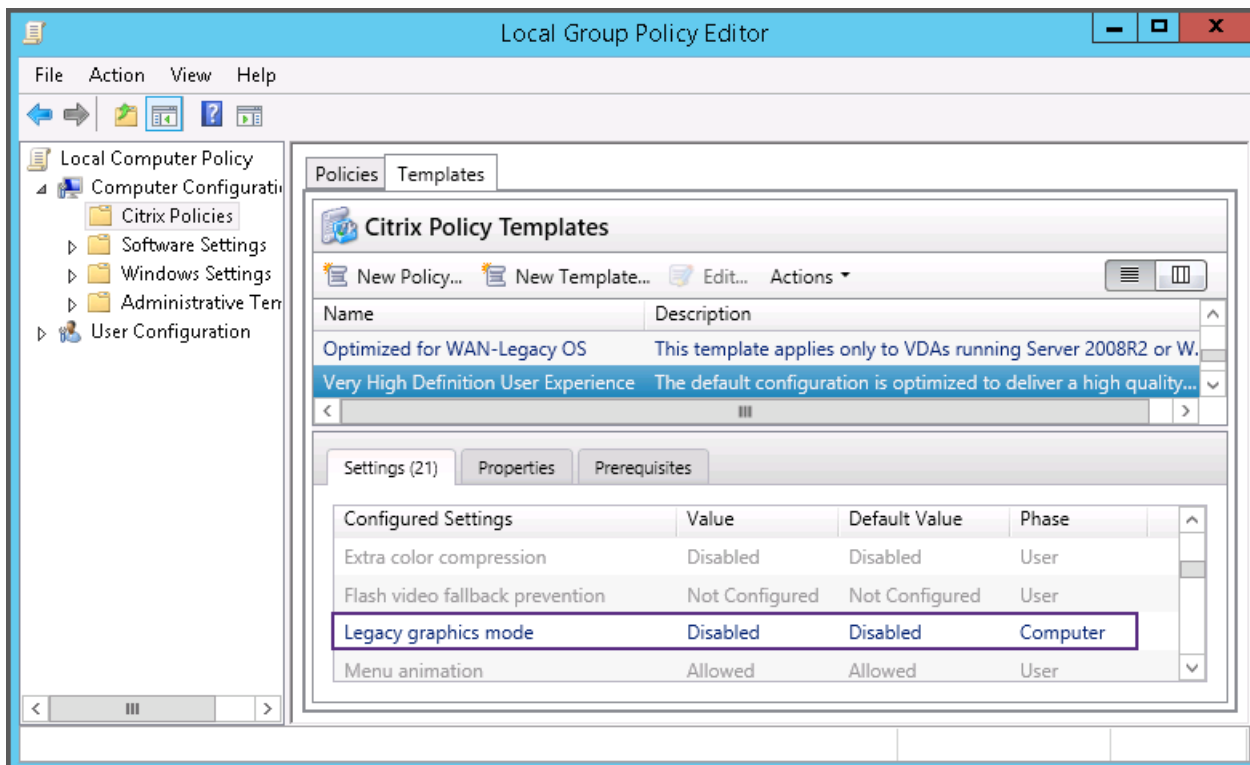
Sie können die Suche verfeinern, indem Sie eine bestimmte Produktversion oder Kategorie auswählen. Für eine ungefilterte Suche wählen Sie Alle Einstellungen.

Eine Richtlinie ist nach ihrer Erstellung völlig unabhängig von der verwendeten Vorlage. Sie können in das Feld "Beschreibung" eingeben, auf welcher Vorlage die neue Richtlinie basiert.

In Studio werden Richtlinien und Vorlagen in einer Liste angezeigt, unabhängig davon, ob sie Benutzer- oder Computereinstellungen oder beide Arten von Einstellungen enthalten, und sie können zudem mit Benutzer- und Computerfiltern angewendet werden.

Im Gruppenrichtlinien-Editor müssen Computer- und Benutzereinstellungen separat angewendet werden, selbst wenn sie auf einer Vorlage basieren, die beide Arten von Einstellungen enthält. In diesem Beispiel wird "Besonders gute High Definition-Benutzererfahrung" in Computerkonfiguration verwendet:

- Der Legacy-Grafikmodus ist eine Computereinstellung, die in einer mit dieser Vorlage erstellten Richtlinie verwendet wird.
- Die Benutzereinstellungen, grau dargestellt, werden nicht in einer mit dieser Vorlage erstellten Richtlinie verwendet.



# Richtlinienvorlagen

May 10, 2016

Vorlagen ermöglichen das Erstellen von Richtlinien von einem vordefinierten Ausgangspunkt aus. Integrierte Citrix Vorlagen sind für bestimmte Umgebungen oder Netzwerkbedingungen optimiert und können für Folgendes verwendet werden:

- Als Ausgangspunkt für das Erstellen Ihrer eigenen Richtlinien und Vorlagen, die Sie für verschiedene Sites freigeben können.
- Eine Referenz für einfacheres Vergleichen der Ergebnisse zwischen Bereitstellungen, da Sie sich auf Ergebnisse beziehen können, zum Beispiel "... wenn Sie die Citrix Vorlage x oder y verwenden ...".
- Eine Methode für das Übermitteln von Richtlinien an Citrix Support oder vertrauenswürdige Dritte durch Importieren oder Exportieren von Vorlagen.

Richtlinienvorlagen können importiert und exportiert werden. Weitere Vorlagen und Updates für die integrierten Vorlagen finden Sie unter [CTX202000](#).

Informationen zum Verwenden von Vorlagen zum Erstellen von Richtlinien finden Sie unter [CTX202330](#).

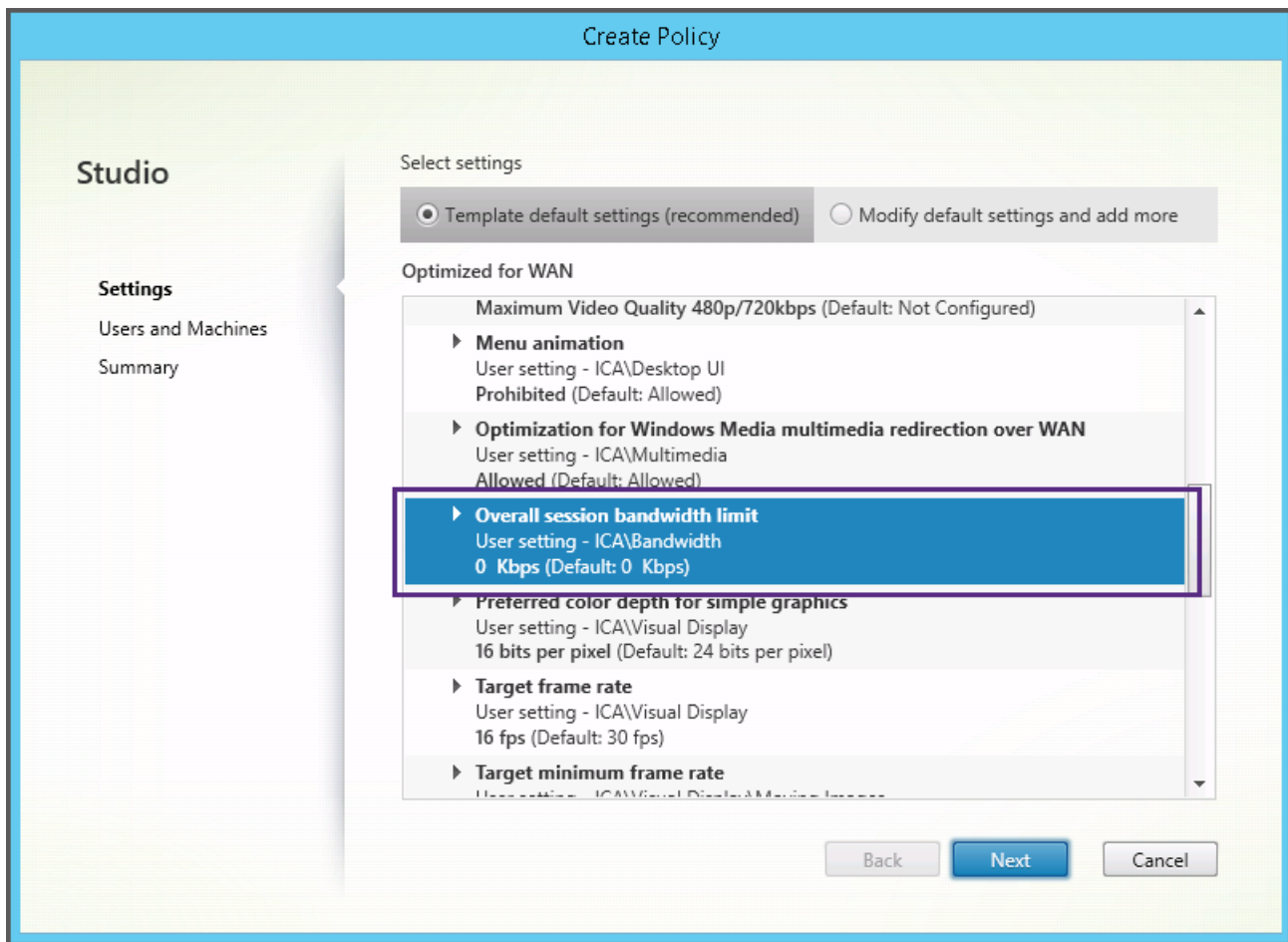
## Integrierte Citrix Vorlagen in 7.6 FP3

Das XenApp und XenDesktop 7.6 FP3 Gruppenrichtlinienverwaltungspaket (2.5.0.0) enthält die folgenden Richtlinienvorlagen, die die zuvor verfügbaren integrierten Citrix Vorlagen ersetzen:

- **Besonders gute High Definition-Benutzererfahrung:** Diese Vorlage erzwingt Standardeinstellungen, die die Benutzererfahrung optimieren. Verwenden Sie diese Vorlage in Szenarios, in denen mehrere Richtlinien in der Reihenfolge der Priorität verarbeitet werden.
- **Hohe Serverskalierbarkeit:** Mit dieser Vorlage können Sie Serverressourcen sparen, da Benutzererfahrung und Serverskalierbarkeit ausbalanciert werden. Die Vorlage ermöglicht eine gute Benutzererfahrung und erhöht gleichzeitig die Anzahl an Benutzern, die auf einem einzelnen Server gehostet werden können. Diese Vorlage verwendet keinen Videocodec für die Komprimierung von Grafiken und verhindert das serverseitige Multimediarendering.
- **Hohe Serverskalierbarkeit – Legacy-OS:** Diese Vorlage für hohe Serverskalierbarkeit gilt nur für VDAs, die auf Server 2008 R2, Windows 7 und älteren Betriebssystemen ausgeführt werden. Die Vorlage stützt sich auf den Legacy-Grafikmodus, der für diese Betriebssysteme effizienter ist.
- **Für WAN optimiert:** Verwenden Sie diese Vorlage bei aufgabenorientierten Mitarbeitern, die in Geschäftsstellen über eine gemeinsam genutzte WAN-Verbindung arbeiten oder bei Remotestandorten, wo über Verbindungen mit geringer Bandbreite auf Anwendungen mit grafisch einfachen Benutzeroberflächen und wenig Multimediainhalt zugegriffen wird. Mit dieser Vorlage werden für optimierte Bandbreiteneffizienz Kompromisse bei der Qualität der Videowiedergabe und der Serverskalierbarkeit gemacht.
- **Für WAN optimiert – Legacy-OS:** Diese Vorlage gilt nur für VDAs, die auf Server 2008 R2, Windows 7 oder älteren Betriebssystemen ausgeführt werden. Die Vorlage stützt sich auf den Legacy-Grafikmodus, der für diese Betriebssysteme effizienter ist.
- **Sicherheit und Steuerung:** Verwenden Sie diese Vorlage in Umgebungen mit niedriger Fehlertoleranz, um die in XenApp und XenDesktop standardmäßig aktivierten Features zu minimieren. Die in dieser Vorlage enthaltenen Einstellungen deaktivieren den Zugriff auf Drucker, Zwischenablage, Peripheriegeräte, Laufwerkzuordnung, Portumleitung und Flash-Beschleunigung auf Benutzergeräten. Bei Anwendung dieser Vorlage wird möglicherweise mehr Bandbreite genutzt und die Benutzerdichte pro Server verringert.

Wir empfehlen zwar, die integrierten Citrix Vorlagen mit den Standardeinstellungen zu verwenden, jedoch haben einige

Einstellungen keine empfohlenen Werte. Ein Beispiel ist die Einstellung "Bandbreitenlimit für Sitzung insgesamt" in der Vorlage "Für WAN optimiert". In diesem Fall wird die Einstellung durch die Vorlage verfügbar gemacht, damit der Administrator die Wirkung dieser Einstellung in diesem Szenario versteht.



Wenn Sie eine Bereitstellung (Richtlinienverwaltung und VDAs) haben, die älter als XenApp und XenDesktop 7.6 FP3 ist und Sie benötigen die Vorlagen "Hohe Serverskalierbarkeit" und "Für WAN optimiert", verwenden Sie ggf. die Legacy OS-Versionen dieser Vorlagen.

## Hinweis

Integrierte Vorlagen werden von Citrix erstellt und aktualisiert. Diese Vorlagen dürfen nicht geändert oder gelöscht werden.

## Erstellen und Verwalten von Vorlagen mit Studio

So erstellen Sie eine neue Vorlage basierend auf einer Vorlage

1. Wählen Sie im Studio-Navigationsbereich **Richtlinien** aus.
2. Wählen Sie die Registerkarte **Vorlagen** und dann die Vorlage, mit der Sie eine neue Vorlage erstellen möchten.
3. Wählen Sie im Aktionsbereich **Vorlage erstellen**.
4. Wählen und konfigurieren Sie die Richtlinieneinstellungen, die Sie in die Vorlage einschließen möchten. Entfernen Sie vorhandene Einstellungen, die nicht eingeschlossen werden sollen. Geben Sie einen Namen für die Vorlage ein.



Nachdem Sie auf **Fertig stellen** geklickt haben, wird die neue Vorlage auf der Registerkarte **Vorlagen** angezeigt.

So erstellen Sie eine neue Vorlage basierend auf einer Richtlinie

1. Wählen Sie im Studio-Navigationsbereich **Richtlinien** aus.
2. Wählen Sie die Registerkarte **Richtlinien** und dann die Richtlinie, mit der Sie die neue Vorlage erstellen möchten.
3. Wählen Sie im Aktionsbereich **Als Vorlage speichern**.
4. Wählen und konfigurieren Sie die neuen Richtlinieneinstellungen, die Sie in die Vorlage einschließen möchten. Entfernen Sie vorhandene Einstellungen, die nicht eingeschlossen werden sollen. Geben Sie einen Namen und eine Beschreibung für die Vorlage ein und klicken Sie auf **Fertig stellen**.

So importieren Sie eine Vorlage

1. Wählen Sie im Studio-Navigationsbereich **Richtlinien** aus.
2. Wählen Sie die Registerkarte **Vorlagen** und dann **Vorlage importieren**.
3. Wählen Sie die Vorlagendatei, die Sie importieren möchten, und klicken Sie auf **Öffnen**. Wenn Sie eine Vorlage importieren, die denselben Namen wie eine vorhandene hat, können Sie die vorhandene Vorlage überschreiben oder die Vorlage unter einem anderen Namen speichern, der automatisch generiert wird.

So exportieren Sie eine Vorlage

1. Wählen Sie im Studio-Navigationsbereich **Richtlinien** aus.
2. Wählen Sie die Registerkarte **Vorlagen** und dann **Vorlage exportieren**.
3. Legen Sie den Speicherort für die Vorlage fest, und klicken Sie auf **Speichern**.

Im angegebenen Speicherort wird eine GPT-Datei erstellt.

## Erstellen und Verwalten von Vorlagen mit dem Gruppenrichtlinien-Editor

Navigieren Sie im Gruppenrichtlinien-Editor zu Computerkonfiguration oder Benutzerkonfiguration. Erweitern Sie den Knoten Richtlinien und wählen Sie dann Citrix Richtlinien. Wählen Sie die entsprechende Aktion aus.

| Aufgabe                                                            | Anweisung                                                                                                                             |
|--------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|
| Erstellen einer Vorlage basierend auf einer vorhandenen Richtlinie | Wählen Sie auf der Registerkarte "Richtlinien" die Richtlinie aus, und wählen Sie dann Aktionen > Als Vorlage speichern.              |
| Erstellen einer Richtlinie basierend auf einer vorhandenen Vorlage | Wählen Sie auf der Registerkarte "Vorlagen" die entsprechende Vorlage aus und klicken Sie dann auf Neue Richtlinie.                   |
| Erstellen einer Vorlage basierend auf einer vorhandenen Vorlage    | Wählen Sie auf der Registerkarte "Vorlagen" die entsprechende Vorlage aus und klicken Sie dann auf Neue Vorlage.                      |
| Importieren einer Vorlage                                          | Wählen Sie auf der Registerkarte "Vorlagen" die Optionen Aktionen > Importieren.                                                      |
| Exportieren einer Vorlage                                          | Wählen Sie auf der Registerkarte "Vorlagen" die Optionen Aktionen > Exportieren.                                                      |
| Anzeigen der Vorlageneinstellungen                                 | Wählen Sie auf der Registerkarte "Vorlagen" die entsprechende Vorlage aus und klicken Sie dann auf die Registerkarte "Einstellungen". |
|                                                                    |                                                                                                                                       |



|                                                                            |                                                                                                                                                           |
|----------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Aufgabe</b><br>Anzeigen einer Zusammenfassung von Vorlageneigenschaften | <b>Anweisung</b><br>Wählen Sie auf der Registerkarte "Vorlagen" die entsprechende Vorlage aus und klicken Sie dann auf die Registerkarte "Eigenschaften". |
| Anzeigen von Vorlagenvoraussetzungen                                       | Wählen Sie auf der Registerkarte "Vorlagen" die entsprechende Vorlage aus und klicken Sie dann auf die Registerkarte "Voraussetzungen".                   |

## Vorlagen und delegierte Administration

Richtlinienvorlagen werden auf der Maschine gespeichert, auf der das Richtlinienverwaltungspaket installiert wurde. Das ist entweder die Maschine mit dem Delivery Controller oder die Maschine für die Verwaltung der Gruppenrichtlinienobjekte, jedoch nicht die Maschine mit der XenApp- und XenDesktop-Sitedatenbank. Daher werden die Richtlinienvorlagen nicht über die Rollen und Geltungsbereiche der delegierten Site-Verwaltung, sondern über Windows-Administratorrechte gesteuert.

Dies bedeutet, dass ein Administrator, der über Leseberechtigungen für die Site verfügt, beispielsweise neue Vorlagen erstellen kann. Da Vorlagen jedoch lokale Dateien sind, werden an der Umgebung keine Änderungen vorgenommen.

Benutzerdefinierte Vorlagen sind für das Benutzerkonto sichtbar, das sie erstellt hat, und werden im Windows-Profil des Benutzers gespeichert. Wenn Sie eine benutzerdefinierte Vorlage umfassender verfügbar machen möchten, erstellen Sie daraus eine Richtlinie oder exportieren Sie die Vorlage in einen freigegebenen Speicherort.

# Erstellen von Richtlinien

May 10, 2016

Legen Sie vor dem Erstellen einer Richtlinie fest, für welche Benutzergruppen oder Geräte sie gelten soll. Sie können Richtlinien basierend auf Aufgabenbereich, Verbindungstyp, Benutzergerät oder geografischer Position erstellen. Alternativ können Sie die gleichen Kriterien verwenden wie für Windows Active Directory-Gruppenrichtlinien.

Wenn Sie bereits eine Richtlinie für eine Gruppe erstellt haben, sollten Sie möglichst die Richtlinie bearbeiten und die entsprechenden Einstellungen konfigurieren, statt eine andere Richtlinie zu erstellen. Vermeiden Sie es, eine neue Richtlinie zu erstellen, deren einziger Zweck ist, eine bestimmte Einstellung zu aktivieren oder bestimmte Benutzer von der Richtlinie auszunehmen.

Wenn Sie eine Richtlinie erstellen, verwenden Sie als Basis eine Richtlinienvorlage und passen Sie die Einstellungen nach Bedarf an. Sie können die Richtlinie natürlich auch ohne Vorlage erstellen und alle benötigten Einstellungen hinzufügen.

## Richtlinieneinstellungen

Richtlinieneinstellungen können deaktiviert, aktiviert oder nicht konfiguriert sein. Standardmäßig sind Richtlinieneinstellungen nicht konfiguriert, d. h. sie wurden keiner Richtlinie hinzugefügt. Einstellungen werden nur angewendet, wenn sie einer Richtlinie hinzugefügt wurden.

Manche Richtlinieneinstellungen können einen der folgenden Zustände haben:

- Mit Zugelassen oder Nicht zugelassen wird die durch die Einstellung gesteuerte Aktion ermöglicht oder verhindert. In manchen Fällen dürfen Benutzer die Aktion der Einstellung in der Sitzung verwalten, in anderen dürfen sie das nicht. Wenn beispielsweise für Menüanimation die Einstellung auf "Zugelassen" festgelegt ist, können Benutzer Menüanimationen in ihrer Clientumgebung steuern.
- Mit Aktiviert oder Deaktiviert schalten Sie die Einstellung ein oder aus. Wenn Sie eine Einstellung deaktivieren, wird sie nicht durch Richtlinien mit geringerer Priorität aktiviert.

Manche Einstellungen steuern außerdem die Wirksamkeit von abhängigen Einstellungen. Die Einstellung Clientlaufwerkumleitung steuert beispielsweise, ob Benutzer auf die Laufwerke ihres Geräts zugreifen können. Damit Benutzer auf Netzlaufwerke zugreifen können, muss sowohl diese Einstellung als auch die Einstellung Clientnetzlaufwerke der Richtlinie hinzugefügt werden. Wenn die Einstellung Clientlaufwerkumleitung deaktiviert ist, können Benutzer nicht auf ihre Netzlaufwerke zugreifen, selbst wenn die Einstellung Clientnetzlaufwerke aktiviert ist.

In der Regel treten Änderungen an Richtlinieneinstellungen, die sich auf Maschinen auswirken, in Kraft, wenn der virtuelle Desktop neu gestartet wird oder wenn sich ein Benutzer anmeldet. Änderungen an Richtlinieneinstellungen, die Auswirkungen auf Benutzer haben, treten in Kraft, wenn sich die Benutzer das nächste Mal anmelden. Wenn Sie Active Directory verwenden, werden die Richtlinieneinstellungen aktualisiert, wenn Active Directory die Richtlinien in 90-Minuten-Intervallen erneut evaluiert. Die Richtlinieneinstellungen werden angewendet, wenn der virtuelle Desktop neu gestartet wird oder wenn sich ein Benutzer anmeldet.

Für manche Richtlinieneinstellungen können Sie einen Wert eingeben oder auswählen, wenn Sie die Einstellung der Richtlinie hinzufügen. Sie können die Konfiguration der Einstellung einschränken, indem Sie Standardwert verwenden wählen. Dadurch deaktivieren Sie die Konfiguration der Einstellung und nur der Standardwert der Einstellung darf verwendet werden, wenn die Richtlinie angewendet wird, unabhängig von dem Wert, der vor dem Aktivieren von Standardwert verwenden eingegeben wurde.

Bewährte Methoden:

- Weisen Sie Richtlinien Gruppen statt einzelnen Benutzern zu. Wenn Sie Richtlinien Gruppen zuweisen, werden Zuweisungen automatisch aktualisiert, wenn Sie Benutzer Gruppen hinzufügen oder sie daraus entfernen.
- Aktivieren Sie nicht widersprechende oder überlappende Einstellungen in der Konfiguration des Remotedesktop-Sitzungshosts. In manchen Fällen bietet die Remotedesktop-Sitzungshostkonfiguration ähnliche Funktionalität wie Citrix Richtlinienereinstellungen. Wählen Sie nach Möglichkeit für alle Einstellungen den gleichen Status (aktiviert oder deaktiviert), um die Problembehandlung zu erleichtern.
- Deaktivieren Sie Richtlinien, die nicht verwendet werden. Richtlinien, denen keine Einstellungen hinzugefügt wurden, verursachen unnötigen Verarbeitungsaufwand.

## Richtlinienzuweisungen

Wenn Sie eine Richtlinie erstellen und bestimmten Benutzer- und Maschinenobjekten zuweisen, wird sie gemäß bestimmter Kriterien oder Regeln auf Verbindungen angewendet. Basierend auf einer Kombination von Kriterien können Sie in der Regel beliebig viele Zuweisungen für eine Richtlinie hinzufügen. Wenn keine Zuweisung angegeben wurde, gilt die Richtlinie für alle Verbindungen.

In der folgenden Tabelle werden verfügbare Zuweisungen aufgelistet:

| Name                      | Anwendung der Richtlinie basierend auf                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Zugriffssteuerung         | Zugriffssteuerungsbedingungen, unter denen Clients eine Verbindung herstellen <ul style="list-style-type: none"> <li>• Verbindungstyp: Ob die Richtlinie auf Verbindungen anzuwenden ist, die mit oder ohne NetScaler Gateway hergestellt wurden.</li> <li>• NetScaler Gateway-Farmname: Name des virtuellen NetScaler Gateway-Servers.</li> <li>• Zugriffsbedingung: Name der zu verwendenden Endpunktanalyserichtlinie oder Sitzungsrichtlinie.</li> </ul> |
| Citrix CloudBridge        | Ob eine Benutzersitzung über Citrix CloudBridge gestartet wird.<br><br>Hinweis: Sie können einer Richtlinie nur eine Citrix CloudBridge-Zuweisung hinzufügen.                                                                                                                                                                                                                                                                                                |
| Client-IP-Adresse         | IP-Adresse des Benutzergeräts, das zum Verbinden der Sitzung verwendet wird <ul style="list-style-type: none"> <li>• IPv4-Beispiele: 12.0.0.0, 12.0.0.*, 12.0.0.1-12.0.0.70, 12.0.0.1/24</li> <li>• IPv6-Beispiele: 2001:0db8:3c4d:0015:0:abcd:ef12, 2001:0db8:3c4d:0015::/54</li> </ul>                                                                                                                                                                     |
| Clientname                | Name des Benutzergeräts <ul style="list-style-type: none"> <li>• Genaue Übereinstimmung: ClientABCName</li> <li>• Verwenden von Platzhalter: Client*Name</li> </ul>                                                                                                                                                                                                                                                                                          |
| Bereitstellungsgruppe     | Bereitstellungsgruppen-Mitgliedschaft                                                                                                                                                                                                                                                                                                                                                                                                                        |
| Bereitstellungsgruppentyp | Desktop- oder Anwendungstyp: privater Desktop, freigegebener Desktop, private Anwendung oder freigegebene Anwendung                                                                                                                                                                                                                                                                                                                                          |
| Organisationseinheit      | Organisationseinheit                                                                                                                                                                                                                                                                                                                                                                                                                                         |

| Tag<br>Name          | Tags<br>Anwendung der Richtlinie basierend auf                                                                                                                                           |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                      | <b>Hinweis:</b> Installieren Sie den unter <a href="#">CTX142439</a> verfügbaren Hotfix, um sicherzustellen, dass die Richtlinien bei der Verwendung von Tags richtig angewendet werden. |
| Benutzer oder Gruppe | Benutzer- oder Gruppenname                                                                                                                                                               |

Alle Richtlinien, die mit den Zuweisungen für die Verbindung übereinstimmen, werden bei der Anmeldung eines Benutzers identifiziert. Die Richtlinien werden nach Priorität sortiert und mehrere Instanzen jeder Einstellung werden verglichen. Die einzelnen Einstellungen werden gemäß der Richtlinien-Prioritätsreihenfolge angewendet. Jede deaktivierte Richtlinieneinstellung hat Vorrang vor einer aktivierten Richtlinieneinstellung, deren Priorität niedriger ist. Richtlinieneinstellungen, die nicht konfiguriert sind, werden ignoriert.

Wichtig: Bei der Konfiguration von Active Directory- und Citrix Richtlinien mit der Gruppenrichtlinien-Verwaltungskonsolle werden Zuweisungen und Einstellungen möglicherweise nicht wie erwartet angewendet. Weitere Informationen finden Sie unter [CTX127461](#).

Eine Richtlinie mit dem Namen "Ungefiltert" ist standardmäßig verfügbar.

- Wenn Sie Studio zur Verwaltung von Citrix Richtlinien verwenden, werden die Einstellungen, die Sie der Richtlinie "Ungefiltert" hinzufügen, auf alle Server, Desktops und Verbindungen einer Site angewendet.
- Wenn Sie mit dem Editor für lokale Gruppenrichtlinien Citrix Richtlinien verwalten, gelten Einstellungen, die Sie der Richtlinie "Ungefiltert" hinzufügen, für alle Sites und Verbindungen, die zu dem Geltungsbereich des Gruppenrichtlinienobjekts gehören, das die Richtlinie enthält. Beispiel: Die Organisationseinheit (OU) "Verkauf" enthält ein Gruppenrichtlinienobjekt "Verkauf-USA", das alle Mitarbeiter des US-Verkaufsteams einschließt. Das Gruppenrichtlinienobjekt "Verkauf-USA" ist mit einer Richtlinie "Ungefiltert" konfiguriert, die mehrere Benutzerrichtlinieneinstellungen enthält. Wenn der US-Verkaufsleiter sich an der Site anmeldet, werden die Einstellungen der Richtlinie "Ungefiltert" automatisch auf die Sitzung angewendet, weil der Benutzer Mitglied des Gruppenrichtlinienobjekts "Verkauf-USA" ist.

Der Modus einer Zuweisung entscheidet, ob die Richtlinie nur auf Verbindungen angewendet wird, die alle Zuweisungskriterien erfüllen. Wenn der Modus Zulassen (Standardwert) ist, wird die Richtlinie nur auf Verbindungen angewendet, die die Zuweisungskriterien erfüllen. Wenn der Modus Verweigern ist, wird die Richtlinie angewendet, wenn eine Verbindung die Zuweisungskriterien nicht erfüllt. Das folgende Beispiel zeigt, wie Zuweisungsmodi sich auf Citrix Richtlinien auswirken, wenn mehrere Zuweisungen vorhanden sind.

- **Beispiel: Zuweisungen des gleichen Typs mit unterschiedlichen Modi:** In Richtlinien mit zwei Zuweisungen des gleichen Typs, eine mit der Einstellung Zulassen und die andere mit der Einstellung Verweigern, hat die Zuweisung mit der Einstellung Verweigern Vorrang, wenn die Verbindung die Kriterien beider Zuweisungen erfüllt. Beispiel:  
Richtlinie 1 enthält die folgenden Zuweisungen:
  - Zuweisung A bestimmt die Verkaufsgruppe, und der Modus ist Zulassen
  - Zuweisung B bestimmt das Konto des Verkaufsleiters, und der Modus ist Verweigern
Da der Modus für Zuweisung B Verweigern ist, wird die Richtlinie nicht angewendet, wenn der Verkaufsleiter sich bei der Site anmeldet, obwohl er Mitglied der Verkaufsgruppe ist.
- **Beispiel: Zuweisungen unterschiedlichen Typs mit gleichen Modi:** In Richtlinien mit zwei oder mehr Zuweisungen unterschiedlichen Typs, für die Zulassen eingestellt ist, muss die Verbindung die Kriterien von mindestens einer Zuweisung jedes Typs erfüllen, damit die Richtlinie angewendet wird. Beispiel:  
Richtlinie 2 enthält die folgenden Zuweisungen:
  - Zuweisung C ist eine Benutzerzuweisung, die die Verkaufsgruppe angibt, und der Modus ist Zulassen

- Zuweisung D ist eine Client-IP-Adressenzuweisung, die 10.8.169.\* festlegt (das Unternehmensnetzwerk), und der Modus ist Zulassen.

Wenn der Verkaufsleiter sich im Büro bei der Site anmeldet, wird die Richtlinie angewendet, weil die Verbindung die Kriterien beider Zuweisungen erfüllt.

Richtlinie 3 enthält die folgenden Zuweisungen:

- Zuweisung E ist eine Benutzerzuweisung, die die Verkaufsgruppe angibt, und der Modus ist Zulassen
- Zuweisung F ist eine Zugriffssteuerungszuweisung, die NetScaler Gateway-Verbindungsbedingungen angibt, und der Modus ist Zulassen.

Wenn der Verkaufsleiter sich im Büro bei der Site anmeldet, wird die Richtlinie nicht angewendet, weil die Verbindung nicht die Kriterien von Zuweisung F erfüllt.

## Erstellen einer Richtlinie basierend auf einer Vorlage mit Studio

1. Wählen Sie im Studio-Navigationsbereich Richtlinien aus.
2. Wählen Sie die Registerkarte "Vorlagen" und wählen Sie dann eine Vorlage.
3. Wählen Sie im Aktionsbereich Richtlinie aus Vorlage erstellen.
4. Standardmäßig verwendet die neue Richtlinie alle Standardeinstellungen der Vorlage (das Optionsfeld Standardeinstellungen der Vorlage ist ausgewählt). Um die Einstellungen zu ändern, wählen Sie das Optionsfeld Standardeinstellungen ändern und Einstellungen hinzufügen, und fügen Sie Einstellungen hinzu oder entfernen Sie sie.
5. Legen Sie fest, wie die Richtlinie angewendet werden soll, indem Sie eine der folgenden Optionen auswählen:
  - Ausgewählten Benutzer- und Maschinenobjekten zuweisen und wählen Sie dann die Benutzer- und Maschinenobjekte, auf die Sie die Richtlinie anwenden möchten.
  - Allen Objekten in der Site zuweisen, damit die Richtlinie auf alle Benutzer- und Maschinenobjekte in der Site angewendet wird.
6. Geben Sie einen Namen für die Richtlinie ein (oder akzeptieren Sie den Standardwert). Empfehlenswert sind Richtlinienamen, die beschreiben, wer von der Richtlinie betroffen ist, z. B. Buchhaltung oder Remotebenutzer. Geben Sie optional eine Beschreibung ein.

Die Richtlinie ist standardmäßig aktiviert. Sie können sie deaktivieren. Wenn die Richtlinie aktiviert ist, kann sie sofort auf Benutzer, die sich anmelden, angewendet werden. Deaktivieren der Richtlinie verhindert, dass sie angewendet wird. Wenn Sie die Priorität der Richtlinie ändern müssen oder später weitere Einstellungen hinzufügen möchten, können Sie die Richtlinie deaktivieren, bis Sie damit fertig sind, und die Richtlinie dann anwenden.

## Erstellen einer Richtlinie mit Studio

1. Wählen Sie im Studio-Navigationsbereich Richtlinien aus.
2. Wählen Sie die Registerkarte Richtlinien.
3. Wählen Sie im Aktionsbereich Richtlinie erstellen.
4. Fügen Sie Richtlinieneinstellungen nach Bedarf hinzu und konfigurieren Sie diese.
5. Legen Sie fest, wie die Richtlinie angewendet werden soll, indem Sie eine der folgenden Optionen auswählen:
  - Ausgewählten Benutzer- und Maschinenobjekten zuweisen und wählen Sie dann die Benutzer- und Maschinenobjekte, auf die Sie die Richtlinie anwenden möchten.
  - Allen Objekten in der Site zuweisen, damit die Richtlinie auf alle Benutzer- und Maschinenobjekte in der Site angewendet wird.
6. Geben Sie einen Namen für die Richtlinie ein (oder akzeptieren Sie den Standardwert). Empfehlenswert sind Richtlinienamen, die beschreiben, wer von der Richtlinie betroffen ist, z. B. Buchhaltung oder Remotebenutzer. Geben Sie optional eine Beschreibung ein.

Die Richtlinie ist standardmäßig aktiviert. Sie können sie deaktivieren. Wenn die Richtlinie aktiviert ist, kann sie sofort auf

Benutzer, die sich anmelden, angewendet werden. Deaktivieren der Richtlinie verhindert, dass sie angewendet wird. Wenn Sie die Priorität der Richtlinie ändern müssen oder später weitere Einstellungen hinzufügen möchten, können Sie die Richtlinie deaktivieren, bis Sie damit fertig sind, und die Richtlinie dann anwenden.

## Erstellen und Verwalten von Richtlinien mit dem Gruppenrichtlinien-Editor

Navigieren Sie im Gruppenrichtlinien-Editor zu Computerkonfiguration oder Benutzerkonfiguration. Erweitern Sie den Knoten Richtlinien und wählen Sie dann Citrix Richtlinien. Wählen Sie die entsprechende Aktion aus.

| Aufgabe                                                            | Anweisung                                                                                                                                                    |
|--------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Erstellen einer Richtlinie                                         | Klicken Sie auf der Registerkarte "Richtlinien" auf Neu.                                                                                                     |
| Bearbeiten einer bestehenden Richtlinie                            | Wählen Sie auf der Registerkarte "Richtlinien" die entsprechende Richtlinie aus und klicken Sie dann auf Bearbeiten.                                         |
| Ändern der Priorität einer bestehenden Richtlinie                  | Wählen Sie auf der Registerkarte "Richtlinien" die entsprechende Richtlinie aus und klicken Sie dann auf Höhere Priorität oder Geringere Priorität.          |
| Anzeigen einer Zusammenfassung über eine Richtlinie                | Wählen Sie auf der Registerkarte "Richtlinien" die entsprechende Richtlinie aus und klicken Sie dann auf die Registerkarte Zusammenfassung.                  |
| Anzeigen und Ändern der Richtlinieneinstellungen                   | Wählen Sie auf der Registerkarte "Richtlinien" die entsprechende Richtlinie aus und klicken Sie dann auf die Registerkarte Einstellungen.                    |
| Anzeigen und Ändern der Richtlinienfilter                          | Wählen Sie auf der Registerkarte "Richtlinien" die entsprechende Richtlinie aus und klicken Sie dann auf die Registerkarte Einstellungen.                    |
| Aktivieren oder Deaktivieren einer Richtlinie                      | Wählen Sie auf der Registerkarte "Richtlinien" die entsprechende Richtlinie aus und klicken Sie dann auf Aktionen > Aktivieren oder Aktionen > Deaktivieren. |
| Erstellen einer Richtlinie basierend auf einer vorhandenen Vorlage | Wählen Sie auf der Registerkarte "Vorlagen" die entsprechende Vorlage aus und klicken Sie dann auf Neue Richtlinie.                                          |

# Vergleichen, Priorisieren, Modellieren und Problembehandlung für Richtlinien

Nov 02, 2016

Sie können mit mehreren Richtlinien Ihre Umgebung an die Anforderungen der Benutzer, basierend auf deren Aufgabengebiet, geografischem Standort oder Verbindungstyp anpassen. Beispielsweise sind Sie vielleicht aus Sicherheitsgründen gezwungen, Benutzergruppen, die regelmäßig mit sensiblen Daten arbeiten, Beschränkungen aufzuerlegen. Sie können eine Richtlinie erstellen, die Benutzer daran hindert, vertrauliche Daten auf ihren lokalen Clientlaufwerken zu speichern. Wenn jedoch manche Mitglieder dieser Benutzergruppe Zugang zu ihren lokalen Laufwerken benötigen, können Sie eine andere Richtlinie für diese Benutzer erstellen. Anschließend können Sie den beiden Richtlinien jeweils eine Priorität zuweisen und damit festlegen, welche Richtlinie Vorrang haben soll.

Wenn Sie mehrere Richtlinien verwenden, müssen Sie festlegen, wie Prioritäten zugewiesen und Ausnahmen erstellt werden und wie die wirksame Richtlinie bei Richtlinienkonflikten angezeigt wird.

In der Regel setzen Richtlinien ähnliche Einstellungen, die für die gesamte Site, für bestimmte Delivery Controller oder auf dem Benutzergerät konfiguriert wurden, außer Kraft. Die Ausnahme von diesem Prinzip sind Sicherheitseinstellungen. Die höchste Verschlüsselungseinstellung in der Umgebung, einschließlich Betriebssystem, und die Spiegelungseinstellung mit der größten Einschränkung haben immer Vorrang vor allen anderen Einstellungen und Richtlinien.

Citrix Richtlinien interagieren mit den Richtlinien, die Sie im Betriebssystem eingestellt haben. In einer Citrix Umgebung überschreiben Citrix Einstellungen die gleichen Einstellungen in einer Active Directory-Richtlinie oder in der Konfiguration des Remotedesktop-Sitzungshosts. Dazu gehören auch Einstellungen, die mit typischen Remotedesktopprotokoll-Clientverbindungseinstellungen zusammenhängen, wie Desktophintergrund, Menüanimation und das Anzeigeverhalten beim Drag & Drop. Manche Richtlinieneinstellungen, wie SecureICA, müssen mit den Richtlinien und Einstellungen im Betriebssystem übereinstimmen. Wenn anderswo ein höherer Verschlüsselungsgrad festgelegt wurde, kann die SecureICA-Richtlinieneinstellung oder die Einstellung beim Veröffentlichen einer Anwendung außer Kraft gesetzt werden.

Die beim Erstellen von Bereitstellungsgruppen angegebenen Verschlüsselungseinstellungen sollten beispielsweise den gleichen Verschlüsselungsgrad verwenden, den Sie an anderer Stelle in der Umgebung verwenden.

Hinweis: Wenn im zweiten Hop eines Double-Hop-Szenarios ein Desktopbetriebssystem-VDA eine Verbindung mit einem Serverbetriebssystem-VDA herstellt, ist die Wirkung der Citrix Richtlinien auf den Desktopbetriebssystem-VDA die gleiche wie beim Benutzergerät. Wenn Richtlinien beispielsweise das Zwischenspeichern von Bildern auf dem Benutzergerät festlegen, werden die Bilder, die während des zweiten Hop in einem Double-Hop-Szenario zwischengespeichert werden, auf der Desktopbetriebssystem-VDA-Maschine zwischengespeichert.

## Vergleichen von Richtlinien und Vorlagen

Sie können die Einstellungen einer Richtlinie oder Vorlage mit denen in anderen Richtlinien oder Vorlagen vergleichen. Beispielsweise ist die Prüfung von Einstellungswerten erforderlich, um sicherzustellen, dass optimale Verfahren eingehalten werden. Außerdem ist ggf. ein Vergleich von Einstellungen in einer Richtlinie oder Vorlage mit den Standardeinstellungen von Citrix erforderlich.

1. Wählen Sie im Studio-Navigationsbereich Richtlinien aus.
2. Klicken Sie auf die Registerkarte "Vergleich" und dann auf Auswählen.
3. Wählen Sie die Richtlinien oder Vorlagen aus, die Sie vergleichen möchten. Aktivieren Sie das Kontrollkästchen Mit Standardeinstellungen vergleichen, um Standardwerte im Vergleich einzuschließen.
4. Wenn Sie auf Vergleichen klicken, werden die konfigurierten Einstellungen in Spalten angezeigt.

5. Zum Anzeigen aller Einstellungen wählen Sie Alle Einstellungen anzeigen. Sie kehren zur Standardansicht zurück, indem Sie Gemeinsame Einstellungen anzeigen auswählen.

## Festlegen der Richtlinienpriorität

Durch Festlegen der Richtlinienpriorität definieren Sie, welche Richtlinie Vorrang hat, wenn es Konflikte gibt. Alle Richtlinien, die mit den Zuweisungen für die Verbindung übereinstimmen, werden bei der Anmeldung eines Benutzers identifiziert. Die Richtlinien werden nach Priorität sortiert und mehrere Instanzen jeder Einstellung werden verglichen. Die einzelnen Einstellungen werden gemäß der Richtlinien-Prioritätsreihenfolge angewendet.

Sie weisen Richtlinien Prioritäten zu, indem Sie ihnen unterschiedliche Prioritätswerte in Studio geben. Neue Richtlinien erhalten standardmäßig die niedrigste Priorität. Falls widersprüchliche Richtlinieneinstellungen auftreten, setzt eine Richtlinie mit einem höheren Prioritätswert (eine Priorität von "1" hat die höchste Priorität) eine Richtlinie mit einem niedrigeren Prioritätswert außer Kraft. Einstellungen werden nach der Priorität und dem Zustand der Einstellung, z. B. ob die Einstellung deaktiviert oder aktiviert ist, zusammengeführt. Jede deaktivierte Einstellung hat Vorrang vor einer aktivierten Einstellung, deren Priorität niedriger ist. Richtlinieneinstellungen, die nicht konfiguriert sind, werden ignoriert und setzen keine Einstellungen mit niedrigerer Priorität außer Kraft.

1. Wählen Sie im Studio-Navigationsbereich Richtlinien aus. Wählen Sie die Registerkarte "Richtlinien" aus.
2. Wählen Sie eine Richtlinie.
3. Wählen Sie im Aktionsbereich Geringere Priorität oder Höhere Priorität.

## Ausnahmen

Wenn Sie Richtlinien für Benutzergruppen, Benutzergeräte oder Maschinen erstellen, werden Sie möglicherweise feststellen, dass für einige Mitglieder einer Gruppe Ausnahmen zu einigen Einstellungen erstellt werden müssen. Sie können Ausnahmen wie folgt erstellen:

- Erstellen Sie eine Richtlinie für die Gruppenmitglieder, für die Ausnahmen erforderlich sind, und stufen Sie die Richtlinie mit höherer Priorität ein als die Richtlinie für die gesamte Gruppe.
- Verwenden Sie den Modus Verweigern in einer Zuweisung, die Sie der Richtlinie hinzufügen.

Die Zuweisung im Modus Verweigern wendet eine Richtlinie nur auf Verbindungen an, die nicht den Zuweisungskriterien entsprechen. Beispielsweise könnte eine Richtlinie folgende Zuweisungen enthalten:

- Zuweisung A ist eine Client-IP-Adressenzuweisung, die den Bereich 208.77.88.\* festlegt, und der Modus ist Zulassen
- Zuweisung B ist eine Benutzerzuweisung, die ein spezifisches Benutzerkonto angibt, und der Modus ist Verweigern

Die Richtlinie wird auf alle Benutzer angewendet, die sich mit IP-Adressen aus dem in Zuweisung A festgelegten Bereich bei der Site anmelden. Die Richtlinie wird aber nicht auf den Benutzer angewendet, der sich mit dem in Zuweisung B festgelegten Konto anmeldet, obwohl dem Computer dieses Benutzers eine IP-Adresse aus dem in Zuweisung A festgelegten Bereich zugewiesen wurde.

## Ermitteln der auf eine Verbindung angewendeten Richtlinien

Manchmal reagiert eine Verbindung nicht wie erwartet, weil mehrere Richtlinien gelten. Wenn eine Richtlinie mit einer höheren Priorität auf eine Verbindung angewendet wird, kann sie Einstellungen, die Sie in der ursprünglichen Richtlinie konfigurieren, außer Kraft setzen. Sie können ermitteln, wie die Richtlinieneinstellungen am Ende für eine Verbindung zusammengeführt werden, indem sie den Richtlinienresultatsatz berechnen.

Sie berechnen den Richtlinienresultatsatz mit folgenden Methoden:

- Verwenden Sie den Assistenten für die Citrix Gruppenrichtlinienmodellierung, um ein Verbindungsszenario zu simulieren und festzustellen, wie Citrix Richtlinien angewendet werden können. Sie können Bedingungen für ein Verbindungsszenario angeben, z. B. Domänencontroller, Benutzer, Citrix Richtlinienzuweisungsbewertungen und simulierte



Umgebungseinstellungen wie langsame Netzwerkverbindungen. Der von dem Assistenten erstellte Bericht listet die Citrix Richtlinien auf, die in dem Szenario wahrscheinlich wirksam werden. Wenn Sie beim Controller als Domänenbenutzer angemeldet sind, berechnet der Assistent den Richtlinienresultatsatz anhand von Richtlinieneinstellungen für die Site und Active Directory-Gruppenrichtlinienobjekten.

- Verwenden Sie das Tool "Gruppenrichtlinienergebnisse", um einen Bericht zu erstellen, der beschreibt, welche Citrix Richtlinien für einen bestimmten Benutzer oder einen bestimmten Controller angewendet werden. Das Tool "Gruppenrichtlinienergebnisse" unterstützt Sie dabei, den aktuellen Zustand von Gruppenrichtlinienobjekten in der Umgebung zu evaluieren, und generiert einen Bericht, in dem beschrieben wird, wie diese Objekte, einschließlich Citrix Richtlinien, derzeit auf einen bestimmten Benutzer und Controller angewendet werden.

Sie können den Assistenten für die Citrix Gruppenrichtlinienmodellierung im Bereich "Aktionen" in Studio starten. Sie können beide Tools in der Gruppenrichtlinien-Verwaltungskonsole von Windows starten.

Wenn Sie den Assistenten für die Citrix Gruppenrichtlinienmodellierung oder das Tool "Gruppenrichtlinienergebnisse" über die Gruppenrichtlinien-Verwaltungskonsole ausführen, werden die mit Studio erstellten Site-Richtlinieneinstellungen nicht in den Richtlinienresultatsatz einbezogen.

Um sicherzustellen, dass Sie den umfassendsten Richtlinienresultatsatz erhalten, empfiehlt Citrix das Starten des Assistenten für die Citrix Gruppenrichtlinienmodellierung über Studio, es sei denn, Sie erstellen Richtlinien nur über die Gruppenrichtlinien-Verwaltungskonsole.

## Verwenden des Assistenten für die Citrix Gruppenrichtlinienmodellierung

Öffnen Sie den Assistenten für die Citrix Gruppenrichtlinienmodellierung mit einer der folgenden Optionen:

- Wählen Sie Richtlinien im Navigationsbereich von Studio, wählen Sie dann die Registerkarte "Modellierung" und dann im Aktionsbereich die Option Modellierungsassistenten starten.
- Starten Sie die Gruppenrichtlinien-Verwaltungskonsole (gpmc.msc), klicken Sie mit der rechten Maustaste in der Konsolenstruktur auf Citrix Gruppenrichtlinienmodellierung und wählen Sie dann Citrix Gruppenrichtlinienmodellierungsassistent aus.

Folgen Sie den Anweisungen des Assistenten, um den Domänencontroller, Benutzer, Computer, Umgebungseinstellungen und Citrix Zuweisungskriterien für die Simulation auszuwählen. Wenn Sie auf Fertig stellen klicken, erstellt der Assistent einen Bericht mit den Modellierungsergebnissen. In Studio wird der Bericht im mittleren Bereich unter der Registerkarte Modellierung angezeigt.

Zum Anzeigen des Berichts wählen Sie Modellierungsbericht anzeigen.

## Problembehandlung bei Richtlinien

Für Benutzer, IP-Adressen und andere zugewiesene Objekte können mehrere Richtlinien gleichzeitig gelten. Dies kann zu Konflikten führen, wenn eine Richtlinie sich nicht wie erwartet verhält. Wenn Sie den Citrix Gruppenrichtlinienmodellierungsassistenten oder das Gruppenrichtlinienergebnisse-Tool ausführen, entdecken Sie möglicherweise, dass keine Richtlinien auf die Benutzerverbindungen angewendet werden. In diesen Fall sind Benutzer nicht von Richtlinieneinstellungen betroffen, wenn sie sich unter Bedingungen mit Anwendungen verbinden, die den Richtlinienkriterien entsprechen. Dies passiert in folgenden Fällen:

- Keine Richtlinie hat eine Zuweisung, die den Richtlinienkriterien entspricht.
- Richtlinien, die der Zuweisung entsprechen, haben keine konfigurierten Einstellungen.
- Richtlinien, die der Zuweisung entsprechen, sind deaktiviert.

Wenn Sie Richtlinieneinstellungen auf Verbindungen anwenden möchten, die bestimmten Kriterien entsprechen, stellen Sie

Folgendes sicher:

- Die Richtlinien, die auf diese Verbindungen angewendet werden sollen, sind aktiviert.
- In den Richtlinien, die Sie anwenden möchten, sind die geeigneten Einstellungen konfiguriert.

# Standardrichtlinieneinstellungen

Nov 02, 2016

Die folgenden Tabellen enthalten Richtlinieneinstellungen, die Standardeinstellungen und die VDA-Versionen, für die sie gelten.

## ICA

| Name                                                          | Standardeinstellung     | VDA                                                                                       |
|---------------------------------------------------------------|-------------------------|-------------------------------------------------------------------------------------------|
| Clientzwischenablagenumleitung                                | Zugelassen              | Alle VDA-Versionen                                                                        |
| Desktop starten                                               | Nicht zugelassen        | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS                                   |
| ICA-Listener - Verbindungstimeout                             | 120.000 Millisekunden   | VDA 5, 5.5, 5.6 Feature Pack 1, VDA für Desktop-OS 7 bis zum aktuellen VDA für Desktop-OS |
| ICA-Listenerportnummer                                        | 1494                    | Alle VDA-Versionen                                                                        |
| Starten nicht-veröffentlichter Programme bei Clientverbindung | Nicht zugelassen        | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS                                   |
| Zum Schreiben in Clientzwischenablage zugelassene Formate     | Keine Formate angegeben | VDA 7.6                                                                                   |
| Schreiben in Clientzwischenablage einschränken                | Nicht zugelassen        | VDA 7.6                                                                                   |
| Schreiben in Sitzungszwischenablage einschränken              | Nicht zugelassen        | VDA 7.6                                                                                   |
| Zum Schreiben in Sitzungszwischenablage zugelassene Formate   | Keine Formate angegeben | VDA 7.6                                                                                   |

## ICA/Adobe Flash-Bereitstellung/Flash-Umleitung

| Name                                                 | Standardeinstellung | VDA         |
|------------------------------------------------------|---------------------|-------------|
| Verhindern von Flash-Videofallback                   | Nicht konfiguriert  | VDA 7.6 FP3 |
| Fehler beim Verhindern von Flash-Videofallback *.swf |                     | VDA 7.6 FP3 |

## ICA/Audio

| Name                    | Standardeinstellung          | VDA                                                     |
|-------------------------|------------------------------|---------------------------------------------------------|
| Audio Plug & Play       | Zugelassen                   | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS |
| Audioqualität           | Hoch - High Definition Audio | Alle VDA-Versionen                                      |
| Clientaudioumleitung    | Zugelassen                   | Alle VDA-Versionen                                      |
| Clientmikrofonumleitung | Zugelassen                   | Alle VDA-Versionen                                      |

#### ICA/Automatische Wiederverbindung von Clients

| Name                                                             | Standardeinstellung                                  | VDA |
|------------------------------------------------------------------|------------------------------------------------------|-----|
| Automatische Wiederverbindung von Clients                        | Zugelassen                                           | VDA |
| Authentifizierung bei automatischer Wiederverbindung von Clients | Keine Authentifizierung erforderlich                 | VDA |
| Protokollierung der automatischen Wiederverbindung von Clients   | Kein Protokollieren von Wiederverbindungsereignissen | VDA |

#### ICA/Bandwidth

| Name                                                      | Standardeinstellung | VDA                                                                                                                                  |
|-----------------------------------------------------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| Bandbreitenlimit für die Audioumleitung                   | 0 KBit/s            | VDA                                                                                                                                  |
| Bandbreitenlimit für die Audioumleitung (Prozent)         | 0                   | VDA                                                                                                                                  |
| Bandbreitenlimit für Client-USB-Geräteumleitung           | 0 KBit/s            | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Bandbreitenlimit für Client-USB-Geräteumleitung (Prozent) | 0                   | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Bandbreitenlimit für Zwischenablagenumleitung             | 0 KBit/s            | Alle VDA-Versionen                                                                                                                   |
| Bandbreitenlimit für Zwischenablagenumleitung (Prozent)   | 0                   | Alle VDA-Versionen                                                                                                                   |

| Name                                                                    | Standardeinstellung | VDA                                                                                                                                  |
|-------------------------------------------------------------------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| Bandbreitenlimit für COM-Portumleitung                                  | 0 KBit/s            | Alle VDA-Versionen; bei VDA 7.x müssen Sie diese Einstellung in der Registrierung konfigurieren.                                     |
| Bandbreitenlimit für COM-Portumleitung (Prozent)                        | 0                   | Alle VDA-Versionen; bei VDA 7.x müssen Sie diese Einstellung in der Registrierung konfigurieren.                                     |
| Bandbreitenlimit für Dateiumleitung                                     | 0 KBit/s            | Alle VDA-Versionen                                                                                                                   |
| Bandbreitenlimit für Dateiumleitung (Prozent)                           | 0                   | Alle VDA-Versionen                                                                                                                   |
| Bandbreitenlimit für HDX MediaStream-Multimediabeschleunigung           | 0 KBit/s            | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Bandbreitenlimit für HDX MediaStream-Multimediabeschleunigung (Prozent) | 0                   | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Bandbreitenlimit für LPT-Portumleitung                                  | 0 KBit/s            | Alle VDA-Versionen; bei VDA 7.x müssen Sie diese Einstellung in der Registrierung konfigurieren.                                     |
| Bandbreitenlimit für LPT-Portumleitung (Prozent)                        | 0                   | Alle VDA-Versionen; bei VDA 7.x müssen Sie diese Einstellung in der Registrierung konfigurieren.                                     |
| Bandbreitenlimit für Sitzung insgesamt                                  | 0 KBit/s            | Alle VDA-Versionen                                                                                                                   |
| Bandbreitenlimit für Druckerumleitung                                   | 0 KBit/s            | Alle VDA-Versionen                                                                                                                   |
| Bandbreitenlimit für Druckerumleitung (Prozent)                         | 0                   | Alle VDA-Versionen                                                                                                                   |
| Bandbreitenlimit für TWAIN-Geräteumleitung                              | 0 KBit/s            | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Bandbreitenlimit für TWAIN-Geräteumleitung (Prozent)                    | 0                   | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |

#### ICA/Clientsensoren

| Name                                                                  | Standardeinstellung | VDA                                                                                                                             |
|-----------------------------------------------------------------------|---------------------|---------------------------------------------------------------------------------------------------------------------------------|
| Anwendungen können den physischen Standort des Clientgeräts verwenden | Nicht zugelassen    | VDA 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |

| Name | Standardeinstellung | VDA |
|------|---------------------|-----|
|------|---------------------|-----|

| Name                                       | Standardeinstellung                                                           | VDA                                                                |
|--------------------------------------------|-------------------------------------------------------------------------------|--------------------------------------------------------------------|
| Desktopgestaltungsumleitung                | Deaktiviert (7.6 FP3 bis aktuelle Version)<br><br>Aktiviert (5.6 bis 7.6 FP2) | VDA 5.6, VDA für Desktopbetriebssysteme 7 bis aktuelle VDA-Version |
| Grafikqualität<br>Desktopgestaltung        | Mittel                                                                        | VDA 5.6, VDA für Desktopbetriebssysteme 7 bis aktuelle VDA-Version |
| Desktophintergrund                         | Zugelassen                                                                    | Alle VDA-Versionen                                                 |
| Menüanimation                              | Zugelassen                                                                    | Alle VDA-Versionen                                                 |
| Fensterinhalt beim<br>Verschieben anzeigen | Zugelassen                                                                    | Alle VDA-Versionen                                                 |

#### ICA/Endbenutzerüberwachung

| Name                                                 | Standardeinstellung | VDA                |
|------------------------------------------------------|---------------------|--------------------|
| ICA-Roundtripberechnung                              | Aktiviert           | Alle VDA-Versionen |
| Intervall für ICA-Roundtripberechnung                | 15 Sekunden         | Alle VDA-Versionen |
| ICA-Roundtrip für Verbindungen im Leerlauf berechnen | Deaktiviert         | Alle VDA-Versionen |

#### ICA/Enhanced Desktop Experience

| Name                        | Standardeinstellung | VDA                                                     |
|-----------------------------|---------------------|---------------------------------------------------------|
| Enhanced Desktop Experience | Zugelassen          | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS |

#### ICA/Dateiumleitung

| Name                                     | Standardeinstellung | VDA                |
|------------------------------------------|---------------------|--------------------|
| Clientlaufwerke<br>automatisch verbinden | Zugelassen          | Alle VDA-Versionen |
| Clientlaufwerkumleitung                  | Zugelassen          | Alle VDA-Versionen |

| Lokale Name                                    | Zugelassen  | Alle VDA-Versionen                                                                                                                   |
|------------------------------------------------|-------------|--------------------------------------------------------------------------------------------------------------------------------------|
| Clientfestplattenlaufwerke                     | Standard    | VDA                                                                                                                                  |
| Clientdiskettenlaufwerke                       | Zugelassen  | Alle VDA-Versionen                                                                                                                   |
| Clientnetzlaufwerke                            | Zugelassen  | Alle VDA-Versionen                                                                                                                   |
| Optische Clientlaufwerke                       | Zugelassen  | Alle VDA-Versionen                                                                                                                   |
| Clientwechsellaufwerke                         | Zugelassen  | Alle VDA-Versionen                                                                                                                   |
| Host-zu-Client-Umleitung                       | Deaktiviert | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS                                                                              |
| Clientlaufwerksbuchstaben erhalten             | Deaktiviert | VDA 5, 5.5, 5.6 Feature Pack 1, VDA für Desktop-OS 7 bis zum aktuellen VDA für Desktop-OS                                            |
| Schreibgeschützter Zugriff auf Clientlaufwerke | Deaktiviert | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Umleitung spezieller Ordner                    | Zugelassen  | Nur Webinterface-Bereitstellungen; VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS                                           |
| Asynchrones Schreiben verwenden                | Deaktiviert | Alle VDA-Versionen                                                                                                                   |

## ICA/Graphics

| Name                                    | Standardeinstellung          | VDA                                                                                                                                  |
|-----------------------------------------|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| Anzeigespeicherlimit                    | 65536 KBit                   | VDA 5, 5.5, 5.6 Feature Pack 1, VDA für Desktop-OS 7 bis zum aktuellen VDA für Desktop-OS                                            |
| Herabsetzungspräferenz für Anzeigemodus | Zuerst Farbtiefe herabsetzen | Alle VDA-Versionen                                                                                                                   |
| Dynamische Fenstervorschau              | Aktiviert                    | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Bildzwischenspeicherung                 | Aktiviert                    | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Legacygrafikmodus                       | Deaktiviert                  | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS                              |
|                                         |                              |                                                                                                                                      |

| Name                                                       | Standardeinstellung | VDA                                                     |
|------------------------------------------------------------|---------------------|---------------------------------------------------------|
| Maximal zugelassene Farbtiefe                              | 32 Bit pro Pixel    | Alle VDA-Versionen                                      |
| Benutzer beim Herabsetzen des Anzeigemodus benachrichtigen | Deaktiviert         | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS |
| Warteschlange und Verwerfen                                | Aktiviert           | Alle VDA-Versionen                                      |

## ICA/Graphics/Caching

| Name                                | Standardeinstellung | VDA                                                     |
|-------------------------------------|---------------------|---------------------------------------------------------|
| Schwellenwert für permanenten Cache | 3000000 Bit/s       | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS |

## ICA/Keep-Alive

| Name                     | Standardeinstellung                   | VDA                |
|--------------------------|---------------------------------------|--------------------|
| ICA-Keep-Alive - Timeout | 60 Sekunden                           | Alle VDA-Versionen |
| ICA-Keep-Alives          | Keine ICA-Keep-Alive-Meldungen senden | Alle VDA-Versionen |

## ICA/Zugriff auf lokale Anwendungen

| Name                         | Standardeinstellung   | VDA                                                                                                     |
|------------------------------|-----------------------|---------------------------------------------------------------------------------------------------------|
| Lokalen App-Zugriff zulassen | Nicht zugelassen      | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| URL-Umleitungssperrliste     | Keine Sites angegeben | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| URL-Umleitungspositivliste   | Keine Sites angegeben | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |

## ICA/Mobilerfahrung

| Name                                  | Standardeinstellung | VDA                                                                                                                             |
|---------------------------------------|---------------------|---------------------------------------------------------------------------------------------------------------------------------|
| Automatische Anzeige der Tastatur     | Nicht zugelassen    | VDA 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Für Fingereingabe optimierten Desktop | Zugelassen          | VDA 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |



| starten<br>Name               | Standardeinstellung | Desktop-OS<br>VDA                                                                                                               |
|-------------------------------|---------------------|---------------------------------------------------------------------------------------------------------------------------------|
|                               |                     | Diese Einstellung ist deaktiviert und für Windows 10-Maschinen nicht verfügbar.                                                 |
| Kombinationsfelder<br>remoten | Nicht zugelassen    | VDA 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |

## ICA/Multimedia

| Name                                                                             | Standardeinstellung | VDA                                                                                                     |
|----------------------------------------------------------------------------------|---------------------|---------------------------------------------------------------------------------------------------------|
| Videoqualität beschränken                                                        | Nicht konfiguriert  | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Multimediakonferenzen                                                            | Zugelassen          | Alle VDA-Versionen                                                                                      |
| Optimierung von Windows Media-Multimediaumleitung über WAN                       | Zugelassen          | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| GPU für die Optimierung von Windows Media-Multimediaumleitung über WAN verwenden | Nicht zugelassen    | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Richtlinieneinstellung für Videolastverwaltung                                   | Nicht konfiguriert  | VDA 7.6 FP3                                                                                             |
| Clientseitiger Abruf von Windows Media-Inhalten                                  | Zugelassen          | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Windows Media-Umleitung                                                          | Zugelassen          | Alle VDA-Versionen                                                                                      |

|                                                    |             |                                   |
|----------------------------------------------------|-------------|-----------------------------------|
| Windows Media-Umleitungspuffergröße                | 5 Sekunden  | VDA 5, 5.5 und 5.6 Feature Pack 1 |
| Verwendung von Windows Media-Umleitungspuffergröße | Deaktiviert | VDA 5, 5.5 und 5.6 Feature Pack 1 |

## ICA/Multistreamverbindungen

| Name                           | Standardeinstellung                     | VDA                                                                                                                                  |
|--------------------------------|-----------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| Audio über UDP                 | Zugelassen                              | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS                                                                              |
| Audio-UDP-Portbereich          | 16500, 16509                            | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Multiportrichtlinie            | Primärer Port (2598) hat hohe Priorität | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Multistreamcomputereinstellung | Deaktiviert                             | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Multistreambenutzereinstellung | Deaktiviert                             | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |

## ICA\Portumleitung

| Name                                   | Standardeinstellung | VDA                                                                                              |
|----------------------------------------|---------------------|--------------------------------------------------------------------------------------------------|
| Client-COM-Ports automatisch verbinden | Deaktiviert         | Alle VDA-Versionen; bei VDA 7.x müssen Sie diese Einstellung in der Registrierung konfigurieren. |
| Client-LPT-Ports automatisch verbinden | Deaktiviert         | Alle VDA-Versionen; bei VDA 7.x müssen Sie diese Einstellung in der Registrierung konfigurieren. |
| Client-COM-Portumleitung               | Nicht zugelassen    | Alle VDA-Versionen; bei VDA 7.x müssen Sie diese Einstellung in der Registrierung konfigurieren. |
| Client-LPT-Portumleitung               | Nicht zugelassen    | Alle VDA-Versionen; bei VDA 7.x müssen Sie diese Einstellung in der Registrierung konfigurieren. |

## ICA/Printing

| Name | Standardeinstellung | VDA |
|------|---------------------|-----|
|------|---------------------|-----|

| Name                                                              | Standardeinstellung                                                                   | VDA                |
|-------------------------------------------------------------------|---------------------------------------------------------------------------------------|--------------------|
| Clientdruckerumleitung                                            | Zugelassen                                                                            | Alle VDA-Versionen |
| Standarddrucker                                                   | Hauptdrucker des Clients als Standarddrucker verwenden                                | Alle VDA-Versionen |
| Druckerzuweisungen                                                | Der aktuelle Drucker des Benutzers wird als Standarddrucker in der Sitzung verwendet. | Alle VDA-Versionen |
| Ereignisprotokollpräferenz für die automatische Druckererstellung | Fehler und Warnungen protokollieren                                                   | Alle VDA-Versionen |
| Sitzungsdrucker                                                   | Keine Drucker angegeben                                                               | Alle VDA-Versionen |
| Warten bis Drucker erstellt sind (Desktop)                        | Deaktiviert                                                                           | Alle VDA-Versionen |

#### ICA/Drucken/Clientdrucker

| Name                                                   | Standardeinstellung                                                 | VDA                               |
|--------------------------------------------------------|---------------------------------------------------------------------|-----------------------------------|
| Clientdrucker automatisch erstellen                    | Alle Clientdrucker automatisch erstellen                            | Alle VDA-Versionen                |
| Generischen universellen Drucker automatisch erstellen | Deaktiviert                                                         | Alle VDA-Versionen                |
| Clientdruckernamen                                     | Standarddruckernamen                                                | Alle VDA-Versionen                |
| Direkte Verbindungen zu Druckservern                   | Aktiviert                                                           | Alle VDA-Versionen                |
| Druckertreiberzuordnung und -kompatibilität            | Keine Regeln angegeben                                              | Alle VDA-Versionen                |
| Speicherung von Druckereigenschaften                   | Im Profil speichern, wenn sie nicht auf dem Client gespeichert sind | Alle VDA-Versionen                |
| Gespeicherte und wiederhergestellte Clientdrucker      | Zugelassen                                                          | VDA 5, 5.5 und 5.6 Feature Pack 1 |

#### ICA/Printing/Drivers

| Name                                                         | Standardeinstellung       | VDA                |
|--------------------------------------------------------------|---------------------------|--------------------|
| Automatische Installation von mitgelieferten Druckertreibern | Aktiviert                 | Alle VDA-Versionen |
| Priorität universeller Treiber                               | EMF, XPS, PCL5c, PCL4, PS | Alle VDA-          |

| Name                                   | Standardeinstellung                                                                | Versionen<br>VDA   |
|----------------------------------------|------------------------------------------------------------------------------------|--------------------|
| Verwendung universeller Druckertreiber | Universelles Drucken nur verwenden, wenn angeforderter Treiber nicht verfügbar ist | Alle VDA-Versionen |

#### ICA/Drucken/Universeller Druckserver

| Name                                                                            | Standardeinstellung | VDA                |
|---------------------------------------------------------------------------------|---------------------|--------------------|
| Universellen Druckserver aktivieren                                             | Deaktiviert         | Alle VDA-Versionen |
| Port für Druckdatenstrom des universellen Druckservers (CGP)                    | 7229                | Alle VDA-Versionen |
| Universeller Druckserver - Eingabebandbreitenlimit für Druckdatenstrom (KBit/s) | 0                   | Alle VDA-Versionen |
| Port für universellen Druckserverwebdienst (HTTP/SOAP)                          | 8080                | Alle VDA-Versionen |

#### ICA/Drucken/Universelles Drucken

| Name                                           | Standardeinstellung                                                                                                                                                                                                                                                                                                                                                                                                                                                | VDA                |
|------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|
| Universelles Drucken - EMF-Verarbeitungsmodus  | Direkt zum Drucker spoolen                                                                                                                                                                                                                                                                                                                                                                                                                                         | Alle VDA-Versionen |
| Universelles Drucken - Bildkomprimierungslimit | Beste Qualität (verlustfreie Komprimierung)                                                                                                                                                                                                                                                                                                                                                                                                                        | Alle VDA-Versionen |
| Universelles Drucken - Optimierungsstandards   | Bildkomprimierung <ul style="list-style-type: none"> <li>• Gewünschte Bildqualität=Standardqualität</li> <li>• Heavyweight-Komprimierung aktivieren=False</li> </ul> Zwischenspeichern von Bildern und Schriftarten <ul style="list-style-type: none"> <li>• Zwischenspeichern eingebetteter Bilder zulassen=True</li> <li>• Zwischenspeichern eingebetteter Schriftarten zulassen=True</li> </ul> Nicht-Administratoren können diese Einstellungen anpassen=False | Alle VDA-Versionen |
| Universelles Drucken - VorschauEinstellung     | Druckvorschau für automatisch erstellte oder generische universelle Drucker nicht verwenden                                                                                                                                                                                                                                                                                                                                                                        | Alle VDA-Versionen |
| Universelles Drucken - Druckqualitätslimit     | Kein Limit                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Alle VDA-Versionen |

| Name                                  | Standardeinstellung | VDA                                                     |
|---------------------------------------|---------------------|---------------------------------------------------------|
| ICA/Security                          |                     |                                                         |
| Name                                  | Standardeinstellung | VDA                                                     |
| SecureICA-Mindestverschlüsselungsgrad | Standard            | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS |

#### ICA/Serverlimits

| Name                         | Standardeinstellung | VDA                                                     |
|------------------------------|---------------------|---------------------------------------------------------|
| Serverleerlauf-Zeitintervall | 0 Millisekunden     | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS |

#### ICA/Sitzungslimits

| Name                                 | Standardeinstellung | VDA                                                                                       |
|--------------------------------------|---------------------|-------------------------------------------------------------------------------------------|
| Timer für getrennte Sitzung          | Deaktiviert         | VDA 5, 5.5, 5.6 Feature Pack 1, VDA für Desktop-OS 7 bis zum aktuellen VDA für Desktop-OS |
| Getrennte Sitzungen - Timerintervall | 1440 Minuten        | VDA 5, 5.5, 5.6 Feature Pack 1, VDA für Desktop-OS 7 bis zum aktuellen VDA für Desktop-OS |
| Sitzungsverbindungstimer             | Deaktiviert         | VDA 5, 5.5, 5.6 Feature Pack 1, VDA für Desktop-OS 7 bis zum aktuellen VDA für Desktop-OS |
| Sitzungsverbindung - Timerintervall  | 1440 Minuten        | VDA 5, 5.5, 5.6 Feature Pack 1, VDA für Desktop-OS 7 bis zum aktuellen VDA für Desktop-OS |
| Sitzungsleerlauf-timer               | Enabledf            | VDA 5, 5.5, 5.6 Feature Pack 1, VDA für Desktop-OS 7 bis zum aktuellen VDA für Desktop-OS |
| Sitzungsleerlauf - Timerintervall    | 1440 Minuten        | VDA 5, 5.5, 5.6 Feature Pack 1, VDA für Desktop-OS 7 bis zum aktuellen VDA für Desktop-OS |

#### ICA/Sitzungszuverlässigkeit

| Name                                   | Standardeinstellung | VDA                |
|----------------------------------------|---------------------|--------------------|
| Sitzungszuverlässigkeit - Verbindungen | Zugelassen          | Alle VDA-Versionen |
| Sitzungszuverlässigkeit - Portnummer   | 2598                | Alle VDA-Versionen |
| Sitzungszuverlässigkeit - Timeout      | 180 Sekunden        | Alle VDA-Versionen |

## ICA/Zeitzonesteuerung

| Name                                       | Standardeinstellung      | VDA                                                     |
|--------------------------------------------|--------------------------|---------------------------------------------------------|
| Lokale Zeitzone für Legacyclients schätzen | Aktiviert                | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS |
| Lokale Zeit des Clients verwenden          | Serverzeitzone verwenden | Alle VDA-Versionen                                      |

## ICA/TWAIN-Geräte

| Name                             | Standardeinstellung | VDA                                                                                                                                  |
|----------------------------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| TWAIN-Geräteumleitung für Client | Zugelassen          | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| TWAIN-Komprimierungsgrad         | Mittel              | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |

## ICA/USB-Geräte

| Name                                              | Standardeinstellung    | VDA                                                                                                     |
|---------------------------------------------------|------------------------|---------------------------------------------------------------------------------------------------------|
| Regeln für die Client-USB-Geräteoptimierung       | Keine Regeln angegeben | VDA 7.6 FP3                                                                                             |
| Client-USB-Geräteumleitung                        | Nicht zugelassen       | Alle VDA-Versionen                                                                                      |
| Regeln für die Client-USB-Geräteumleitung         | Keine Regeln angegeben | Alle VDA-Versionen                                                                                      |
| Client-USB-Geräteumleitung für Plug & Play-Geräte | Zugelassen             | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |

## ICA/Visuelle Anzeige

| Name                                       | Standardeinstellung | VDA                |
|--------------------------------------------|---------------------|--------------------|
| Bevorzugte Farbtiefe für einfache Grafiken | 24 Bit pro Pixel    | VDA 7.6 FP3        |
| Frameratesollwert                          | 30 F/s              | Alle VDA-Versionen |

|                                                |                                         |                                                                                                         |
|------------------------------------------------|-----------------------------------------|---------------------------------------------------------------------------------------------------------|
| Bildqualität                                   | Mittel                                  | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Verwenden von Videocodec für die Komprimierung | Verwenden von Videocodec wenn verfügbar | VDA 7.6 FP3                                                                                             |

#### ICA/Visuelle Anzeige/Bewegtbilder

| Name                                        | Standardeinstellung | VDA                                                                                                                                  |
|---------------------------------------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| Mindestbildqualität                         | Normal              | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Bewegtbildkomprimierung                     | Aktiviert           | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Grad der progressiven Komprimierung         | Keine               | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Schwellenwert für progressive Komprimierung | 2147483647 KBit/s   | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Mindestframeratesollwert                    | 10 F/s              | VDA 5.5, 5.6 Feature Pack 1, VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |

#### ICA/Visuelle Anzeige/Festbilder

| Name                                            | Standardeinstellung | VDA                |
|-------------------------------------------------|---------------------|--------------------|
| Zusätzliche Farbkomprimierung                   | Deaktiviert         | Alle VDA-Versionen |
| Schwellenwert für zusätzliche Farbkomprimierung | 8192 KBit/s         | Alle VDA-Versionen |
| Heavyweight-Komprimierung                       | Deaktiviert         | Alle VDA-Versionen |
| Grad der verlustreichen Komprimierung           | Mittel              | Alle VDA-Versionen |
| Schwellenwert für verlustreiche Komprimierung   | 2147483647 KBit/s   | Alle VDA-Versionen |

## ICA/WebSockets

| Name                                              | Standardeinstellung                                                          | VDA                                                                                                     |
|---------------------------------------------------|------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------|
| WebSockets-Verbindungen                           | Nicht zugelassen                                                             | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| WebSockets-Portnummer                             | 8008                                                                         | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Vertrauenswürdige WebSockets-Ursprungsserverliste | Bei Verwendung des Platzhalters * wird allen Receiver für Web-URLs vertraut. | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |

## Load Management

| Name                                        | Standardeinstellung      | VDA                                                     |
|---------------------------------------------|--------------------------|---------------------------------------------------------|
| Toleranzwert für gleichzeitige Anmeldungen  | 2                        | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS |
| CPU-Nutzung                                 | Deaktiviert              | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS |
| CPU-Nutzung ausschließlich Prozesspriorität | Unter normal oder gering | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS |
| Datenträgenutzung                           | Deaktiviert              | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS |
| Sitzungshöchstanzahl                        | 250                      | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS |
| Speichernutzung                             | Deaktiviert              | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS |
| Speichernutzung - Ausgangslast              | Nulllast: 768 MB         | VDA für Server-OS 7 bis zum aktuellen VDA für Server-OS |

## Profilverwaltung/Erweiterte Einstellungen

| Name                                    | Standardeinstellung | VDA                |
|-----------------------------------------|---------------------|--------------------|
| Automatische Konfiguration deaktivieren | Deaktiviert         | Alle VDA-Versionen |
| Benutzer bei Problem abmelden           | Deaktiviert         | Alle VDA-Versionen |



|                                                                         |                                 |                                  |
|-------------------------------------------------------------------------|---------------------------------|----------------------------------|
| Anzahl Wiederholungen beim Zugriff auf gesperrte Dateien<br><b>Name</b> | 5<br><b>Standardeinstellung</b> | Alle VDA-Versionen<br><b>VDA</b> |
| Internet-Cookiedateien bei Abmeldung verarbeiten                        | Deaktiviert                     | Alle VDA-Versionen               |

## Profilverwaltung/Grundeinstellungen

| <b>Name</b>                                     | <b>Standardeinstellung</b>                                        | <b>VDA</b>         |
|-------------------------------------------------|-------------------------------------------------------------------|--------------------|
| Aktiv zurückschreiben                           | Deaktiviert                                                       | Alle VDA-Versionen |
| Profilverwaltung aktivieren                     | Deaktiviert                                                       | Alle VDA-Versionen |
| Ausgeschlossene Gruppen                         | Deaktiviert: Mitglieder aller Benutzergruppen werden verarbeitet. | Alle VDA-Versionen |
| Unterstützung von Offlineprofilen               | Deaktiviert                                                       | Alle VDA-Versionen |
| Pfad zum Benutzerspeicher                       | Windows                                                           | Alle VDA-Versionen |
| Anmeldungen lokaler Administratoren verarbeiten | Deaktiviert                                                       | Alle VDA-Versionen |
| Verarbeitete Gruppen                            | Deaktiviert: Mitglieder aller Benutzergruppen werden verarbeitet. | Alle VDA-Versionen |

## Profilverwaltung/Plattformübergreifende Einstellungen

| <b>Name</b>                                                | <b>Standardeinstellung</b>                                                               | <b>VDA</b>         |
|------------------------------------------------------------|------------------------------------------------------------------------------------------|--------------------|
| Benutzergruppen für plattformübergreifende Einstellungen   | Deaktiviert: Alle in Verarbeitete Gruppen angegebenen Benutzergruppen werden verarbeitet | Alle VDA-Versionen |
| Plattformübergreifende Einstellungen aktivieren            | Deaktiviert                                                                              | Alle VDA-Versionen |
| Pfad zu plattformübergreifenden Definitionen               | Deaktiviert: Kein Pfad angegeben.                                                        | Alle VDA-Versionen |
| Pfad zum Speicher für plattformübergreifende Einstellungen | Deaktiviert: Windows\PM_CM wird verwendet.                                               | Alle VDA-Versionen |
| Quelle für plattformübergreifende Einstellungen            | Deaktiviert                                                                              | Alle VDA-Versionen |

## Profilverwaltung/Dateisystem/Ausschlüsse

| Name                            | Standardeinstellung                                                | VDA                |
|---------------------------------|--------------------------------------------------------------------|--------------------|
| Ausschlussliste - Verzeichnisse | Deaktiviert: Alle Ordner im Benutzerprofil werden synchronisiert.  | Alle VDA-Versionen |
| Ausschlussliste - Dateien       | Deaktiviert: Alle Dateien im Benutzerprofil werden synchronisiert. | Alle VDA-Versionen |

## Profilverwaltung/Dateisystem/Synchronisierung

| Name                               | Standardeinstellung                                                   | VDA                |
|------------------------------------|-----------------------------------------------------------------------|--------------------|
| Zu synchronisierende Verzeichnisse | Deaktiviert: Nur nicht ausgeschlossene Ordner werden synchronisiert.  | Alle VDA-Versionen |
| Zu synchronisierende Dateien       | Deaktiviert: Nur nicht ausgeschlossene Dateien werden synchronisiert. | Alle VDA-Versionen |
| Zu spiegelnde Ordner               | Deaktiviert: Es werden keine Ordner gespiegelt.                       | Alle VDA-Versionen |

## Profilverwaltung/Ordnerumleitung

| Name                          | Standardeinstellung | VDA                |
|-------------------------------|---------------------|--------------------|
| Administratorzugriff gewähren | Deaktiviert         | Alle VDA-Versionen |
| Domänennamen einschließen     | Deaktiviert         | Alle VDA-Versionen |

## Profilverwaltung/Ordnerumleitung/AppData(Roaming)

| Name                                         | Standardeinstellung                                                                                        | VDA                |
|----------------------------------------------|------------------------------------------------------------------------------------------------------------|--------------------|
| AppData(Roaming)-Pfad                        | Deaktiviert: Kein Speicherort angegeben.                                                                   | Alle VDA-Versionen |
| Umleitungseinstellungen für AppData(Roaming) | Inhalte werden zu dem in der Richtlinieneinstellung AppData(Roaming)-Pfad angegebenen UNC-Pfad umgeleitet. | Alle VDA-Versionen |

## Profilverwaltung/Ordnerumleitung/Kontakte

| Name            | Standardeinstellung                      | VDA                |
|-----------------|------------------------------------------|--------------------|
| 'Kontakte'-Pfad | Deaktiviert: Kein Speicherort angegeben. | Alle VDA-Versionen |
|                 |                                          |                    |

| Name                                   | Standardeinstellung                                                                                  | VDA                |
|----------------------------------------|------------------------------------------------------------------------------------------------------|--------------------|
| Umleitungseinstellungen für 'Kontakte' | Inhalte werden zu dem in der Richtlinieneinstellung 'Kontakte'-Pfad angegebenen UNC-Pfad umgeleitet. | Alle VDA-Versionen |

#### Profilverwaltung/Ordnerumleitung/Desktop

| Name                                  | Standardeinstellung                                                                                 | VDA                |
|---------------------------------------|-----------------------------------------------------------------------------------------------------|--------------------|
| 'Desktop'-Pfad                        | Deaktiviert: Kein Speicherort angegeben.                                                            | Alle VDA-Versionen |
| Umleitungseinstellungen für 'Desktop' | Inhalte werden zu dem in der Richtlinieneinstellung 'Desktop'-Pfad angegebenen UNC-Pfad umgeleitet. | Alle VDA-Versionen |

#### Profilverwaltung/Ordnerumleitung/Dokumente

| Name                                    | Standardeinstellung                                                                                   | VDA                |
|-----------------------------------------|-------------------------------------------------------------------------------------------------------|--------------------|
| 'Dokumente'-Pfad                        | Deaktiviert: Kein Speicherort angegeben.                                                              | Alle VDA-Versionen |
| Umleitungseinstellungen für 'Dokumente' | Inhalte werden zu dem in der Richtlinieneinstellung 'Dokumente'-Pfad angegebenen UNC-Pfad umgeleitet. | Alle VDA-Versionen |

#### Profilverwaltung/Ordnerumleitung/Downloads

| Name                                    | Standardeinstellung                                                                                   | VDA                |
|-----------------------------------------|-------------------------------------------------------------------------------------------------------|--------------------|
| 'Downloads'-Pfad                        | Deaktiviert: Kein Speicherort angegeben.                                                              | Alle VDA-Versionen |
| Umleitungseinstellungen für 'Downloads' | Inhalte werden zu dem in der Richtlinieneinstellung 'Downloads'-Pfad angegebenen UNC-Pfad umgeleitet. | Alle VDA-Versionen |

#### Profilverwaltung/Ordnerumleitung/Favoriten

| Name                                    | Standardeinstellung                                                                                   | VDA                |
|-----------------------------------------|-------------------------------------------------------------------------------------------------------|--------------------|
| 'Favoriten'-Pfad                        | Deaktiviert: Kein Speicherort angegeben.                                                              | Alle VDA-Versionen |
| Umleitungseinstellungen für 'Favoriten' | Inhalte werden zu dem in der Richtlinieneinstellung 'Favoriten'-Pfad angegebenen UNC-Pfad umgeleitet. | Alle VDA-Versionen |

#### Profilverwaltung/Ordnerumleitung/Links

| Name         | Standardeinstellung                      | VDA                |
|--------------|------------------------------------------|--------------------|
| 'Links'-Pfad | Deaktiviert: Kein Speicherort angegeben. | Alle VDA-Versionen |

| Name                                | Standardeinstellung                                                                               | VDA                |
|-------------------------------------|---------------------------------------------------------------------------------------------------|--------------------|
| Umleitungseinstellungen für 'Links' | Inhalte werden zu dem in der Richtlinieneinstellung 'Links'-Pfad angegebenen UNC-Pfad umgeleitet. | Alle VDA-Versionen |

#### Profilverwaltung/Ordnerumleitung/Musik

| Name                                | Standardeinstellung                                                                               | VDA                |
|-------------------------------------|---------------------------------------------------------------------------------------------------|--------------------|
| 'Musik'-Pfad                        | Deaktiviert: Kein Speicherort angegeben.                                                          | Alle VDA-Versionen |
| Umleitungseinstellungen für 'Musik' | Inhalte werden zu dem in der Richtlinieneinstellung 'Musik'-Pfad angegebenen UNC-Pfad umgeleitet. | Alle VDA-Versionen |

#### Profilverwaltung/Ordnerumleitung/Bilder

| Name                                 | Standardeinstellung                                                                                | VDA                |
|--------------------------------------|----------------------------------------------------------------------------------------------------|--------------------|
| 'Bilder'-Pfad                        | Deaktiviert: Kein Speicherort angegeben.                                                           | Alle VDA-Versionen |
| Umleitungseinstellungen für 'Bilder' | Inhalte werden zu dem in der Richtlinieneinstellung 'Bilder'-Pfad angegebenen UNC-Pfad umgeleitet. | Alle VDA-Versionen |

#### Profilverwaltung/Ordnerumleitung/Gespeicherte Spiele

| Name                                              | Standardeinstellung                                                                                             | VDA                |
|---------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|--------------------|
| 'Gespeicherte Spiele'-Pfad                        | Deaktiviert: Kein Speicherort angegeben.                                                                        | Alle VDA-Versionen |
| Umleitungseinstellungen für 'Gespeicherte Spiele' | Inhalte werden zu dem in der Richtlinieneinstellung 'Gespeicherte Spiele'-Pfad angegebenen UNC-Pfad umgeleitet. | Alle VDA-Versionen |

#### Profilverwaltung/Ordnerumleitung/Suchen

| Name                                 | Standardeinstellung                                                                                | VDA                |
|--------------------------------------|----------------------------------------------------------------------------------------------------|--------------------|
| 'Suchen'-Pfad                        | Deaktiviert: Kein Speicherort angegeben.                                                           | Alle VDA-Versionen |
| Umleitungseinstellungen für 'Suchen' | Inhalte werden zu dem in der Richtlinieneinstellung 'Suchen'-Pfad angegebenen UNC-Pfad umgeleitet. | Alle VDA-Versionen |

#### Profilverwaltung/Ordnerumleitung/Startmenü

| Name           | Standardeinstellung                      | VDA       |
|----------------|------------------------------------------|-----------|
| Startmenü-Pfad | Deaktiviert: Kein Speicherort angegeben. | Alle VDA- |

| Name                                    | Standardeinstellung                                                                                   | Versionen<br>VDA   |
|-----------------------------------------|-------------------------------------------------------------------------------------------------------|--------------------|
| Umleitungseinstellungen für 'Startmenü' | Inhalte werden zu dem in der Richtlinieneinstellung 'Startmenü'-Pfad angegebenen UNC-Pfad umgeleitet. | Alle VDA-Versionen |

## Profilverwaltung/Ordnerumleitung/Videos

| Name                                 | Standardeinstellung                                                                                | VDA                |
|--------------------------------------|----------------------------------------------------------------------------------------------------|--------------------|
| 'Videos'-Pfad                        | Deaktiviert: Kein Speicherort angegeben.                                                           | Alle VDA-Versionen |
| Umleitungseinstellungen für 'Videos' | Inhalte werden zu dem in der Richtlinieneinstellung 'Videos'-Pfad angegebenen UNC-Pfad umgeleitet. | Alle VDA-Versionen |

## Profilverwaltung/Protokolleinstellungen

| Name                          | Standardeinstellung                                                                                                         | VDA                |
|-------------------------------|-----------------------------------------------------------------------------------------------------------------------------|--------------------|
| Active Directory-Aktionen     | Deaktiviert                                                                                                                 | Alle VDA-Versionen |
| Allgemeine Informationen      | Deaktiviert                                                                                                                 | Alle VDA-Versionen |
| Allgemeine Warnungen          | Deaktiviert                                                                                                                 | Alle VDA-Versionen |
| Protokollierung aktivieren    | Deaktiviert                                                                                                                 | Alle VDA-Versionen |
| Dateisystemaktionen           | Deaktiviert                                                                                                                 | Alle VDA-Versionen |
| Dateisystembenachrichtigungen | Deaktiviert                                                                                                                 | Alle VDA-Versionen |
| Abmeldung                     | Deaktiviert                                                                                                                 | Alle VDA-Versionen |
| Anmeldung                     | Deaktiviert                                                                                                                 | Alle VDA-Versionen |
| Max. Größe der Protokolldatei | 1048576                                                                                                                     | Alle VDA-Versionen |
| Pfad zur Protokolldatei       | Deaktiviert: Protokolldateien werden im Standardspeicherort gespeichert: %SystemRoot%\System32\Logfiles\UserProfileManager. | Alle VDA-Versionen |
| Persönliche                   | Deaktiviert                                                                                                                 | Alle VDA-          |

| Benutzerinformationen<br><b>Name</b>         | <b>Standardeinstellung</b> | <b>Versionen<br/>VDA</b> |
|----------------------------------------------|----------------------------|--------------------------|
| Richtlinienwerte bei Anmeldung und Abmeldung | Deaktiviert                | Alle VDA-Versionen       |
| Registrierungsaktionen                       | Deaktiviert                | Alle VDA-Versionen       |
| Registrierungsunterschiede bei der Abmeldung | Deaktiviert                | Alle VDA-Versionen       |

## Profilverwaltung/Profilverarbeitung

| <b>Name</b>                                                                    | <b>Standardeinstellung</b>                                                                                                                     | <b>VDA</b>         |
|--------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|
| Verzögerung vor dem Löschen von zwischengespeicherten Profilen                 | 0                                                                                                                                              | Alle VDA-Versionen |
| Lokal zwischengespeicherte Profile nach Abmeldung löschen                      | Deaktiviert                                                                                                                                    | Alle VDA-Versionen |
| Behandlung von Konflikten lokaler Profile                                      | Lokales Profil verwenden                                                                                                                       | Alle VDA-Versionen |
| Migration vorhandener Profile                                                  | Lokal und Roaming                                                                                                                              | Alle VDA-Versionen |
| Pfad zum Vorlagenprofil                                                        | Deaktiviert: Neue Benutzerprofile werden von dem Standardbenutzerprofil auf dem Gerät erstellt, auf dem sich ein Benutzer als Erstes anmeldet. | Alle VDA-Versionen |
| Vorlagenprofil überschreibt lokales Profil                                     | Deaktiviert                                                                                                                                    | Alle VDA-Versionen |
| Vorlagenprofil überschreibt Roamingprofil                                      | Deaktiviert                                                                                                                                    | Alle VDA-Versionen |
| Als verbindliches Citrix Profil für alle Anmeldungen verwendete Vorlagenprofil | Deaktiviert                                                                                                                                    | Alle VDA-Versionen |

## Profilverwaltung/Registrierung

| <b>Name</b>     | <b>Standardeinstellung</b>                                                                                          | <b>VDA</b>         |
|-----------------|---------------------------------------------------------------------------------------------------------------------|--------------------|
| Ausschlussliste | Deaktiviert: Alle Registrierungsschlüssel in der HKCU-Struktur werden verarbeitet, wenn ein Benutzer sich abmeldet. | Alle VDA-Versionen |
| Aufnahmeliste   | Deaktiviert: Alle Registrierungsschlüssel in der HKCU-Struktur werden verarbeitet, wenn ein Benutzer sich abmeldet. | Alle VDA-Versionen |

| Name                                          | Standardeinstellung                                                                        | VDA                |
|-----------------------------------------------|--------------------------------------------------------------------------------------------|--------------------|
| Profilverwaltung/Gestreamte Benutzerprofile   |                                                                                            |                    |
| Name                                          | Standardeinstellung                                                                        | VDA                |
| Immer zwischenspeichern                       | Deaktiviert                                                                                | Alle VDA-Versionen |
| Immer Cachegröße                              | 0 MBit                                                                                     | Alle VDA-Versionen |
| Profilstreaming                               | Deaktiviert                                                                                | Alle VDA-Versionen |
| Gruppen mit gestreamten Benutzerprofilen      | Deaktiviert: Alle Benutzerprofile in einer Organisationseinheit werden normal verarbeitet. | Alle VDA-Versionen |
| Timeout für bevorstehende Sperrdateien (Tage) | 1 Tag                                                                                      | Alle VDA-Versionen |

## Receiver

| Name                   | Standardeinstellung    | VDA                                                                                                     |
|------------------------|------------------------|---------------------------------------------------------------------------------------------------------|
| StoreFront-Kontenliste | Keine Stores angegeben | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |

## Virtual Delivery Agent

| Name                                           | Standardeinstellung            | VDA                                                                                                     |
|------------------------------------------------|--------------------------------|---------------------------------------------------------------------------------------------------------|
| IPv6-Netzwerkmaske für Controllerregistrierung | Keine Netzwerkmaske angegeben. | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Controllerregistrierungsport                   | 80                             | Alle VDA-Versionen                                                                                      |
| Controller-SIDs                                | Keine SIDs angegeben           | Alle VDA-Versionen                                                                                      |
| Controller                                     | Keine Controller angegeben     | Alle VDA-Versionen                                                                                      |
| Automatische Controllerupdates aktivieren      | Aktiviert                      | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |
| Nur IPv6-Controllerregistrierung               | Deaktiviert                    | VDA für Server-OS 7 und VDA für Desktop-OS 7 bis zum aktuellen VDA für Server-OS und VDA für Desktop-OS |

| verwenden<br>Name | Standardeinstellung  | VDA                |
|-------------------|----------------------|--------------------|
| Site-GUID         | Kein GUID angegeben. | Alle VDA-Versionen |

## Virtual IP

| Name                                             | Standardeinstellung | VDA     |
|--------------------------------------------------|---------------------|---------|
| Virtuelle IP - Loopbackunterstützung             | Deaktiviert         | VDA 7.6 |
| Virtuelle IP - Programme für virtuelles Loopback | Keine               | VDA 7.6 |

## HDX 3D Pro

| Name                            | Standardeinstellung | VDA                            |
|---------------------------------|---------------------|--------------------------------|
| Verlustfrei aktivieren          | Aktiviert           | VDA 5.5 und 5.6 Feature Pack 1 |
| HDX3DPro-Qualitätseinstellungen |                     | VDA 5.5 und 5.6 Feature Pack 1 |



# Referenz für Richtlinienereinstellungen

May 10, 2016

Richtlinien enthalten Einstellungen, die gelten, wenn die Richtlinie angewendet wird. Die Beschreibungen in diesem Abschnitt geben auch an, ob zusätzliche Einstellungen zum Aktivieren eines Features erforderlich sind oder ob Einstellungen sich ähnlich sind.

## Kurzanleitung

Die folgenden Tabellen listen die Einstellungen auf, die Sie in einer Richtlinie konfigurieren können. In der linken Spalte finden Sie die Aufgaben, in der rechten die dazugehörigen Einstellungen.

## Audio

| Aufgabe                                                                    | Richtlinieneinstellung  |
|----------------------------------------------------------------------------|-------------------------|
| Steuern der Verwendung mehrerer Audiogeräte                                | Audio Plug & Play       |
| Steuern, ob Audioeingaben vom Mikrofon auf dem Benutzergerät zulässig sind | Clientmikrofonumleitung |
| Steuern der Audioqualität auf dem Benutzergerät                            | Audioqualität           |
| Steuern der Audiozuordnung für Lautsprecher am Benutzergerät               | Clientaudioumleitung    |

## Bandbreite für Benutzergeräte

| Beschränken der Bandbreite                           | Richtlinieneinstellung                                                                                                                                                 |
|------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Clientaudiozuordnung                                 | <ul style="list-style-type: none"><li>• Bandbreitenlimit für die Audioumleitung oder</li><li>• Bandbreitenlimit für die Audioumleitung (Prozent)</li></ul>             |
| Kopieren und Einfügen mit der lokalen Zwischenablage | <ul style="list-style-type: none"><li>• Bandbreitenlimit für Zwischenablagenumleitung oder</li><li>• Bandbreitenlimit für Zwischenablagenumleitung (Prozent)</li></ul> |
| Zugriff auf lokale Clientlaufwerke in einer Sitzung  | <ul style="list-style-type: none"><li>• Bandbreitenlimit für Dateiumleitung oder</li><li>• Bandbreitenlimit für Dateiumleitung (Prozent)</li></ul>                     |
| HDX MediaStream-Multimediabeschleunigung             | <ul style="list-style-type: none"><li>• Bandbreitenlimit für HDX MediaStream-</li></ul>                                                                                |

| Beschränken der Bandbreite              | Richtlinieneinstellung<br>• Multimediabeschleunigung oder<br>• Bandbreitenlimit für HDX MediaStream-Multimediabeschleunigung (Prozent) |
|-----------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| Clientsitzung                           | Bandbreitenlimit für Sitzung insgesamt                                                                                                 |
| Drucken                                 | • Bandbreitenlimit für Druckerumleitung oder<br>• Bandbreitenlimit für Druckerumleitung (Prozent)                                      |
| TWAIN-Geräte (wie Kameras oder Scanner) | • Bandbreitenlimit für TWAIN-Geräteumleitung oder<br>• Bandbreitenlimit für TWAIN-Geräteumleitung (Prozent)                            |
| USB-Geräte                              | • Bandbreitenlimit für Client-USB-Geräteumleitung oder<br>• Bandbreitenlimit für Client-USB-Geräteumleitung (Prozent)                  |

## Umleitung von Clientlaufwerken und Benutzergeräten

| Aufgabe                                                                                                            | Richtlinieneinstellung                                                  |
|--------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------|
| Steuern, ob Laufwerke des Benutzergeräts verbunden werden, wenn Benutzer sich am Server anmelden                   | Clientlaufwerke automatisch verbinden                                   |
| Steuern der Datenübertragung mit Kopier- und Einfügeoperationen zwischen dem Server und der lokalen Zwischenablage | Clientzwischenablagenumleitung                                          |
| Steuern der Laufwerkzuordnung des Benutzergeräts                                                                   | Clientlaufwerkumleitung                                                 |
| Steuern, ob die lokalen Festplatten des Benutzers in einer Sitzung verfügbar sind                                  | • Lokale<br>• Clientlaufwerke und Clientlaufwerkumleitung               |
| Steuern, ob die lokalen Diskettenlaufwerke des Benutzers in einer Sitzung verfügbar sind                           | • Clientdiskettenlaufwerke<br>• und Clientlaufwerkumleitung redirection |
| Steuern, ob die Netzlaufwerke des Benutzers in einer Sitzung verfügbar sind                                        | • Clientnetzlaufwerke<br>• und Clientlaufwerkumleitung redirection      |

| Aufgabe                                                                                                                                            | Richtlinieneinstellung                                                                                                                  |
|----------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| Steuern, ob die lokalen CD-, DVD- oder Blu-ray-Laufwerke des Benutzers in einer Sitzung verfügbar sind                                             | <ul style="list-style-type: none"> <li>• Optische Clientlaufwerke und Clientlaufwerkumleitung</li> </ul>                                |
| Steuern, ob die lokalen Clientwechseldatenträger des Benutzers in einer Sitzung verfügbar sind                                                     | <ul style="list-style-type: none"> <li>• Clientwechsellaufwerke</li> <li>• und Clientlaufwerkumleitung redirection</li> </ul>           |
| Steuern, ob TWAIN-Geräte, wie Scanner und Kameras, in einer Sitzung verfügbar sind und Steuern der Komprimierung bei der Übertragung von Bilddaten | <ul style="list-style-type: none"> <li>• TWAIN-Geräteumleitung für Client</li> <li>• TWAIN-Umleitung</li> </ul>                         |
| Steuern, ob die USB-Geräte in einer Sitzung verfügbar sind                                                                                         | <ul style="list-style-type: none"> <li>• Client-USB-Geräteumleitung und</li> <li>• Regeln für die Client-USB-Geräteumleitung</li> </ul> |
| Geschwindigkeit beim Schreiben und Kopieren von Dateien auf einen Clientdatenträger über ein WAN erhöhen                                           | Asynchrones Schreiben verwenden                                                                                                         |

## Inhaltsumleitung

| Aufgabe                                                                  | Richtlinieneinstellung   |
|--------------------------------------------------------------------------|--------------------------|
| Steuern der Verwendung der Inhaltsumleitung vom Server zum Benutzergerät | Host-zu-Client-Umleitung |

## Desktopbenutzeroberfläche

| Aufgabe                                                                | Richtlinieneinstellung                  |
|------------------------------------------------------------------------|-----------------------------------------|
| Steuern, ob der Desktophintergrund in Benutzersitzungen angezeigt wird | Desktophintergrund                      |
| Anzeigen des Fensterinhalts beim Verschieben des Fensters              | Fensterinhalt beim Verschieben anzeigen |

## Grafiken & Multimedia

| Aufgabe                                                                                                            | Richtlinieneinstellung |
|--------------------------------------------------------------------------------------------------------------------|------------------------|
| Steuern der maximalen Anzahl von Frames pro Sekunde, die an Benutzergeräte von virtuellen Desktops gesendet werden | Frameratesollwert      |

| Aufgabe                                                                     | Richtlinieneinstellung                                                                                                                                                                                                                                                               |
|-----------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Steuern der visuellen Qualität der auf dem Benutzergerät angezeigten Bilder | Bildqualität                                                                                                                                                                                                                                                                         |
| Steuern, ob Flash-Inhalte in Sitzungen wiedergegeben werden                 | Flash-Standardverhalten                                                                                                                                                                                                                                                              |
| Steuern, ob Flash-Inhalte auf Websites in Sitzungen angezeigt werden        | <ul style="list-style-type: none"> <li>• URL-Liste für serverseitigen Flash-Inhaltsabruf</li> <li>• Flash-URL-Kompatibilitätsliste</li> <li>• Einstellung der Richtlinie zum Verhindern von Videofallback</li> <li>• Fehler beim Verhindern von Flash-Videofallback *.swf</li> </ul> |

## Priorisieren des Multistream-Netzwerkdatenverkehrs

| Aufgabe                                                                                                      | Richtlinieneinstellung                            |
|--------------------------------------------------------------------------------------------------------------|---------------------------------------------------|
| Angaben der Ports für ICA-Datenübertragungen über mehrere Verbindungen und Festlegen der Netzwerkprioritäten | Multiportrichtlinie                               |
| Aktivieren der Unterstützung von Multistreamverbindungen zwischen Servern und Benutzergeräten                | Multistream (Computer- und Benutzereinstellungen) |

## Drucken

| Aufgabe                                                                                                                 | Richtlinieneinstellung                                                                                                        |
|-------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| Steuern der Clientdruckererstellung auf dem Benutzergerät                                                               | <ul style="list-style-type: none"> <li>• Clientdrucker automatisch erstellen und</li> <li>• Clientdruckerumleitung</li> </ul> |
| Steuern des Speicherorts für die Druckereigenschaften                                                                   | Speicherung von Druckereigenschaften                                                                                          |
| Steuern, ob Druckanfragen vom Client oder vom Server verarbeitet werden                                                 | Direkte Verbindungen zu Druckservern                                                                                          |
| Steuern, ob Benutzer auf Drucker zugreifen können, die an die Benutzergeräte angeschlossen sind                         | Clientdruckerumleitung                                                                                                        |
| Steuern, ob bei der automatischen Erstellung von Client- und Netzwerkdruckern native Windows-Treiber installiert werden | Automatische Installation von mitgelieferten Druckertreibern                                                                  |

| <b>Aufgabe</b><br>Steuern, wann der universelle Druckertreiber verwendet wird | <b>Richtlinieneinstellung</b><br>Verwendung universeller Druckertreiber |
|-------------------------------------------------------------------------------|-------------------------------------------------------------------------|
| Wählen des Druckers anhand von Sitzungsinformationen eines mobilen Benutzers  | Standarddrucker                                                         |

Hinweis: Richtlinien können nicht zum Aktivieren eines Bildschirmschoners in einer Desktop- oder Anwendungssitzung verwendet werden. Wenn Benutzer einen Bildschirmschoner benötigen, muss dieser auf dem Benutzergerät eingerichtet werden.

# Einstellungen der Richtlinie "ICA"

May 10, 2016

Der Abschnitt "ICA" enthält Richtlinieneinstellungen für ICA-Listenerverbindungen und die Zwischenablagenzuordnung.

## Clientzwischenablagenumleitung

Mit dieser Einstellung legen Sie fest, ob die Zwischenablage auf dem Clientgerät der Zwischenablage auf dem Server zugeordnet wird.

Standardmäßig ist die Umleitung der Zwischenablage zugelassen.

Wenn Sie verhindern möchten, dass Daten durch Kopieren und Einfügen über die Zwischenablage zwischen einer Sitzung und der lokalen Zwischenablage übertragen werden, wählen Sie Nicht zugelassen. Benutzer können weiterhin die Zwischenablage für das Kopieren von Daten zwischen Anwendungen einsetzen, die in Sitzungen ausgeführt werden.

Nachdem diese Einstellung zugelassen wurde, konfigurieren Sie die maximal zulässige Bandbreite, die die Zwischenablage in einer Clientverbindung verbrauchen darf. Verwenden Sie dazu die Einstellung Bandbreitenlimit für Zwischenablagenumleitung oder Bandbreitenlimit für Zwischenablagenumleitung (Prozent).

## Zum Schreiben in Clientzwischenablage zugelassene Formate

Wenn die Einstellung Schreiben in Clientzwischenablage einschränken aktiviert ist, können Hostzwischenablagendaten nicht für den Clientendpunkt freigegeben werden. Mit dieser Einstellung können jedoch bestimmte Datenformate für die Zwischenablage des Clientendpunkts freigegeben werden. Um diese Einstellung zu verwenden, aktivieren Sie sie und fügen Sie die zulässigen Formate hinzu.

Die folgenden Zwischenablageformate sind vom System definiert:

- CF\_TEXT
- CF\_BITMAP
- CF\_METAFILEPICT
- CF\_SYLK
- CF\_DIF
- CF\_TIFF
- CF\_OEMTEXT
- CF\_DIB
- CF\_PALETTE
- CF\_PENDATA
- CF\_RIFF
- CF\_WAVE
- CF\_UNICODETEXT
- CF\_ENHMETAFILE
- CF\_HDROP
- CF\_LOCALE
- CF\_DIBV5
- CF\_OWNERDISPLAY
- CF\_DSPTEXT
- CF\_DSPBITMAP

- CF\_DSPMETAFILEPICT
- CF\_DISPENHMETAFILE

Die folgenden benutzerdefinierten Formate sind in XenApp und XenDesktop vordefiniert:

- CFX\_RICHTEXT
- CFX\_OfficeDrawingShape
- CFX\_BIFF8

Zusätzliche benutzerdefinierte Formate können hinzugefügt werden. Der Name des benutzerdefinierten Formats muss mit den Formaten übereinstimmen, die mit dem System registriert werden. Bei Formatnamen muss die Groß- und Kleinschreibung beachtet werden.

Diese Einstellung wird nicht angewendet, wenn Clientzwischenablagenumleitung oder Schreiben in Clientzwischenablage einschränken auf "Nicht zugelassen" festgelegt sind.

## Desktop starten

Mit dieser Einstellung legen Sie fest, ob Benutzer ohne Administratorrechte, die in der Gruppe der Benutzer mit direktem Zugriff eines VDAs sind, über eine ICA-Verbindung eine Verbindung zu einer Sitzung auf dem VDA herstellen können.

Standardmäßig können Benutzer ohne Administratorrechte keine Verbindung zu diesen Sitzungen herstellen.

Diese Einstellung hat keine Auswirkungen auf Benutzer ohne Administratorrechte, die in der Gruppe der Benutzer mit direktem Zugriff eines VDAs sind und eine RDP-Verbindung verwenden. Diese Benutzer können eine Verbindung zum VDA herstellen, unabhängig davon, ob diese Einstellung aktiviert ist. Diese Einstellung hat keine Auswirkungen auf Benutzer ohne Administratorrechte, die nicht in der Gruppe der Benutzer mit direktem Zugriff eines VDAs sind. Diese Benutzer können unabhängig vom Status dieser Einstellung keine Verbindung zum VDA herstellen.

## ICA-Listener - Verbindungstimeout

**Hinweis:** Diese Einstellung gilt nur für Virtual Delivery Agents 5.0, 5.5 und 5.6 Feature Pack 1.

Mit dieser Einstellung geben Sie die maximale Wartezeit an, bis eine Verbindung mit dem ICA-Protokoll abgeschlossen wird.

Standardmäßig ist die maximale Wartezeit 120000 Millisekunden oder zwei Minuten.

## ICA-Listenerportnummer

Mit dieser Einstellung konfigurieren Sie die TCP/IP-Portnummer, die vom ICA-Protokoll auf dem Server verwendet wird.

Die Standardeinstellung der Portnummer ist "1494".

Gültige Portnummern müssen zwischen 0 und 65535 liegen. Sie dürfen keinen Konflikt mit anderen gängigen Portnummern verursachen. Wenn Sie die Portnummer ändern, muss der Server neu gestartet werden, damit der neue Wert wirksam werden kann. Wenn Sie die Portnummer auf dem Server ändern, müssen Sie sie auch in jedem Receiver oder Plug-In ändern, das eine Verbindung zu diesem Server herstellt.

## Starten nicht-veröffentlichter Programme bei Clientverbindung

Mit dieser Einstellung geben Sie an, ob Startanwendungen über RDP auf dem Server gestartet werden.

Standardmäßig ist das Starten von Startanwendungen über RDP auf dem Server nicht zulässig.

## Schreiben in Clientzwischenablage einschränken

Wenn diese Einstellung zugelassen ist, können Hostzwischenablagedaten nicht für den Clientendpunkt freigegeben werden. Durch Aktivieren der Einstellung Zum Schreiben in Clientzwischenablage zugelassene Formate können Sie bestimmte Formate zulassen.

Die Standardeinstellung ist "Nicht zugelassen".

## Schreiben in Sitzungszwischenablage einschränken

Wenn diese Einstellung zugelassen ist, können Clientzwischenablagedaten nicht für die Benutzersitzung freigegeben werden. Durch Aktivieren der Einstellung Zum Schreiben in Sitzungszwischenablage zugelassene Formate können Sie bestimmte Formate zulassen.

Die Standardeinstellung ist "Nicht zugelassen".

## Zum Schreiben in Sitzungszwischenablage zugelassene Formate

Wenn die Einstellung Schreiben in Sitzungszwischenablage einschränken aktiviert ist, können Clientzwischenablagedaten nicht für Sitzungsanwendungen freigegeben werden. Mit dieser Einstellung können jedoch bestimmte Datenformate für die Sitzungszwischenablage freigegeben werden.

Die folgenden Zwischenablageformate sind vom System definiert:

- CF\_TEXT
- CF\_BITMAP
- CF\_METAFILEPICT
- CF\_SYLK
- CF\_DIF
- CF\_TIFF
- CF\_OEMTEXT
- CF\_DIB
- CF\_PALETTE
- CF\_PENDATA
- CF\_RIFF
- CF\_WAVE
- CF\_UNICODETEXT
- CF\_ENHMETAFILE
- CF\_HDROP
- CF\_LOCALE
- CF\_DIBV5
- CF\_OWNERDISPLAY
- CF\_DSPTEXT
- CF\_DSPBITMAP
- CF\_DSPMETAFILEPICT
- CF\_DISPENHMETAFILE

Die folgenden benutzerdefinierten Formate sind in XenApp und XenDesktop vordefiniert:

- CFX\_RICHTEXT
- CFX\_OfficeDrawingShape
- CFX\_BIFF8



Zusätzliche benutzerdefinierte Formate können hinzugefügt werden. Der Name des benutzerdefinierten Formats muss mit den Formaten übereinstimmen, die mit dem System registriert werden. Bei Formatnamen muss die Groß- und Kleinschreibung beachtet werden.

Diese Einstellung wird nicht angewendet, wenn die Einstellung Clientzwischenablagenumleitung oder Schreiben in Sitzungszwischenablage einschränken auf "Nicht zugelassen" festgelegt sind.

# Einstellungen der Richtlinie "Automatische Wiederverbindung von Clients"

May 10, 2016

Der Abschnitt "Automatische Wiederverbindung von Clients" enthält Richtlinieneinstellungen, mit denen Sie die automatische Wiederverbindung von Sitzungen steuern.

## Automatische Wiederverbindung von Clients

Diese Einstellung legt fest, ob die automatische Wiederverbindung des gleichen Clients zulässig ist, nachdem eine Verbindung unterbrochen wurde.

Standardmäßig ist die automatische Wiederverbindung zugelassen.

Ist die automatische Wiederverbindung zulässig, können Benutzer ihre Arbeit an der Stelle wiederaufnehmen, an der die Verbindung unterbrochen wurde. Die automatische Wiederverbindung erkennt unterbrochene Verbindungen und verbindet die Benutzer wieder mit ihren Sitzungen.

Bei der automatischen Wiederverbindung kann es aber dazu kommen, dass eine neue Sitzung gestartet wird (statt die Verbindung mit der bestehenden Sitzung wiederherzustellen), wenn das Receiver-Cookie, das den Schlüssel für Sitzungs-ID und Anmeldeinformationen enthält, nicht verwendet wird. Das Cookie wird nicht verwendet, wenn es abgelaufen ist, z. B. weil die Wiederverbindung verzögert wird, oder wenn die Anmeldeinformationen neu eingegeben werden müssen. Die automatische Wiederverbindung von Clients wird nicht ausgelöst, wenn Benutzer die Sitzung absichtlich trennen.

Wenn die automatische Wiederverbindung zugelassen ist, versucht Receiver bei Anwendungssitzungen, die Verbindung mit der Sitzung wiederherzustellen, bis die Wiederverbindung erfolgreich war oder der Benutzer die Wiederverbindung abbricht.

Wenn die automatische Wiederverbindung zugelassen ist, versucht Receiver bei Desktopsitzungen eine festgelegte Zeit lang, die Verbindung mit der Sitzung wiederherzustellen, bis die Wiederverbindung erfolgreich war oder der Benutzer die Wiederverbindung abbricht. Der Standardwert für diese Zeit ist fünf Minuten. Um die Zeit zu ändern, bearbeiten Sie die Registrierung auf dem Benutzergerät:

`HKLM\Software\Citrix\ICA Client\TransportReconnectRetryMaxTimeSeconds; DWORD;<Sekunden>`

wobei <Sekunden> die Zeit in Sekunden ist, nach deren Ablauf keine weiteren Versuche der Sitzungswiederherstellung erfolgen.

## Authentifizierung bei automatischer Wiederverbindung von Clients

Mit dieser Einstellung ist die Authentifizierung erforderlich, wenn die Verbindung zum Client automatisch wiederhergestellt wird.

Standardmäßig ist die Authentifizierung nicht erforderlich.

Wenn sich ein Benutzer erstmals anmeldet, werden seine Anmeldeinformationen verschlüsselt und gespeichert und es wird ein Cookie mit dem Schlüssel erstellt, der an Receiver gesendet wird. Wenn diese Einstellung konfiguriert ist, werden keine Cookies verwendet. Stattdessen wird ein Dialogfeld mit der Aufforderung zur Eingabe der Anmeldeinformationen angezeigt, wenn Receiver versucht, die Verbindung automatisch wiederherzustellen.

## Protokollierung der automatischen Wiederverbindung von Clients

Mit dieser Einstellung legen Sie fest, ob die automatischen Wiederverbindungen im Ereignisprotokoll aufgezeichnet werden.

Standardmäßig ist die Protokollierung deaktiviert.

Wenn die Protokollierung aktiviert ist, werden Informationen über erfolgreiche und fehlgeschlagene Wiederverbindungsereignisse im Systemprotokoll des Servers aufgezeichnet. Eine Site stellt kein kombiniertes Protokoll zu Wiederverbindungsereignissen auf allen Servern zur Verfügung.

# Einstellungen der Richtlinie "Audio"

May 10, 2016

Der Abschnitt "Audio" enthält Richtlinieneinstellungen, mit denen Sie das Senden und Empfangen von Audiodaten auf dem Benutzergerät konfigurieren können, ohne dass es dabei zu einer unerwünschten Verschlechterung der Leistung kommt.

## Audio über UDP - Real-time Transport

Diese Einstellung aktiviert bzw. deaktiviert die Audioübertragung und den Audioempfang zwischen VDA und Benutzergeräten über RTP mit UDP (User Datagram Protocol). Wenn diese Einstellung deaktiviert ist, wird Audio über TCP gesendet und empfangen.

Standardmäßig ist Audio über UDP zugelassen.

## Audio Plug & Play

Mit dieser Einstellung lassen Sie die Verwendung mehrerer Audiogeräte zum Aufzeichnen und zum Wiedergeben von Ton zu oder verhindern sie.

Standardmäßig ist die Verwendung mehrerer Audiogeräte zulässig.

Diese Einstellung gilt nur für Windows-Serverbetriebssystemmaschinen.

## Audioqualität

Mit dieser Einstellung legen Sie die Tonqualität fest, die in Benutzersitzungen empfangen wird.

In der Standardeinstellung ist die Tonqualität auf Hoch - High Definition-Audio eingestellt.

Um die Tonqualität zu steuern, wählen Sie eine der folgenden Optionen:

- Wählen Sie Gering - für langsame Verbindungen für Verbindungen mit geringer Bandbreite. An das Benutzergerät gesendete Audiodaten werden bis auf 16 KBit/s komprimiert. Diese Komprimierung führt zu einer erheblichen Verringerung der Tonqualität, ermöglicht aber eine akzeptable Leistung bei einer Verbindung mit geringer Bandbreite.
- Wählen Sie Mittel - für Sprache optimiert, um VoIP-Anwendungen oder Medienanwendungen bei schwierigen Netzwerkverbindungen mit Leitungen unter 512 KBit/s oder bei erheblicher Überlastung und Paketverlust bereitzustellen. Dieses Codec bietet eine schnelle Codierung und ist daher ideal für Softphones und Unified Communications-Anwendungen geeignet, wenn Sie eine serverseitige Medienverarbeitung benötigen.  
Die an das Benutzergerät gesendeten Audiodaten werden bis auf 64 KBit/s komprimiert; diese Komprimierung führt zu einer moderaten Verringerung der Tonqualität auf dem Benutzergerät mit niedriger Latenz und geringem Bandbreitenverbrauch. Wenn die Einstellung eine unbefriedigende VoIP-Qualität liefert, stellen Sie sicher, dass die Richtlinie Audio über UDP - Real-time Transport auf Zugelassen eingestellt ist.
- Wählen Sie Hoch - High Definition Audio für Verbindungen, bei denen die Bandbreite keine Rolle spielt und bei denen die Tonqualität wichtig ist. Clients können Audiodaten mit der nativen Abspielrate wiedergeben. Audiodaten werden mit einer hohen Qualitätsstufe bei Erhaltung der CD-Qualität komprimiert, die bis zu 112 KBit/s Bandbreite benötigt. Die Übertragung dieser Datenmenge kann zu einer höheren CPU-Last und Engpässen im Netzwerk führen.

Die Bandbreite wird nur verbraucht, während Audio aufgenommen oder abgespielt wird. Wenn beides gleichzeitig stattfindet, verdoppelt sich der Bandbreitenverbrauch.

Konfigurieren Sie die Einstellungen Bandbreitenlimit für die Audioumleitung oder Bandbreitenlimit für die Audioumleitung

(Prozent), um die maximale Bandbreite anzugeben.

## Clientaudioumleitung

Mit dieser Einstellung legen Sie fest, ob auf dem Server gehostete Anwendungen Audiodateien über ein auf dem Benutzergerät installiertes Audiogerät wiedergeben können. Diese Einstellung gibt auch an, ob Benutzer Audio aufzeichnen können.

Standardmäßig ist die Audioumleitung zugelassen.

Nachdem Sie diese Einstellung zugelassen haben, können Sie die Bandbreite beschränken, die durch die Wiedergabe oder das Aufzeichnen von Audio verbraucht wird. Durch Beschränken der Bandbreite, die durch Audio verbraucht wird, kann sich die Anwendungsleistung steigern, die Audioqualität wird aber herabgesetzt. Die Bandbreite wird nur verbraucht, während Audio aufgenommen oder abgespielt wird. Wenn beides gleichzeitig stattfindet, verdoppelt sich der Bandbreitenverbrauch. Konfigurieren Sie die Einstellungen Bandbreitenlimit für die Audioumleitung oder Bandbreitenlimit für die Audioumleitung (Prozent), um die maximale Bandbreite anzugeben.

Auf Windows-Serverbetriebssystemmaschinen müssen Sie sicherstellen, dass für Audio Plug & Play die Unterstützung mehrerer Audiogeräte aktiviert ist.

Wichtig: Wenn die Clientaudioumleitung nicht zugelassen ist, sind alle HDX-Audiofunktionen deaktiviert.

## Clientmikrofonumleitung

Mit dieser Einstellung aktivieren oder deaktivieren Sie die Umleitung von Clientmikrofonen. Wenn aktiviert, können Benutzer Mikrofone für die Aufnahme von Audioeingaben in einer Sitzung verwenden.

Standardmäßig ist die Clientmikrofonumleitung zugelassen.

Aus Sicherheitsgründen werden Benutzer darauf hingewiesen, wenn Server, die keine vertrauenswürdige Beziehung zu den Geräten haben, auf Mikrofone zugreifen. Benutzer können den Zugriff ermöglichen oder ablehnen. Benutzer können die Warnung in Citrix Receiver deaktivieren.

Auf Windows-Serverbetriebssystemmaschinen müssen Sie sicherstellen, dass für Audio Plug & Play die Unterstützung mehrerer Audiogeräte aktiviert ist.

Wenn die Einstellung Clientaudioumleitung auf dem Benutzergerät deaktiviert ist, hat diese Regel keine Auswirkung.

# Einstellungen der Richtlinie "Bandbreite"

May 10, 2016

Der Abschnitt "Bandbreite" enthält Richtlinieneinstellungen, mit denen Sie Leistungsprobleme vermeiden können, die sich aus der Bandbreitenverwendung in der Clientsitzung ergeben.

Wichtig: Die Verwendung dieser Richtlinieneinstellungen mit den Richtlinieneinstellungen Multistream kann zu unerwarteten Ergebnissen führen. Wenn Sie Multistreameinstellungen in einer Richtlinie verwenden, stellen Sie sicher, dass diese Richtlinieneinstellungen für das Bandbreitenlimit nicht eingeschlossen sind.

## Bandbreitenlimit für die Audioumleitung

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite für die Wiedergabe oder die Aufnahme von Audio in einer Benutzersitzung in Kilobits pro Sekunde an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für die Audioumleitung (Prozent) einen Wert angeben, wird die restriktivere Einstellung angewendet.

## Bandbreitenlimit für die Audioumleitung (Prozent)

Mit dieser Einstellung geben Sie das maximal zulässige Bandbreitenlimit für die Wiedergabe oder die Aufnahme von Audio in als Prozentsatz der Gesamtsitzungsbandbreite an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für die Audioumleitung einen Wert angeben, wird die restriktivere Einstellung angewendet.

Wenn Sie die Einstellung konfigurieren, müssen Sie auch die Einstellung Bandbreitenlimit für Sitzung insgesamt konfigurieren, mit der die Gesamtbandbreite definiert wird, die für Clientsitzungen verfügbar ist.

## Bandbreitenlimit für Client-USB-Geräteumleitung

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite für die Umleitung von USB-Geräten zum und vom Client in Kilobit pro Sekunde an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für Client-USB-Geräteumleitung (Prozent) einen Wert angeben, wird die restriktivere Einstellung angewendet.

## Bandbreitenlimit für Client-USB-Geräteumleitung (Prozent)

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite für die Umleitung von USB-Geräten zum und vom Client als Prozentsatz der Gesamtsitzungsbandbreite an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für Client-USB-Geräteumleitung einen Wert angeben, wird die restriktivere Einstellung angewendet.

Wenn Sie die Einstellung konfigurieren, müssen Sie auch die Einstellung Bandbreitenlimit für Sitzung insgesamt konfigurieren,

mit der die Gesamtbandbreite definiert wird, die für Clientsitzungen verfügbar ist.

### Bandbreitenlimit für Zwischenablagenumleitung

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite in Kilobits pro Sekunde für Datenübertragungen zwischen einer Sitzung und der lokalen Zwischenablage an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für Zwischenablagenumleitung (Prozent) einen Wert angeben, wird die restriktivere Einstellung angewendet.

### Bandbreitenlimit für Zwischenablagenumleitung (Prozent)

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite für Datenübertragungen zwischen einer Sitzung und der lokalen Zwischenablage als Prozentsatz der Gesamtsitzungsbandbreite an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für Zwischenablagenumleitung einen Wert angeben, wird die restriktivere Einstellung angewendet.

Wenn Sie die Einstellung konfigurieren, müssen Sie auch die Einstellung Bandbreitenlimit für Sitzung insgesamt konfigurieren, mit der die Gesamtbandbreite definiert wird, die für Clientsitzungen verfügbar ist.

### Bandbreitenlimit für COM-Portumleitung

Hinweis: Konfigurieren Sie bei Virtual Delivery Agent 7.x diese Einstellung über die Registrierung (siehe [Konfigurieren von COM-Port- und LPT-Portumleitungseinstellungen in der Registrierung](#)).

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite für den Zugriff auf einen COM-Port in einer Clientverbindung in Kilobits pro Sekunde an. Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für LPT-Portumleitung (Prozent) einen Wert angeben, wird die restriktivere Einstellung angewendet.

### Bandbreitenlimit für COM-Portumleitung (Prozent)

Hinweis: Konfigurieren Sie bei Virtual Delivery Agent 7.x diese Einstellung über die Registrierung (siehe [Konfigurieren von COM-Port- und LPT-Portumleitungseinstellungen in der Registrierung](#)).

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite für den Zugriff auf COM-Ports in einer Clientverbindung als Prozentsatz der Gesamtsitzungsbandbreite an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und auch für die Einstellung Bandbreitenlimit für LPT-Portumleitung einen Wert angeben, wird die restriktivere Einstellung angewendet.

Wenn Sie die Einstellung konfigurieren, müssen Sie auch die Einstellung Bandbreitenlimit für Sitzung insgesamt konfigurieren, mit der die Gesamtbandbreite definiert wird, die für Clientsitzungen verfügbar ist.

### Bandbreitenlimit für Dateiumleitung

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite für den Zugriff auf Clientlaufwerke in einer Clientverbindung in Kilobits pro Sekunde an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für Dateiumleitung (Prozent) einen Wert angeben, wird die restriktivere Einstellung (mit dem niedrigeren Wert) angewendet.

### Bandbreitenlimit für Dateiumleitung (Prozent)

Mit dieser Einstellung geben Sie das maximal zulässige Bandbreitenlimit für den Zugriff auf Clientlaufwerke als Prozentsatz der Gesamtsitzungsbandbreite an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für Dateiumleitung einen Wert angeben, wird die restriktivere Einstellung angewendet.

Wenn Sie die Einstellung konfigurieren, müssen Sie auch die Einstellung Bandbreitenlimit für Sitzung insgesamt konfigurieren, mit der die Gesamtbandbreite definiert wird, die für Clientsitzungen verfügbar ist.

### Bandbreitenlimit für HDX MediaStream-Multimediabeschleunigung

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite für die Bereitstellung von Streamingaudio und -video mit HDX MediaStream-Multimediabeschleunigung in Kilobit pro Sekunde an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für HDX MediaStream-Multimediabeschleunigung (Prozent) einen Wert angeben, wird die restriktivere Einstellung angewendet.

### Bandbreitenlimit für HDX MediaStream-Multimediabeschleunigung (Prozent)

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite für die Bereitstellung von Streamingaudio und -video mit HDX MediaStream-Multimediabeschleunigung als Prozentsatz der Gesamtsitzungsbandbreite an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung einen Wert angeben und auch für die Einstellung "Bandbreitenlimit für HDX MediaStream-Multimediabeschleunigung", wird die restriktivere Einstellung angewendet.

Wenn Sie die Einstellung konfigurieren, müssen Sie auch die Einstellung Bandbreitenlimit für Sitzung insgesamt konfigurieren, mit der die Gesamtbandbreite definiert wird, die für Clientsitzungen verfügbar ist.

### Bandbreitenlimit für LPT-Portumleitung

Hinweis: Konfigurieren Sie bei Virtual Delivery Agent 7.x diese Einstellung über die Registrierung (siehe [Konfigurieren von COM-Port- und LPT-Portumleitungseinstellungen in der Registrierung](#)).

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite in Kilobits pro Sekunde an, die für Druckaufträge über den LPT-Port in einer Benutzersitzung verwendet werden kann.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für COM-Portumleitung (Prozent) einen Wert angeben, wird die restriktivere Einstellung angewendet.

### Bandbreitenlimit für LPT-Portumleitung (Prozent)

Hinweis: Konfigurieren Sie bei Virtual Delivery Agent 7.x diese Einstellung über die Registrierung (siehe [Konfigurieren von](#)



#### COM-Port- und LPT-Portumleitungseinstellungen in der Registrierung).

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite, die für Druckaufträge über den LPT-Port in einer Sitzung verwendet werden darf, als Prozent der Gesamtsitzungsbandbreite an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für COM-Portumleitung einen Wert angeben, wird die restriktivere Einstellung angewendet.

Wenn Sie die Einstellung konfigurieren, müssen Sie auch die Einstellung Bandbreitenlimit für Sitzung insgesamt konfigurieren, mit der die Gesamtbandbreite definiert wird, die für Clientsitzungen verfügbar ist.

#### Bandbreitenlimit für Sitzung insgesamt

Mit dieser Einstellung geben Sie Gesamtbandbreite in Kilobits pro Sekunde an, die für Benutzersitzungen verwendet werden kann.

Die maximal erzwingbare Bandbreitenbeschränkung ist 10 MBit/s (10.000 KBit/s). Standardmäßig ist kein Maximalwert (Null) angegeben.

Durch Beschränken der Bandbreite, die von einer Clientverbindung verbraucht wird, kann zu einer Leistungsverbesserung führen, wenn andere Anwendungen außerhalb der Clientverbindung auch auf die Bandbreite zugreifen.

#### Bandbreitenlimit für Druckerumleitung

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite für den Zugriff auf Clientdrucker in einer Benutzersitzung in Kilobits pro Sekunde an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für Druckerumleitung (Prozent) einen Wert angeben, wird die restriktivere Einstellung angewendet.

#### Bandbreitenlimit für Druckerumleitung (Prozent)

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite für den Zugriff auf Clientdrucker als Prozentsatz der Gesamtsitzungsbandbreite an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für Druckerumleitung einen Wert angeben, wird die restriktivere Einstellung angewendet.

Wenn Sie die Einstellung konfigurieren, müssen Sie auch die Einstellung Bandbreitenlimit für Sitzung insgesamt konfigurieren, mit der die Gesamtbandbreite definiert wird, die für Clientsitzungen verfügbar ist.

#### Bandbreitenlimit für TWAIN-Geräteumleitung

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite in Kilobits pro Sekunde für die Steuerung von TWAIN-Bildverarbeitungsgeräten in veröffentlichten Anwendungen an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für TWAIN-Geräteumleitung (Prozent) einen Wert

angeben, wird die restriktivere Einstellung angewendet.

### Bandbreitenlimit für TWAIN-Geräteumleitung (Prozent)

Mit dieser Einstellung geben Sie die maximal zulässige Bandbreite für die Steuerung von TWAIN-Bildverarbeitungsgeräten in veröffentlichten Anwendungen als Prozentsatz der Gesamtsitzungsbandbreite an.

Standardmäßig ist kein Maximalwert (Null) angegeben.

Wenn Sie für diese Einstellung und für die Einstellung Bandbreitenlimit für TWAIN-Geräteumleitung einen Wert angeben, wird die restriktivere Einstellung angewendet.

Wenn Sie die Einstellung konfigurieren, müssen Sie auch die Einstellung Bandbreitenlimit für Sitzung insgesamt konfigurieren, mit der die Gesamtbandbreite definiert wird, die für Clientsitzungen verfügbar ist.

# Einstellungen der Richtlinie "Clientsensoren"

May 10, 2016

Der Abschnitt "Clientsensoren" enthält Richtlinieneinstellungen, mit denen gesteuert wird, wie Informationen über den Mobilgerätsensor in einer Benutzersitzung gehandhabt werden.

Anwendungen können den physischen Standort des Clientgeräts verwenden

Diese Einstellung legt fest, ob Anwendungen, die in einer Sitzung auf einem Mobilgerät ausgeführt werden, den physischen Standort des Benutzergeräts verwenden können.

In der Standardeinstellung ist die Verwendung von Standortinformationen nicht zugelassen.

Wenn diese Einstellung nicht zugelassen ist und eine Anwendung versucht, die Standortinformationen abzurufen, wird ein Wert von "Zugriff verweigert" zurückgegeben.

Wenn diese Einstellung nicht zugelassen ist, kann ein Benutzer die Verwendung von Standortinformationen verhindern und eine Receiver-Anforderung für den Zugriff auf den Standort ablehnen. Android- und iOS-Geräte senden am Anfang jeder Sitzung eine Anforderung für die Standortinformationen.

Berücksichtigen Sie beim Entwickeln von gehosteten Anwendungen, die die Einstellung Anwendungen können den physischen Standort des Clientgeräts verwenden enthalten Folgendes:

- Eine standortaktivierte Anwendung sollte sich aus den folgenden Gründen nicht darauf verlassen, dass Standortinformationen verfügbar sind:
  - Ein Benutzer gewährt möglicherweise keinen Zugriff auf die Standortinformationen.
  - Der Standort ist ggf. nicht verfügbar oder ändert sich, während die Anwendung ausgeführt wird.
  - Ein Benutzer stellt möglicherweise eine Verbindung mit der Anwendungssitzung von einem anderen Gerät her, das keine Standortinformationen unterstützt.
- Anforderungen für eine standortaktivierte Anwendung:
  - Das Standortfeature muss in der Standardeinstellung deaktiviert sein.
  - Eine Benutzeroption für das Zulassen oder Ablehnen des Features muss bei Ausführung der Anwendung verfügbar sein.
  - Eine Benutzeroption muss verfügbar sein, mit der von der Anwendung zwischengespeicherte Standortdaten gelöscht werden. (Receiver speichert keine Standortdaten im Cache.)
- Eine standortaktivierte Anwendung muss die Granularität der Standortinformationen verwalten, damit die abgefragten Daten dem Zweck der Anwendung entsprechen und die entsprechenden Gesetze einhalten.
- Bei der Verwendung der Standortdienste sollte eine sichere Verbindung (zum Beispiel mit SSL/TLS oder einem VPN) erzwungen werden. Citrix Receiver sollte eine Verbindung mit vertrauenswürdigen Servern herstellen.
- Sie sollten eine Rechtsberatung hinsichtlich der Verwendung von Standortdiensten erwägen.

# Einstellungen der Richtlinie "Desktopbenutzeroberfläche"

Nov 02, 2016

Der Abschnitt "Desktopbenutzeroberfläche" enthält Richtlinieneinstellungen für visuelle Effekte, wie Desktophintergrund, Menüanimationen und das Verhalten von Fensterinhalten beim Drag & Drop, um die für Clientverbindungen verbrauchte Bandbreite zu steuern. Die Anwendungsleistung über ein WAN lässt sich durch Beschränken des Bandbreitenverbrauchs verbessern.

## Desktopgestaltungsumleitung

Mit dieser Einstellung geben Sie an, ob die Verarbeitung des Grafikprozessors (GPU) oder des integrierten Grafikprozessors (IGP) auf dem Benutzergerät für die lokale DirectX-Grafikwiedergabe verwendet werden soll, um eine nahtlosere Windows-Desktopdarstellung zu erzielen. Wenn die Desktopgestaltungsumleitung aktiviert ist, wird eine hoch reaktionsfähige Windows-Benutzererfahrung bei Beibehaltung einer hohen Skalierbarkeit auf dem Server gewährleistet.

Standardmäßig ist Desktopgestaltungsumleitung ab Version 7.6 FP3 bis zur aktuellen Version deaktiviert und ab Version 5.6 bis 7.6 FP2 aktiviert.

Um die Desktopgestaltungsumleitung auszuschalten und die für Benutzersitzungen erforderliche Bandbreite zu reduzieren, wählen Sie Deaktiviert aus, wenn Sie diese Einstellung einer Richtlinie möchten.

## Grafikqualität Desktopgestaltung

Mit dieser Einstellung wird die Qualität der für die Desktopgestaltungsumleitung verwendeten Grafiken angegeben.

Die Standardeinstellung ist "Hoch".

Wählen Sie die Qualität Hoch, Mittel, Niedrig oder Verlustfrei aus.

## Desktophintergrund

Mit dieser Einstellung legen Sie fest, ob Hintergründe in Benutzersitzungen angezeigt werden.

Standardmäßig kann der Desktophintergrund in Benutzersitzungen angezeigt werden.

Um den Desktophintergrund auszuschalten und die für Benutzersitzungen erforderliche Bandbreite zu reduzieren, wählen Sie die Einstellung Nicht zugelassen, wenn Sie diese Einstellung einer Richtlinie hinzufügen.

## Menüanimation

Mit dieser Einstellung legen Sie fest, ob Menüanimation in Benutzersitzungen zugelassen oder verhindert wird.

Standardmäßig ist Menüanimation zugelassen.

Menüanimation ist eine Microsoft-Einstellung für erleichterte Bedienung. Ist die Einstellung aktiviert, werden Menüs nach einer kurzen Verzögerung durch Bildlauf- oder Einblendeffekt angezeigt. Unten im Menü wird ein Pfeil angezeigt. Das Menü wird eingeblendet, wenn Sie mit der Maus auf diesen Pfeil zeigen.

Menüanimation ist auf einem Desktop aktiviert, wenn diese Richtlinieneinstellung auf Zugelassen festgelegt ist und die Microsoft-Einstellung für Menüanimation aktiviert ist.

Hinweis: Änderungen an der Microsoft-Einstellung für Menüanimation sind Desktopänderungen. Dies bedeutet, dass einem Benutzer, der Menüanimation in einer Sitzung aktiviert hat, in späteren Sitzungen auf dem Desktop keine Menüanimation zur Verfügung steht, wenn die Desktopeinstellungen so festgelegt sind, dass am Desktop vorgenommene Änderungen nach dem Beenden der Sitzung verworfen werden. Aktivieren Sie daher für Benutzer, die Menüanimation benötigen, die Microsoft-Einstellung im Masterimage für den Desktop oder stellen Sie sicher, dass der Desktop vom Benutzer vorgenommene Änderungen beibehält.

#### Fensterinhalt beim Verschieben anzeigen

Mit dieser Einstellung legen Sie fest, ob Fensterinhalte beim Verschieben des Fensters auf dem Bildschirm angezeigt werden.

Standardmäßig ist die Anzeige des Fensterinhalts beim Verschieben zugelassen.

Wenn Zugelassen ausgewählt ist, wird beim Verschieben das ganze Fenster angezeigt. Wenn Nicht zugelassen ausgewählt ist, wird bis zum Ablegen nur der Fensterrahmen beim Verschieben angezeigt.

# Einstellungen der Richtlinie "Endbenutzerüberwachung"

May 10, 2016

Der Abschnitt "Endbenutzerüberwachung" enthält Richtlinien zum Messen von Sitzungsnetzwerkverkehr.

## ICA-Roundtripberechnung

Mit dieser Einstellung legen Sie fest, ob ICA-Roundtripberechnungen für aktive Verbindungen durchgeführt werden.

Standardmäßig sind die Berechnungen für aktive Verbindungen aktiviert.

Standardmäßig wird die Initiierung des ICA-Roundtripmessung verzögert, bis Netzwerkverkehr auf eine Benutzeraktion hinweist. Diese Verzögerung kann eine unbestimmte Länge haben und verhindert, dass die ICA-Roundtripmessung der einzige Grund für den ICA-Verkehr ist

## Intervall für ICA-Roundtripberechnung

Mit dieser Einstellung geben Sie die Häufigkeit an, in Sekunden, mit der ICA-Roundtripberechnungen durchgeführt werden

Standardmäßig wird der ICA-Roundtrip alle 15 Sekunden berechnet.

## ICA-Roundtrip für Verbindungen im Leerlauf berechnen

Mit dieser Einstellung legen Sie fest, ob ICA-Roundtripberechnungen für Verbindungen im Leerlauf durchgeführt werden.

Standardmäßig werden Berechnungen nicht für Verbindungen im Leerlauf durchgeführt.

Standardmäßig wird die Initiierung des ICA-Roundtripmessung verzögert, bis Netzwerkverkehr auf eine Benutzeraktion hinweist. Diese Verzögerung kann eine unbestimmte Länge haben und verhindert, dass die ICA-Roundtripmessung der einzige Grund für den ICA-Verkehr ist

# Richtlinieneinstellung für Enhanced Desktop Experience

May 10, 2016

Durch die Richtlinieneinstellung "Enhanced Desktop Experience" werden Sitzungen auf Serverbetriebssystemen so konfiguriert, dass sie wie lokale Windows 7-Desktops aussehen, damit Benutzer in den Genuss einer verbesserten Desktopdarstellung kommen.

Standardmäßig ist diese Einstellung zugelassen.

Wenn ein Benutzerprofil mit dem Design "Windows – klassisch" bereits auf dem virtuellen Desktop vorhanden ist, wird durch Aktivieren dieser Richtlinie nicht die verbesserte Desktopdarstellung für diesen Benutzer bereitgestellt. Wenn sich ein Benutzer, dessen Benutzerprofil mit einem Windows 7-Design konfiguriert ist, bei einem virtuellen Desktop unter Windows Server 2012 anmeldet, für den diese Richtlinie deaktiviert oder nicht konfiguriert ist, wird eine Fehlermeldung angezeigt, die angibt, dass das Design nicht angewendet werden kann.

In beiden Fällen kann das Problem durch Zurücksetzen des Benutzerprofils gelöst werden.

Wenn die Richtlinie auf einem virtuellen Desktop mit aktiven Benutzersitzungen deaktiviert wird, sind Aussehen und Verhalten von Sitzungen nicht mit der Desktopdarstellung von Windows 7 und Windows – klassisch konsistent. Wenn Sie dies vermeiden möchten, starten Sie den virtuellen Desktop neu, nachdem Sie die Richtlinieneinstellung geändert haben. Sie müssen auch sämtliche Roamingprofile auf dem virtuellen Desktop löschen. Citrix empfiehlt außerdem, alle anderen Benutzerprofile auf dem virtuellen Desktop zu löschen, um Inkonsistenzen zwischen Profilen zu vermeiden.

Wenn Sie Roamingbenutzerprofile in der Umgebung verwenden, stellen Sie sicher, dass das Feature "Enhanced Desktop Experience" für alle virtuellen Desktops, die sich ein Profil teilen, entweder aktiviert oder deaktiviert ist.

Citrix rät davon ab, Roamingprofile zwischen virtuellen Desktops, auf denen Serverbetriebssysteme und Clientbetriebssysteme ausgeführt werden, freizugeben. Die Profile für Client- und Serverbetriebssysteme sind unterschiedlich und das Freigeben von Roamingprofilen zwischen beiden Systemtypen kann zu Inkonsistenzen in den Profileigenschaften führen, wenn ein Benutzer zwischen den Systemen wechselt.

# Einstellungen der Richtlinie "Dateiumleitung"

May 10, 2016

Der Abschnitt "Dateiumleitung" enthält Richtlinieneinstellungen für die Clientlaufwerkzuordnung und die Clientlaufwerkoptimierung.

## Clientlaufwerke automatisch verbinden

Mit dieser Einstellung legen Sie fest, ob die automatische Verbindung von Clientlaufwerken bei der Benutzeranmeldung zugelassen ist.

In der Standardeinstellung ist die automatische Verbindung zugelassen.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, müssen Sie die Einstellungen für die Laufwerktypen aktivieren, die automatisch verbunden werden. Konfigurieren Sie beispielsweise Optische Clientlaufwerke, damit CD-Laufwerke auf dem Clientgerät automatisch verbunden werden.

Die folgenden Richtlinieneinstellungen hängen zusammen:

- Clientlaufwerkumleitung
- Clientdiskettenlaufwerke
- Optische Clientlaufwerke
- Lokale Clientfestplattenlaufwerke
- Clientnetzlaufwerke
- Clientwechsellaufwerke

## Clientlaufwerkumleitung

Mit dieser Einstellung aktivieren oder deaktivieren Sie die Dateiumleitung von und zu Laufwerken auf dem Benutzergerät.

In der Standardeinstellung ist die Dateiumleitung aktiviert.

Wenn aktiviert, können Benutzer ihre Dateien auf allen Clientlaufwerken speichern. Wenn deaktiviert, wird jegliche Dateiumleitung verhindert, unabhängig von den Dateiumleitungseinstellungen für einzelne Laufwerkstypen, z. B. Clientdiskettenlaufwerke und Clientnetzlaufwerke.

Die folgenden Richtlinieneinstellungen hängen zusammen:

- Clientdiskettenlaufwerke
- Optische Clientlaufwerke
- Lokale Clientfestplattenlaufwerke
- Clientnetzlaufwerke
- Clientwechsellaufwerke

## Lokale Clientfestplattenlaufwerke

Mit dieser Einstellung legen Sie fest, ob Benutzer auf die lokalen Festplattenlaufwerke des Benutzergeräts zugreifen oder Dateien darauf speichern können.

In der Standardeinstellung ist der Zugriff auf lokale Festplattenlaufwerke zugelassen.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, müssen Sie auch für die Einstellung Clientlaufwerkumleitung die Option Zugelassen wählen. Wenn diese Einstellungen deaktiviert sind, können lokale Festplattenlaufwerke nicht



zugeordnet werden und Benutzer können auch nicht manuell auf diese Laufwerke zugreifen – unabhängig von der Einstellung für Lokale Festplattenlaufwerke.

Konfigurieren Sie außerdem die Einstellung Clientlaufwerke automatisch verbinden, damit lokale Festplattenlaufwerke automatisch verbunden werden.

### Clientdiskettenlaufwerke

Mit dieser Einstellung legen Sie fest, ob Benutzer auf die Diskettenlaufwerke des Benutzergeräts zugreifen oder Dateien darauf speichern können.

In der Standardeinstellung ist der Zugriff auf Diskettenlaufwerke zugelassen.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, müssen Sie auch für die Einstellung Clientlaufwerkumleitung die Option Zugelassen wählen. Wenn diese Einstellungen deaktiviert sind, können Diskettenlaufwerke nicht zugeordnet werden und Benutzer können auch nicht manuell auf diese Laufwerke zugreifen – unabhängig von der Einstellung für Clientdiskettenlaufwerke.

Konfigurieren Sie außerdem die Einstellung Clientlaufwerke automatisch verbinden, damit Diskettenlaufwerke automatisch verbunden werden.

### Clientnetzlaufwerke

Mit dieser Einstellung legen Sie fest, ob Benutzer auf die (remoten) Netzlaufwerke des Benutzergeräts zugreifen oder Dateien speichern können.

In der Standardeinstellung ist der Zugriff auf Netzlaufwerke zugelassen.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, müssen Sie auch für die Einstellung Clientlaufwerkumleitung die Option Zugelassen wählen. Wenn diese Einstellungen deaktiviert sind, können Netzlaufwerke nicht zugeordnet werden und Benutzer können auch nicht manuell auf diese Laufwerke zugreifen – unabhängig von der Einstellung für Clientnetzlaufwerke.

Konfigurieren Sie außerdem die Einstellung Clientlaufwerke automatisch verbinden, damit Netzlaufwerke automatisch verbunden werden.

### Optische Clientlaufwerke

Mit dieser Einstellung legen Sie fest, ob Benutzer auf CD-, DVD- und BD-Laufwerke des Clientgeräts zugreifen oder Dateien dort speichern können.

In der Standardeinstellung ist der Zugriff auf optische Clientlaufwerke zugelassen.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, müssen Sie auch für die Einstellung Clientlaufwerkumleitung die Option Zugelassen wählen. Wenn diese Einstellungen deaktiviert sind, können optische Clientlaufwerke nicht zugeordnet werden und Benutzer können auch nicht manuell auf diese Laufwerke zugreifen – unabhängig von der Einstellung für Optische Clientlaufwerke.

Konfigurieren Sie außerdem die Einstellung Clientlaufwerke automatisch verbinden, damit optische Clientlaufwerke automatisch verbunden werden.

### Clientwechsellaufwerke

Mit dieser Einstellung legen Sie fest, ob Benutzer auf die USB-Laufwerke des Benutzergeräts zugreifen oder Dateien speichern können.

In der Standardeinstellung ist der Zugriff auf Clientwechsellaufwerke zugelassen.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, müssen Sie auch für die Einstellung Clientlaufwerkumleitung die Option Zugelassen wählen. Wenn diese Einstellungen deaktiviert sind, können Clientwechsellaufwerke nicht zugeordnet werden und Benutzer können auch nicht manuell auf diese Laufwerke zugreifen – unabhängig von der Einstellung für Clientwechsellaufwerke.

Konfigurieren Sie außerdem die Einstellung Clientlaufwerke automatisch verbinden, damit Clientwechsellaufwerke automatisch verbunden werden.

## Host-zu-Client-Umleitung

Mit dieser Einstellung aktivieren oder deaktivieren Sie Dateitypzuordnungen für URLs und manche Medieninhalte, damit sie auf dem Clientgerät geöffnet werden. Wenn deaktiviert, werden Inhalte auf dem Server geöffnet.

In der Standardeinstellung ist die Dateitypzuordnung deaktiviert.

Diese Art von URLs werden lokal geöffnet, wenn Sie die Einstellung aktivieren:

- Hypertext Transfer Protocol (HTTP)
- Secure Hypertext Transfer Protocol (HTTPS)
- Real Player und QuickTime (RTSP)
- Real Player und QuickTime (RTSPU)
- Ältere Real Player-URLs (PNM)
- Microsoft Media Server (MMS)

## Clientlaufwerksbuchstaben erhalten

Mit dieser Einstellung aktivieren oder deaktivieren Sie, ob die Clientlaufwerksbuchstaben erhalten bleiben.

In der Standardeinstellung bleiben die Clientlaufwerksbuchstaben nicht erhalten.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, müssen Sie auch für die Einstellung Clientlaufwerkumleitung die Option Zugelassen wählen.

## Schreibgeschützter Zugriff auf Clientlaufwerke

Diese Einstellung erlaubt oder verhindert, dass Benutzer und Anwendungen Dateien oder Ordner auf zugeordneten Clientlaufwerken erstellen oder ändern.

Standardmäßig können Dateien und Ordner auf zugeordneten Clientlaufwerken geändert werden.

Wenn die Einstellung auf Aktiviert gesetzt wird, ist Lesezugriff auf die Dateien und Verzeichnisse möglich.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, müssen Sie auch für die Einstellung Clientlaufwerkumleitung die Option Zugelassen wählen.

## Umleitung spezieller Ordner

Mit dieser Einstellung legen Sie fest, ob Benutzer von Citrix Receiver und dem Webinterface ihre lokalen speziellen Ordner in einer Sitzung sehen, z. B. "Dokumente" und "Desktop".

In der Standardeinstellung ist die Umleitung spezieller Ordner zugelassen.

Diese Einstellung verhindert, dass jegliche Objekte, die durch eine Richtlinie gefiltert werden, die Umleitung spezieller Ordner verwenden. Einstellungen an anderer Stelle werden nicht beachtet. Wenn diese Einstellung nicht zugelassen ist, werden verwandte Einstellungen im Webinterface, in StoreFront oder Citrix Receiver ignoriert.

Sie legen fest, welche Benutzer die Umleitung spezieller Ordner erhalten, indem Sie Zugelassen wählen und diese Einstellung in eine Richtlinie aufnehmen, die nach den Benutzern gefiltert wird, denen diese Funktion zur Verfügung stehen soll. Diese Einstellung überschreibt alle anderen Einstellungen für die Umleitung spezieller Ordner.

Die Umleitung spezieller Ordner interagiert mit dem Clientgerät. Daher verhindern Einstellungen, die den Benutzerzugriff auf lokale Festplatten untersagen, auch die Umleitung spezieller Ordner.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, müssen Sie auch für die Einstellung Lokale Clientfestplattenlaufwerke die Option Zugelassen wählen.

### Asynchrones Schreiben verwenden

Mit dieser Einstellung aktivieren oder deaktivieren Sie asynchrones Schreiben auf Laufwerke.

Standardmäßig ist das asynchrone Schreiben deaktiviert.

Für Verbindungen über WANs, die normalerweise durch relativ hohe Bandbreite und hohe Latenz charakterisiert sind, können Sie durch asynchrone Schreibvorgänge die Dateiübertragungen und Schreibvorgänge auf Clientlaufwerke beschleunigen. Sollte jedoch ein Verbindungsfehler oder Datenträgerfehler auftreten, können die Clientdateien, die geschrieben werden, in einem nicht definierten Zustand enden. Dem Benutzer werden dann in einem Pop-up-Fenster die betroffenen Dateien angezeigt. Der Benutzer kann das Problem beheben, z. B. durch Neustart einer unterbrochenen Dateiübertragung bei der Wiederverbindung oder nach Beheben eines Datenträgerfehlers.

Citrix empfiehlt, dass asynchrone Schreibvorgänge auf Datenträgern nur für Benutzer implementiert werden, die eine Remoteverbindung mit guter Geschwindigkeit für die Dateiübertragungen benötigen, und die verlorene Dateien oder Daten problemlos wiederherstellen können, sollten Fehler bei der Verbindung oder dem Datenträger auftreten.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, müssen Sie auch für die Einstellung Clientlaufwerkumleitung die Option Zugelassen wählen. Wenn diese Einstellung deaktiviert ist, finden keine asynchronen Schreibvorgänge statt.

# Einstellungen der Richtlinie "Flash-Umleitung"

May 10, 2016

Der Abschnitt "Flash-Umleitung" enthält Richtlinieneinstellungen für die Behandlung von Flash-Inhalten in Benutzersitzungen.

## Flash-Beschleunigung

Aktiviert oder deaktiviert die Wiedergabe von Flash-Inhalten auf Benutzergeräten statt auf dem Server. Standardmäßig ist die clientseitige Wiedergabe von Flash-Inhalten aktiviert.

Hinweis: Diese Einstellung wird für die Legacy-Flash-Umleitung mit dem Citrix Online Plug-In 12.1 verwendet. Wenn aktiviert, reduziert diese Einstellung die Netzwerk- und Serverlast, indem Flash-Inhalte auf dem Clientgerät wiedergegeben werden. Zusätzlich erreichen Sie mit der Einstellung Flash-URL-Kompatibilitätsliste, dass Flash-Inhalte von bestimmten Websites auf dem Server wiedergegeben werden.

Auf dem Benutzergerät muss die Einstellung HDX MediaStream Flash-Umleitung auf dem Benutzergerät aktivieren auch aktiviert sein.

Ist diese Einstellung deaktiviert, werden Flash-Inhalte von allen Websites, unabhängig von deren URLs, auf dem Server wiedergegeben. Damit nur die Flash-Inhalte bestimmter Websites auf dem Benutzergerät wiedergegeben werden, konfigurieren Sie die Einstellung Flash-URL-Kompatibilitätsliste.

## Flash-Hintergrundfarbenliste

Mit dieser Einstellung können Sie Schlüsselfarben für bestimmte URLs festlegen.

In der Standardeinstellung sind keine Schlüsselfarben angegeben.

Schlüsselfarben werden hinter clientseitig wiedergegebenem Flash angezeigt und unterstützen die visuelle Erkennung von Bereichen. Die angegebene Schlüsselfarbe sollte selten sein, sonst funktioniert die visuelle Bereichserkennung ggf. nicht richtig.

Gültige Eingaben bestehen aus einer URL (mit optionalen Platzhaltern am Anfang und Ende), der ein 24-Bit-RGB-Farbcodex folgt. Zum Beispiel: <http://citrix.com> 000003.

Stellen Sie sicher, dass die angegebene URL die URL für den Flash-Inhalt ist. Sie weicht möglicherweise von der URL der Website ab.

## Warnung

Die unsachgemäße Verwendung des Registrierungs-Editors kann zu schwerwiegenden Problemen führen, die nur durch eine Neuinstallation des Betriebssystems gelöst werden können. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine falsche Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

Auf VDA-Maschinen, auf denen Windows 8 oder Windows 2012 ausgeführt wird, werden mit dieser Einstellung möglicherweise nicht die Schlüsselfarben für die URL festgelegt. Wenn dieses Problem auftritt, bearbeiten Sie die

Registrierung auf der VDA-Maschine.

Verwenden Sie bei 32-Bit-Maschinen die folgende Registrierungseinstellung:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\HdxMediaStreamForFlash\Server\PseudoServer]
"ForceHDXFlashEnabled"=dword:00000001
```

Verwenden Sie bei 64-Bit-Maschinen die folgende Registrierungseinstellung:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\HdxMediaStreamForFlash\Server\PseudoServer]
"ForceHDXFlashEnabled"=dword:00000001
```

## Flash-Abwärtskompatibilität

Mit dieser Einstellung aktivieren oder deaktivieren Sie die Verwendung von Originalfeatures für die Legacy-Flash-Umleitung mit älteren Versionen von Citrix Receiver (früher Citrix Online Plug-In).

Standardmäßig ist diese Einstellung aktiviert.

Auf dem Benutzergerät muss die Einstellung HDX MediaStream Flash-Umleitung auf dem Benutzergerät aktivieren auch aktiviert sein.

Flash-Umleitungsfeatures der zweiten Generation können mit Citrix Receiver 3.0 verwendet werden. Legacyumleitungsfeatures werden mit dem Citrix Online Plug-In 12.1 unterstützt. Um sicherzustellen, dass die Flash-Umleitungsfeatures der zweiten Generation verwendet werden, muss die Flash-Umleitung der zweiten Generation auf dem Server und dem Benutzergerät aktiviert sein. Wenn die Legacyumleitung entweder auf dem Server oder dem Benutzergerät aktiviert ist, werden die Legacyumleitungsfeatures verwendet.

## Flash-Standardverhalten

Mit dieser Einstellung wird das Standardverhalten für die Flash-Beschleunigung der zweiten Generation festgelegt.

Standardmäßig ist die Flash-Beschleunigung aktiviert.

Wählen Sie für die Konfiguration dieser Einstellung eine der folgenden Optionen:

- Flash-Beschleunigung aktivieren: Flash-Umleitung wird verwendet.
- Flash Player blockieren: Flash-Umleitung sowie serverseitiges Rendering werden nicht verwendet. Der Benutzer kann keine Flash-Inhalte anzeigen.
- Flash-Beschleunigung deaktivieren: Flash-Umleitung wird nicht verwendet. Der Benutzer kann auf dem Server wiedergegebene Flash-Inhalte anzeigen, wenn ein mit den Inhalten kompatibler Adobe Flash Player für Windows Internet Explorer auf dem Server installiert ist.

Diese Einstellung kann für einzelne Webseiten und Flash-Instanzen basierend auf der Einstellung Flash-URL-Kompatibilitätsliste überschrieben werden. Außerdem muss auf dem Benutzergerät die Einstellung HDX MediaStream Flash-Umleitung auf dem Benutzergerät aktivieren aktiviert sein.

## Flash-Ereignisprotokollierung

Mit dieser Einstellung aktivieren oder deaktivieren Sie die Aufzeichnung von Flash-Ereignissen im Windows-Anwendungsereignisprotokoll.

Standardmäßig ist die Protokollierung aktiviert.

Auf Computern unter Windows 7 oder Windows Vista wird ein spezielles Protokoll für die Flash-Umleitung im Knoten "Anwendungs- und Dienstprotokolle" angezeigt.

## Flash - intelligentes Fallback

Mit dieser Einstellung aktivieren oder deaktivieren Sie automatische Versuche, das serverseitige Rendering für Flash Player-Instanzen zu verwenden, wenn das clientseitige Rendering entweder unnötig ist oder eine schlechte Benutzererfahrung ergibt.

Standardmäßig ist diese Einstellung aktiviert.

## Flash-Latenzschwellenwert

Mit dieser Einstellung geben Sie einen Schwellenwert zwischen 0-30 Millisekunden an, um festzulegen, wo Adobe Flash-Inhalte wiedergegeben werden.

In der Standardeinstellung ist der Schwellenwert 30.

Beim Start wird die aktuelle Latenz zwischen Server und Benutzergerät von HDX MediaStream für Flash gemessen. Wenn die Latenz unter dem Schwellenwert liegt, werden Flash-Inhalte mit HDX MediaStream für Flash auf dem Benutzergerät wiedergegeben. Wenn die Latenz den Schwellenwert überschreitet, werden die Inhalte auf dem Netzwerkserver wiedergegeben, wenn Adobe Flash Player dort vorhanden ist.

Wenn Sie diese Einstellung aktivieren, muss die Einstellung Flash-Abwärtskompatibilität vorhanden und auf Zugelassen eingestellt sein.

Hinweis: Dies gilt nur, wenn HDX MediaStream Flash-Umleitung im Legacymodus verwendet wird.

## Verhindern von Flash-Videofallback

Mit dieser Einstellung geben Sie an, ob und wie "kleine" Flash-Inhalte gerendert und angezeigt werden.

Standardmäßig ist diese Einstellung nicht konfiguriert.

Wählen Sie für die Konfiguration dieser Einstellung eine der folgenden Optionen:

- **Only small content:** Es werden nur Intelligentes-Fallback-Inhalte auf dem Server gerendert, andere Flash-Inhalte werden durch eine Fehler-SWF-Datei ersetzt.
- **Only small content with a supported client:** Es werden nur Intelligentes-Fallback-Inhalte auf dem Server gerendert, wenn der Client aktuell die Flash-Umleitung verwendet, andere Flash-Inhalte werden durch eine Fehler-SWF-Datei ersetzt.
- **No server side content:** Alle Inhalte auf dem Server werden durch eine Fehler-SWF-Datei ersetzt.

Zum Verwenden dieser Richtlinieneinstellung müssen Sie eine Fehler-SWF-Datei angeben. Diese Fehler-SWF-Datei ersetzt alle Flash-Inhalte, die nicht auf dem VDA gerendert werden sollen.

## Fehler beim Verhindern von Flash-Videofallback \*.swf

Diese Einstellung legt die URL der Fehlermeldung fest, die anstelle von Flash-Instanzen angezeigt wird, wenn Server-Lastverwaltungsrichtlinien verwendet werden. Beispiel:

<http://domainName.tld/sample/path/error.swf>

## URL-Liste für serverseitigen Flash-Inhaltsabruf

Mit dieser Einstellung geben Sie die Websites an, deren Flash-Inhalte auf den Server heruntergeladen und dann auf das Benutzergerät zur Wiedergabe übertragen werden.

In der Standardeinstellung sind keine Sites angegeben.

Diese Einstellung wird verwendet, wenn das Benutzergerät keinen direkten Internetzugang hat; der Server stellt die Verbindung bereit. Auf dem Benutzergerät muss die Einstellung Serverseitigen Inhaltsabruf aktivieren auch aktiviert sein.

Die Flash-Umleitung der zweiten Generation enthält ein Fallback für den serverseitigen Flash-Inhaltsabruf für SWF-Dateien. Wenn ein Benutzergerät Flash-Inhalte nicht von einer Website abrufen kann, und die Website in der URL-Liste für serverseitigen Flash-Inhaltsabruf angegeben ist, erfolgt automatisch der serverseitige Inhaltsabruf.

Hinzufügen von URLs zur Liste:

- Fügen Sie die URL der Flash-Anwendung und nicht die HTML-Seite auf oberster Ebene hinzu, die Flash Player initiiert.
- Verwenden Sie ein Sternchen (\*) am Anfang oder am Ende der URL als Platzhalterzeichen.
- Verwenden Sie den Platzhalter am Ende, um alle untergeordneten URLs zuzulassen (<http://www.citrix.com/>).
- Die Präfixe <http://> und <https://> werden verwendet (falls vorhanden), sind jedoch für gültige Listeneinträge nicht vorgeschrieben.

### Flash-URL-Kompatibilitätsliste

Mit dieser Einstellung geben Sie die Regeln an, die festlegen, ob die Wiedergabe von Flash-Inhalten auf bestimmten Websites auf dem Benutzergerät oder auf dem Server erfolgt oder ob sie blockiert wird.

In der Standardeinstellung sind keine Regeln angegeben.

Hinzufügen von URLs zur Liste:

- Sortieren Sie die Liste nach Priorität, sodass die wichtigsten URLs, Aktionen und Wiedergabeorte ganz oben stehen.
- Verwenden Sie ein Sternchen (\*) am Anfang oder am Ende der URL als Platzhalterzeichen.
- Verwenden Sie den Platzhalter am Ende, um auf alle untergeordneten URLs zu verweisen (<http://www.citrix.com/>).
- Die Präfixe <http://> und <https://> werden verwendet (falls vorhanden), sind jedoch für gültige Listeneinträge nicht vorgeschrieben.
- Fügen Sie der Liste Websites hinzu, deren Flash-Inhalte nicht richtig auf dem Benutzergerät wiedergegeben werden, und wählen Sie entweder die Optionen Auf Server rendern oder Blockieren.

# Einstellungen der Richtlinie "Grafiken"

May 10, 2016

Der Abschnitt "Grafiken" enthält Richtlinieneinstellungen, mit denen Sie steuern, wie Bilder in Benutzersitzungen behandelt werden.

## Anzeigespeicherlimit

Mit dieser Einstellung geben Sie die maximale Größe des Videopuffers (in Kilobytes) für die Sitzung an.

Das Standardlimit ist 65536 Kilobytes.

Für Verbindungen, die eine größere Farbtiefe und eine höhere Auflösung erfordern, erhöhen Sie den Grenzwert. Berechnen Sie den maximal erforderlichen Arbeitsspeicher mit dieser Formel:

$\text{Speicher in Byte} = (\text{Farbtiefe in Bits pro Pixel} / 8) * (\text{vertikale Auflösung in Pixel}) * (\text{horizontale Auflösung in Pixel})$

Beispiel: Bei einer Farbtiefe von 32, einer vertikalen Auflösung von 600 und einer horizontalen Auflösung von 800 ergibt dies einen maximal erforderlichen Arbeitsspeicher von  $(32 / 8) * (600) * (800) = 1920000$  Byte, was ein Anzeigespeicherlimit von 1920 KB ergibt.

Andere Farbtiefen als 32 Bit sind nur verfügbar, wenn die Richtlinieneinstellung Legacygrafikmodus aktiviert ist.

HDX weist Benutzern nur den pro Sitzung erforderlichen Anzeigespeicher zu. Wenn also nur einige Benutzer mehr als den Standardspeicher benötigen, hat das Erhöhen des Anzeigespeicherlimits keine negativen Auswirkungen auf die Skalierbarkeit.

## Herabsetzungspräferenz für Anzeigemodus

Hinweis: Bei Virtual Delivery Agent 7.x gilt diese Richtlinieneinstellung nur, wenn die Richtlinieneinstellung Legacygrafikmodus aktiviert ist.

Diese Einstellung gibt an, ob die Farbtiefe oder die Auflösung zuerst herabgesetzt werden soll, wenn das Speicherlimit für die Sitzung erreicht wird.

Standardmäßig wird die Farbtiefe zuerst herabgesetzt.

Wenn das Speicherlimit der Sitzung erreicht wird, können Sie die Bildqualität verringern, indem Sie erst die Farbtiefe oder erst die Auflösung herabsetzen. Wird erst die Farbtiefe herabgesetzt, werden Bilder mit weniger Farben dargestellt. Wird erst die Auflösung herabgesetzt, werden Bilder mit weniger Pixel pro Zoll angezeigt.

Wenn Benutzer in dem Fall benachrichtigt werden sollen, dass entweder die Farbtiefe oder die Auflösung herabgesetzt werden muss, konfigurieren Sie die Einstellung Benutzer beim Herabsetzen des Anzeigemodus benachrichtigen.

## Dynamische Fenstervorschau

Mit dieser Einstellung kann die Anzeige von Seamlessfenster in den Fenstervorschau Modi Flip, Flip 3D, Taskleistenvorschau und Einsehen aktiviert bzw. deaktiviert werden.

| Windows Aero-Vorschauoption | Beschreibung                                                                                  |
|-----------------------------|-----------------------------------------------------------------------------------------------|
| Symbolleistenvorschau       | Wenn der Benutzer auf das Symbol eines Fensters zeigt, wird ein Bild dieses Fensters über der |



|                                                        |                                                                                                                                                  |
|--------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Windows Aero-Vorschauoption</b><br>Fenster-Vorschau | Symbolleiste angezeigt.                                                                                                                          |
|                                                        | <b>Beschreibung</b><br>Wenn der Benutzer auf ein Symbolleistenvorschaubild zeigt, wird das Bild in voller Größe auf dem Bildschirm angezeigt.    |
| Flip                                                   | Wenn der Benutzer Alt + Tab drückt, werden kleine Vorschausymbole für jedes geöffnete Fenster angezeigt.                                         |
| Flip-3D                                                | Wenn der Benutzer die Tabulator- und Windows-Tasten drückt, werden große Bilder der geöffneten Fenster überlappend auf dem Bildschirm angezeigt. |

Standardmäßig ist diese Einstellung aktiviert.

## Bildzwischenspeicherung

Hinweis: Bei Virtual Delivery Agent 7.x gilt diese Richtlinieneinstellung nur, wenn die Richtlinieneinstellung Legacygrafikmodus aktiviert ist.

Mit dieser Einstellung aktivieren oder deaktivieren Sie das Zwischenspeichern und Abrufen von Bildabschnitten in Sitzungen. Durch das Zwischenspeichern von Bildern in Abschnitten und das Abrufen dieser Abschnitte wird das Ruckeln beim Bildlauf verringert. Darüber hinaus werden weniger Daten über das Netzwerk übertragen und die Verarbeitung auf dem Benutzergerät wird reduziert.

Standardmäßig ist die Einstellung für die Bildzwischenspeicherung aktiviert.

Hinweis: Die Einstellung für die Bildzwischenspeicherung steuert, wie Bilder zwischengespeichert und abgerufen werden, jedoch nicht, ob sie zwischengespeichert werden. Wenn die Einstellung "Legacygrafikmodus" aktiviert ist, werden Bilder zwischengespeichert.

## Legacygrafikmodus

Diese Einstellung deaktiviert die umfassende Grafikdarstellung und verwendet den Legacygrafikmodus, um die Skalierbarkeit über ein WAN oder eine mobile Verbindung zu verbessern.

Die Einstellung ist standardmäßig deaktiviert und die umfassende Grafikdarstellung wird verwendet.

## Maximal zugelassene Farbtiefe

Hinweis: Bei Virtual Delivery Agent 7.x gilt diese Richtlinieneinstellung nur, wenn die Richtlinieneinstellung Legacygrafikmodus aktiviert ist.

Mit dieser Einstellung geben Sie die maximale Farbtiefe an, die für eine Sitzung zulässig ist.

Die Standardeinstellung für die maximal zulässige Farbtiefe ist 32 Bits pro Pixel.

Diese Einstellung gilt nur für ThinWire-Treiber und -Verbindungen. Sie gilt nicht für VDAs mit einem anderen Treiber als ThinWire für die primäre Anzeige, z. B. VDAs mit WDDM-Treiber (Windows Display Driver Model). Bei Desktopbetriebssystem-VDAs mit einem WDDM-Treiber als primären Anzeigetreiber, z. B. Windows 8, hat diese Einstellung keine Auswirkung. Bei Windows-Serverbetriebssystem-VDAs mit WDDM-Treiber, z. B. Windows Server 2012 R2, kann diese Einstellung verhindern, dass Benutzer eine Verbindung mit dem VDA herstellen.

Für eine hohe Farbtiefe ist mehr Speicher erforderlich. Damit die Farbtiefe herabgesetzt wird, wenn das Speicherlimit erreicht wurde, konfigurieren Sie die Einstellung Herabsetzungspräferenz für Anzeigemodus. Wird die Farbtiefe herabgesetzt, werden Bilder mit weniger Farben dargestellt.

## Benutzer beim Herabsetzen des Anzeigemodus benachrichtigen

Hinweis: Bei Virtual Delivery Agent 7.x gilt diese Richtlinieneinstellung nur, wenn die Richtlinieneinstellung Legacygrafikmodus aktiviert ist.

Mit dieser Einstellung erzielen Sie, dass Benutzer eine kurze Erklärung erhalten, wenn die Farbtiefe oder die Auflösung herabgesetzt wird.

Standardmäßig werden Benutzer nicht benachrichtigt.

## Warteschlange und Verwerfen

Hinweis: Bei Virtual Delivery Agent 7.x gilt diese Richtlinieneinstellung nur, wenn die Richtlinieneinstellung Legacygrafikmodus aktiviert ist.

Mit dieser Einstellung werden Bilder in der Warteschlange verworfen, die durch ein anderes Bild ersetzt wurden.

Standardmäßig ist diese Einstellung aktiviert.

Dies verbessert die Reaktionszeit, wenn Grafiken an das Benutzergerät gesendet werden. Wenn Sie diese Einstellung konfigurieren, ruckeln Animationen möglicherweise, weil Frames ausgelassen werden.

# Einstellungen der Richtlinie "Zwischenspeichern"

May 10, 2016

Der Abschnitt "Zwischenspeichern" enthält Einstellungen, mit denen Bilddaten auf Benutzergeräten zwischengespeichert werden können, wenn Clientverbindungen eine beschränkte Bandbreite haben.

## Schwellenwert für permanenten Cache

Hinweis: Bei Virtual Delivery Agent 7.x gilt diese Richtlinieneinstellung nur, wenn die Richtlinieneinstellung Legacygrafikmodus aktiviert ist.

Mit dieser Einstellung werden Bitmaps auf der Festplatte des Benutzergeräts zwischengespeichert. Dies ermöglicht eine Wiederverwendung von großen, oft verwendeten Bildern aus früheren Sitzungen.

Der Standardschwellenwert ist 3000000 Bits pro Sekunde.

Der Schwellenwert ist der Wert, unter dem das Feature "Permanentcache" angewendet wird. Beispielsweise werden mit dem Standardwert Bitmaps auf der Festplatte des Benutzergeräts zwischengespeichert, wenn die Bandbreite unter 3000000 Bit/s fällt.

# Einstellungen der Richtlinie "Keep-Alive"

May 10, 2016

Der Abschnitt "Keep-Alive" enthält Richtlinieneinstellungen für die Verwaltung der ICA-Keep-Alive-Meldungen.

## ICA-Keep-Alive - Timeout

Mit dieser Einstellung geben Sie die Anzahl der Sekunden zwischen aufeinanderfolgenden ICA-Keep-Alive-Meldungen an.

Das Standardintervall zwischen Keep-Alive-Meldungen ist 60 Sekunden.

Geben Sie ein Intervall zwischen 1-3600 Sekunden an, in dem ICA-Keep-Alive-Meldungen gesendet werden. Konfigurieren Sie diese Einstellung nicht, wenn Sie eine Netzwerküberwachungssoftware zum Schließen inaktiver Verbindungen verwenden.

## ICA-Keep-Alives

Mit dieser Einstellung legen Sie fest, ob ICA-Keep-Alive-Meldungen in regelmäßigen Abständen gesendet werden sollen.

Standardmäßig werden keine Keep-Alive-Meldungen gesendet.

Wenn Sie diese Einstellung aktivieren, wird verhindert, dass unterbrochene Verbindungen getrennt werden. Wenn der Server keine Aktivität feststellt, verhindert diese Einstellung, dass die Sitzung durch die Remotedesktopdienste getrennt wird. Der Server sendet alle paar Sekunden Keep-Alive-Meldungen, um zu ermitteln, ob die Sitzung aktiv ist. Wenn die Sitzung nicht mehr aktiv ist, wird die Sitzung vom Server als "Getrennt" gekennzeichnet.

ICA-Keep-Alive funktioniert nicht, wenn Sie die Sitzungszuverlässigkeit verwenden. Konfigurieren Sie daher ICA-Keep-Alive nur für Verbindungen, die die Sitzungszuverlässigkeit nicht verwenden.

Verwandte Richtlinieneinstellungen: Sitzungszuverlässigkeit - Verbindungen.

# Einstellungen der Richtlinie "Lokaler App-Zugriff"

May 10, 2016

Der Abschnitt "Lokaler App-Zugriff" enthält Richtlinieneinstellungen, mit denen Sie die Integration lokal installierter Anwendungen mit gehosteten Anwendungen in einer gehosteten Desktopumgebung konfigurieren können.

## Lokalen App-Zugriff zulassen

Mit dieser Einstellung legen Sie fest, ob die Integration lokal installierter Anwendungen mit gehosteten Anwendungen in einer gehosteten Desktopumgebung zugelassen oder verweigert werden soll.

Wenn ein Benutzer eine lokal installierte Anwendung startet, wirkt es so, als ob diese auf dem virtuellen Desktop des Benutzers ausgeführt würde, obwohl sie tatsächlich lokal ausgeführt wird.

Standardmäßig ist der lokale App-Zugriff nicht zulässig.

## URL-Umleitungssperrliste

Mit dieser Einstellung geben Sie Websites an, die Ziel einer Weiterleitung sind und im lokalen Webbrowser gestartet werden sollen. Dies kann auch Websites umfassen, für die Gebietsschema-Informationen erforderlich sind (z. B. msn.com oder newsgoogle.com) oder Websites mit reichhaltigen Medieninhalten, die besser auf dem Benutzergerät wiedergegeben werden.

In der Standardeinstellung sind keine Sites angegeben.

## URL-Umleitungspositivliste

Mit dieser Einstellung geben Sie die Websites an, die in der Umgebung, in der sie gestartet werden, wiedergegeben werden sollen.

In der Standardeinstellung sind keine Sites angegeben.

# Einstellungen der Richtlinie "Mobilerfahrung"

May 10, 2016

Der Abschnitt "Mobilerfahrung" enthält Richtlinieneinstellungen für die Handhabung des Citrix Mobility Packs.

## Automatische Anzeige der Tastatur

Diese Einstellung aktiviert oder deaktiviert die automatische Anzeige der Tastatur auf Bildschirmen von Mobilgeräten.

Standardmäßig ist die automatische Anzeige der Tastatur deaktiviert.

## Für Fingereingabe optimierten Desktop starten

Diese Einstellung ist deaktiviert und für Windows 10-Maschinen nicht verfügbar.

Diese Einstellung bestimmt das allgemeine Verhalten der Receiver-Benutzeroberfläche, indem sie eine für die Fingereingabe optimierte Benutzeroberfläche zulässt oder verweigert, die für Tablet-Geräte optimiert ist.

Standardmäßig wird eine für die Fingereingabe optimierte Benutzeroberfläche verwendet.

Setzen Sie diese Richtlinie auf "Nicht zugelassen", um nur die Windows-Benutzeroberfläche zu verwenden.

## Kombinationsfelder remoten

Diese Einstellung bestimmt die Typen von Kombinationsfeldern, die in Sitzungen auf mobilen Geräten angezeigt werden können. Stellen Sie diese Richtlinie auf Zugelassen ein, um das gerätenative Kombinationsfeld-Steuererelement anzuzeigen. Wenn diese Einstellung zugelassen ist, kann ein Benutzer eine Sitzungseinstellung in Receiver für iOS ändern und das Windows-Kombinationsfeld verwenden.

Standardmäßig wird die Funktion zum Remoten von Kombinationsfeldern verweigert.

# Einstellungen der Richtlinie "Multimedia"

May 10, 2016

Der Abschnitt "Multimedia" enthält Richtlinieneinstellungen, mit denen Sie das Streaming von Audio und Video in Benutzersitzungen verwalten.

## Videoqualität beschränken

Mit dieser Einstellung geben Sie die maximale Videoqualitätsstufe für eine HDX-Verbindung an. Wird die Einstellung konfiguriert, dann wird die Videoqualität auf den angegebenen Wert beschränkt, sodass die Dienstqualität für Multimedia in der Umgebung gewährleistet ist.

Standardmäßig ist diese Einstellung nicht konfiguriert.

Zum Festlegen der maximalen Qualität wählen Sie eine der folgenden Optionen:

- 1080p/8,5 MBit/s
- 720p/4,0 MBit/s
- 480p/720 KBit/s
- 380p/400 KBit/s
- 240p/200 KBit/s

Hinweis: Die gleichzeitige Wiedergabe mehrerer Videos auf einem Server verbraucht viele Ressourcen und kann die Skalierbarkeit des Servers beeinträchtigen.

## Multimediakonferenzen

Mit dieser Einstellung legen Sie fest, ob die Unterstützung von Videokonferenzanwendungen zugelassen ist.

Standardmäßig ist die Unterstützung für Videokonferenzen zugelassen.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, muss die Einstellung Windows Media-Umleitung vorhanden und auf Zugelassen gesetzt sein.

Für Multimediakonferenzen müssen folgende Bedingungen erfüllt sein:

- Der Gerätetreiber des Herstellers für die in der Multimediakonferenz verwendete Webcam muss installiert sein.
- Die Webcam muss mit dem Clientgerät verbunden werden, bevor eine Videokonferenzsitzung initiiert wird. Der Server verwendet zu jedem Zeitpunkt nur eine installierte Webcam. Wenn mehrere Webcams auf dem Benutzergerät installiert sind, versucht der Server nacheinander jede Webcam zu verwenden, bis eine Videokonferenzsitzung steht.

## Optimierung von Windows Media-Multimediaumleitung über WAN

Mit dieser Einstellung aktivieren Sie die Multimediatranscodierung in Echtzeit. Damit wird Audio- und Videostreaming für mobile Geräte ermöglicht und die Benutzererfahrung durch eine verbesserte Übermittlung von Windows Media-Inhalt über WAN optimiert.

Standardmäßig wird die Bereitstellung von Windows Media-Inhalt über das WAN optimiert.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, muss die Einstellung Windows Media-Umleitung vorhanden und auf Zugelassen gesetzt sein.

Wenn diese Einstellung aktiviert ist, wird die Multimediatranscodierung nach Bedarf automatisch bereitgestellt, sodass Audio- und Videostreaming zur Verbesserung der Benutzererfahrung auch bei schlechten Netzwerkbedingungen ermöglicht

wird.

## GPU für die Optimierung von Windows Media Multimediaumleitung über WAN verwenden

Mit dieser Einstellung wird die Transcodierung von Multimediainhalten in Echtzeit im Grafikprozessor (GPU) des Virtual Delivery Agent (VDA) ermöglicht, um die Serverskalierbarkeit zu verbessern. Die GPU-Transcodierung ist nur verfügbar, wenn der VDA eine unterstützte GPU für die Hardwarebeschleunigung hat. Andernfalls erfolgt die Transcodierung automatisch in der CPU.

Hinweis: GPU-Transcodierung wird nur von NVIDIA-GPUs unterstützt.

Standardmäßig ist die Verwendung der GPU auf dem VDA zum Optimieren der Bereitstellung von Windows Media-Inhalt über das WAN nicht zulässig.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, stellen Sie sicher, dass die Einstellungen Windows Media-Umleitung und Optimierung von Windows Media-Multimediaumleitung über WAN vorhanden und auf Zugelassen gesetzt sind.

## Verhindern von Videofallback

Administratoren können über die Einstellung der Richtlinie "Verhindern von Videofallback" die Methoden für die Übertragung gestreamter Inhalte an Benutzer festlegen.

Standardmäßig ist diese Einstellung nicht konfiguriert. Dies ermöglicht ein Fallback vom clientseitigen Abruf auf RAVE und von dort auf den Server.

Wählen Sie für die Konfiguration dieser Einstellung eine der folgenden Optionen:

- **Serverseitiger Abruf - Serverseitige Wiedergabe:** ermöglicht ein Fallback vom clientseitigen Abruf auf RAVE und von dort auf den Server.
- **Serverseitiger Abruf - Clientseitige Wiedergabe:** ermöglicht ein Fallback vom clientseitigen Abruf auf RAVE, jedoch kein Fallback von RAVE auf den Server.
- **Clientseitiger Abruf - Clientseitige Wiedergabe:** lässt kein Fallback vom clientseitigen Abruf auf RAVE und von dort auf den Server zu.

Wird der Inhalt nicht wiedergegeben, wird im Playerfenster gemeldet, dass das Video wegen mangelnder Ressourcen blockiert wurde.

## Clientseitiger Abruf von Windows Media-Inhalten

Diese Einstellung ermöglicht das Streamen von Multimediadateien anstatt über den Hostserver direkt vom Quellenanbieter im Internet oder Intranet auf Benutzergeräte.

Standardmäßig ist das direkte Streaming von Multimediadateien vom Quellenanbieter auf Benutzergeräte zulässig.

Zulassen dieser Einstellung verbessert die Netzwerknutzung und Serverskalierbarkeit durch Verschieben der gesamten Medienverarbeitung vom Hostserver auf das Benutzergerät. Dadurch wird auch das Erfordernis der Installation eines erweiterten Multimedia-Frameworks, z. B. von Microsoft DirectShow oder Media Foundation, auf Benutzergeräten hinfällig. Diese müssen lediglich eine Datei von einer URL abspielen können.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, muss die Einstellung Windows Media-Umleitung vorhanden und auf Zugelassen gesetzt sein. Wenn diese Einstellung deaktiviert ist, ist das direkte Streaming von Multimediadateien auf Benutzergeräte ebenfalls deaktiviert.



## Windows Media-Umleitung (Windows Media Redirection)

Diese Einstellung steuert und optimiert die Art und Weise, mit der Streamingaudio und -video Benutzern von Servern bereitgestellt wird.

Standardmäßig ist das Streamen von Audio und Video für Benutzer zulässig.

Zulassen dieser Einstellung erhöht die Qualität von auf dem Server wiedergegebenem Audio und Video auf ein Niveau, das dem der lokalen Wiedergabe auf dem Benutzergerät entspricht. Der Server streamt Multimediainhalte komprimiert im Originalformat zum Client, Dekomprimierung und Wiedergabe der Medien übernimmt das Benutzergerät.

Die Windows Media-Umleitung optimiert Multimediadateien, die mit Codecs verschlüsselt sind, die den Standards von Microsoft DirectShow, DirectX Media Objects (DMO) und Media Foundation entsprechen. Um eine Multimediadatei wiederzugeben, muss ein mit dem Codierungsformat der Multimediadatei kompatibler Codec auf dem Benutzergerät vorhanden sein.

In der Standardeinstellung ist Audio auf Citrix Receiver deaktiviert. Wenn Benutzer Multimedia-Anwendungen in ICA-Sitzungen ausführen können, aktivieren Sie Audio oder geben Sie den Benutzern in der Receiver-Benutzeroberfläche die Berechtigung, Audio zu aktivieren.

Wählen Sie Verweigert nur, wenn die Wiedergabe von Medien mit der Windows Media-Umleitung schlechter zu sein scheint, als mit der ICA-Komprimierung und regulärem Audio. Dies ist selten, kann aber mit geringer Bandbreite vorkommen, beispielsweise bei Medien, in denen die Schlüsselbilder (Keyframes) sehr weit auseinander liegen.

## Windows Media-Umleitungspuffergröße

Mit dieser Einstellung geben Sie für die Multimediabeschleunigung eine Puffergröße zwischen 1 und 10 Sekunden an.

Die Standardpuffergröße ist 5 Sekunden.

## Verwendung von Windows Media-Umleitungspuffergröße

Mit dieser Einstellung aktivieren oder deaktivieren Sie die Verwendung der unter Windows Media-Umleitungspuffergröße angegebenen Puffergröße.

In der Standardeinstellung wird die angegebene Puffergröße nicht verwendet.

Wenn diese Einstellung deaktiviert oder die Einstellung für die Windows Media-Umleitungspuffergröße nicht konfiguriert ist, verwendet der Server den Standardwert für die Puffergröße (5 Sekunden).

# Einstellungen der Richtlinie "Multistreamverbindungen"

May 10, 2016

Im Abschnitt "Multistreamverbindungen" finden Sie Richtlinieneinstellungen zum Verwalten der Quality-of-Service-Priorität (QoS) für mehrere ICA-Verbindungen in einer Sitzung.

## Audio über UDP

Mit dieser Einstellung legen Sie fest, ob Audio über UDP auf dem Server zugelassen wird.

Standardmäßig ist Audio über UDP auf dem Server zugelassen.

Wenn diese Einstellung aktiviert ist, wird ein UDP-Port auf dem Server geöffnet, sodass alle Verbindungen, die zur Verwendung von Audio über UDP - Real-time Transport konfiguriert sind, unterstützt werden.

## Audio-UDP-Portbereich

Mit dieser Einstellung geben Sie den Bereich der Portnummern an (im Format niedrigste Portnummer,höchste Portnummer), die von Virtual Desktop Agent (VDA) zum Austausch von Audiopaketsdaten mit dem Benutzergerät verwendet werden. Der VDA versucht, jedes UDP-Portpaar für den Austausch von Daten mit dem Benutzergerät zu verwenden. Dabei wird mit dem Port, der die niedrigste Nummer hat, begonnen und die Zahl für jeden folgenden Versuch um 2 erhöht. Alle Ports übernehmen eingehende und ausgehende Datenübertragungen.

Die Standardeinstellung ist 16500,16509.

## Multiportrichtlinie

Mit dieser Einstellung geben Sie die TCP-Ports an, die für den ICA-Verkehr verwendet werden sollen, und legen eine Netzwerkpriorität für jeden Port fest.

Standardmäßig hat der primäre Port (2598) eine hohe Priorität.

Wenn Sie Ports konfigurieren, können Sie die folgenden Prioritäten zuweisen:

- Sehr hoch: für Echtzeitvorgänge, z. B. Webkonferenzen.
- Hoch: für interaktive Elemente, z. B. Bildschirm, Tastatur und Maus.
- Mittel: für Massenvorgänge, z. B. Clientlaufwerkzuordnung.
- Niedrig: für Hintergrundaufgaben, z. B. Drucken.

Jeder Port muss eine eindeutige Priorität haben. Sie können also nicht eine sehr hohe Priorität sowohl für CGP-Port 1 als auch für CGP-Port 3 zuweisen.

Wenn Sie für einen Port keine Priorität einstellen möchten, setzen Sie den Wert für den Port auf 0. Sie können den primären Port nicht entfernen und Sie können seine Prioritätsstufe nicht ändern.

Wenn Sie diese Einstellung konfigurieren, starten Sie den Server neu. Diese Einstellung wird nur angewendet, wenn die Richtlinie Multistreamcomputereinstellung aktiviert ist.

## Multistreamcomputereinstellung

Mit dieser Einstellung aktivieren oder deaktivieren Sie Multistream auf dem Server.

Standardmäßig ist Multistream deaktiviert.

Wenn Sie Citrix Cloudbridge mit Multistream-Unterstützung in Ihrer Umgebung verwenden, müssen Sie diese Einstellung nicht konfigurieren. Konfigurieren Sie diese Richtlinieneinstellung, wenn Sie Router von Drittanbietern oder Legacy-Branch Repeater verwenden, um die gewünschte Quality of Service (QoS) zu erzielen.

Wenn Sie diese Einstellung konfigurieren, starten Sie den Server neu, um sicherzustellen, dass die Änderungen wirksam werden.

Wichtig: Das Verwenden dieser Richtlinieneinstellung mit den Richtlinieneinstellungen für das Bandbreitenlimit, wie beispielsweise Bandbreitenlimit für Sitzung insgesamt kann zu unerwarteten Ergebnissen führen. Wenn Sie diese Einstellung in eine Richtlinie aufnehmen, stellen Sie sicher, dass Sie keine Bandbreitenlimit-Einstellungen einschließen.

### Multistreambenutzereinstellung

Mit dieser Einstellung aktivieren oder deaktivieren Sie Multistream auf dem Benutzergerät.

Standardmäßig ist Multistream für alle Benutzer deaktiviert.

Diese Einstellung wird nur auf Hosts angewendet, für die die Richtlinie Multistreamcomputereinstellung aktiviert ist.

Wichtig: Das Verwenden dieser Richtlinieneinstellung mit den Richtlinieneinstellungen für das Bandbreitenlimit, z. B. Bandbreitenlimit für Sitzung insgesamt, kann zu unerwarteten Ergebnissen führen. Wenn Sie diese Einstellung in eine Richtlinie aufnehmen, stellen Sie sicher, dass Sie keine Bandbreitenlimit-Einstellungen einschließen.

# Einstellungen der Richtlinie "Portumleitung"

May 10, 2016

Der Abschnitt "Portumleitung" enthält Richtlinieneinstellungen für die LPT- und COM-Portzuordnung auf dem Client.

Hinweis: Konfigurieren Sie bei Virtual Delivery Agent 7.x diese Einstellungen über die Registrierung (siehe [Konfigurieren von COM-Port- und LPT-Portumleitungseinstellungen in der Registrierung](#)).

## Client-COM-Ports automatisch verbinden

Mit dieser Einstellung aktivieren oder deaktivieren Sie die automatische Verbindung von COM-Ports auf dem Benutzergerät, wenn Benutzer sich bei der Site anmelden.

Standardmäßig werden COM-Ports nicht automatisch verbunden.

## Client-LPT-Ports automatisch verbinden

Mit dieser Einstellung aktivieren oder deaktivieren Sie die automatische Verbindung von LPT-Ports auf dem Benutzergerät, wenn Benutzer sich bei der Site anmelden.

Standardmäßig werden LPT-Ports nicht automatisch verbunden.

## Client-COM-Portumleitung

Mit dieser Einstellung legen Sie fest, ob der Zugriff auf COM-Ports des Benutzergeräts zulässig ist.

Standardmäßig ist die COM-Portumleitung nicht zugelassen.

Die folgenden Richtlinieneinstellungen hängen zusammen:

- Bandbreitenlimit für COM-Portumleitung
- Bandbreitenlimit für COM-Portumleitung (Prozent)

## Client-LPT-Portumleitung

Mit dieser Einstellung legen Sie fest, ob der Zugriff auf LPT-Ports des Benutzergeräts zulässig ist.

Standardmäßig ist die LPT-Portumleitung nicht zugelassen.

LPT-Ports werden nur von Legacyanwendungen verwendet, die Druckaufträge an LPT-Ports senden und nicht an die Druckerobjekte auf dem Benutzergerät. Die meisten Geräte können heute Druckaufträge und Druckerobjekte senden. Diese Richtlinieneinstellung ist nur für Server erforderlich, auf denen Legacyanwendungen gehostet werden, die für das Drucken LPT-Ports verwenden.

Die folgenden Richtlinieneinstellungen hängen zusammen:

- Bandbreitenlimit für LPT-Portumleitung
- Bandbreitenlimit für LPT-Portumleitung (Prozent)

# Einstellungen der Richtlinie "Drucken"

May 10, 2016

Der Abschnitt "Drucken" enthält Richtlinieneinstellungen für die Verwaltung des Clientdrucks.

## Clientdruckerumleitung

Mit dieser Einstellung legen Sie fest, ob Clientdrucker einem Server zugeordnet werden können, wenn sich ein Benutzer an einer Sitzung anmeldet.

Standardmäßig ist die Clientdruckerzuordnung zugelassen. Wenn diese Einstellung deaktiviert ist, wird der PDF-Drucker für die Sitzung nicht automatisch erstellt.

Verwandte Richtlinieneinstellungen: Clientdrucker automatisch erstellen

## Standarddrucker

Mit dieser Einstellung geben Sie an, wie der Standarddrucker in einer ICA-Sitzung ermittelt wird.

Standardmäßig wird der aktuelle Standarddrucker auf dem Clientgerät als Standarddrucker in der Sitzung verwendet.

Mit Standarddrucker des Benutzers nicht anpassen werden die aktuellen Einstellungen für den Standarddrucker in den Remotedesktopdiensten oder im Windows-Benutzerprofil verwendet. Wenn Sie diese Option auswählen, wird der Standarddrucker nicht im Profil gespeichert und ändert sich nicht entsprechend den anderen Sitzungs- oder Clienteigenschaften. Der Standarddrucker in einer Sitzung ist der erste Drucker, der in der Sitzung automatisch erstellt wird. Das ist:

- Der erste Drucker, der lokal auf dem Windows-Server unter Systemsteuerung > Geräte und Drucker hinzugefügt wurde.
- Der erste automatisch erstellte Drucker, wenn auf dem Server keine Drucker lokal hinzugefügt wurden.

Verwenden Sie diese Option, um Benutzern über Profileinstellungen den nächstgelegenen Drucker anzubieten (Proximitydrucken).

## Druckerzuweisungen

Diese Einstellung bietet eine Alternative zu den Einstellungen Standarddrucker und Sitzungsdrucker. Mit den einzelnen Einstellungen für Standarddrucker und Sitzungsdrucker können Sie das Verhalten einer Site, einer großen Gruppe oder einer Organisationseinheit konfigurieren. Mit der Einstellung Druckerzuweisungen weisen Sie eine große Gruppe Drucker mehreren Benutzern zu.

Mit dieser Einstellung geben Sie an, wie der Standarddrucker auf den aufgeführten Benutzergeräten in einer Sitzung ermittelt wird.

Standardmäßig wird der aktuelle Standarddrucker auf dem Clientgerät als Standarddrucker in der Sitzung verwendet.

Mit dieser Einstellung geben Sie außerdem die Netzwerkdrucker an, die in einer Sitzung für jedes Benutzergerät automatisch erstellt werden sollen. In der Standardeinstellung sind keine Drucker angegeben.

- Beim Einstellen des Standarddruckerwerts:  
Wenn Sie den aktuellen Standarddrucker für das Benutzergerät verwenden möchten, wählen Sie Nicht anpassen.

Mit Nicht anpassen werden die aktuellen Einstellungen für den Standarddrucker in den Remotedesktopdiensten oder im

Windows-Benutzerprofil verwendet. Wenn Sie diese Option auswählen, wird der Standarddrucker nicht im Profil gespeichert und ändert sich nicht entsprechend den anderen Sitzungs- oder Clienteigenschaften. Der Standarddrucker in einer Sitzung ist der erste Drucker, der in der Sitzung automatisch erstellt wird. Das ist:

- Der erste Drucker, der lokal auf dem Windows-Server unter Systemsteuerung > Geräte und Drucker hinzugefügt wurde.
- Der erste automatisch erstellte Drucker, wenn auf dem Server keine Drucker lokal hinzugefügt wurden.
- Beim Einstellen des Sitzungsdruckerwerts: Zum Hinzufügen eines Druckers geben Sie den UNC-Pfad des Druckers ein, der automatisch erstellt werden soll. Nach dem Hinzufügen des Druckers können Sie angepasste Einstellungen für die aktuelle Sitzung bei jeder Anmeldung anwenden.

### Ereignisprotokollpräferenz für die automatische Druckererstellung

Mit dieser Einstellung geben Sie an, welche Ereignisse bei der automatischen Druckererstellung protokolliert werden. Sie haben die Option, keine Fehler oder Warnungen, nur Fehler oder Fehler und Warnungen zu protokollieren.

Standardmäßig werden Fehler und Warnungen protokolliert.

Ein Beispiel für eine Warnung ist ein Ereignis, bei dem der native Druckertreiber für einen Drucker nicht installiert werden konnte und stattdessen der universelle Druckertreiber installiert wurde. Damit der universelle Druckertreiber in diesem Szenario verwendet werden kann, stellen Sie für Verwendung universeller Druckertreiber entweder Nur universelles Drucken verwenden oder Universelles Drucken nur verwenden, wenn angeforderter Treiber nicht verfügbar ist ein.

### Sitzungsdrucker

Mit dieser Einstellung geben Sie die Netzwerkdrucker an, die in einer ICA-Sitzung automatisch erstellt werden sollen.

In der Standardeinstellung sind keine Drucker angegeben.

Um Drucker hinzuzufügen, geben Sie den UNC-Pfad des Druckers ein, der automatisch erstellt werden soll. Nach dem Hinzufügen des Druckers können Sie angepasste Einstellungen für die aktuelle Sitzung bei jeder Anmeldung anwenden.

### Warten bis Drucker erstellt sind (Serverdesktop)

Mit dieser Einstellung legen Sie fest, ob es eine Verzögerung bei der Sitzungsverbindung geben soll, sodass Serverdesktopdrucker automatisch erstellt werden können.

Standardmäßig findet keine Verbindungsverzögerung statt.

# Einstellungen der Richtlinie "Clientdrucker"

May 10, 2016

Der Abschnitt "Clientdrucker" enthält Richtlinieneinstellungen für Clientdrucker, einschließlich solcher zur automatischen Erstellung von Clientdruckern, zum Speichern von Druckereigenschaften und zum Verbinden mit Druckservern.

## Clientdrucker automatisch erstellen

Mit dieser Einstellung geben Sie die Clientdrucker an, die automatisch erstellt werden. Diese Einstellung überschreibt die Standardeinstellungen für die automatische Clientdruckererstellung.

Standardmäßig werden alle Clientdrucker automatisch erstellt.

Diese Einstellung gilt nur, wenn die Einstellung Clientdruckerumleitung vorhanden und Zugelassen ist.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, wählen Sie eine der Optionen:

- Mit Alle Clientdrucker automatisch erstellen werden alle Drucker auf dem Clientgerät erstellt.
- Mit Nur Standarddrucker des Clients automatisch erstellen wird der Drucker automatisch erstellt, der auf dem Clientgerät als Standarddrucker angegeben wurde.
- Mit Nur lokale Clientdrucker (keine Netzwerkdrucker) automatisch erstellen werden nur die Drucker automatisch erstellt, die über einen LPT-, COM-, USB-, TCP/IP- oder anderen lokalen Port direkt mit dem Clientgerät verbunden sind.
- Mit Clientdrucker nicht automatisch erstellen wird die automatische Erstellung von Clientdruckern beim Anmelden der Benutzer deaktiviert. Die Remotedesktopdienste-Einstellungen für die automatische Erstellung von Clientdruckern überschreiben dann diese Einstellung in Richtlinien mit niedrigerer Priorität.

## Generischen universellen Drucker automatisch erstellen

Hinweis: Hotfixes für Probleme aufgrund dieser Richtlinieneinstellung sind in den Knowledge Center-Artikeln CTX141565 und CTX141566 verfügbar.

Mit dieser Einstellung aktivieren oder deaktivieren Sie die automatische Erstellung des generischen universellen Citrix Druckerobjekts für Sitzungen, in denen ein Benutzergerät verwendet wird, das mit der universellen Drucklösung kompatibel ist.

Standardmäßig wird das generische universelle Druckerobjekt nicht automatisch erstellt.

Die folgenden Richtlinieneinstellungen hängen zusammen:

- Verwendung universeller Druckertreiber
- Priorität universeller Treiber

## Clientdruckernamen

Mit dieser Einstellung legen Sie die Namenskonvention für automatisch erstellte Drucker fest.

Standardmäßig werden die Standardnamen der Drucker verwendet.

Wählen Sie Standarddruckernamen, um Druckernamen im Format "HPLaserJet 4 von Clientname in Sitzung 3" zu verwenden.

Wählen Sie Legacydruckernamen, um Clientdruckernamen im alten Stil zuzulassen und die Rückwärtskompatibilität für Benutzer oder Gruppen zu erhalten, die MetaFrame Presentation Server 3.0 oder früher verwenden. Ein Beispiel für einen Legacydruckernamen ist "Client/clientname#/HPLaserJet 4". Diese Option ist weniger sicher.

Hinweis: Diese Option wird nur für Abwärtskompatibilität mit älteren Versionen von XenApp und XenDesktop bereitgestellt.

## Direkte Verbindungen zu Druckservern

Mit dieser Einstellung aktivieren oder deaktivieren Sie direkte Verbindungen vom virtuellen Desktop oder Server, auf dem Anwendungen gehostet werden, zu einem Druckserver für Clientdrucker in einer zugänglichen Netzwerkfreigabe.

Standardmäßig sind direkte Verbindungen aktiviert.

Aktivieren Sie direkte Verbindungen, wenn der Netzwerkdruckserver für virtuelle Desktops bzw. für Server, auf denen Anwendungen gehostet werden, nicht über ein WAN zugänglich ist. Direkte Verbindungen gestatten schnelleres Drucken, wenn der Netzwerkdruckserver und der virtuelle Desktop oder Anwendungsserver sich im gleichen LAN befinden.

Deaktivieren Sie direkte Verbindungen, wenn das Netzwerk über ein WAN verläuft oder hohe Latenz oder beschränkte Bandbreite aufweist. Druckaufträge werden durch das Benutzergerät und den Netzwerkdruckserver geleitet. Daten werden komprimiert an das Benutzergerät gesendet, es wird somit weniger Bandbreite bei der Übertragung der Daten über das WAN gebraucht.

Wenn zwei Netzwerkdrucker den gleichen Namen haben, wird der Drucker benutzt, der im gleichen Netzwerk ist wie der Client.

## Druckertreiberzuordnung und -kompatibilität

Mit dieser Einstellung legen Sie Regeln für die Treiberersetzung bei automatisch erstellten Druckern fest.

In der Standardeinstellung sind keine Regeln angegeben.

Wenn Sie Regeln für die Treiberersetzung definieren, können Sie zulassen oder verhindern, dass Drucker mit dem angegebenen Treiber erstellt werden. Außerdem können Sie für erstellte Drucker nur universelle Druckertreiber zulassen. Bei der Treiberersetzung werden die Namen der Druckertreiber, die das Benutzergerät bereitstellt, überschrieben oder zugeordnet und ein äquivalenter Treiber auf dem Server wird ersetzt. So können Serveranwendungen auf Clientdrucker zuzugreifen, die denselben Treiber wie der Server, aber unterschiedliche Treibernamen verwenden.

Sie können eine Treiberzuordnung hinzufügen, eine bestehende Zuordnung bearbeiten, benutzerdefinierte Einstellungen für eine Zuordnung überschreiben oder die Reihenfolge der Treibereinträge in der Liste ändern. Um eine Zuordnung hinzuzufügen, geben Sie den Clientdruckertreibernamen an und wählen dann den Ersatztreiber auf dem Server aus.

## Speicherung von Druckereigenschaften

Mit dieser Einstellung geben Sie an, ob die Druckereigenschaften gespeichert werden und wo.

Standardmäßig ermittelt das System, ob Druckereigenschaften auf dem Clientgerät (falls verfügbar) gespeichert werden oder im Benutzerprofil.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, wählen Sie eine der Optionen:

- Wählen Sie Nur auf dem Clientgerät speichern, wenn Sie ein vorgeschriebenes oder servergespeichertes Profil verwenden, das nicht gespeichert wird. Wählen Sie diese Option nur aus, wenn auf allen Servern in der Farm XenApp 5 oder höher ausgeführt wird und die Benutzer die Citrix Online Plug-In-Versionen 9 bis 12.x oder Citrix Receiver 3.x verwenden.
- Wählen Sie Nur im Benutzerprofil speichern, wenn das System durch die Bandbreite (diese Option reduziert den Datenverkehr im Netzwerk) und die Anmeldegeschwindigkeit begrenzt ist oder die Benutzer Legacy-Plug-Ins verwenden. Bei dieser Option werden die Druckereigenschaften im Benutzerprofil auf dem Server gespeichert. Die Eigenschaften werden nicht mit dem Clientgerät ausgetauscht. Verwenden Sie diese Option für MetaFrame Presentation Server 3.0 oder früher und MetaFrame Presentation Server Client 8.x oder früher. Dies gilt nur, wenn ein servergespeichertes Profil für Remotedesktopdienste (RDS) verwendet wird.



- Mit Nur im Profil speichern, wenn sie nicht auf dem Client gespeichert sind kann das System festlegen, wo die Druckereigenschaften gespeichert werden. Die Druckereigenschaften werden auf dem Clientgerät gespeichert, sofern es verfügbar ist, ansonsten im Benutzerprofil. Diese Option bietet zwar die größte Flexibilität, kann jedoch die Anmeldezeit verlangsamen und zusätzliche Bandbreite für die Systemprüfung verbrauchen.
- Druckereigenschaften nicht speichern verhindert das Speichern von Druckereigenschaften.

### Gespeicherte und wiederhergestellte Clientdrucker

Mit dieser Einstellung aktivieren oder deaktivieren Sie das Speichern und Neuerstellen von Clientdruckern. Standardmäßig werden Clientdrucker automatisch gespeichert und automatisch wiederhergestellt.

Gespeicherte Drucker sind vom Benutzer erstellte Drucker, die beim Start der nächsten Sitzung wiederhergestellt werden. Wenn XenApp einen gespeicherten Drucker wiederherstellt, werden alle RichtlinienEinstellungen außer Clientdrucker automatisch erstellen berücksichtigt.

Gespeicherte Drucker sind Drucker die von einem Administrator vollständig angepasst wurden und deren gespeicherter Zustand permanent mit einem Clientport verbunden ist.

# Einstellungen der Richtlinie "Treiber"

May 10, 2016

Der Abschnitt "Treiber" enthält Richtlinieneinstellungen für Druckertreiber.

## Automatische Installation von mitgelieferten Druckertreibern

Mit dieser Einstellung aktivieren oder deaktivieren Sie die automatische Installation von Druckertreibern vom standardmäßigen Windows-Treibersatz oder von Treiberpaketen, die auf dem Host mit pnputil.exe /a bereitgestellt wurden.

Standardmäßig werden diese Treiber bei Bedarf installiert.

## Priorität universeller Treiber

Mit dieser Einstellung geben Sie an, in welcher Reihenfolge die universellen Druckertreiber verwendet werden, angefangen mit dem ersten Eintrag in der Liste.

Standardmäßig ist die Prioritätsreihenfolge wie folgt:

- EMF
- XPS
- PCL5c
- PCL4
- PS

Sie können Treiber hinzufügen, bearbeiten oder entfernen und die Reihenfolge der Treiber in der Liste ändern.

## Verwendung universeller Druckertreiber

Mit dieser Einstellung geben Sie an, wann universelles Drucken verwendet wird.

Standardmäßig wird universelles Drucken nur verwendet, wenn der angeforderte Treiber nicht verfügbar ist.

Universelles Drucken verwendet allgemeine Druckertreiber statt modellspezifischer Standardtreiber; dies verringert potentiell den Aufwand für die Treiberverwaltung auf Hostcomputern. Die Verfügbarkeit universeller Druckertreiber hängt von den Funktionen des Benutzergeräts, des Hosts und der Druckersoftware ab. In bestimmten Konfigurationen steht universelles Drucken möglicherweise nicht zur Verfügung.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, wählen Sie eine der Optionen:

- Mit Nur druckerspezifische Treiber verwenden verwendet der Clientdrucker nur die modellspezifischen Standardtreiber, die bei der Anmeldung automatisch erstellt wurden. Wenn der erforderliche Treiber nicht verfügbar ist, kann der Clientdrucker nicht automatisch erstellt werden.
- Nur universelles Drucken verwenden gibt an, dass keine modellspezifischen Standardtreiber verwendet werden. Nur universelle Druckertreiber werden zum Erstellen von Druckern verwendet.
- Mit Universelles Drucken nur verwenden, wenn angeforderter Treiber nicht verfügbar ist werden modellspezifische Standardtreiber für die Druckererstellung verwendet, wenn sie verfügbar sind. Wenn der Treiber auf dem Server nicht zur Verfügung steht, wird der Clientdrucker automatisch mit dem entsprechenden universellen Treiber erstellt.
- Mit Druckerspezifische Treiber nur verwenden, wenn universelles Drucken nicht verfügbar ist werden universelle Druckertreiber verwendet, wenn sie verfügbar sind. Wenn der Treiber auf dem Server nicht zur Verfügung steht, wird der Clientdrucker automatisch mit dem entsprechenden modellspezifischen Druckertreiber erstellt.

# Einstellungen der Richtlinie "Universeller Druckserver"

May 10, 2016

Der Abschnitt "Universeller Druckserver" enthält Richtlinieneinstellungen für die Behandlung des universellen Druckservers.

## Universellen Druckserver aktivieren

Diese Einstellung aktiviert oder deaktiviert das Feature "Universeller Druckserver" auf dem virtuellen Desktop oder dem Server, auf dem Anwendungen gehostet werden. Wenden Sie die Richtlinie auf Organisationseinheiten an, die den virtuellen Desktop oder Server enthalten, auf dem Anwendungen gehostet werden.

Standardmäßig ist das Feature deaktiviert.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, wählen Sie eine der folgenden Optionen:

- **Aktiviert mit Fallback auf systemeigenen Windows-Remotedruck:** Netzwerkdruckerverbindungen werden nach Möglichkeit vom universellen Druckserver bedient. Wenn dieser nicht verfügbar ist, wird der Windows-Druckanbieter verwendet. Der Windows-Druckanbieter handhabt weiterhin alle Drucker, die vorher mit dem Windows-Druckanbieter erstellt wurden.
- **Aktiviert ohne Fallback auf systemeigenen Windows-Remotedruck:** Netzwerkdruckerverbindungen werden ausschließlich vom universellen Druckserver bedient. Wenn dieser nicht verfügbar ist, schlägt die Netzwerkdrucker Verbindung fehl. Mit dieser Einstellung wird der Netzwerkdruck über den Windows-Druckanbieter effektiv deaktiviert. Drucker, die vorher mit dem Windows-Druckanbieter erstellt wurden, werden nicht erstellt, solange eine Richtlinie mit dieser Einstellung aktiv ist.
- **Deaktiviert:** Das Feature "Universeller Druckserver" ist deaktiviert. Beim Herstellen einer Verbindung mit einem Netzwerkdrucker, der einen UNC-Namen hat, wird keine Verbindung mit dem universellen Druckserver versucht. Verbindungen mit Remotedruckern verwenden weiterhin den Windows-Remotedruck.

## Port für Druckdatenstrom des universellen Druckservers (CGP)

Diese Einstellung gibt die Nummer des TCP-Ports an, die vom Druckdatenstrom-Listener (CGP) des universellen Druckservers verwendet wird. Wenden Sie diese Richtlinie nur für Organisationseinheiten an, die den Druckserver enthalten.

Die Standardeinstellung der Portnummer ist "7229".

Gültige Portnummern müssen im Bereich von 1 bis 65535 liegen.

## Universeller Druckserver - Eingabebandbreitenlimit für Druckdatenstrom (KBit/s)

Diese Einstellung gibt das obere Limit (in Kilobit pro Sekunde) für die Übertragungsrate der Druckdaten an, die von jedem Druckauftrag mit CGP an den universellen Druckserver übergeben werden. Wenden Sie die Richtlinie auf Organisationseinheiten an, die den virtuellen Desktop oder Server enthalten, auf dem Anwendungen gehostet werden.

In der Standardeinstellung ist der Wert 0, was angibt, dass es kein oberes Limit gibt.

## Port für universellen Druckserverwebdienst (HTTP/SOAP)

Diese Einstellung gibt die Nummer des TCP-Ports an, die vom Webdienstlistener des universellen Druckservers für eingehende HTTP/SOAP-Anforderungen verwendet wird. Der universelle Druckserver ist eine optionale Komponente, mit der die Verwendung universeller Druckertreiber von Citrix für den Netzwerkdruck ermöglicht wird. Wird der universelle Druckserver verwendet, werden die Druckbefehle von den XenApp- und XenDesktop-Hosts mit SOAP über HTTP an den

universellen Druckserver gesendet. Durch diese Einstellung ändert sich die Nummer des Standard-TCP-Ports, der vom Webdienstlistener des universellen Druckservers für eingehende HTTP/SOAP-Anforderungen überwacht wird.

Sie müssen den gleichen HTTP- Port für Host und Druckserver konfigurieren. Wenn Sie nicht den gleichen Port konfigurieren, stellt die Hostsoftware keine Verbindung mit dem universellen Druckserver her. Durch diese Einstellung ändert sich der VDA unter XenApp und XenDesktop. Außerdem müssen Sie den Standardport auf dem Computer mit dem universellen Druckserver ändern.

Die Standardeinstellung der Portnummer ist "8080".

Gültige Portnummern müssen im Bereich von 0 bis 65535 liegen.

# Einstellungen der Richtlinie "Universelles Drucken"

May 10, 2016

Der Abschnitt "Universelles Drucken" enthält Richtlinieneinstellungen für die Verwaltung des universellen Drucks.

## Universelles Drucken - EMF-Verarbeitungsmodus

Mit dieser Einstellung steuern Sie die Verarbeitungsmethode für die EMF-Spooldatei auf dem Windows-Benutzergerät.

Standardmäßig werden EMF-Datensätze direkt zum Drucker gespoolt.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, wählen Sie eine der Optionen:

- EMF-Datensätze für Drucker neu verarbeiten erzwingt die Neuverarbeitung der EMF-Spooldatei und sendet sie durch das GDI-Teilsystem auf dem Benutzergerät. Sie können diese Einstellung für Treiber verwenden, für die eine EMF-Neuverarbeitung erforderlich ist, die jedoch nicht unbedingt in der Sitzung automatisch ausgewählt werden.
- Wenn Direkt zum Drucker spoolen mit dem universellen Citrix Druckertreiber verwendet wird, werden die EMF-Datensätze garantiert gespoolt und an das Benutzergerät für die Verarbeitung übergeben. Diese EMF-Spooldateien werden normalerweise direkt in die Spoolwarteschlange des Clients gesetzt. Für Drucker und Treiber, die mit dem EMF-Format kompatibel sind, ist dies die schnellste Druckmethode.

## Universelles Drucken - Bildkomprimierungslimit

Mit dieser Einstellung geben Sie die maximale Qualität und minimale Komprimierung für Bilder an, die mit dem universellen Citrix Druckertreiber gedruckt werden.

Das Limit für Bildkomprimierung ist standardmäßig auf Beste Qualität (verlustfreie Komprimierung) gesetzt.

Wenn Keine Komprimierung ausgewählt ist, wird die Komprimierung nur für den EMF-Druck deaktiviert.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, wählen Sie eine der Optionen:

- Keine Komprimierung
- Beste Qualität (verlustfreie Komprimierung)
- Hohe Qualität
- Standardqualität
- Geringere Qualität (maximale Komprimierung)

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, die auch die Einstellung "Universelles Drucken - Optimierungstandards" enthält, achten Sie auf Folgendes:

- Wenn die Komprimierungsstufe in der Einstellung Universelles Drucken - Komprimierungslimit niedriger ist als die in der Einstellung Universelles Drucken - Optimierungstandards, werden Bilder basierend auf der Einstellung Universelles Drucken - Bildkomprimierungslimit komprimiert.
- Wenn die Komprimierung deaktiviert ist, haben die Optionen Gewünschte Bildqualität und Heavyweight-Komprimierung aktivieren in der Einstellung Universelles Drucken - Optimierungstandards keine Auswirkung in der Richtlinie.

## Universelles Drucken - Optimierungstandards

Mit dieser Einstellung geben Sie die Standardwerte für die Druckoptimierung an, wenn der universelle Druckertreiber für eine Sitzung erstellt wurde.

- Mit Gewünschte Bildqualität geben Sie das standardmäßige Bildkomprimierungslimit an, das auf universelles Drucken angewendet wird. In der Standardeinstellung ist Standardqualität aktiviert, d. h. Benutzer können Bilder nur mit der

Standardqualitäts- oder geringeren Qualitätskomprimierung drucken.

- Mit Heavyweight-Komprimierung aktivieren oder deaktivieren Sie das Verringern der Bandbreite unter den Komprimierungsgrad, der von Gewünschte Bildqualität festgelegt ist; Bildqualität geht nicht verloren. Standardmäßig ist die Heavyweight-Komprimierung deaktiviert.
- Mit den Einstellungen Zwischenspeichern von Bildern und Schriftarten legen Sie fest, ob Bilder und Schriftarten, die mehrmals im Druckdatenstrom vorhanden sind, zwischengespeichert werden. Sie stellen damit sicher, dass jedes eindeutige Bild oder jede Schriftart nur einmal zum Drucker gesendet wird. Standardmäßig werden eingebettete Bilder und Schriftarten zwischengespeichert. Hinweis: Diese Einstellungen gelten nur, wenn das Benutzergerät dieses Verhalten unterstützt.
- Mit Nicht-Administratoren können diese Einstellungen ändern legen Sie fest, ob Benutzer die Standardeinstellungen für die Druckoptimierung in einer Sitzung ändern können. Standardmäßig können Benutzer die Standardeinstellungen für die Druckoptimierung nicht ändern.

Hinweis: Alle diese Optionen werden für den EMF-Druck unterstützt. Für XPS-Druck wird nur die Option Gewünschte Bildqualität unterstützt.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, die auch die Einstellung "Universelles Drucken - Optimierungsstandards" enthält, achten Sie auf Folgendes:

- Wenn die Komprimierungsstufe in der Einstellung Universelles Drucken - Komprimierungslimit niedriger ist als die in der Einstellung Universelles Drucken - Optimierungsstandards werden Bilder basierend auf der Einstellung Universelles Drucken - Bildkomprimierungslimit komprimiert.
- Wenn die Komprimierung deaktiviert ist, haben die Optionen Gewünschte Bildqualität und Heavyweight-Komprimierung aktivieren in der Einstellung Universelles Drucken - Optimierungsstandards keine Auswirkung in der Richtlinie.

### Universelles Drucken - Vorschau-einstellung

Mit dieser Einstellung geben Sie an, ob die Druckvorschau für automatisch erstellte oder universelle Drucker verwendet werden soll.

Standardmäßig wird die Druckvorschau für automatisch erstellte oder generische universelle Drucker nicht verwendet.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, wählen Sie eine der Optionen:

- Druckvorschau für automatisch erstellte oder generische universelle Drucker nicht verwenden
- Druckvorschau nur für automatisch erstellte Drucker verwenden
- Druckvorschau nur für generische universelle Drucker verwenden
- Druckvorschau für automatisch erstellte und generische universelle Drucker verwenden

### Universelles Drucken - Druckqualitätslimit

Diese Einstellung legt den Höchstwert für Punkte pro Zoll (dpi) zum Erstellen von Ausdrucken in einer Sitzung fest.

Standardmäßig ist Kein Limit aktiviert, d. h. Benutzer können die höchste Druckqualität auswählen, die vom Drucker zugelassen wird, mit dem sie eine Verbindung herstellen.

Wenn diese Einstellung konfiguriert ist, wird die maximale Druckqualität, die Benutzern zur Verfügung steht, hinsichtlich Ausgabeauflösung beschränkt. Sowohl die Druckqualität und die Druckqualitätsmerkmale des Druckers, mit dem sich die Benutzer verbinden, werden auf die konfigurierte Einstellung beschränkt. Beispiel: Wenn Mittlere Auflösung (600 dpi) konfiguriert ist, können Benutzer eine Ausgabe nur mit einer maximalen Qualität von 600 drucken, und die Einstellung "Druckqualität" auf der Registerkarte "Erweitert" im Dialogfeld "Universeller Drucker" enthält nur Auflösungseinstellungen bis zu "Mittlere Qualität (600 dpi).

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, wählen Sie eine der Optionen:

- Entwurf (150 dpi)
- Niedrige Auflösung (300 dpi)
- Mittlere Auflösung (600 dpi)
- Hohe Auflösung (1200 dpi)
- Kein Limit

# Einstellungen der Richtlinie "Sicherheit"

May 10, 2016

Der Abschnitt "Sicherheit" enthält die Richtlinieneinstellung zum Konfigurieren der Sitzungsverschlüsselung und der Anmeldedatenverschlüsselung.

## SecureICA-Mindestverschlüsselungsgrad

Mit dieser Einstellung geben Sie das Minimum für den Verschlüsselungsgrad der Sitzungsdaten an, die zwischen dem Server und einem Clientgerät ausgetauscht werden.

### Wichtig:

Bei Virtual Delivery Agent 7.x kann mit dieser Richtlinieneinstellung nur die Anmeldedatenverschlüsselung mit RC5 128-Bit-Verschlüsselung aktiviert werden. Die anderen Einstellungen werden nur für Abwärtskompatibilität mit älteren Versionen von XenApp und XenDesktop bereitgestellt.

Bei VDA 7.x wird die Sitzungsdatenverschlüsselung mit den Grundeinstellungen der Bereitstellungsgruppe des VDAs festgelegt. Wenn für die Bereitstellungsgruppe die Option "Secure ICA aktivieren" ausgewählt ist, werden Sitzungsdaten mit der RC5-Verschlüsselung (128 Bit) verschlüsselt. Wenn die Option "Secure ICA aktivieren" für die Bereitstellungsgruppe nicht ausgewählt ist, werden Sitzungsdaten mit der Basic-Verschlüsselung verschlüsselt.

Wenn Sie diese Einstellung einer Richtlinie hinzufügen, wählen Sie eine der Optionen:

- Basic verschlüsselt die Clientverbindung mit einem nicht RC5-konformen Algorithmus. Mit diesem Verschlüsselungsverfahren kann der Datenstrom zwar vor direktem Lesen geschützt werden, ein Entschlüsseln ist aber möglich. Standardmäßig verwendet der Server für den Client-Server-Netzwerkverkehr den Verschlüsselungsgrad "Basic".
- RC5 (128 Bit) nur Anmeldung verschlüsselt die Anmeldedaten mit der RC5-128-Bit-Verschlüsselung und die Clientverbindung mit dem Verschlüsselungsgrad "Basic".
- RC5 (40 Bit) verschlüsselt die Verbindung mit der RC5-40-Bit-Verschlüsselung.
- RC5 (56 Bit) verschlüsselt die Verbindung mit der RC5-56-Bit-Verschlüsselung.
- RC5 (128 Bit) verschlüsselt die Verbindung mit der RC5-128-Bit-Verschlüsselung.

Die Einstellungen, die Sie für die Verschlüsselung zwischen Client und Server festlegen, können mit Verschlüsselungseinstellungen des Windows-Betriebssystems interagieren. Wenn ein höherer Verschlüsselungsgrad auf dem Server oder Benutzergerät eingestellt ist, können Einstellungen überschrieben werden, die Sie für veröffentlichte Ressourcen angegeben haben.

Sie können den Verschlüsselungsgrad erhöhen, um die Kommunikation und Datenintegrität für bestimmte Benutzer stärker zu sichern. Wenn für eine Richtlinie ein höherer Verschlüsselungsgrad erforderlich ist, wird Receivern mit einem niedrigeren Verschlüsselungsgrad die Verbindung verweigert.

SecureICA führt keine Authentifizierung durch und prüft auch nicht die Datenintegrität. Verwenden Sie SecureICA mit SSL/TLS-Verschlüsselung, um eine vollständige Verschlüsselung für die Site bereitzustellen.

SecureICA verwendet nicht FIPS-konforme Algorithmen. Wenn dies ein Problem ist, konfigurieren Sie den Server und die Receiver, um zu verhindern, dass SecureICA verwendet wird.



# Einstellungen der Richtlinie "Serverlimits"

May 10, 2016

Der Abschnitt "Serverlimits" enthält die Richtlinieneinstellung zum Steuern von Sitzungen im Leerlauf.

## Serverleerlauf-Zeitintervall

Mit dieser Einstellung geben Sie in Millisekunden an, wie lange eine ununterbrochene Benutzersitzung erhalten bleibt, wenn keine Benutzereingaben stattfinden.

Standardmäßig werden Leerlaufsitzungen nicht getrennt (Serverleerlaufzeitintervall = 0)

### Hinweis

Bei Verwendung dieser Richtlinie wird Benutzern u. U. ein Dialogfeld mit der Meldung "Leerlauf-Timer abgelaufen" angezeigt, wenn die Sitzung die angegebene Zeit lang im Leerlauf war. Dies ist ein Microsoft Dialogfeld, das nicht von Citrix Richtlinieneinstellungen gesteuert wird. Weitere Informationen finden Sie unter <http://support.citrix.com/article/CTX118618>.

# Einstellungen der Richtlinie "Sitzungslimits"

May 10, 2016

Der Abschnitt "Sitzungslimits" enthält Richtlinieneinstellungen, mit denen Sie die Dauer der Sitzungen festlegen können, bevor diese getrennt werden.

## Timer für getrennte Sitzung

Mit dieser Einstellung aktivieren oder deaktivieren Sie einen Timer, der angibt, wie lange ein getrennter, gesperrter Desktop gesperrt bleibt, bevor die Sitzung abgemeldet wird.

Standardmäßig werden getrennte Sitzungen nicht abgemeldet.

## Getrennte Sitzungen - Timerintervall

Mit dieser Einstellung legen Sie fest, wie viele Minuten ein getrennter, gesperrter Desktop gesperrt bleibt, bevor die Sitzung abgemeldet wird.

Standardmäßig sind es 1440 Minuten (24 Stunden).

## Sitzungsverbindungstimer

Mit dieser Einstellung aktivieren oder deaktivieren Sie einen Timer, mit dem die maximale Dauer einer ununterbrochenen Sitzung zwischen einem Benutzergerät und einem Desktop festgelegt wird.

Standardmäßig ist dieser Timer aktiviert.

## Sitzungsverbindung - Timerintervall

Diese Einstellung legt die Höchstdauer einer ununterbrochenen Verbindung zwischen einem Benutzergerät und einem Desktop in Minuten fest.

Standardmäßig ist die maximale Dauer 1440 Minuten (24 Stunden).

## Sitzungsleerlauftimer

Mit dieser Einstellung aktivieren oder deaktivieren Sie einen Timer, der angibt, wie lange eine ununterbrochene Benutzergeräteverbindung mit einem Desktop erhalten bleibt, wenn keine Benutzereingaben stattfinden.

Standardmäßig ist dieser Timer deaktiviert.

## Sitzungsleerlauf - Timerintervall

Diese Einstellung legt fest, wie viele Minuten eine ununterbrochene Verbindung zwischen einem Benutzergerät und einem Desktop aufrechterhalten wird, wenn keine Eingabe vom Benutzer erfolgt.

Standardmäßig bleiben Leerlaufsitzen 1440 Minuten (24 Stunden) erhalten.

# Einstellungen der Richtlinie "Sitzungszuverlässigkeit"

May 10, 2016

Der Abschnitt "Sitzungszuverlässigkeit" enthält Richtlinieneinstellungen zum Verwalten von Verbindungen, für die die Sitzungszuverlässigkeit verwendet wird.

## Sitzungszuverlässigkeit - Verbindungen

Mit dieser Einstellung legen Sie fest, ob Sitzungen bei dem Verlust der Netzwerkkonnektivität offen bleiben sollen.

Standardmäßig ist die Sitzungszuverlässigkeit zugelassen.

Durch die Sitzungszuverlässigkeit bleiben Sitzungen aktiv und auf dem Bildschirm des Benutzers, wenn die Netzwerkverbindung unterbrochen wird. Die Benutzer sehen so lange weiterhin die Anwendung, die sie verwenden, bis die Netzwerkkonnektivität wiederhergestellt ist.

Mit der Sitzungszuverlässigkeit bleibt die Sitzung auf dem Server aktiv. Wenn die Netzwerkverbindung unterbrochen ist, friert der Bildschirm auf dem Client ein und der Mauszeiger wird als Sanduhr angezeigt, bis die Verbindung wiederhergestellt ist. Der Benutzer kann während der Unterbrechung weiterhin auf die Anzeige zugreifen und mit der Anwendung weiterarbeiten, wenn die Netzwerkverbindung wiederhergestellt ist. Die Sitzungszuverlässigkeit verbindet Benutzer ohne Neuauthentifizierung wieder. Wenn Sie möchten, dass Benutzer eine Verbindung zu unterbrochenen Sitzungen nur mit einer Neuauthentifizierung wiederherstellen können, stellen Sie Authentifizierung bei automatischer Wiederverbindung von Clients entsprechend ein. Benutzer werden dann aufgefordert, sich neu zu authentifizieren, wenn sie sich mit der unterbrochenen Sitzung wiederverbinden.

Wenn Sie sowohl Sitzungszuverlässigkeit als auch die Funktion zur automatischen Wiederverbindung verwenden, werden beide Funktionen nacheinander ausgeführt. Die Sitzungszuverlässigkeit beendet oder trennt die Benutzersitzung, nachdem der mit der Option Sitzungszuverlässigkeit - Timeout festgelegte Zeitraum abgelaufen ist. Anschließend werden die Richtlinieneinstellungen für die automatische Wiederverbindung von Clients wirksam und es wird versucht, eine Verbindung mit der unterbrochenen Sitzung wiederherzustellen.

## Sitzungszuverlässigkeit - Portnummer

Mit dieser Einstellung geben Sie die TCP-Portnummer für eingehende Sitzungszuverlässigkeitsverbindungen an.

Die Standardeinstellung der Portnummer ist "2598".

## Sitzungszuverlässigkeit - Timeout

Mit dieser Einstellung legen Sie fest, wie viele Sekunden der Sitzungszuverlässigkeitsproxy auf die Wiederverbindung der Sitzung wartet, bevor die Sitzung getrennt wird.

Die Standardeinstellung ist 180 Sekunden (3 Minuten).

Sie können zwar eine Sitzung länger offen lassen, beachten Sie jedoch, dass Benutzer nicht zu einer Neuauthentifizierung aufgefordert werden, um den Bedienungskomfort zu erhöhen. Je länger eine Sitzung offen gelassen wird, desto höher ist das Risiko, dass der Benutzer abgelenkt wird und das Benutzergerät verlässt. Benutzer ohne Berechtigung hätten in dem Fall möglicherweise Zugriff auf die Sitzung.

# Einstellungen der Richtlinie "Zeitzonesteuerung"

May 10, 2016

Der Abschnitt "Zeitzonesteuerung" enthält Richtlinieneinstellungen für die Zeitzone in Sitzungen.

## Lokale Zeitzone für Legacyclients schätzen

Mit dieser Einstellung aktivieren oder deaktivieren Sie das Schätzen der lokalen Zeitzone auf Clientgeräten, die falsche Zeitzoneinformationen an den Server senden.

Standardmäßig schätzt der Server die lokale Zeitzone, wenn erforderlich.

Diese Einstellung ist für die Verwendung mit älteren Receiver-Versionen oder ICA- Clients vorgesehen, die keine detaillierten Zeitzoneinformationen an den Server senden. Bei Verwendung mit Receiver-Versionen, die detaillierte Zeitzoneinformationen an den Server senden, beispielsweise die unterstützten Versionen von Receiver für Windows, hat diese Einstellung keine Auswirkung.

## Lokale Zeit des Clients verwenden

Mit dieser Einstellung legen Sie die Zeitzoneneinstellung der Benutzersitzung fest. Dies kann entweder die Zeitzone der Sitzung des Benutzers oder die des Benutzergeräts sein.

Standardmäßig wird die Zeitzone der Sitzung des Benutzers verwendet.

Damit diese Einstellung wirksam wird, aktivieren Sie die Einstellung Zeitzonenumleitung zulassen im Gruppenrichtlinien-Editor (Benutzerkonfiguration > Administrative Vorlagen > Windows-Komponenten > Remotedesktopdienste > Remotedesktop-Sitzungshost > Remotesitzungsumgebung > Geräte- und Ressourcenumleitung).

# Einstellungen der Richtlinie "TWAIN-Geräte"

May 10, 2016

Der Abschnitt "TWAIN-Geräte" enthält Richtlinieneinstellungen für die Zuordnung von TWAIN-Geräten, wie Digitalkameras oder Scanner, und für das Optimieren der Bildübertragung vom Server zum Client.

## Hinweis

TWAIN 2.0 wird derzeit nicht unterstützt

### TWAIN-Geräteumleitung für Client

Mit dieser Einstellung legen Sie fest, ob Benutzer auf TWAIN-Geräte auf dem Benutzergerät aus Bildverarbeitungsanwendungen auf Servern zugreifen können. Standardmäßig ist die TWAIN-Geräteumleitung zugelassen.

Die folgenden Richtlinieneinstellungen hängen zusammen:

- TWAIN-Komprimierungsgrad
- Bandbreitenlimit für TWAIN-Geräteumleitung
- Bandbreitenlimit für TWAIN-Geräteumleitung (Prozent)

### TWAIN-Komprimierungsgrad

Mit dieser Einstellung geben Sie den Komprimierungsgrad für Bildübertragungen vom Client zum Server an. Verwenden Sie Gering für die beste Bildqualität, Mittel für eine gute Bilderqualität und Hoch für eine geringe Bildqualität. Standardmäßig wird die mittlere Komprimierung angewendet.

# Einstellungen der Richtlinie "USB-Geräte"

May 10, 2016

Der Abschnitt "USB-Geräte" enthält Richtlinieneinstellungen für die Verwaltung der Dateiumleitung bei USB-Geräten.

## Regeln für die Client-USB-Geräteoptimierung

In XenApp und XenDesktop 7.6 FP3 können die Regeln für die Client-USB-Geräteoptimierung auf Geräte angewendet werden, um die Optimierung zu deaktivieren oder den Optimierungsmodus zu ändern.

Wenn ein Benutzer ein USB-Gerät anschließt, prüft der Host, ob das Gerät gemäß den USB-Richtlinieneinstellungen zulässig ist. Ist das Gerät zulässig, prüft der Host die **Regeln für die Client-USB-Geräteoptimierung** für das Gerät. Wenn keine Regel angegeben ist, wird das Gerät im interaktiven Modus behandelt (02). Der Erfassungsmodus (04) ist der empfohlene Modus für Signaturgeräte. Eine Beschreibung der Modi finden Sie unten.

### Gut zu wissen

- Für die Verwendung von Wacom Signatur-Tablets sollten Sie den Bildschirmschoner deaktivieren. Anweisungen hierzu finden Sie am Ende dieses Abschnitts.
- Unterstützung für die Optimierung von Wacom-Signatur-Tablets der STU-Reihe ist in der Installation von Richtlinien bei XenApp und XenDesktop 7.6 FP3 vorkonfiguriert.
- Signaturgeräte funktionieren lückenlos in XenApp und XenDesktop und erfordern zur Verwendung als Signaturgerät keine Treiber. Wacom bietet zusätzliche Software an, die zur weiteren Anpassung des Geräts installiert werden kann. Informationen finden Sie unter <http://www.wacom.com/>.
- Grafiktablets: Bestimmte Grafik-Eingabegeräte werden als HID-Gerät an einem PCI/ACPI-Bus präsentiert und nicht unterstützt. Diese Geräte müssen an einen USB-Hostcontroller auf dem Client angeschlossen werden, damit sie innerhalb der XenDesktop-Sitzung umgeleitet werden.

Richtlinienregeln haben das Format von durch Leerzeichen getrennten Tag=Wert-Ausdrücken. Die folgenden Tags werden unterstützt:

| Tagname | Beschreibung                                                                                                                                                                                                                                                                                                                                                    |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Modus   | <p>Der Optimierungsmodus wird für Eingabegeräte der Klasse 3 (class=<b>03</b>) unterstützt. Unterstützte Modi:</p> <p>Keine Optimierung: Wert <b>01</b>.</p> <p>Interaktiver Modus: Wert <b>02</b>. Empfohlen für Geräte wie Stift-Tablets und 3D Pro-Mäuse.</p> <p>Erfassungsmodus: Wert <b>04</b>. Vorzugsmodus für Signatur-Tablets und ähnliche Geräte.</p> |
| VID     | Vendor-ID vom Gerätedeskriptor                                                                                                                                                                                                                                                                                                                                  |
| PID     | Produkt-ID vom Gerätedeskriptor                                                                                                                                                                                                                                                                                                                                 |

|          |                                                                    |
|----------|--------------------------------------------------------------------|
| REL      | Release-ID vom Gerätedeskriptor                                    |
| Class    | Klasse vom Gerätedeskriptor oder einem Schnittstellendeskriptor    |
| SubClass | Unterklasse vom Gerätedeskriptor oder ein Schnittstellendeskriptor |
| Prot     | Protokoll vom Gerätedeskriptor oder ein Schnittstellendeskriptor   |

## Beispiele

Mode=00000004 VID=1230 PID=1230 class=03 (Eingabegerät im Erfassungsmodus)

Mode=00000002 VID=1230 PID=1230 class=03 (Eingabegerät im interaktiven Modus, Standardeinstellung)

Mode=00000001 VID=1230 PID=1230 class=03 (Eingabegerät ohne Optimierung)

Mode=00000100 VID=1230 PID=1230 (Setuptoptimierung deaktiviert, Standardeinstellung)

Mode=00000200 VID=1230 PID=1230 (Setuptoptimierung aktiviert)

## Deaktivieren des Optimierungsmodus mittels einer Einstellung in der Registrierung

Der Optimierungsmodus kann systemweit über folgendes Registrierungskennzeichen deaktiviert werden:

HKLM\System\CurrentControlSet\Services\Icausb\Parameters

**DisableInputOptimization** DWORD – Wert auf **1** festlegen

Es ist ein Systemneustart erforderlich, damit diese Änderung wirksam wird.

## Deaktivieren des Bildschirmschoners für Wacom-Signatur-Tablets

Für die Verwendung von Wacom Signatur-Tablets sollten Sie den Bildschirmschoner wie folgt deaktivieren:

1. Installieren Sie den Wacom-STU-Treiber nach der Umleitung des Geräts.
2. Installieren Sie das Wacom-STU-Display-MSI, um Zugriff auf die Systemsteuerung des Signatur-Tablets zu erhalten.
3. Navigieren Sie zu **Control Panel > Wacom STU Display > STU430** oder **STU530** und wählen Sie die Registerkarte für das jeweilige Modell aus.
4. Klicken Sie auf **Change** und wählen Sie **Yes**, wenn das Fenster für die UAC-Sicherheit angezeigt wird.
5. Wählen Sie **Disable slideshow** und klicken Sie auf **Apply**.

Wenn die Einstellung für ein Signatur-Tabletmodell festgelegt ist, wird sie auf alle Modelle angewendet.

## Client-USB-Geräteumleitung

Mit dieser Einstellung legen Sie fest, ob die Umleitung von USB-Geräten zu und von Benutzergeräten zulässig ist.

Standardmäßig werden USB-Geräte nicht umgeleitet.

## Regeln für die Client-USB-Geräteumleitung

Mit dieser Einstellung legen Sie die Umleitungsregeln für USB-Geräte fest.

In der Standardeinstellung sind keine Regeln angegeben.

Schließt ein Benutzer ein USB-Gerät an, prüft das Hostgerät jede Richtlinienregel, bis eine Übereinstimmung vorliegt. Die erste Übereinstimmung für ein beliebiges Gerät ist entscheidend. Ist es eine Zulassen-Regel, kann das Gerät remote auf dem virtuellen Desktop verwendet werden. Ist es eine Verweigern-Regel, kann das Gerät nur auf dem lokalen Desktop verwendet werden. Wenn keine Übereinstimmung gefunden wird, werden die Standardregeln verwendet.

Richtlinienregeln haben das Format {Allow:|Deny:} plus Tag=Wert, durch Leerzeichen getrennt. Die folgenden Tags werden unterstützt:

| Tagname  | Beschreibung                                                       |
|----------|--------------------------------------------------------------------|
| VID      | Vendor-ID vom Gerätedeskriptor                                     |
| PID      | Produkt-ID vom Gerätedeskriptor                                    |
| REL      | Release-ID vom Gerätedeskriptor                                    |
| Class    | Klasse vom Gerätedeskriptor oder einem Schnittstellendeskriptor    |
| SubClass | Unterklasse vom Gerätedeskriptor oder ein Schnittstellendeskriptor |
| Prot     | Protokoll vom Gerätedeskriptor oder ein Schnittstellendeskriptor   |

Wenn Sie neue Richtlinienregeln erstellen, beachten Sie Folgendes:

- Bei Regeln wird die Groß- und Kleinschreibung nicht berücksichtigt.
- Regeln können optional von einem Kommentar gefolgt werden, der mit # eingeleitet wird.
- Leere Zeilen und Kommentare werden ignoriert.
- Tags müssen den Übereinstimmungsoperator = verwenden, z. B. VID=1230.
- Jede Regel muss auf einer neuen Zeile beginnen oder Teil einer durch Semikolon getrennten Liste sein.
- USB-Klassencodes finden Sie auf der Website von USB Implementers Forum, Inc.

Beispiel für administratordefinierte USB-Richtlinienregeln:

- Allow: VID=1230 PID=0007 # Weitere Industrie, Weiteres Flash-Laufwerk
- Deny: Class=08 SubClass=05 # Massenspeichergeräte
- Eine Regel, die alle USB-Geräte verweigert, erstellen Sie mit "DENY:" ohne weitere Tags.

## Client-USB-Geräteumleitung für Plug & Play-Geräte

Mit dieser Einstellung legen Sie fest, ob Plug & Play-Geräte, wie Kameras oder POS-Geräte (Point of Sale) in einer Clientsitzung verwendet werden können.

In der Standardeinstellung ist die Umleitung von Plug & Play-Geräten zugelassen. Bei der Einstellung Zugelassen werden alle Plug & Play-Geräte für einen bestimmten Benutzer oder eine bestimmte Benutzergruppe umgeleitet. Bei der Einstellung Nicht zugelassen werden keine Geräte umgeleitet.





# Einstellungen der Richtlinie "Visuelle Anzeige"

May 10, 2016

Der Abschnitt "Visuelle Anzeige" enthält Richtlinieneinstellungen, mit denen die Qualität der von virtuellen Desktops an das Benutzergerät gesendeten Bilder gesteuert wird.

## Bevorzugte Farbtiefe für einfache Grafiken

Gilt nur für XenApp und XenDesktop **7.6 FP3**.

Ermöglicht die Verringerung des Farbtiefewerts für einfache Grafiken auf **16 Bit pro Pixel**, was potenziell unter geringen Einbußen bei der Bildqualität die Reaktion bei Verbindungen mit geringer Bandbreite verbessert. Diese Option wird nur unterstützt, wenn kein Videocodec zum Komprimieren von Grafiken verwendet wird.

Die Standardeinstellung ist 24 Bit pro Pixel.

## Frameratesollwert

Mit dieser Einstellung geben Sie die maximale Anzahl von Frames pro Sekunde an, die vom virtuellen Desktop zum Benutzergerät gesendet werden.

In der Standardeinstellung ist die Höchstanzahl 30 Frames pro Sekunde.

Die Festlegung auf eine hohe Anzahl von Frames pro Sekunde (z. B. 30) führt zu einer besseren Benutzererfahrung, erfordert aber mehr Bandbreite. Wenn Sie die Anzahl von Frames pro Sekunde herabsetzen (z. B. auf 10), wird die Serverskalierbarkeit auf Kosten der Benutzererfahrung erhöht. Bei Benutzergeräten mit langsamen CPUs erzielen Sie durch Festlegen eines niedrigeren Werts eine bessere Benutzererfahrung.

## Verwenden von Videocodec für die Komprimierung

Gilt nur für XenApp und XenDesktop **7.6 FP3**.

Ermöglicht die Verwendung eines Videocodecs zum Komprimieren von Grafiken, wenn am Endpunkt eine Videodecodierung verfügbar ist. Ist am Endpunkt keine Videodecodierung verfügbar oder wenn Sie festlegen, dass kein Videocodec verwendet werden soll, wird eine Kombination aus Standbildkomprimierung und Bitmapcaching verwendet.

Die Standardeinstellung ist "Verwenden von Videocodec wenn verfügbar".

## Bildqualität

Mit dieser Einstellung legen Sie die Bildqualität für auf dem Benutzergerät angezeigte Bilder fest.

Die Standardeinstellung ist "Mittel".

Zum Festlegen der Bildqualität wählen Sie eine der folgenden Optionen:

- **Niedrig**
- **Mittel:** bietet die beste Leistung und Bandbreiteneffizienz in den meisten Anwendungsfällen.
- **Hoch:** empfiehlt sich, wenn visuell verlustfreie Bildqualität gewünscht wird.
- **Zu verlustfrei verbessern:** sendet verlustreiche Bilder in Zeiträumen mit hoher Netzwerkaktivität und verlustfreie Bilder bei verringerter Netzwerkaktivität an das Benutzergerät. Mit dieser Einstellung wird die Leistung bei Netzwerkverbindungen mit beschränkter Bandbreite verbessert.

- **Immer verlustfrei:** In Situationen, in denen kein Qualitätsverlust akzeptabel ist (z. B. bei der Anzeige von Röntgenbildern), wählen Sie Immer verlustfrei, um sicherzustellen, dass keine verlustreichen Daten an das Benutzergerät gesendet werden.

Wenn die Einstellung **Legacygrafikmodus** aktiviert ist, hat die Einstellung **Bildqualität** keine Auswirkungen in der Richtlinie.

# Einstellungen der Richtlinie "Bewegtbilder"

May 10, 2016

Der Abschnitt "Bewegtbilder" enthält Einstellungen, mit denen Sie die Komprimierung für dynamische Bilder entfernen oder ändern können.

## Mindestbildqualität

Hinweis: Bei Virtual Delivery Agent 7.x gilt diese Richtlinieneinstellung nur, wenn die Richtlinieneinstellung Legacygrafikmodus aktiviert ist.

Mit dieser Einstellung wird die zulässige Mindestbildqualität für den adaptiven Bildschirm angegeben. Je geringer die verwendete Komprimierung ist, desto höher ist die Qualität der angezeigten Bilder. Es stehen folgende Komprimierungen zur Verfügung: Ultrahoch, Sehr hoch, Hoch, Normal und Niedrig.

Die Standardeinstellung ist "Normal".

## Bewegtbildkomprimierung

Mit dieser Einstellung wird angegeben, ob der adaptive Bildschirm aktiviert ist. Der adaptive Bildschirm passt die Bildqualität von Videos und Bildübergängen in Diashows auf der Grundlage der verfügbaren Bandbreite automatisch an. Bei aktiviertem adaptivem Bildschirm werden Benutzern gleichmäßig ausgeführte Präsentationen ohne Qualitätseinbußen angezeigt. Standardmäßig ist der adaptive Bildschirm aktiviert.

Bei VDAs der Version 7.0 bis 7.6 gilt diese Einstellung nur, wenn der Legacygrafikmodus aktiviert ist. Bei VDAs der Version 7.6 FP1, FP2 und FP3 gilt diese Einstellung, wenn der Legacygrafikmodus aktiviert ist oder wenn der Legacygrafikmodus deaktiviert ist und kein Videocodex zum Komprimieren von Grafiken verwendet wird.

Wenn der Legacygrafikmodus aktiviert ist, muss die Sitzung neu gestartet werden, damit die Richtlinienänderungen wirksam werden. Adaptive Anzeige und progressive Anzeige schließen einander aus, d. h. durch Aktivieren der adaptiven Anzeige wird die progressive Anzeige deaktiviert und umgekehrt. Allerdings können progressive und adaptive Anzeige zur gleichen Zeit deaktiviert sein. Die progressive Anzeige wird als Legacyfeature für XenApp und XenDesktop nicht empfohlen. Durch Festlegen des Schwellenwerts für die progressive Komprimierung wird die adaptive Anzeige deaktiviert.

## Grad der progressiven Komprimierung

Hinweis: Bei Virtual Delivery Agent 7.x gilt diese Richtlinieneinstellung nur, wenn die Richtlinieneinstellung Legacygrafikmodus aktiviert ist.

Mit dieser Einstellung wird zuerst ein weniger detailliertes Bild angezeigt, das dafür aber schneller dargestellt werden kann.

In der Standardeinstellung wird keine progressive Komprimierung angewendet.

Sobald es verfügbar ist, wird ein detailreicheres Bild angezeigt, das die normale Einstellung für verlustreiche Komprimierung verwendet. Verwenden Sie sehr hohe oder ultrahohe Komprimierung für die verbesserte Anzeige von bandbreitenintensiven Grafiken, wie etwa Fotografien.

Die progressive Komprimierung ist nur wirksam, wenn der Komprimierungsgrad höher ist als die Einstellung für Grad der verlustreichen Komprimierung.

Hinweis: Der stärkere Komprimierungsgrad für die progressive Komprimierung verbessert auch die Interaktivität von dynamischen Bildern über Clientverbindungen. Die Qualität eines dynamischen Bilds, z. B. ein sich drehendes dreidimensionales Modell, wird temporär verringert, bis das Bild stehen bleibt. Zu dem Zeitpunkt wird dann die reguläre

Einstellung der verlustreichen Komprimierung angewendet.

Die folgenden Richtlinieneinstellungen hängen zusammen:

- Schwellenwert für progressive Komprimierung
- Progressive Heavyweight-Komprimierung

### Schwellenwert für progressive Komprimierung

Hinweis: Bei Virtual Delivery Agent 7.x gilt diese Richtlinieneinstellung nur, wenn die Richtlinieneinstellung Legacygrafikmodus aktiviert ist.

Mit dieser Einstellung geben Sie die maximale Bandbreite in Kilobits pro Sekunde für eine Verbindung an, auf die progressive Komprimierung angewendet wird. Die Komprimierung wird nur für Clientverbindungen unter diesem Bandbreitenwert verwendet.

Der Standardschwellenwert ist 2147483647 Kilobits pro Sekunde.

Die folgenden Richtlinieneinstellungen hängen zusammen:

- Schwellenwert für progressive Komprimierung
- Progressive Heavyweight-Komprimierung

### Mindestframeratesollwert

Mit dieser Einstellung wird die Framerate pro Sekunde eingestellt, die das System für dynamische Bilder in Netzwerken mit geringer Bandbreite versucht beizubehalten.

Die Standardeinstellung für diesen Parameter ist 10 F/s.

Bei VDAs der Version 7.0 bis 7.6 gilt diese Einstellung nur, wenn der Legacygrafikmodus aktiviert ist. Bei VDAs der Version 7.6 FP1 bis FP3 gilt diese Einstellung, wenn der Legacygrafikmodus deaktiviert oder aktiviert ist.

# Einstellungen der Richtlinie "Standbilder"

May 10, 2016

Der Abschnitt "Standbilder" enthält Einstellungen, mit denen Sie die Komprimierung für statische Bilder entfernen oder ändern können.

Mit dieser Einstellung aktivieren oder deaktivieren Sie die Verwendung der zusätzlichen Farbkomprimierung für Bilder, die über Clientverbindungen mit beschränkter Bandbreite bereitgestellt werden; dies verbessert die Reaktionszeit, da die Bilder in geringerer Qualität angezeigt werden.

Standardmäßig ist die zusätzliche Farbkomprimierung deaktiviert.

Bei Aktivierung wird die zusätzliche Farbkomprimierung nur angewendet, wenn die Bandbreite der Clientverbindung unter dem für Schwellenwert für zusätzliche Farbkomprimierung festgelegten Wert liegt. Wenn die Bandbreite der Clientverbindung über dem Schwellenwert liegt oder Deaktiviert ausgewählt ist, wird die zusätzliche Farbkomprimierung nicht angewendet.

Hinweis: Bei Virtual Delivery Agent 7.x gilt diese Richtlinieneinstellung nur, wenn die Richtlinieneinstellung Legacygrafikmodus aktiviert ist.

Mit dieser Einstellung geben Sie die maximale Bandbreite in Kilobits pro Sekunde für eine Verbindung an, unter der die zusätzliche Farbkomprimierung angewendet wird. Wenn die Bandbreite der Clientverbindung unter den eingestellten Wert abfällt, wird die zusätzliche Farbkomprimierung (falls aktiviert) angewendet.

Der Standardschwellenwert ist 8192 Kilobits pro Sekunde.

Hinweis: Bei Virtual Delivery Agent 7.x gilt diese Richtlinieneinstellung nur, wenn die Richtlinieneinstellung Legacygrafikmodus aktiviert ist.

Mit dieser Einstellung reduzieren Sie die erforderliche Bandbreite noch stärker als mit der progressiven Komprimierung, ohne dabei an Bildqualität zu verlieren, indem ein verbesserter grafischer Algorithmus verwendet wird, der aber mehr CPU beansprucht.

Standardmäßig ist die Heavyweight-Komprimierung deaktiviert.

Wenn die Heavyweight-Komprimierung aktiviert ist, gilt sie für alle verlustreichen Komprimierungen. Diese Einstellung wird von Citrix Receiver unterstützt, hat aber keine Auswirkung auf andere Plug-Ins.

Die folgenden Richtlinieneinstellungen hängen zusammen:

- Grad der progressiven Komprimierung
- Schwellenwert für progressive Komprimierung

Hinweis: Bei Virtual Delivery Agent 7.x gilt diese Richtlinieneinstellung nur, wenn die Richtlinieneinstellung Legacygrafikmodus aktiviert ist.

Mit dieser Einstellung steuern Sie den Grad der verlustreichen Komprimierung, der für Grafiken verwendet wird, die über Clientverbindungen mit beschränkter Bandbreite bereitgestellt werden. In solchen Fällen kann die Anzeige von Bildern ohne

Komprimierung sehr langsam sein.

Standardmäßig wird eine mittlere Komprimierung ausgewählt.

Bessere Reaktionszeiten bei bandbreitenintensiven Bildern erzielen Sie mit hoher Komprimierung. In Fällen, in denen die Bilddaten erhalten bleiben müssen, beispielsweise bei der Anzeige von Röntgenbildern, wo kein Qualitätsverlust akzeptabel ist, sollten Sie die verlustreiche Komprimierung nicht einsetzen.

Verwandte Richtlinieneinstellung: Schwellenwert für verlustreiche Komprimierung

Hinweis: Bei Virtual Delivery Agent 7.x gilt diese Richtlinieneinstellung nur, wenn die Richtlinieneinstellung Legacygrafikmodus aktiviert ist.

Mit dieser Einstellung geben Sie die maximale Bandbreite in Kilobits pro Sekunde für eine Verbindung an, auf die die verlustreiche Komprimierung angewendet wird.

Der Standardschwellenwert ist 2147483647 Kilobits pro Sekunde.

Wenn Sie die Einstellung Grad der verlustreichen Komprimierung einer Richtlinie hinzufügen, ohne einen Schwellenwert anzugeben, kann sich dadurch die Anzeigegeschwindigkeit für detailreiche Bitmaps, wie Fotografien, über ein LAN verbessern.

Verwandte Richtlinieneinstellung: Grad der verlustreichen Komprimierung

# Einstellungen der Richtlinie "WebSockets"

May 10, 2016

Der Abschnitt "WebSockets" enthält Richtlinieneinstellungen für den Zugriff auf virtuelle Desktops und gehostete Anwendungen mit Receiver für HTML5. Das Feature WebSockets erhöht die Sicherheit und verringert die Last durch bidirektionale Kommunikation zwischen browserbasierten Anwendungen und Servern ohne Öffnen von mehreren HTTP-Verbindungen.

Diese Einstellung lässt WebSockets-Verbindungen zu oder lehnt sie ab.

Standardmäßig sind WebSocket-Verbindungen nicht zulässig.

Mit dieser Einstellung wird der Port für eingehende WebSocket-Verbindungen festgelegt.

In der Standardeinstellung ist der Wert 8008.

Diese Einstellung bietet eine durch Trennzeichen getrennte Liste der vertrauenswürdigen Ursprungsserver, normalerweise Receiver für Web, in Form von URLs. Nur WebSockets-Verbindungen, die von einer dieser Adressen stammen, werden vom Server akzeptiert.

Standardmäßig wird der Platzhalter \* verwendet. Damit wird allen Receiver für Web-URLs vertraut.

Wenn Sie eine Adresse in die Liste eingeben möchten, verwenden Sie folgende Syntax:

<Protokoll>://<Vollqualifizierter Domänenname des Hosts>:[Port]

Das Protokoll muss HTTP oder HTTPS sein. Wenn der Port nicht angegeben wird, wird Port 80 für HTTP und Port 443 für HTTPS verwendet.

Der Platzhalter \* kann innerhalb der URL verwendet werden, außer als Teil einer IP-Adresse (10.105. \*. \*).



# Einstellungen der Richtlinie "Lastverwaltung"

May 10, 2016

Der Abschnitt "Lastverwaltung" enthält Richtlinieneinstellungen für das Aktivieren und Konfigurieren des Lastausgleichs zwischen Servern, über die Windows Server-Betriebssystemmaschinen bereitgestellt werden.

Mit dieser Einstellung geben Sie die maximal zulässige Anzahl gleichzeitiger Anmeldungen bei einem Server an.

Die Standardeinstellung ist 2.

Mit dieser Einstellung geben Sie den Prozentsatz der CPU-Nutzung an, bei dem der Server Volllast meldet. Ist diese Einstellung aktiviert, beträgt der Standardwert, bei dem der Server Volllast meldet, 90 %.

Standardmäßig ist diese Einstellung deaktiviert und die CPU-Nutzung wird bei der Lastberechnung nicht berücksichtigt.

Mit dieser Einstellung geben Sie die Prioritätsstufe an, bei der die Prozess-CPU-Auslastung vom Lastindex der CPU-Nutzung ausgeschlossen wird.

Die Standardeinstellung ist "Unter normal" oder "Niedrig".

Mit dieser Einstellung geben Sie die Länge der Datenträgerwarteschlange an, zu der der Server 75 % Volllast meldet. Der Standardwert dieser Einstellung ist 8.

Standardmäßig ist diese Einstellung deaktiviert und die Datenträgernutzung wird bei der Lastberechnung nicht berücksichtigt.

Mit dieser Einstellung geben Sie die maximale Anzahl von Sitzungen an, die von einem Server gehostet werden können. Ist die Einstellung aktiviert, beträgt der Standardwert für die maximale Anzahl Sitzungen an, die von einem Server gehostet werden können, 250.

Standardmäßig ist diese Einstellung aktiviert.

Mit dieser Einstellung geben Sie den Prozentsatz der Speichernutzung an, bei dem der Server Volllast meldet. Ist diese Einstellung aktiviert, beträgt der Standardwert, bei dem der Server Volllast meldet, 90 %.

Standardmäßig ist diese Einstellung deaktiviert und die Speichernutzung wird bei der Lastberechnung nicht berücksichtigt.

Mit dieser Einstellung geben Sie einen Näherungswert der Speichernutzung durch das Basisbetriebssystem in MB an, unterhalb dessen die Speichernutzung bei einem Server als Nulllast interpretiert wird.

Der Standardwert dieser Einstellung ist 768 MB.

# Einstellungen der Richtlinie "Profilverwaltung"

Nov 02, 2016

Der Abschnitt "Profilverwaltung" enthält Richtlinieneinstellungen zum Aktivieren der Profilverwaltung und zum Konfigurieren der Gruppen, die in die Verarbeitung der Profilverwaltung eingeschlossen bzw. ausgeschlossen werden sollen.

Weitere Informationen, wie die Namen der entsprechenden INI-Dateieinstellungen und die für eine Richtlinieneinstellung erforderliche Version der Profilverwaltung, finden Sie unter [Profilverwaltungsrichtlinien](#).

# Einstellungen der Richtlinie "Erweitert"

May 10, 2016

Der Abschnitt "Erweitert" enthält Richtlinieneinstellungen für die erweiterte Konfiguration der Profilverwaltung.

Mit dieser Einstellung legen Sie fest, dass die Umgebung von der Profilverwaltung untersucht werden kann, z. B., um zu überprüfen, ob persönliche vDisks vorhanden sind, und um die Gruppenrichtlinie entsprechend zu konfigurieren. Nur Richtlinien für die Profilverwaltung mit dem Status "Nicht konfiguriert" werden angepasst, sodass alle zuvor vorgenommenen Anpassungen beibehalten werden. Dieses Feature beschleunigt die Bereitstellung und vereinfacht die Optimierung. Es ist keine Konfiguration des Features erforderlich, aber Sie können die automatische Konfiguration bei Upgrades (zum Beibehalten der Einstellungen von früheren Versionen) oder bei der Problembehandlung deaktivieren. Die automatische Konfiguration funktioniert in XenApp oder anderen Umgebungen nicht.

In der Standardeinstellung ist die automatische Konfiguration zugelassen.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, wird die automatische Konfiguration aktiviert. Einstellungen der Profilverwaltung können sich dann möglicherweise ändern, wenn sich die Umgebung ändert.

Mit dieser Einstellung wird eine Abmeldung eines Benutzers durch die Profilverwaltung ermöglicht, wenn ein Problem auftritt, z. B. wenn der Benutzerspeicher nicht verfügbar ist. Wenn diese Einstellung aktiviert ist, wird dem Benutzer eine Fehlermeldung angezeigt, bevor er abgemeldet wird. Wenn diese Einstellung deaktiviert ist, erhalten Benutzer ein temporäres Profil.

Standardmäßig ist diese Einstellung deaktiviert und Benutzer erhalten ein temporäres Profil, wenn ein Problem auftritt.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, wird ein temporäres Profil bereitgestellt.

Mit dieser Einstellung geben Sie die Anzahl der Zugriffsversuche auf gesperrte Dateien durch die Profilverwaltung an.

Der Standardwert dieser Einstellung ist fünf Versuche.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, wird der Standardwert verwendet.

Mit dieser Einstellung legen Sie fest, dass die Datei index.dat beim Abmelden von der Profilverwaltung verarbeitet werden kann, damit Internet-Cookies im Dateisystem nach längerem Browsen entfernt werden. Auf diese Weise wird eine Aufblähung von Profilen vermieden. Eine Aktivierung der Einstellung verlängert die Abmeldezeiten. Aktivieren Sie sie daher nur, wenn das entsprechende Problem bei Ihnen auftritt.

Standardmäßig ist diese Einstellung deaktiviert und die Profilverwaltung verarbeitet die Datei index.dat beim Abmelden nicht.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, wird Index.dat nicht verarbeitet.

# Grundlegende Richtlinienereinstellungen

May 10, 2016

Der Abschnitt "Grundlegend" enthält Richtlinienereinstellungen für die grundlegende Konfiguration der Profilverwaltung.

Mit dieser Einstellung können geänderte Dateien und Ordner (aber keine Registrierungseinträge) während einer Sitzung mit dem Benutzerspeicher synchronisiert werden bevor die Abmeldung erfolgt.

Standardmäßig ist die Synchronisierung mit dem Benutzerspeicher während laufender Sitzungen deaktiviert.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, ist sie aktiviert.

Mit dieser Einstellung wird die Profilverwaltung zur Verarbeitung von An- und Abmeldungen aktiviert.

Standardmäßig ist diese Einstellung deaktiviert, um die Bereitstellung zu vereinfachen.

Wichtig: Citrix empfiehlt, eine Aktivierung der Profilverwaltung erst dann durchzuführen, wenn alle anderen Setupaufgaben ausgeführt wurden und getestet wurde, wie sich Citrix Benutzerprofile in der Umgebung verhalten.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, verarbeitet die Profilverwaltung keine Windows-Benutzerprofile.

Mit dieser Einstellung geben Sie an, welche lokalen Computergruppen und welche Domänengruppen (lokale, globale und universelle) nicht von der Profilverwaltung verarbeitet werden sollen.

Wenn diese Option aktiviert ist, verarbeitet die Profilverwaltung keine Mitglieder der angegebenen Benutzergruppen.

Standardmäßig ist diese Einstellung deaktiviert und Mitglieder aller Benutzergruppen werden verarbeitet.

Geben Sie Domänengruppen im Format <DOMÄNENNAME>\<GRUPPENNAME> an.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden Mitglieder aller Benutzergruppen verarbeitet.

Mit dieser Einstellung wird die Unterstützung von Offlineprofilen aktiviert, durch die Profile nach einer Unterbrechung der Verbindung mit dem Netzwerk zum frühestmöglichen Zeitpunkt mit dem Benutzerspeicher synchronisiert werden.

Standardmäßig ist die Unterstützung von Offlineprofilen deaktiviert.

Diese Einstellung gilt für Benutzer von Laptops und mobile Benutzer im Roamingmodus. Wenn die Verbindung zum

Netzwerk unterbrochen wird, bleiben die Profile auf dem Laptop oder Gerät intakt, selbst wenn das Gerät neu gestartet wird oder im Ruhezustand gewesen ist. Während mobile Benutzer arbeiten, werden ihre Profile lokal aktualisiert und am Ende mit dem Benutzerspeicher synchronisiert, wenn die Netzwerkverbindung wiederhergestellt ist.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, ist die Unterstützung von Offlineprofilen deaktiviert.

Mit dieser Einstellung geben Sie den Pfad zu dem Verzeichnis (Benutzerspeicher) an, in dem Benutzereinstellungen, z. B. Registrierungseinstellungen und synchronisierte Dateien, gespeichert werden.

Standardmäßig wird das Windows-Verzeichnis auf dem Basislaufwerk verwendet.

Wenn diese Einstellung deaktiviert ist, werden die Benutzereinstellungen im Windows-Unterverzeichnis des Basisverzeichnisses gespeichert.

Mögliche Pfade:

- **Relativer Pfad:** Dieser Pfad muss relativ zum Basisverzeichnis sein. Dieses ist normalerweise mit dem Attribut #homeDirectory# für einen Benutzer in Active Directory konfiguriert.
- **Absoluter UNC-Pfad:** Hiermit wird üblicherweise eine Serverfreigabe oder ein DFS-Namespace angegeben.
- **Deaktiviert oder nicht konfiguriert:** In diesem Fall wird als Wert #homeDirectory#\Windows angenommen.

Verwenden Sie die folgenden Variablentypen beim Konfigurieren dieser Richtlinieneinstellung:

- Systemumgebungsvariablen in Prozentzeichen (z. B. %ProfVer%). Beachten Sie, dass Systemumgebungsvariablen normalerweise weitere Einrichtung erfordern.
- Attribute des Active Directory-Benutzerobjekts in Rauten (z. B. #sAMAccountName#).
- Profilverwaltungsvariablen. Weitere Informationen finden Sie in der Dokumentation zur Profilverwaltung.

Sie können auch mit den Benutzerumgebungsvariablen %username% und %userdomain% benutzerdefinierte Attribute erstellen, um Organisationsvariablen wie Standort und Benutzer vollständig zu definieren. Bei Attributen muss Groß- und Kleinschreibung beachtet werden.

Beispiele:

- \\Server\Freigabe\#sAMAccountName# speichert die Benutzereinstellungen unter dem UNC-Pfad \\Server\Freigabe\KurtMeier (wenn #sAMAccountName# zu Kurt Meier als aktuellem Benutzer aufgelöst wird).
- \\server\profiles\$\%USERNAME%.%USERDOMAIN%\!CTX\_PROFILEVER!!CTX\_OSBITNESS! kann erweitert werden zu \\server\profiles\$\JohnSmith.DOMAINCONTROLLER1\v2x64

Wichtig: Unabhängig davon, welche Attribute oder Variablen Sie verwenden, müssen Sie sicherstellen, dass diese Einstellung zu einem Ordner über dem Ordner, der NTUSER.DAT enthält, aufgelöst wird. Wenn sich diese Datei z. B. in \\Server\profiles\$\KurtMeier.BuHa\v2x64\UPM\_Profile befindet, geben Sie den Pfad zum Benutzerspeicher als \\Server\profiles\$\KurtMeier.BuHa\v2x64 an (ohne den Unterordner \UPM\_Profile) an.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, wird das Windows-Verzeichnis auf dem Basislaufwerk verwendet.

Mit dieser Einstellung wird angegeben, ob Anmeldungen von Mitgliedern der Gruppe "BUILTIN\Administrators" verarbeitet

werden. Dies ermöglicht es Domänenbenutzern mit lokalen Administratorrechten, d. h. normalerweise Benutzer mit zugewiesenen virtuellen Desktops, die Verarbeitung zu umgehen, sich anzumelden und Probleme mit der Profilverwaltung bei einem Desktop zu behandeln.

Wenn diese Einstellung unter Serverbetriebssystemen deaktiviert oder nicht konfiguriert ist, nimmt die Profilverwaltung an, dass Anmeldungen von Domänenbenutzern, aber nicht von lokalen Administratoren, verarbeitet werden müssen. Unter Desktopbetriebssystemen werden Anmeldungen lokaler Administratoren verarbeitet.

Diese Einstellung ist standardmäßig deaktiviert und Anmeldungen lokaler Administratoren werden nicht verarbeitet.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden Anmeldungen lokaler Administratoren nicht verarbeitet.

Mit dieser Einstellung geben Sie an, welche lokalen Computergruppen und welche Domänengruppen (lokale, globale und universelle) von der Profilverwaltung verarbeitet werden sollen.

Wenn diese Option aktiviert ist, verarbeitet die Profilverwaltung nur Mitglieder der angegebenen Benutzergruppen.

Standardmäßig ist diese Einstellung deaktiviert und Mitglieder aller Benutzergruppen werden verarbeitet.

Geben Sie Domänengruppen im Format <DOMÄNENNAME>\<GRUPPENNAME> an.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden Mitglieder aller Benutzergruppen verarbeitet.



# Plattformübergreifende Richtlinienereinstellungen

May 10, 2016

Der Abschnitt "Plattformübergreifende Einstellungen" enthält Richtlinienereinstellungen für die Konfiguration der plattformübergreifenden Einstellungen der Profilverwaltung.

Mit dieser Einstellung geben Sie die Benutzergruppen an, deren Profile verarbeitet werden sollen, wenn das Feature für plattformübergreifende Einstellungen aktiviert ist.

Standardmäßig ist diese Einstellung deaktiviert und alle Benutzergruppen in der Richtlinieneinstellung Verarbeitete Gruppen werden verarbeitet.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden alle Benutzergruppen verarbeitet.

Mit dieser Einstellung aktivieren oder deaktivieren Sie das Feature "Plattformübergreifende Einstellungen". Dieses ermöglicht das Migrieren und Roamen von Benutzerprofilen, wenn ein Benutzer eine Verbindung mit einer Anwendung herstellt, die unter mehreren Betriebssystemen ausgeführt wird.

Standardmäßig ist das Feature "Plattformübergreifende Einstellungen" deaktiviert.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden keine plattformübergreifenden Einstellungen angewendet.

Mit dieser Einstellung geben Sie in Form eines UNC-Pfads den Netzwerkspeicherort an, der die aus dem Downloadpaket kopierten Definitionsdateien enthält.

Hinweis: Benutzer benötigen Lesezugriff und Administratoren Schreibzugriff auf diesen Speicherort, bei dem es sich entweder um eine Server Message Block-Dateifreigabe (SMB) oder eine Common Internet File System-Dateifreigabe (CIFS) handeln muss.

In der Standardeinstellung ist kein Pfad angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden keine plattformübergreifenden Einstellungen angewendet.

Diese Einstellung gibt den Pfad zum Speicher für plattformübergreifende Einstellungen an. Hier werden die plattformübergreifenden Einstellungen der Benutzer gespeichert. Der Pfad kann ein UNC-Pfad oder ein relativer Pfad zum Basisverzeichnis sein.

Hinweis: Benutzer müssen über Schreibzugriff auf den Speicher für plattformübergreifende Einstellungen verfügen. Standardmäßig ist diese Einstellung deaktiviert und der Pfad "Windows\PM\_CP" wird verwendet.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, wird der Standardwert verwendet.

Mit dieser Einstellung legen Sie eine Plattform als Basisplattform fest, wenn die entsprechende Einstellung für die Organisationseinheit der Plattform aktiviert ist. Daten von den Profilen der Basisplattform werden in den Speicher für plattformübergreifende Einstellungen migriert.

Die Profile jeder Plattform werden in einer separaten Organisationseinheit gespeichert. Dies bedeutet, dass Sie die Plattform auswählen müssen, deren Profildaten zum Füllen des Speichers für plattformübergreifende Einstellungen verwendet werden sollen. Dies wird als Basisplattform bezeichnet.

Wenn diese Option aktiviert ist, migriert die Profilverwaltung die Daten des Einzelplattformprofils in den Speicher, wenn der Speicher für plattformübergreifende Einstellungen eine Definitionsdatei ohne Daten enthält oder die zwischengespeicherten Daten eines Einzelplattformprofils neuer sind als die Definitionsdaten im Speicher.

Wichtig: Wenn diese Einstellung in mehreren Organisationseinheiten für mehrere Benutzer oder Maschinenobjekte aktiviert ist, wird die Plattform, bei der sich der erste Benutzer anmeldet, zum Basisprofil. Standardmäßig ist diese Einstellung deaktiviert und die Profilverwaltung migriert die Daten des Einzelplattformprofils nicht in den Speicher.

# Einstellungen der Richtlinie "Dateisystem"

May 10, 2016

Der Abschnitt "Dateisystem" enthält Richtlinieneinstellungen zum Angeben der Dateien und Verzeichnisse in einem Benutzerprofil, die zwischen dem System, auf dem das Profil installiert ist, und dem Benutzerspeicher synchronisiert werden sollen.

# Einstellungen der Richtlinie "Ausschlüsse"

May 10, 2016

Der Abschnitt "Ausschlüsse" enthält Richtlinieneinstellungen zum Konfigurieren der Dateien und Verzeichnisse in einem Benutzerprofil, die von der Synchronisierung ausgeschlossen werden sollen.

Mit dieser Einstellung geben Sie eine Liste der Ordner im Benutzerprofil an, die bei der Synchronisierung ignoriert werden sollen.

Geben Sie die Ordernamen als Pfad relativ zum Benutzerprofil (% USERPROFILE%) an.

Standardmäßig ist diese Einstellung deaktiviert und alle Ordner in Benutzerprofilen werden synchronisiert.

Beispiel: Desktop ignoriert den Ordner "Desktop" im Benutzerprofil.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden alle Ordner im Benutzerprofil synchronisiert.

Mit dieser Einstellung geben Sie eine Liste der Dateien im Benutzerprofil an, die bei der Synchronisierung ignoriert werden sollen.

Standardmäßig ist diese Einstellung deaktiviert und alle Dateien in Benutzerprofilen werden synchronisiert.

Geben Sie die Dateinamen als Pfad relativ zum Benutzerprofil (% USERPROFILE%) an. Hinweis: Platzhalter sind zulässig und werden rekursiv angewendet.

Beispiel: "Desktop\Desktop.ini" ignoriert die Datei Desktop.ini im Verzeichnis "Desktop".

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden alle Dateien im Benutzerprofil synchronisiert.

# Einstellungen der Richtlinie "Synchronisierung"

May 10, 2016

Der Abschnitt "Synchronisierung" enthält Richtlinieneinstellungen für die Angabe, welche Dateien und Ordner in einem Benutzerprofil zwischen dem System, auf dem das Profil installiert ist, und dem Benutzerspeicher synchronisiert werden.

Diese Einstellung gibt die Dateien an, die sich in ausgeschlossenen Ordnern befinden und von der Profilverwaltung in die Synchronisierung eingeschlossen werden sollen. Standardmäßig synchronisiert die Profilverwaltung alle Elemente im Benutzerprofil. Es ist nicht erforderlich, Unterordner des Benutzerprofils einzuschließen, indem Sie sie dieser Liste hinzufügen. Weitere Informationen finden Sie unter [Aufnehmen und Ausschließen von Objekten](#).

Pfade in dieser Liste müssen relativ zum Benutzerprofil sein.

Beispiel: "Desktop\ausschließen\aufnehmen" stellt sicher, dass der Unterordner "aufnehmen" synchronisiert wird, selbst wenn das Verzeichnis "Desktop\ausschließen" nicht synchronisiert wird.

Standardmäßig ist diese Einstellung deaktiviert und keine Ordner sind angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden nur nicht ausgeschlossene Ordner im Benutzerprofil synchronisiert.

Diese Einstellung gibt die Dateien an, die sich in ausgeschlossenen Ordnern befinden und von der Profilverwaltung in die Synchronisierung eingeschlossen werden sollen. Standardmäßig synchronisiert die Profilverwaltung alle Elemente im Benutzerprofil. Es ist nicht erforderlich, Dateien in dem Benutzerprofil einzuschließen, indem Sie sie dieser Liste hinzufügen. Weitere Informationen finden Sie unter [Aufnehmen und Ausschließen von Objekten](#).

Pfade in dieser Liste müssen relativ zum Benutzerprofil sein. Relative Pfade werden als relativ zum Benutzerprofil interpretiert. Platzhalter können nur für Dateinamen verwendet werden. Platzhalter können nicht verschachtelt werden und werden rekursiv angewendet.

Beispiele:

- Anwendungsdaten\Lokal\Microsoft\Office\Access.qat gibt eine Datei unter einem Ordner an, der in der Standardkonfiguration ausgeschlossen wurde.
- Anwendungsdaten\Lokal\Anwendungen\\*.cfg gibt alle Dateien mit der Erweiterung .cfg im Profilordner Anwendungsdaten\Lokal\Anwendungen und dessen Unterordnern an.

Standardmäßig ist diese Einstellung deaktiviert und keine Dateien sind angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden nur nicht ausgeschlossene Dateien im Benutzerprofil synchronisiert.

Mit dieser Einstellung geben Sie an, welche Ordner relativ zum Stammordner eines Benutzerprofils gespiegelt werden sollen. Diese Richtlinie kann beim Lösen von Problemen bei Transaktionsordnern (auch "Referenzordner" genannt) helfen. Dies sind Ordner, die voneinander abhängige Dateien enthalten, wobei eine Datei Verweise auf andere Dateien enthält.

Durch das Spiegeln von Ordnern kann die Profilverwaltung einen Transaktionsordner und seinen Inhalt als eine Entität verarbeiten. So wird das Aufblähen von Profilen verhindert. Beachten Sie, dass in solchen Situationen der "letzte Schreibvorgang Priorität hat", d. h. dass Dateien in gespiegelten Ordnern, die in mehr als einer Sitzung geändert wurden, von der letzten Aktualisierung überschrieben werden. Hierdurch gehen Profiländerungen verloren.

Sie können z. B. den Cookies-Ordner von Internet Explorer spiegeln, damit Index.dat mit den Cookies synchronisiert wird, auf die die Datei verweist.

Wenn ein Benutzer zwei Internet Explorer-Sitzungen auf unterschiedlichen Servern hat und er in jeder Sitzung auf andere Websites zugreift, werden Cookies dieser Sites auf dem entsprechenden Server hinzugefügt. Wenn sich der Benutzer von der ersten Sitzung abmeldet (oder auch mitten in der Sitzung, wenn das Feature für aktives Zurückschreiben konfiguriert ist), sollten die Cookies der zweiten Sitzung die Cookies der ersten Sitzung ersetzen. Stattdessen werden sie jedoch zusammengeführt und die Verweise auf die Cookies in Index.dat sind infolgedessen veraltet. Weiteres Browsen in neuen Sitzungen kann zum wiederholten Zusammenführen und einem aufgeblähten Cookie-Ordner führen.

Durch Spiegeln des Cookie-Ordners wird dieses Problem gelöst, indem Cookies, jedes Mal wenn der Benutzer sich abmeldet, mit denen der letzten Sitzung überschrieben werden. So bleibt Index.dat aktuell.

Standardmäßig ist diese Einstellung deaktiviert und es werden keine Ordner gespiegelt.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Richtlinie weder hier noch in der INI-Datei konfiguriert ist, werden keine Ordner gespiegelt.

# Einstellungen der Richtlinie "Ordnerumleitung"

May 10, 2016

Der Abschnitt "Ordnerumleitung" enthält Richtlinieneinstellungen für die Angabe, ob häufig in Profilen erscheinende Ordner an einen freigegebenen Speicherort im Netzwerk umgeleitet werden sollen.

Diese Einstellung ermöglicht einem Administrator den Zugriff auf den Inhalt von umgeleiteten Ordnern der Benutzer.

Diese Einstellung ist standardmäßig deaktiviert und es haben ausschließlich Benutzer Zugriff auf den Inhalt ihrer umgeleiteten Ordner.

Diese Einstellung ermöglicht die Verwendung der Umgebungsvariablen %userdomain% als Teil des für umgeleitete Ordner angegebenen UNC-Pfads.

Standardmäßig ist diese Einstellung deaktiviert und die Umgebungsvariable %userdomain% ist nicht Teil der UNC-Pfad-Angabe für umgeleitete Ordner.

# Einstellungen der Richtlinie "AppData(Roaming)"

May 10, 2016

Der Abschnitt "AppData(Roaming)" enthält Richtlinieneinstellungen für die Angabe, ob der Inhalt des Ordners "AppData(Roaming)" an einen freigegebenen Speicherort im Netzwerk umgeleitet werden soll.

Mit dieser Einstellung geben Sie den Netzwerkspeicherort an, zu dem der Inhalt des Ordners "AppData(Roaming)" umgeleitet werden soll.

Standardmäßig ist diese Einstellung deaktiviert und kein Speicherort wird angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

Mit dieser Einstellung geben Sie an, wie der Inhalt des Ordners "AppData(Roaming)" umgeleitet werden soll.

Standardmäßig erfolgt die Umleitung an einen UNC-Pfad.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.



# Einstellungen der Richtlinie "Kontakte"

May 10, 2016

Der Abschnitt "Kontakte" enthält Richtlinieneinstellungen für die Angabe, ob der Inhalt des Ordners "Kontakte" an einen freigegebenen Speicherort im Netzwerk umgeleitet werden soll.

Mit dieser Einstellung geben Sie den Netzwerkspeicherort an, zu dem der Inhalt des Ordners "Kontakte" umgeleitet werden soll.

Standardmäßig ist diese Einstellung deaktiviert und kein Speicherort wird angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

Mit dieser Einstellung geben Sie an, wie der Inhalt des Ordners "Kontakte" umgeleitet werden soll.

Standardmäßig erfolgt die Umleitung an einen UNC-Pfad.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

# Einstellungen der Richtlinie "Desktop"

May 10, 2016

Der Abschnitt "Desktop" enthält Richtlinieneinstellungen für die Angabe, ob der Inhalt des Ordners "Desktop" an einen freigegebenen Speicherort im Netzwerk umgeleitet werden soll.

Mit dieser Einstellung geben Sie den Netzwerkspeicherort an, zu dem der Inhalt des Ordners "Desktop" umgeleitet werden soll.

Standardmäßig ist diese Einstellung deaktiviert und kein Speicherort wird angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

Mit dieser Einstellung geben Sie an, wie der Inhalt des Ordners "Desktop" umgeleitet werden soll.

Standardmäßig erfolgt die Umleitung an einen UNC-Pfad.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

# Einstellungen der Richtlinie "Dokumente"

May 10, 2016

Der Abschnitt "Dokumente" enthält Richtlinieneinstellungen für die Angabe, ob der Inhalt des Ordners "Dokumente" an einen freigegebenen Speicherort im Netzwerk umgeleitet werden soll.

Mit dieser Einstellung geben Sie den Netzwerkspeicherort an, zu dem die Dateien im Ordner "Dokumente" umgeleitet werden sollen.

Standardmäßig ist diese Einstellung deaktiviert und kein Speicherort wird angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

Die Einstellung 'Dokumente'-Pfad muss aktiviert sein, damit Dateien sowohl in den Ordner "Dokumente" als auch in die Ordner "Bilder", "Musik" und "Videos" umgeleitet werden.

Mit dieser Einstellung geben Sie an, wie der Inhalt des Ordners "Dokumente" umgeleitet werden soll.

Standardmäßig erfolgt die Umleitung an einen UNC-Pfad.

Wählen Sie eine der folgenden Optionen, um zu steuern, wie der Inhalt des Ordners "Dokumente" umgeleitet werden soll:

- Zum folgenden UNC-Pfad umleiten: leitet den Inhalt zu dem in der Richtlinieneinstellung 'Dokumente'-Pfad angegebenen UNC-Pfad.
- Zum Basisverzeichnis des Benutzers umleiten: leitet den Inhalt zu dem Basisverzeichnis des Benutzers. Dieses ist normalerweise mit dem Attribut #homeDirectory# für einen Benutzer in Active Directory konfiguriert.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

# Einstellungen der Richtlinie "Downloads"

May 10, 2016

Der Abschnitt "Downloads" enthält Richtlinieneinstellungen für die Angabe, ob der Inhalt des Ordners "Downloads" an einen freigegebenen Speicherort im Netzwerk umgeleitet werden soll.

Mit dieser Einstellung geben Sie den Netzwerkspeicherort an, zu dem die Dateien im Ordner "Downloads" umgeleitet werden.

Standardmäßig ist diese Einstellung deaktiviert und kein Speicherort wird angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

Mit dieser Einstellung geben Sie an, wie der Inhalt des Ordners "Downloads" umgeleitet werden soll.

Standardmäßig erfolgt die Umleitung an einen UNC-Pfad.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

# Einstellungen der Richtlinie "Favoriten"

May 10, 2016

Der Abschnitt "Favoriten" enthält Richtlinieneinstellungen für die Angabe, ob der Inhalt des Ordners "Favoriten" an einen freigegebenen Speicherort im Netzwerk umgeleitet werden soll.

Mit dieser Einstellung geben Sie die Netzwerkfreigabe an, zu der der Inhalt des Ordners "Favoriten" umgeleitet werden soll.

Standardmäßig ist diese Einstellung deaktiviert und kein Speicherort wird angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

Mit dieser Einstellung geben Sie an, wie der Inhalt des Ordners "Favoriten" umgeleitet werden soll.

Standardmäßig erfolgt die Umleitung an einen UNC-Pfad.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

# Einstellungen der Richtlinie "Links"

May 10, 2016

Der Abschnitt "Links" enthält Richtlinieneinstellungen für die Angabe, ob der Inhalt des Ordners "Links" an einen freigegebenen Speicherort im Netzwerk umgeleitet werden soll.

Mit dieser Einstellung geben Sie die Netzwerkfreigabe an, zu der der Inhalt des Ordners "Links" umgeleitet werden soll.

Standardmäßig ist diese Einstellung deaktiviert und kein Speicherort wird angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

Mit dieser Einstellung geben Sie an, wie der Inhalt des Ordners "Links" umgeleitet werden soll.

Standardmäßig erfolgt die Umleitung an einen UNC-Pfad.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

# Einstellungen der Richtlinie "Musik"

May 10, 2016

Der Abschnitt "Musik" enthält Richtlinieneinstellungen für die Angabe, ob der Inhalt des Ordners "Musik" an einen freigegebenen Speicherort im Netzwerk umgeleitet werden soll.

Mit dieser Einstellung geben Sie die Netzwerkfreigabe an, zu der der Inhalt des Ordners "Musik" umgeleitet werden soll.

Standardmäßig ist diese Einstellung deaktiviert und kein Speicherort wird angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

Mit dieser Einstellung geben Sie an, wie der Inhalt des Ordners "Musik" umgeleitet werden soll.

Standardmäßig erfolgt die Umleitung an einen UNC-Pfad.

Wählen Sie eine der folgenden Optionen, um zu steuern, wie der Inhalt des Ordners "Musik" umgeleitet werden soll:

- Zum folgenden UNC-Pfad umleiten: Leitet den Inhalt zu dem in der Richtlinieneinstellung 'Musik'-Pfad angegebenen UNC-Pfad.
- Relativ zum Ordner 'Dokumente' umleiten: Leitet den Inhalt in einen Ordner relativ zu dem Ordner "Dokumente" um.

Damit Inhalte in einen Ordner relativ zum Ordner "Dokumente" umgeleitet werden, muss die Einstellung 'Dokumente'-Pfad aktiviert sein.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

# Einstellungen der Richtlinie "Bilder"

May 10, 2016

Der Abschnitt "Bilder" enthält Richtlinieneinstellungen für die Angabe, ob der Inhalt des Ordners "Bilder" an einen freigegebenen Speicherort im Netzwerk umgeleitet werden soll.

Mit dieser Einstellung geben Sie die Netzwerkfreigabe an, zu der der Inhalt des Ordners "Bilder" umgeleitet werden soll.

Standardmäßig ist diese Einstellung deaktiviert und kein Speicherort wird angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

Mit dieser Einstellung geben Sie an, wie der Inhalt des Ordners "Bilder" umgeleitet werden soll.

Standardmäßig erfolgt die Umleitung an einen UNC-Pfad.

Wählen Sie eine der folgenden Optionen, um zu steuern, wie der Inhalt des Ordners "Bilder" umgeleitet werden soll:

- Zum folgenden UNC-Pfad umleiten: Leitet den Inhalt zu dem in der Richtlinieneinstellung 'Bilder'-Pfad angegebenen UNC-Pfad.
- Relativ zum Ordner 'Dokumente' umleiten: Leitet den Inhalt in einen Ordner relativ zu dem Ordner "Dokumente" um.

Damit Inhalte in einen Ordner relativ zum Ordner "Dokumente" umgeleitet werden, muss die Einstellung 'Dokumente'-Pfad aktiviert sein.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.



# Einstellungen der Richtlinie "Gespeicherte Spiele"

May 10, 2016

Der Abschnitt "Gespeicherte Spiele" enthält Richtlinieneinstellungen für die Angabe, ob der Inhalt des Ordners "Gespeicherte Spiele" an einen freigegebenen Speicherort im Netzwerk umgeleitet werden soll.

## Umleitungseinstellungen für 'Gespeicherte Spiele'

Mit dieser Einstellung geben Sie an, wie der Inhalt des Ordners "Gespeicherte Spiele" umgeleitet werden soll.

Standardmäßig erfolgt die Umleitung an einen UNC-Pfad.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

## 'Gespeicherte Spiele'-Pfad

Mit dieser Einstellung geben Sie die Netzwerkfreigabe an, zu der der Inhalt des Ordners "Gespeicherte Spiele" umgeleitet werden soll.

Standardmäßig ist diese Einstellung deaktiviert und kein Speicherort wird angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

# Einstellungen der Richtlinie "Suchen"

May 10, 2016

Der Abschnitt "Suchen" enthält Richtlinieneinstellungen für die Angabe, ob der Inhalt des Ordners "Suchen" an einen freigegebenen Speicherort im Netzwerk umgeleitet werden soll.

## Umleitungseinstellungen für 'Suchen'

Mit dieser Einstellung geben Sie an, wie der Inhalt des Ordners "Suchen" umgeleitet werden soll.

Standardmäßig erfolgt die Umleitung an einen UNC-Pfad.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

## 'Suchen'-Pfad

Mit dieser Einstellung geben Sie die Netzwerkfreigabe an, zu der der Inhalt des Ordners "Suchen" umgeleitet werden soll.

Standardmäßig ist diese Einstellung deaktiviert und kein Speicherort wird angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

# Einstellungen der Richtlinie "Startmenü"

May 10, 2016

Der Abschnitt "Startmenü" enthält Richtlinieneinstellungen für die Angabe, ob der Inhalt des Ordners "Startmenü" an einen freigegebenen Speicherort im Netzwerk umgeleitet werden soll.

## Umleitungseinstellungen für 'Startmenü'

Mit dieser Einstellung geben Sie an, wie der Inhalt des Ordners "Startmenü" umgeleitet werden soll.

Standardmäßig erfolgt die Umleitung an einen UNC-Pfad.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

## Startmenü-Pfad

Mit dieser Einstellung geben Sie die Netzwerkfreigabe an, zu der der Inhalt des Ordners "Startmenü" umgeleitet werden soll.

Standardmäßig ist diese Einstellung deaktiviert und kein Speicherort wird angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

# Einstellungen der Richtlinie "Videos"

May 10, 2016

Der Abschnitt "Videos" enthält Richtlinieneinstellungen für die Angabe, ob der Inhalt des Ordners "Videos" an einen freigegebenen Speicherort im Netzwerk umgeleitet werden soll.

## Umleitungseinstellungen für 'Videos'

Mit dieser Einstellung geben Sie an, wie der Inhalt des Ordners "Videos" umgeleitet werden soll.

Standardmäßig erfolgt die Umleitung an einen UNC-Pfad.

Wählen Sie eine der folgenden Optionen, um zu steuern, wie der Inhalt des Ordners "Videos" umgeleitet werden soll:

- Zum folgenden UNC-Pfad umleiten: Leitet den Inhalt zu dem in der Richtlinieneinstellung 'Videos'-Pfad angegebenen UNC-Pfad.
- Relativ zum Ordner 'Dokumente' umleiten: Leitet den Inhalt in einen Ordner relativ zu dem Ordner "Dokumente" um.

Damit Inhalte in einen Ordner relativ zum Ordner "Dokumente" umgeleitet werden, muss die Einstellung 'Dokumente'-Pfad aktiviert sein.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

## 'Videos'-Pfad

Mit dieser Einstellung geben Sie die Netzwerkfreigabe an, zu der der Inhalt des Ordners "Videos" umgeleitet werden soll.

Standardmäßig ist diese Einstellung deaktiviert und kein Speicherort wird angegeben.

Wenn diese Einstellung hier nicht konfiguriert ist, erfolgt keine Umleitung durch die Profilverwaltung.

# Einstellungen der Richtlinie "Protokollierung"

May 10, 2016

Der Abschnitt "Protokollierung" enthält Richtlinieneinstellungen zum Konfigurieren der Protokollierung der Profilverwaltung.

## Active Directory-Aktionen

Mit dieser Einstellung aktivieren oder deaktivieren Sie die ausführliche Protokollierung der in Active Directory ausgeführten Aktionen.

Diese Einstellung ist standardmäßig deaktiviert.

Wenn Sie diese Einstellung aktivieren, stellen Sie sicher, dass die Einstellung Protokollierung aktivieren auch aktiviert ist.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden Fehler und allgemeine Informationen protokolliert.

## Allgemeine Informationen

Mit dieser Einstellung aktivieren oder deaktivieren Sie die ausführliche Protokollierung allgemeiner Informationen.

Diese Einstellung ist standardmäßig deaktiviert.

Wenn Sie diese Einstellung aktivieren, stellen Sie sicher, dass die Einstellung Protokollierung aktivieren auch aktiviert ist.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden Fehler und allgemeine Informationen protokolliert.

## Allgemeine Warnungen

Mit dieser Einstellung aktivieren oder deaktivieren Sie die ausführliche Protokollierung allgemeiner Warnungen.

Diese Einstellung ist standardmäßig deaktiviert.

Wenn Sie diese Einstellung aktivieren, stellen Sie sicher, dass die Einstellung Protokollierung aktivieren auch aktiviert ist.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden Fehler und allgemeine Informationen protokolliert.

## Protokollierung aktivieren

Mit dieser Einstellung aktivieren oder deaktivieren Sie die Protokollierung der Profilverwaltung im Debugmodus (ausführliche Protokollierung). Im Debugmodus werden umfangreiche Statusinformationen in den Protokolldateien unter "%SystemRoot%\System32\Logfiles\UserProfileManager" aufgezeichnet.

Standardmäßig ist diese Einstellung deaktiviert und es werden nur Fehler protokolliert.

Citrix empfiehlt, dass Sie diese Einstellung nur aktivieren, wenn Sie eine Problembehandlung für die Profilverwaltung durchführen.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden nur Fehler protokolliert.

### Dateisystemaktionen

Mit dieser Einstellung aktivieren oder deaktivieren Sie die ausführliche Protokollierung der im Dateisystem ausgeführten Aktionen.

Diese Einstellung ist standardmäßig deaktiviert.

Wenn Sie diese Einstellung aktivieren, stellen Sie sicher, dass die Einstellung Protokollierung aktivieren auch aktiviert ist.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden Fehler und allgemeine Informationen protokolliert.

### Dateisystembenachrichtigungen

Mit dieser Einstellung aktivieren oder deaktivieren Sie die ausführliche Protokollierung von Dateisystembenachrichtigungen.

Diese Einstellung ist standardmäßig deaktiviert.

Wenn Sie diese Einstellung aktivieren, stellen Sie sicher, dass die Einstellung Protokollierung aktivieren auch aktiviert ist.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden Fehler und allgemeine Informationen protokolliert.

### Abmeldung

Mit dieser Einstellung aktivieren oder deaktivieren Sie die ausführliche Protokollierung von Benutzerabmeldungen.

Diese Einstellung ist standardmäßig deaktiviert.

Wenn Sie diese Einstellung aktivieren, stellen Sie sicher, dass die Einstellung Protokollierung aktivieren auch aktiviert ist.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden Fehler und allgemeine Informationen protokolliert.

### Anmeldung

Mit dieser Einstellung aktivieren oder deaktivieren Sie die ausführliche Protokollierung von Benutzeranmeldungen.

Diese Einstellung ist standardmäßig deaktiviert.

Wenn Sie diese Einstellung aktivieren, stellen Sie sicher, dass die Einstellung Protokollierung aktivieren auch aktiviert ist.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden Fehler und allgemeine Informationen protokolliert.

### Max. Größe der Protokolldatei

Mit dieser Einstellung geben Sie die maximal zulässige Größe für die Protokolldatei der Profilverwaltung in Bytes an.

Der Standardwert dieser Einstellung ist 1048576 Bytes (1 MB).

Citrix empfiehlt, dass die Größe dieser Datei auf 5 MB oder mehr erhöht wird, sofern Sie ausreichend Speicherplatz auf dem Datenträger haben. Wenn die Protokolldatei die maximale Größe überschreitet, wird eine vorhandene Sicherungskopie der Datei (.bak) gelöscht, die Protokolldatei erhält die Erweiterung .bak und eine neue Protokolldatei wird erstellt.

Die Protokolldatei wird unter "%SystemRoot%\System32\Logfiles\UserProfileManager" erstellt.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, wird der Standardwert verwendet.

### Pfad zur Protokolldatei

Mit dieser Einstellung geben Sie einen alternativen Pfad an, der zum Speichern der Protokolldatei der Profilverwaltung verwendet wird.

Standardmäßig ist diese Einstellung deaktiviert und Protokolldateien werden im Standardspeicherort %SystemRoot%\System32\Logfiles\UserProfileManager gespeichert.

Der Pfad kann zu einem lokalen Laufwerk oder einem Remotelaufwerk im Netzwerk (UNC-Pfad) führen. Remotepfade können in großen, verteilten Umgebungen von Nutzen sein, können aber zu erheblichem Netzwerkverkehr führen, was für Protokolldateien unangebracht ist. Geben Sie für bereitgestellte virtuelle Maschinen mit einer beständigen Festplatte einen lokalen Pfad zu diesem Laufwerk an. Hierdurch wird sichergestellt, dass Protokolldateien beim Neustart der Maschine beibehalten werden. Geben Sie für virtuelle Maschinen ohne eine beständige Festplatte einen UNC-Pfad an. So werden Protokolldateien beibehalten. Das Systemkonto für die Maschinen muss aber Schreibzugriff auf die UNC-Freigabe haben. Verwenden Sie für Laptops, die vom Feature für Offlineprofile verwaltet werden, einen lokalen Pfad.

Wenn für Protokolldateien ein UNC-Pfad verwendet wird, empfiehlt Citrix, entsprechende Zugriffssteuerungslisten auf den Ordner mit den Protokolldateien anzuwenden, um sicherzustellen, dass nur autorisierte Benutzer- oder Computerkonten auf die gespeicherten Dateien zugreifen können.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, wird der Standardspeicherort "%SystemRoot%\System32\Logfiles\UserProfileManager" verwendet.

### Persönliche Benutzerinformationen

Mit dieser Einstellung aktivieren oder deaktivieren Sie die ausführliche Protokollierung persönlicher Benutzerinformationen.

Diese Einstellung ist standardmäßig deaktiviert.

Wenn Sie diese Einstellung aktivieren, stellen Sie sicher, dass die Einstellung Protokollierung aktivieren auch aktiviert ist.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden Fehler und allgemeine Informationen protokolliert.

### Richtlinienwerte bei Anmeldung und Abmeldung

Mit dieser Einstellung aktivieren oder deaktivieren Sie ausführliche Protokollierung der Richtlinienwerte beim An- und Abmelden von Benutzern.

Diese Einstellung ist standardmäßig deaktiviert.

Wenn Sie diese Einstellung aktivieren, stellen Sie sicher, dass die Einstellung Protokollierung aktivieren auch aktiviert ist.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden Fehler und allgemeine Informationen protokolliert.

### Registrierungsaktionen

Mit dieser Einstellung aktivieren oder deaktivieren Sie die ausführliche Protokollierung der in der Registrierung ausgeführten Aktionen.

Diese Einstellung ist standardmäßig deaktiviert.

Wenn Sie diese Einstellung aktivieren, stellen Sie sicher, dass die Einstellung Protokollierung aktivieren auch aktiviert ist.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden Fehler und allgemeine Informationen protokolliert.

### Registrierungsunterschiede bei der Abmeldung

Mit dieser Einstellung aktivieren oder deaktivieren Sie ausführliche Protokollierung aller Registrierungsunterschiede bei der Abmeldung von Benutzern.

Diese Einstellung ist standardmäßig deaktiviert.

Wenn Sie diese Einstellung aktivieren, stellen Sie sicher, dass die Einstellung Protokollierung aktivieren auch aktiviert ist.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden Fehler und allgemeine Informationen protokolliert.



# Einstellungen der Richtlinie "Profilverarbeitung"

May 10, 2016

Der Abschnitt "Profilverarbeitung" enthält Richtlinieneinstellungen zum Konfigurieren, wie die Profilverwaltung Benutzerprofile verarbeitet.

## Verzögerung vor dem Löschen von zwischengespeicherten Profilen

Mit dieser Einstellung geben Sie optional eine Verlängerung für die Verzögerung (in Minuten) ein, nach der die Profilverwaltung lokal zwischengespeicherte Profile bei der Abmeldung löscht.

Bei einem Wert von 0 werden die Profile am Ende der Abmeldung sofort gelöscht. Die Profilverwaltung prüft jede Minute auf Abmeldungen, sodass ein Wert von 60 sicherstellt, dass Profile zwischen einer und zwei Minuten (je nachdem, wann die letzte Überprüfung stattgefunden hat) nach dem Abmelden gelöscht werden. Das Erweitern der Verzögerung ist nützlich, wenn Sie wissen, dass ein Prozess Dateien oder die Registrierungsstruktur während der Abmeldung geöffnet hält. Bei großen Profilen kann dies auch den Abmeldungsprozess beschleunigen.

Die Standardeinstellung ist 0, lokal zwischengespeicherte Profile werden von der Profilverwaltung sofort gelöscht.

Wenn Sie diese Einstellung aktivieren, müssen Sie sicherstellen, dass die Einstellung Lokal zwischengespeicherte Profile nach Abmeldung löschen auch aktiviert ist.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden die Profile sofort gelöscht.

## Lokal zwischengespeicherte Profile nach Abmeldung löschen

Mit dieser Einstellung geben Sie an, ob lokal zwischengespeicherte Profile gelöscht werden, nachdem Benutzer sich abmelden.

Wenn diese Einstellung aktiviert ist, wird der lokale Profilcache der Benutzer nach der Abmeldung gelöscht. Citrix empfiehlt, dass Sie diese Einstellung für Terminalserver aktivieren.

Standardmäßig ist diese Einstellung deaktiviert und der lokale Profilcache von Benutzern wird nach der Abmeldung beibehalten.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden zwischengespeicherte Profile nicht gelöscht.

## Behandlung von Konflikten lokaler Profile

Mit dieser Einstellung wird konfiguriert, wie die Profilverwaltung verfährt, wenn ein Benutzerprofil sowohl im Benutzerspeicher als auch als lokales Windows-Benutzerprofil (kein Citrix Benutzerprofil) vorhanden ist.

Standardmäßig verwendet die Profilverwaltung lokale Windows-Profile, ohne diese jedoch zu ändern.

Zum Steuern, wie die Profilverwaltung verfahren soll, wählen Sie eine der folgenden Optionen:

- Lokales Profil verwenden: Die Profilverwaltung verwendet lokale Windows-Profile, ohne diese jedoch zu ändern.
- Lokales Profil löschen: Die Profilverwaltung löscht das lokale Windows-Benutzerprofil und importiert dann das Citrix

Benutzerprofil aus dem Benutzerspeicher.

- Lokales Profil umbenennen: Die Profilverwaltung benennt das lokale Windows-Benutzerprofil um (als Sicherungskopie) und importiert dann das Citrix Benutzerprofil aus dem Benutzerspeicher.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden vorhandene lokale Profile verwendet.

## Migration vorhandener Profile

Mit dieser Einstellung geben Sie den Typ des Profils an, das bei der Anmeldung eines Benutzers in den Benutzerspeicher migriert wird, wenn der Speicher kein aktuelles Profil für den Benutzer enthält.

Die Profilverwaltung kann vorhandene Profile während der Anmeldung spontan migrieren, wenn der Benutzer kein Profil im Benutzerspeicher hat. Anschließend verwendet die Profilverwaltung das Profil im Benutzerspeicher in der aktuellen Sitzung und in allen künftigen Sitzungen, die mit dem Pfad zu demselben Benutzerspeicher konfiguriert sind.

Standardmäßig werden lokale Profile und Roamingprofile während der Anmeldung in den Benutzerspeicher migriert.

Um anzugeben welche Profiltypen bei der Anmeldung in den Benutzerspeicher migriert werden sollen, wählen Sie eine der folgenden Optionen:

- Lokal und Roaming
- Lokal
- Roaming
- Keine (deaktiviert)

Wenn Sie Keine auswählen, wird der vorhandene Windows-Mechanismus für die Erstellung neuer Profile verwendet, genau wie in einer Umgebung, in der die Profilverwaltung nicht installiert ist.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden vorhandene lokale und servergespeicherte Profile migriert.

## Pfad zum Vorlagenprofil

Mit dieser Einstellung geben Sie den Pfad zu dem Profil an, das die Profilverwaltung als Vorlage zum Erstellen neuer Benutzerprofile verwenden soll.

Dies muss der vollständige Pfad zu dem Ordner sein, der die Registrierungsdatei NTUSER.DAT und sämtliche anderen für das Vorlagenprofil erforderlichen Dateien und Ordner enthält.

Hinweis: Geben Sie mit dem Pfad nicht NTUSER.DAT ein. Geben Sie für die Datei \\Server\Profile\Vorlage\ntuser.dat den Speicherort als \\Server\Profile\Vorlage an.

Verwenden Sie einen absoluten Pfad (entweder einen UNC-Pfad oder einen Pfad auf dem lokalen Computer). Sie können einen lokalen Pfad verwenden, um z. B. ein Vorlagenprofil auf einem Citrix Provisioning Services-Image dauerhaft anzugeben. Relative Pfade werden nicht unterstützt.

Hinweis: Beachten Sie, dass diese Richtlinie nicht die Erweiterung von Active Directory-Attributen, Systemumgebungsvariablen oder der Variablen %USERNAME% und %USERDOMAIN% unterstützt.

Standardmäßig ist diese Einstellung deaktiviert und neue Benutzerprofile werden auf der Basis des Standardbenutzerprofils auf dem Gerät, auf dem sich der Benutzer als erstes anmeldet, erstellt.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden keine Vorlagen verwendet.

### Vorlagenprofil überschreibt lokales Profil

Diese Einstellung ermöglicht eine Überschreibung des lokalen Profils durch das Vorlagenprofil bei der Erstellung neuer Benutzerprofile.

Wenn ein Benutzer kein Citrix Benutzerprofil hat, aber ein lokales Windows-Benutzerprofil vorhanden ist, wird standardmäßig das lokale Profil verwendet (und in den Benutzerspeicher migriert, wenn diese Option nicht deaktiviert ist). Durch Aktivieren dieser Einstellung kann das Vorlagenprofil das lokale Profil bei der Erstellung neuer Benutzerprofile überschreiben.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden keine Vorlagen verwendet.

### Vorlagenprofil überschreibt Roamingprofil

Diese Einstellung ermöglicht eine Überschreibung eines Roamingprofils durch das Vorlagenprofil bei der Erstellung neuer Benutzerprofile.

Wenn ein Benutzer kein Citrix Benutzerprofil hat aber ein lokales Windows-Benutzerprofil vorhanden ist, wird standardmäßig das Roamingprofil verwendet (und in den Benutzerspeicher migriert, wenn diese Option nicht deaktiviert ist). Durch Aktivieren dieser Einstellung kann das Vorlagenprofil das Roamingprofil bei der Erstellung neuer Benutzerprofile überschreiben.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden keine Vorlagen verwendet.

### Als verbindliches Citrix Profil für alle Anmeldungen verwendete Vorlagenprofil

Bei Auswahl dieser Einstellung verwendet die Profilverwaltung das Vorlagenprofil als Standardprofil bei der Erstellung aller neuen Benutzerprofile.

Standardmäßig ist diese Einstellung deaktiviert und neue Benutzerprofile werden auf der Basis des Standardbenutzerprofils auf dem Gerät, auf dem sich der Benutzer als erstes anmeldet, erstellt.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden keine Vorlagen verwendet.

# Einstellungen der Richtlinie "Registrierung"

May 10, 2016

Der Abschnitt "Registrierung" enthält Richtlinieneinstellungen, mit denen Sie festlegen können, welche Registrierungsschlüssel bei der Verarbeitung der Profilverwaltung berücksichtigt und welche ausgeschlossen werden sollen.

## Ausschlussliste

Mit dieser Einstellung geben Sie die Liste der Registrierungsschlüssel in der HKCU-Struktur an, die nicht von der Profilverwaltung verarbeitet werden sollen, wenn sich ein Benutzer abmeldet.

Wenn diese Option aktiviert ist, werden die in dieser Liste aufgeführten Schlüssel von der Verarbeitung ausgeschlossen, wenn sich ein Benutzer abmeldet.

Standardmäßig ist diese Einstellung deaktiviert, und alle Registrierungsschlüssel in der HKCU-Struktur werden verarbeitet.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden keine Registrierungsschlüssel von der Verarbeitung ausgeschlossen.

## Aufnahmeliste

Mit dieser Einstellung geben Sie die Liste der Registrierungsschlüssel in der HKCU-Struktur an, die von der Profilverwaltung verarbeitet werden sollen, wenn sich ein Benutzer abmeldet.

Wenn diese Option aktiviert ist, werden die in dieser Liste aufgeführten Schlüssel in die Verarbeitung eingeschlossen, wenn sich ein Benutzer abmeldet.

Standardmäßig ist diese Einstellung deaktiviert, und alle Registrierungsschlüssel in der HKCU-Struktur werden verarbeitet.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, wird die gesamte HKCU-Struktur verarbeitet.

# Einstellungen der Richtlinie "Gestreamte Benutzerprofile"

May 10, 2016

Der Abschnitt "Gestreamte Benutzerprofile" enthält Richtlinieneinstellungen zum Konfigurieren, wie die Profilverwaltung Benutzerprofile verarbeitet.

## Immer zwischenspeichern

Mit dieser Einstellung geben Sie an, ob die Profilverwaltung gestreamte Dateien so bald wie möglich zwischenspeichern soll, wenn sich ein Benutzer anmeldet. Durch das Zwischenspeichern von Dateien, nachdem sich ein Benutzer anmeldet, wird Netzwerkbandbreite gespart und die Benutzererfahrung optimiert.

Verwenden Sie diese Einstellung mit der Einstellung Profilstreaming.

Standardmäßig ist diese Einstellung deaktiviert und gestreamte Dateien werden nicht so schnell wie möglich zwischengespeichert, wenn sich ein Benutzer anmeldet.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, ist sie deaktiviert.

## Immer Cachegröße

Mit dieser Einstellung geben Sie eine Untergrenze (in Megabytes) für die Größe der Dateien an, die gestreamt werden. Die Profilverwaltung speichert Dateien dieser Größe bzw. größere Dateien so bald wie möglich zwischen, wenn ein Benutzer sich anmeldet.

Die Standardeinstellung ist 0 (null) und die Funktion zum Zwischenspeichern des gesamten Profils wird verwendet. Wenn das Feature zum Zwischenspeichern des gesamten Profils aktiviert ist, ruft die Profilverwaltung den gesamten Inhalt des Profils im Benutzerspeicher als Hintergrundaufgabe ab, nachdem sich ein Benutzer anmeldet.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, ist sie deaktiviert.

## Profilstreaming

Diese Einstellung aktiviert oder deaktiviert das Feature für gestreamte Citrix Benutzerprofile. Wenn diese Option aktiviert ist, werden Dateien und Ordner in einem Profil nur dann aus dem Benutzerspeicher auf den lokalen Computer abgerufen, wenn auf sie von Benutzern nach der Anmeldung zugegriffen wird. Registrierungseinträge und Dateien im Bereich für ausstehende Dateien werden sofort abgerufen.

Standardmäßig ist das Profilstreaming deaktiviert.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, ist sie deaktiviert.

## Gruppen mit gestreamten Benutzerprofilen

Mit dieser Einstellung geben Sie die Benutzerprofile in einer Organisationseinheit gestreamt werden, basierend auf Windows Benutzergruppen.

Wenn diese Option aktiviert ist, werden nur die Benutzerprofile in den angegebenen Benutzergruppen gestreamt. Alle anderen Benutzerprofile werden normal verarbeitet.

Standardmäßig ist diese Einstellung deaktiviert und alle Dateien in Benutzerprofilen werden normal verarbeitet.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, werden alle Benutzergruppen verarbeitet.

### Timeout für gesperrte Dateien im ausstehenden Bereich

Mit dieser Einstellung geben Sie einen Zeitraum (in Tagen) an, nach dem Dateien der Benutzer aus dem Bereich für ausstehende Dateien in den Benutzerspeicher zurückgeschrieben werden, wenn ein Server nicht mehr reagiert und der Benutzerspeicher gesperrt bleibt. Dies verhindert ein Aufblähen des ausstehenden Bereichs und stellt sicher, dass der Benutzerstore immer die aktuellen Dateien enthält.

Die Standardeinstellung ist 1 Tag.

Wenn diese Einstellung hier nicht konfiguriert ist, wird der Wert in der INI-Datei verwendet.

Wenn diese Einstellung weder hier noch in der INI-Datei konfiguriert ist, wird der Standardwert verwendet.

# Einstellungen der Richtlinie "Receiver"

May 10, 2016

Der Abschnitt "Receiver" enthält Richtlinieneinstellungen zum Angeben einer Liste von StoreFront-Adressen, die für auf dem virtuellen Desktop ausgeführte Windows-Betriebssysteme per Push an Receiver für Windows übertragen werden sollen.

## StoreFront-Kontenliste

Mit dieser Einstellung geben Sie eine Liste der StoreFront-Stores an, die Administratoren für die Übertragung per Push an Receiver für Windows auswählen können. Beim Erstellen einer Bereitstellungsgruppe können Administratoren auswählen, welche Stores für Receiver für Windows auf virtuellen Desktops, der in dieser Gruppe ausgeführt werden, per Push übertragen werden sollen.

In der Standardeinstellung sind keine Stores angegeben.

Geben Sie für jeden Store die folgenden Informationen in Form eines durch Semikolons getrennten Eintrags an:

- Storename: der Name des Stores wie er Benutzern angezeigt wird.
- Store-URL: die URL des Stores.
- Storeaktivierungszustand: Ob der Store für Benutzer verfügbar ist. Dies ist entweder ein oder aus.
- Store-Beschreibung: die Beschreibung des Stores, die Benutzern angezeigt wird.

Beispiel: Vertriebsstore;https://sales.mycompany.com/Citrix/Store/discovery;On;Store für Vertriebsmitarbeiter

# Einstellungen der Richtlinie "Virtual Delivery Agent"

May 10, 2016

Der Abschnitt "Virtual Delivery Agent" (VDA) enthält Richtlinieneinstellungen, mit denen Sie die Kommunikation zwischen VDA und Controllern einer Site steuern können.

Wichtig: Der VDA benötigt die in diesen Einstellungen enthaltenen Informationen für die Registrierung bei einem Delivery Controller, wenn das Feature für automatische Controllerupdates nicht verwendet wird. Da die Informationen für die Registrierung erforderlich sind, müssen Sie sie mit dem Gruppenrichtlinien-Editor konfigurieren, sofern Sie sie nicht bei der VDA-Installation angeben.

- IPv6-Netzwerkmaske für Controllerregistrierung
- Controllerregistrierungsport
- Controller-SIDs
- Controller
- Nur IPv6-Controllerregistrierung verwenden
- Site-GUID

## IPv6-Netzwerkmaske für Controllerregistrierung

Mit dieser Richtlinieneinstellung kann der VDA auf ein bevorzugtes Subnetz (anstelle einer globalen IP, sofern registriert) limitiert werden. Mit dieser Einstellung geben Sie die IPv6-Adresse und das Netzwerk an, in dem der VDA registriert wird. Der VDA wird nur an der ersten Adresse registriert, die mit der angegebenen Netzmaske übereinstimmt. Diese Einstellung ist nur gültig, wenn die Richtlinieneinstellung Nur IPv6-Controllerregistrierung verwenden aktiviert ist.

Diese Einstellung ist standardmäßig leer.

## Controllerregistrierungsport

Verwenden Sie diese Einstellung nur, wenn die Einstellung Automatische Controllerupdates aktivieren deaktiviert ist.

Mit dieser Einstellung geben Sie die TCP/IP-Portnummer an, die der VDA für die Registrierung bei einem Controller verwendet, wenn die registrierungsbasierte Registrierung verwendet wird.

Die Standardeinstellung der Portnummer ist "80".

## Controller-SIDs

Verwenden Sie diese Einstellung nur, wenn die Einstellung Automatische Controllerupdates aktivieren deaktiviert ist.

Mit dieser Einstellung geben Sie eine durch Leerzeichen getrennte Liste von Controller-SIDs an, die der VDA für die Registrierung bei einem Controller verwendet, wenn die registrierungsbasierte Registrierung verwendet wird. Dies ist eine optionale Einstellung, die mit der Einstellung "Controller" verwendet werden kann, um die für die Registrierung verwendete Liste von Controllern zu beschränken.

Diese Einstellung ist standardmäßig leer.

## Controller

Verwenden Sie diese Einstellung nur, wenn die Einstellung Automatische Controllerupdates aktivieren deaktiviert ist.

Mit dieser Einstellung geben Sie eine durch Leerzeichen getrennte Liste von vollständig qualifizierten Domännennamen



(FQDN) für Controller an, die der VDA für die Registrierung bei einem Controller verwendet, wenn die registrierungsbasierte Registrierung verwendet wird. Dies ist eine optionale Einstellung, die mit der Einstellung "Controller-SIDs" verwendet werden kann.

Diese Einstellung ist standardmäßig leer.

### Automatische Controllerupdates aktivieren

Mit dieser Einstellung ist eine automatische Registrierung des VDAs bei einem Controller nach der Installation möglich.

Nach der Registrierung wird von dem Controller, bei dem der VDA registriert ist, eine Liste der aktuellen Controller-FQDNs und -SIDs an den VDA gesendet. Diese Liste wird in den persistenten Speicher des VDAs geschrieben. Jeder Controller prüft die Datenbank alle 90 Minuten auf Controllerinformationen. Wurde seit der letzten Prüfung ein Controller hinzugefügt oder entfernt oder ist eine Richtlinienänderung erfolgt, sendet der Controller eine aktualisierte Liste an die bei ihm registrierten VDAs. Der VDA nimmt alle Verbindungen von allen Controllern in der aktuellen Liste an.

Standardmäßig ist diese Einstellung aktiviert.

### Nur IPv6-Controllerregistrierung verwenden

Diese Einstellung steuert die Form der Adresse, die vom VDA für die Registrierung beim Controller verwendet wird:

- Bei aktivierter Einstellung wird der VDA mit der IPv6-Adresse der Maschine für die Kommunikation beim Controller registriert. Wenn der VDA mit dem Controller kommuniziert, wird eine Adresse verwendet, deren Auswahl folgender Reihenfolge unterliegt: globale IP-Adresse, ULA-Adresse, Link-Local-Adresse (wenn keine anderen IPv6-Adressen verfügbar sind).
- Bei deaktivierter Einstellung wird der VDA mit der IPv4-Adresse der Maschine für die Kommunikation beim Controller registriert.

Diese Einstellung ist standardmäßig deaktiviert.

### Site-GUID

Verwenden Sie diese Einstellung nur, wenn die Einstellung Automatische Controllerupdates aktivieren deaktiviert ist.

Diese Einstellung gibt den Globally Unique Identifier (GUID) der Site an, den der VDA für die Registrierung bei einem Controller verwendet, wenn die Active Directory-basierte Registrierung verwendet wird.

Diese Einstellung ist standardmäßig leer.

# Einstellungen der Richtlinie "HDX 3D Pro"

May 10, 2016

Der Bereich "HDX 3D Pro" enthält Richtlinieneinstellungen, mit denen Sie das Tool zum Konfigurieren der Bildqualität für Benutzer aktivieren und konfigurieren können. Mit dem Tool können Benutzer die Verwendung der verfügbaren Bandbreite durch Anpassen des Verhältnisses zwischen Bildqualität und Reaktionszeit in Echtzeit optimieren.

## Verlustfrei aktivieren

Mit dieser Einstellung wird angegeben, ob Benutzer verlustfreie Komprimierung mit dem Tool zum Konfigurieren der Bildqualität aktivieren oder deaktivieren können. In der Standardeinstellung wird den Benutzern die Möglichkeit zum Aktivieren der verlustfreien Komprimierung nicht eingeräumt.

Aktiviert ein Benutzer die verlustfreie Komprimierung, wird die Bildqualität automatisch auf den höchsten Wert eingestellt, der im Bildkonfigurationstool verfügbar ist. Standardmäßig kann je nach Leistungsfähigkeit des Benutzergeräts und des Hostcomputers entweder die GPU-basierte oder die CPU-basierte Komprimierung verwendet werden.

## HDX3DPro-Qualitätseinstellungen

Mit dieser Einstellung geben Sie den Mindest- und den Höchstwert an, mit denen der Bildqualitätsanpassungsbereich, der den Benutzern im Tool zum Konfigurieren der Bildqualität zur Verfügung steht, festgelegt wird.

Geben Sie für die Bildqualität Werte zwischen 0 und 100 an. Der Höchstwert muss größer oder gleich dem Mindestwert sein.

# Einstellungen der Richtlinie "Virtuelle IP"

May 10, 2016

Der Abschnitt "Virtuelle IP" enthält Richtlinieneinstellungen für die Angabe, ob Sitzungen eine eigene virtuelle Loopbackadresse haben.

## Virtuelle IP - Loopbackunterstützung

Wenn diese Einstellung aktiviert ist, hat jede Sitzung eine eigene virtuelle Loopbackadresse. Wenn diese Einstellung deaktiviert ist, haben Sitzungen keine individuellen Loopbackadressen.

Diese Einstellung ist standardmäßig deaktiviert.

## Virtuelle IP - Programme für virtuelles Loopback

Mit dieser Einstellung geben Sie die ausführbaren Dateien der Anwendungen an, die virtuelle Loopbackadressen verwenden können. Wenn Sie der Liste Programme hinzufügen, geben Sie nur den Namen der ausführbaren Datei an. Sie müssen nicht den gesamten Pfad angeben.

In der Standardeinstellung sind keine ausführbaren Dateien angegeben.

# Konfigurieren von COM-Port- und LPT-Portumleitungseinstellungen in der Registrierung

May 10, 2016

Richtlinieneinstellungen für COM-Port- und LPT-Portumleitung befinden sich unter HKLM\Software\Citrix\GroupPolicy\Defaults\Deprecated auf dem VDA-Image oder Computer.

Zum Aktivieren der COM-Port- und LPT-Portumleitung, fügen Sie neue Registrierungsschlüssel vom Typ REG\_DWORD wie folgt hinzu:

Achtung: Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

| Registrierungsschlüssel   | Beschreibung                                                                              | Zulässige Werte                     |
|---------------------------|-------------------------------------------------------------------------------------------|-------------------------------------|
| AllowComPortRedirection   | Zulassen oder Verhindern der COM-Portumleitung                                            | 1 (Zulassen) oder 0 (Verhindern)    |
| LimitComBw                | Bandbreitenlimit für COM-Portumleitungskanal                                              | Numerischer Wert                    |
| LimitComBWPercent         | Bandbreitenlimit für COM-Portumleitungskanal als Prozentsatz der Gesamtsitzungsbandbreite | Numerischer Wert zwischen 0 und 100 |
| AutoConnectClientComPorts | Automatische Verbindung von COM-Ports auf dem Benutzergerät                               | 1 (Zulassen) oder 0 (Verhindern)    |
| AllowLptPortRedirection   | Zulassen oder Verhindern der LPT-Portumleitung                                            | 1 (Zulassen) oder 0 (Verhindern)    |
| LimitLptBw                | Bandbreitenlimit für LPT-Portumleitungskanal                                              | Numerischer Wert                    |
| LimitLptBWPercent         | Bandbreitenlimit für LPT-Portumleitungskanal als Prozentsatz der Gesamtsitzungsbandbreite | Numerischer Wert zwischen 0 und 100 |
| AutoConnectClientLptPorts | Automatische Verbindung von LPT-Ports auf dem Benutzergerät                               | 1 (Zulassen) oder 0 (Verhindern)    |

Nach dem Konfigurieren dieser Einstellungen ändern Sie die Maschinenkataloge, damit sie das neue Masterimage oder die aktualisierte physische Maschine verwenden. Wenn sich die Benutzer das nächste Mal abmelden, werden die Desktops mit den neuen Einstellungen aktualisiert.

# Richtlinieneinstellungen für Connector für Configuration Manager 2012

May 10, 2016

Der Abschnitt "Connector für Configuration Manager 2012" enthält Richtlinieneinstellungen zum Konfigurieren des Citrix Connector 7.5-Agents.

Wichtig: Richtlinien für Warnungs-, Abmeldungs- und Neustartmeldungen gelten nur für Bereitstellungen für Serverbetriebssystemmaschinenkataloge, die manuell oder über Provisioning Services verwaltet werden. Bei solchen Maschinenkatalogen benachrichtigt der Connector-Dienst Benutzer über ausstehende Anwendungsinstallationen oder Softwareupdates.

Verwenden Sie bei über MCS verwalteten Katalogen Studio zur Benachrichtigung der Benutzer. Verwenden Sie bei manuell verwalteten Maschinenkatalogen mit Desktopbetriebssystemmaschinen Configuration Manager zur Benachrichtigung der Benutzer. Verwenden Sie bei mit Provisioning Services verwalteten Maschinenkatalogen mit Desktopbetriebssystemmaschinen Provisioning Services zur Benachrichtigung der Benutzer.

## Zeitraum für Vorabwarnung

Diese Einstellung gibt das Intervall an, mit dem Benutzern Vorabmeldungen angezeigt werden.

Intervalle werden im Format ttt.hh:mm:ss festgelegt. Dabei gilt Folgendes:

- ttt steht für die Tage. Dieser Parameter ist optional und kann Werte von 0 bis 999 annehmen.
- hh steht für die Stunden und kann Werte von 0 bis 23 annehmen.
- mm steht für Minuten und kann Werte von 0 bis 59 annehmen.
- ss steht für Sekunden und kann Werte von 0 bis 59 annehmen.

Das Standardintervall ist 1 Stunde (01:00:00).

## Meldungsfeldtext für Vorabwarnung

Diese Einstellung enthält den editierbaren Text für die Vorabmeldung, die Benutzer vor anstehenden Softwareupdates oder Wartungsaufgaben erhalten, für die sie sich abmelden müssen.

Die Standardmeldung lautet: {TIMESTAMP} Please save your work. The server will go offline for maintenance in {TIMELEFT}

## Meldungsfeldtitel für Vorabwarnung

Diese Einstellung enthält den editierbaren Titel für die Titelleiste der Vorabmeldung, die Benutzer erhalten.

Der Standardtitel ist: Upcoming Maintenance

## Zeitraum für Vorabwarnung

Diese Einstellung definiert, wie lange vor Wartungsaufgaben die Vorabwarnmeldung zum ersten Mal angezeigt wird.

Der Zeitraum wird im Format ttt.hh:mm:ss angegeben, mit den folgenden Variablen:

- ttt steht für die Tage. Dieser Parameter ist optional und kann Werte von 0 bis 999 annehmen.
- hh steht für die Stunden und kann Werte von 0 bis 23 annehmen.
- mm steht für Minuten und kann Werte von 0 bis 59 annehmen.

- ss steht für Sekunden und kann Werte von 0 bis 59 annehmen.

In der Standardeinstellung ist der Wert 16 Stunden (16:00:00), d. h. dass die erste Vorabwarnmeldung ca. 16 Stunden vor der Wartung angezeigt wird.

#### Letzter Meldungsfeldtext für Abmeldung erzwingen

Diese Einstellung enthält den editierbaren Text für die Meldung, die Benutzer warnt, dass die Abmeldung erzwungen wird.

Die Standardmeldung lautet: The server is currently going offline for maintenance

#### Letzter Meldungsfeldtitel für Abmeldung erzwingen

Diese Einstellung enthält den editierbaren Titel für die Titelleiste der letzten Meldung für Abmeldung erzwingen.

Der Standardtitel lautet: Notification From IT Staff

#### Kulanzzeitraum für erzwungenes Abmelden

Diese Einstellung definiert den Kulanzzeitraum, der Benutzern zugestanden wird, nachdem sie gewarnt wurden, dass die Abmeldung erzwungen wird, und dem tatsächlichen erzwungenen Abmelden, damit die ausstehenden Wartungsaufgaben gestartet werden können.

Der Zeitraum wird im Format ttt.hh:mm:ss angegeben, mit den folgenden Variablen:

- ttt steht für die Tage. Dieser Parameter ist optional und kann Werte von 0 bis 999 annehmen.
- hh steht für die Stunden und kann Werte von 0 bis 23 annehmen.
- mm steht für Minuten und kann Werte von 0 bis 59 annehmen.
- ss steht für Sekunden und kann Werte von 0 bis 59 annehmen.

In der Standardeinstellung ist der Kulanzzeitraum für erzwungenes Abmelden auf 5 Minuten (00:05:00) festgelegt.

#### Letzter Meldungsfeldtext für Abmeldung erzwingen

Diese Einstellung enthält den editierbaren Text für die letzte Warnmeldung, die Benutzer auffordert, ihre Arbeit zu speichern und sich vor der erzwungenen Abmeldung abzumelden.

Die Standardmeldung enthält den folgenden Text: "{TIMESTAMP} Please save your work and log off. The server will go offline for maintenance in {TIMELEFT}"

#### Meldungsfeldtitel für Abmeldung erzwingen

Diese Einstellung enthält den editierbaren Text für die Titelleiste der Meldung für Abmeldung erzwingen.

Der Standardtitel lautet: Notification From IT Staff

#### Imageverwalteter Modus

Der Connector-Agent erkennt automatisch, wenn er auf einem von Provisioning Services oder MCS verwalteten Maschinenklon ausgeführt wird. Der Agent blockiert Configuration Manager-Updates auf imageverwalteten Klonen und installiert die Updates automatisch auf dem Masterimage des Katalogs.

Nachdem ein Masterimage aktualisiert wurde, verwenden Sie Studio zum Orchestrieren des Neustarts der MCS-Klone. Der Connector-Agent orchestriert automatisch den Neustart von PVS-Katalogklonen während der Configuration Manager-Wartung. Zur Außerkraftsetzung dieses Verhaltens, damit Software auf Katalogklonen von Configuration Manager

installiert wird, ändern Sie den Modus von "Imageverwaltet" in Deaktiviert.

### Meldungsfeldtext für Neustarten

Diese Einstellung enthält den editierbaren Text der Benutzermeldung, dass der Server bald neu gestartet wird.

Die Standardmeldung lautet: The server is currently going offline for maintenance

### Normales Zeitintervall, in dem die Agent-Aufgabe ausgeführt wird

Durch diese Einstellung wird festgelegt, wie häufig der Citrix Connector Agent-Aufgabe ausgeführt wird.

Der Zeitraum wird im Format ttt.hh:mm:ss angegeben, mit den folgenden Variablen:

- ttt steht für die Tage. Dieser Parameter ist optional und kann Werte von 0 bis 999 annehmen.
- hh steht für die Stunden und kann Werte von 0 bis 23 annehmen.
- mm steht für Minuten und kann Werte von 0 bis 59 annehmen.
- ss steht für Sekunden und kann Werte von 0 bis 59 annehmen.

In der Standardeinstellung ist das Intervall auf fünf Minuten (00:05:00) festgelegt.

# Drucken

May 10, 2016

Die Druckerverwaltung in Ihrer Umgebung umfasst verschiedene Stufen:

1. Machen Sie sich, falls erforderlich, mit den Druckkonzepten vertraut.
2. Planen der Druckarchitektur. Dazu gehört die Analyse folgender Faktoren: Unternehmensanforderungen, vorhandene Druckinfrastruktur, derzeitige Interaktion von Benutzern und Anwendungen mit Druckvorgängen und das für Ihre Umgebung am besten geeignete Druckverwaltungsmodell.
3. Konfigurieren Sie die Druckumgebung, indem Sie eine Druckerbereitstellungsmethode auswählen und dann Richtlinien zur Bereitstellung Ihres Druckkonzepts erstellen. Aktualisieren Sie Richtlinien, wenn neue Mitarbeiter oder Server hinzugefügt werden.
4. Testen einer Druckkonfiguration, bevor sie den Benutzern bereitgestellt wird.
5. Pflegen Sie die Citrix Druckumgebung durch Verwalten von Druckertreibern und Optimieren der Druckleistung.
6. Beseitigen Sie evtl. auftretende Probleme.

## Druckkonzepte

Bevor Sie die Bereitstellung planen, sollten Sie mit folgenden Hauptkonzepten des Druckens vertraut sein:

- Arten der Druckerbereitstellung
- Wie Druckaufträge weitergeleitet werden
- Grundlagen der Druckertreiberverwaltung

Die Druckkonzepte bauen auf denen von Windows auf. Um das Drucken in Ihrer Umgebung zu konfigurieren und erfolgreich zu verwalten, müssen Sie verstehen, wie das Netzwerk- und Clientdrucken in Windows funktioniert und wie das Druckverhalten in dieser Umgebung umgesetzt wird.

## Ablauf des Druckprozesses

In dieser Umgebung werden alle Druckvorgänge (durch den Benutzer) auf Maschinen initiiert, auf denen Anwendungen gehostet werden. Druckaufträge werden über den Netzwerkdruckserver oder das Benutzergerät an das Druckgerät weitergeleitet.

Für Benutzer von virtuellen Desktops und Anwendungen gibt es keinen persistenten Arbeitsbereich. Bei Sitzungsende wird der Arbeitsbereich des Benutzers gelöscht, demnach müssen alle Einstellungen zu Beginn jeder Sitzung neu erstellt werden. Bei jedem Start einer neuen Sitzung muss daher die Neuerstellung des Arbeitsbereichs durch das System erfolgen.

Wenn ein Benutzer druckt, übernimmt das System folgende Aufgaben:

- Entscheidung darüber, welche Drucker dem Benutzer bereitgestellt werden. Dies wird als Druckerprovisioning bezeichnet.
- Wiederherstellen der Druckereinstellungen des Benutzers.
- Ermitteln des Standarddruckers für die Sitzung.

Sie können festlegen, wie diese Aufgaben durchgeführt werden, indem Sie die Optionen für das Druckerprovisioning, die Weiterleitung von Druckaufträgen, das Speichern von Druckereigenschaften und die Treiberverwaltung konfigurieren. Bedenken Sie dabei, wie die verschiedenen Einstellungen möglicherweise die Druckleistung in der Umgebung und die Benutzererfahrung beeinflussen.

## Druckerprovisioning



Der Prozess, durch den Drucker in einer Sitzung verfügbar gemacht werden, wird als Provisioning bezeichnet. Das Druckerprovisioning wird normalerweise dynamisch abgewickelt. Das heißt, die in einer Sitzung angezeigten Drucker sind nicht vordefiniert und gespeichert. Stattdessen werden die Drucker gemäß der Richtlinien beim Entstehen der Sitzung während der Anmeldung und Wiederverbindung zusammengestellt. Folglich können sich die Drucker je nach Richtlinie, Benutzerort und Netzwerkänderungen ändern, vorausgesetzt, dies spiegelt sich in den Richtlinien wider. Benutzer, die an einen anderen Ort wechseln, bemerken daher möglicherweise Änderungen in ihrem Arbeitsbereich.

Das System überwacht auch clientseitige Drucker und passt automatisch erstellte Drucker in Sitzungen dynamisch an, je nachdem, welche Hinzufügungen, Löschungen und Änderungen an den clientseitigen Druckern vorgenommen werden. Von dieser dynamischen Druckerermittlung profitieren mobile Benutzer, wenn sie über verschiedene Geräte eine Verbindung herstellen.

Die gängigsten Methoden der Druckerbereitstellung sind folgende:

- **Universeller Druckserver:** Der [universelle Druckserver](#) von Citrix bietet universelle Druckunterstützung für Netzwerkdrucker. Der universelle Druckserver verwendet den universellen Druckertreiber. Diese Lösung ermöglicht die Verwendung eines einzelnen Treibers auf einer Serverbetriebssystemmaschine und damit den Netzwerkdruck von jedem Gerät aus.

Citrix empfiehlt den Einsatz des universellen Druckservers für Szenarios mit Remote-Druckerservern. Der universelle Druckserver überträgt den Druckauftrag über das Netzwerk in einem optimierten und komprimierten Format, wodurch der Netzwerkverkehr reduziert und die Benutzererfahrung verbessert wird.

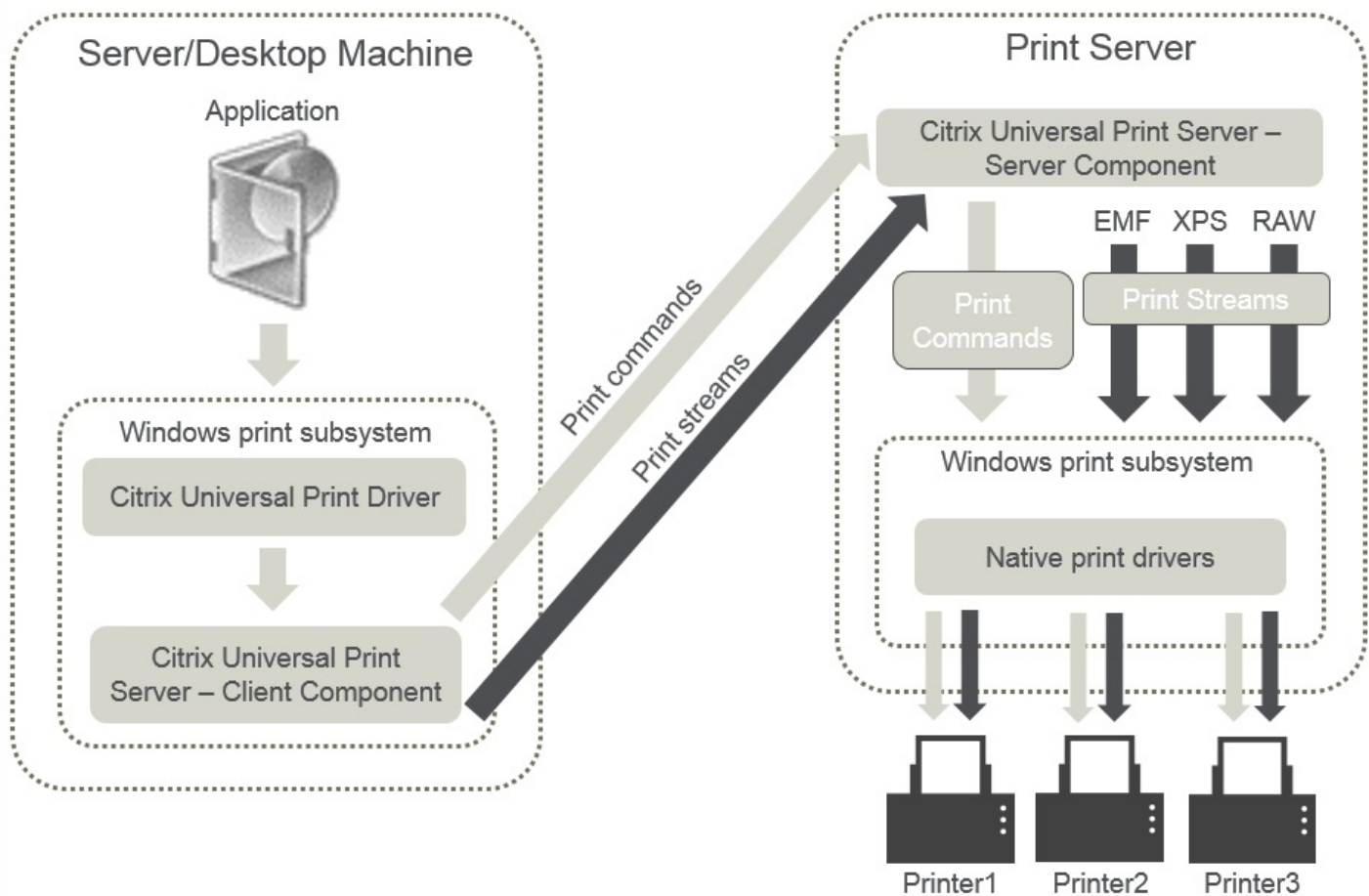
Der universelle Druckserver umfasst als Feature die folgenden Komponenten:

- Clientkomponente "UPClient": Aktivieren Sie UPClient auf jeder Serverbetriebssystemmaschine, die Netzwerksitzungsdrucker bereitstellt und den universellen Druckertreiber verwendet.
- Serverkomponente "UPServer": Installieren Sie UPServer auf jedem Druckserver, der Netzwerksitzungsdrucker bereitstellt und den universellen Druckertreiber für die Sitzungsdrucker verwendet (unabhängig davon, ob die Sitzungsdrucker zentral bereitgestellt werden).

Informationen zu den Anforderungen und zum Setup des universellen Druckservers finden Sie in der Dokumentation zu Systemanforderungen und Installation.

Hinweis: Der universelle Druckserver wird auch für VDI-in-a-Box 5.3 unterstützt. Informationen zur Installation des universellen Druckservers mit VDI-in-a-Box finden Sie in der Dokumentation zu VDI-in-a-Box.

Die folgende Abbildung zeigt den typischen Workflow eines Netzwerkdruckers in einer Umgebung mit universellem Druckserver.



Wenn Sie das Citrix Feature "Universeller Druckserver" aktivieren, wird es von allen verbundenen Netzwerkdruckern automatisch über Autodiscovery genutzt.

- **Automatische Erstellung:** *Automatische Erstellung* bezieht sich auf Drucker, die automatisch zu Beginn jeder Sitzung erstellt werden. Sowohl Remotedrucker als auch lokal angeschlossene Drucker können automatisch erstellt werden. Bei Umgebungen mit einer großen Anzahl von Druckern pro Benutzer ist es u. U. besser, nur den Standarddrucker automatisch zu erstellen. Wenn weniger Drucker automatisch erstellt werden, entsteht auf den Serverbetriebssystemmaschinen weniger Mehraufwand (Arbeitsspeicher und CPU). Eine reduzierte Anzahl an automatisch erstellten Druckern kann auch die Anmeldedauer der Benutzer verkürzen.

Automatisch erstellte Drucker basieren auf:

- Den auf dem Benutzergerät installierten Druckern.
- Den auf die Sitzung angewendeten Richtlinien.

Durch Richtlinieneinstellungen für die automatische Erstellung können Sie Anzahl oder Art der automatisch erstellten Drucker beschränken. Standardmäßig sind die Drucker in Sitzungen verfügbar, wenn alle Drucker auf dem Benutzergerät automatisch konfiguriert werden, einschließlich der lokal angeschlossenen und der Netzwerkdrucker.

Nachdem der Benutzer die Sitzung beendet, werden die Drucker für diese Sitzung gelöscht.

Mit der automatischen Erstellung von Client- und Netzwerkdruckern sind Wartungsarbeiten verbunden. Bei Hinzufügen eines Druckers muss beispielsweise auch Folgendes durchgeführt werden:

- Aktualisieren der Richtlinieneinstellung "Sitzungsdrucker"

- Hinzufügen des Treibers zu allen Serverbetriebssystemmaschinen über die Richtlinieneinstellung "Druckertreiberzuordnung und -kompatibilität"

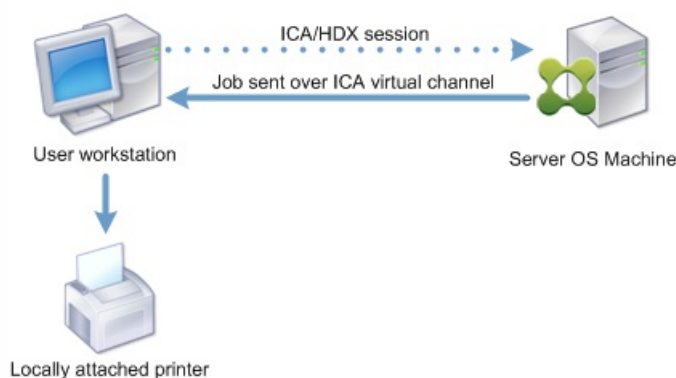
## Weiterleiten von Druckaufträgen

Der Begriff Druckpfad umfasst den Pfad, über den Druckaufträge weitergeleitet werden, und den Speicherort, an dem Druckaufträge gespoolt werden. Beide Aspekte dieses Konzepts sind wichtig. Die Weiterleitung wirkt sich auf den Netzwerk-Datenverkehr aus. Das Spooling wirkt sich auf die Auslastung der lokalen Ressourcen an dem Gerät, das den Auftrag verarbeitet, aus.

In dieser Umgebung können Druckaufträge auf zwei Wegen zu einem Druckgerät gelangen: über den Client oder über einen Netzwerkdruckserver. Dafür werden die Bezeichnungen Clientdruckpfad und Netzwerkdruckpfad verwendet. Welcher Pfad standardmäßig ausgewählt wird, hängt vom verwendeten Drucker ab.

### Lokal angeschlossene Drucker

Das System leitet Aufträge von der Serverbetriebssystemmaschine über den Client an den Drucker. Der Druckdatenverkehr wird über das ICA-Protokoll optimiert und komprimiert. Wenn ein Druckgerät lokal an das Benutzergerät angeschlossen ist, werden Druckaufträge über den virtuellen ICA-Kanal weitergeleitet.



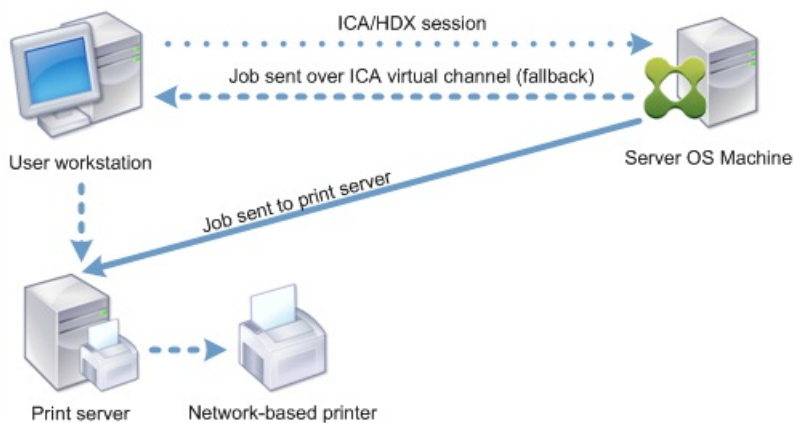
### Netzwerkdrucker

Standardmäßig werden alle für Netzwerkdrucker bestimmten Druckaufträge von der Serverbetriebssystemmaschine über das Netzwerk direkt an den Druckserver weitergeleitet. In folgenden Situationen werden jedoch Druckaufträge automatisch über die ICA-Verbindung geleitet:

- Wenn der virtuelle Desktop oder die Anwendung keine Verbindung mit dem Druckserver herstellen kann.
- Wenn der systemeigene Druckertreiber auf der Serverbetriebssystemmaschine nicht verfügbar ist.

Wenn der universelle Druckserver nicht aktiviert ist, empfiehlt sich die Konfiguration des Clientdruckpfads für den Netzwerkdruck bei Verbindungen mit geringer Bandbreite, z. B. WANs, die von der Optimierung und Komprimierung des Datenverkehrs beim Senden von Aufträgen über die ICA-Verbindung profitieren.

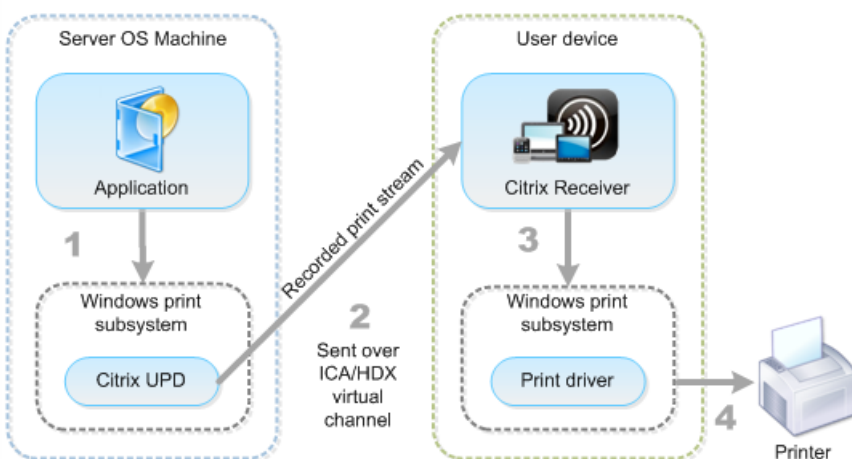
Der Clientdruckpfad ermöglicht auch die Begrenzung des Datenverkehrs oder der für Druckaufträge zugeordneten Bandbreite. Wenn die Auftragsleitung über das Benutzergerät nicht möglich ist, z. B. bei Thin Clients ohne Druckerfunktionen, muss die Servicequalität so konfiguriert werden, dass ICA/HDX-Verkehr Vorrang hat und eine gute Benutzererfahrung bei der Sitzung gewährleistet ist.



## Druckertreiberverwaltung

Zur Vereinfachung der Druckprozesse in dieser Umgebung empfiehlt Citrix die Verwendung des universellen Citrix Druckertreibers. Der universelle Druckertreiber ist ein geräteunabhängiger, für jedes Druckgerät geeigneter Treiber. Durch die so erzielte Reduzierung der Treiberanzahl wird die Verwaltung erheblich vereinfacht.

Die folgende Abbildung zeigt die universellen Druckertreiberkomponenten und einen typischen Workflow für ein lokal angeschlossenes Druckgerät.



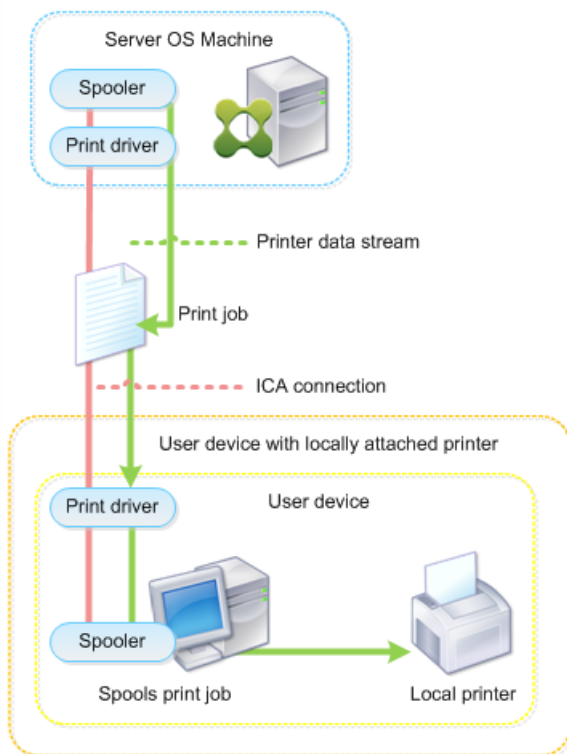
Legen Sie bei der Planung der Strategie zur Treiberverwaltung fest, ob Sie gerätespezifische Treiber, den universellen Druckertreiber oder beides unterstützen wollen. Wenn Sie Standardtreiber unterstützen, müssen Sie außerdem Folgendes festlegen:

- Zu unterstützende Treibertypen
- Aktivieren oder Deaktivieren der automatischen Installation der Druckertreiber (falls auf Serverbetriebssystemmaschinen nicht vorhanden)
- Erstellen der Treiberkompatibilitätslisten

Wenn das System während der automatischen Druckererstellung erkennt, dass ein neuer lokaler Drucker an einem Benutzergerät angeschlossen ist, wird die Serverbetriebssystemmaschine auf den erforderlichen Druckertreiber hin überprüft. Ist kein Windows-systemeigener Treiber verfügbar, wird vom System standardmäßig der universelle Druckertreiber verwendet.

Der Druckvorgang kann nur dann erfolgreich ausgeführt werden, wenn der Druckertreiber auf der Serverbetriebssystemmaschine und der Treiber auf dem Benutzergerät übereinstimmen. In der folgenden Abbildung wird

dargestellt, wie der Druckertreiber an zwei Orten für den Clientdruck verwendet wird.



#### Verwandter Inhalt

- [Beispiel einer Druckkonfiguration](#)
- [Bewährte Methoden, Sicherheitsüberlegungen und Standardvorgänge](#)
- [Druckrichtlinien und Einstellungen](#)
- [Bereitstellen von Druckern](#)
- [Pflege der Druckumgebung](#)
- [Anforderungen für den universellen Druckserver](#)

# Beispiel einer Druckkonfiguration

May 10, 2016

Die Auswahl der am besten geeigneten Druckkonfigurationsoptionen für die Anforderungen und die Umgebung kann die Verwaltung vereinfachen. Obwohl die Standarddruckkonfiguration für die meisten Umgebungen geeignet ist, gewährleisten die Standardwerte möglicherweise nicht die erwartete Benutzererfahrung oder die optimale Netzwerkverwendung und den gewünschten Verwaltungsaufwand für die Umgebung.

Die Druckkonfiguration hängt von folgenden Faktoren ab:

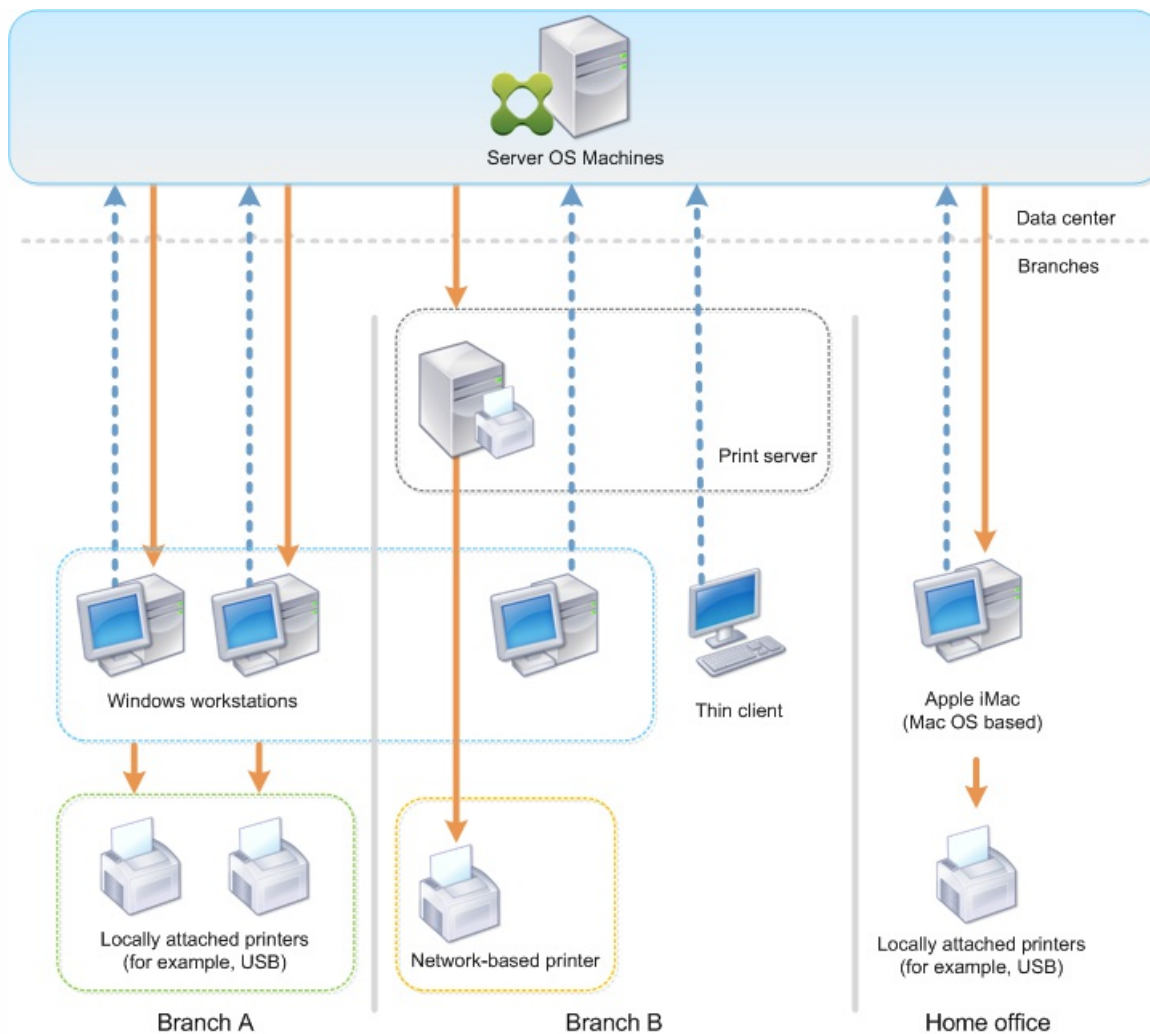
- Den Unternehmensanforderungen und der vorhandenen Druckinfrastruktur.  
Berücksichtigen Sie bei der Druckkonfiguration die Anforderungen der Organisation. Die vorhandene Druckimplementierung (ob Benutzer Drucker hinzufügen können, welche Benutzer Zugriff auf welche Drucker haben usw.) kann bei der Definition der Druckkonfiguration ein nützlicher Leitfaden sein.
- Ob in Ihrer Organisation Sicherheitsrichtlinien gelten, die Drucker für bestimmte Benutzer reservieren (z. B. Drucker für die Personalabteilung oder die Gehaltsabrechnung).
- Ob Benutzer drucken müssen, wenn sie nicht an ihrem primären Arbeitsort sind, z. B. Mitarbeiter, die verschiedene Arbeitsstationen verwenden oder auf Geschäftsreisen gehen.

Achten Sie beim Entwerfen der Druckkonfiguration darauf, den Benutzern die gleiche Erfahrung in einer Sitzung zu bieten, wie sie es beim Drucken von lokalen Benutzergeräten aus gewohnt sind.

## Beispiel einer Druckbereitstellung

Die folgende Abbildung zeigt die Bereitstellung dieser Anwendungsfälle:

- **Zweigstelle A:** Kleine Auslandsniederlassung mit einigen Windows-Arbeitsstationen. Jede Benutzerarbeitsstation hat einen lokal angeschlossenen, privaten Drucker.
- **Zweigstelle B:** Großes Zweigstellenbüro mit Thin Clients und auf Windows basierenden Arbeitsstationen. Aus Effizienzgründen teilen sich die Benutzer dieser Zweigstelle die Netzwerkdrucker (einen pro Stockwerk). Die Druckwarteschlangen werden über Windows-Druckserver der Zweigstelle gesteuert.
- **Home Office:** Büro im Haus eines Mitarbeiters mit einem Mac OS-basierten Benutzergerät, über das auf die Citrix Infrastruktur des Unternehmens zugegriffen wird. Das Benutzergerät hat einen lokal angeschlossenen Drucker.



In den folgenden Abschnitten werden Konfigurationen beschrieben, die die Komplexität der Umgebung minimieren und die Verwaltung vereinfachen.

### Automatisch erstellte Clientdrucker und der universelle Citrix Druckertreiber

In Branch A arbeiten alle Benutzer auf Arbeitsstationen unter Windows und verwenden daher automatisch erstellte Clientdrucker und den universellen Druckertreiber. Dies bietet folgende Vorteile:

- **Leistung:** Druckaufträge werden über den ICA-Druckkanal geleitet, sodass die Druckdaten komprimiert werden können und Bandbreite eingespart wird.

Um sicherzustellen, dass ein einzelner Benutzer durch den Druck eines großen Dokuments nicht die Sitzungsleistung anderer Benutzer beeinträchtigt, wird eine Citrix Richtlinie für die maximale Druckbandbreite konfiguriert.

Eine andere Lösung wäre die Multistream-ICA-Verbindung, bei der der Druckverkehr innerhalb einer separaten TCP-Verbindung mit niedriger Priorität übertragen wird. Multistream-ICA kann verwendet werden, wenn Quality of Service (QoS) über die WAN-Verbindung nicht implementiert ist.

- **Flexibilität:** Der universelle Citrix Druckertreiber gewährleistet, dass alle mit dem Client verbundenen Drucker auch von virtuellen Desktop- oder Anwendungssitzungen verwendet werden können, ohne dass ein neuer Druckertreiber im Datacenter integriert werden muss.

### Universeller Citrix Druckserver

In Branch B werden alle Netzwerkdrucker und ihre Warteschlangen auf einem Windows-Druckerserver verwaltet. Somit



erweist sich der universelle Citrix Druckserver als die effizienteste Konfiguration.

Alle erforderlichen Druckertreiber werden von lokalen Administratoren auf dem Druckserver installiert und verwaltet. Das Zuordnen von Druckern in virtuellen Desktop- oder Anwendungssitzungen funktioniert wie folgt:

- Arbeitsstationen unter Windows: Das IT-Team vor Ort hilft den Benutzern beim Herstellen der Verbindung mit dem geeigneten Netzwerkdrucker auf ihren Windows-Arbeitsstationen. Dies ermöglicht Benutzern, über lokal installierte Anwendungen zu drucken.

Bei virtuellen Desktop- oder Anwendungssitzungen werden die lokal konfigurierten Drucker über die automatische Erstellung aufgelistet. Der virtuelle Desktop oder die virtuelle Anwendung stellt dann eine Verbindung mit dem Druckserver her, falls möglich, als Direktnetzwerkverbindung.

Die Komponenten des universellen Citrix Druckservers werden installiert und aktiviert, systemeigene Druckertreiber sind nicht erforderlich. Falls ein Treiber aktualisiert oder eine Druckerwarteschlange geändert wird, ist im Datacenter keine weitere Konfiguration nötig.

- Thin Clients: Für Thin Clients-Benutzer müssen die Drucker in den virtuellen Desktop- oder Anwendungssitzungen angeschlossen werden. Um den Benutzern das Drucken so einfach wie möglich zu machen, konfigurieren die Administratoren eine einzige Citrix Sitzungsdruckerrichtlinie pro Stockwerk, damit der jeweilige Drucker als Standarddrucker festgelegt wird.  
Damit sichergestellt ist, dass die Benutzer stets mit dem richtigen Drucker verbunden sind, auch wenn sie in einem anderen Stockwerk sind, werden die Richtlinien nach Subnetz oder Thin Client-Namen gefiltert. Diese Konfiguration, die auch als "Proximitydrucken" bezeichnet wird, lässt die Wartung lokaler Druckertreiber zu (gemäß der delegierten Administration).

Wenn eine Druckerwarteschlange geändert oder hinzugefügt werden muss, müssen Citrix Administratoren die entsprechende Richtlinie für Sitzungsdrucker in der Umgebung ändern.

Da der Netzwerkdruckdatenverkehr außerhalb des virtuellen ICA-Kanals gesendet wird, wird QoS implementiert. Eingehende und ausgehende Netzwerkdaten an Ports für ICA/HDX-Datenverkehr haben Vorrang vor sonstigem Netzwerkdatenverkehr. Diese Konfiguration gewährleistet, dass Benutzersitzungen von großen Druckaufträgen nicht beeinträchtigt werden.

### Automatisch erstellte Clientdrucker und der universelle Citrix Druckertreiber

Bei Heimbüros mit nicht standardmäßigen Arbeitsstationen und nicht verwalteten Druckgeräten ist es am einfachsten, automatisch erstellte Drucker und den universellen Druckertreiber zu verwenden.

### Zusammenfassung der Bereitstellung

Zusammenfassend lässt sich die Konfiguration dieses Bereitstellungsbeispiels wie folgt beschreiben:

- Auf Serverbetriebssystemmaschinen werden keine Druckertreiber installiert. Es wird nur der universelle Citrix Druckertreiber verwendet. Fallback auf systemeigene Druckertreiber und die automatische Installation von Druckertreibern sind deaktiviert.
- Die automatische Erstellung von Clientdruckern für alle Benutzer wird über eine Richtlinie konfiguriert. Serverbetriebssystemmaschinen werden standardmäßig direkt mit dem Druckserver verbunden. Zur Konfiguration müssen lediglich die Komponenten des universellen Druckservers aktiviert werden.
- Eine Sitzungsdruckerrichtlinie wird für jedes Stockwerk von Branch B konfiguriert und gilt für alle Thin Clients des jeweiligen Stockwerks.
- Die Implementierung von QoS für Branch B gewährleistet eine hervorragende Benutzenerfahrung.



# Bewährte Methoden, Sicherheitsüberlegungen und Standardvorgänge

May 10, 2016

## Bewährte Methoden

Viele Faktoren bestimmen die beste Drucklösung für eine bestimmte Umgebung. Einige dieser bewährten Methoden sind möglicherweise für Ihre Site nicht geeignet.

- Verwenden Sie das Citrix Feature "universeller Druckserver".
- Verwenden Sie den universellen Druckertreiber oder Windows-systemeigene Treiber.
- Reduzieren Sie die Anzahl der installierten Druckertreiber auf den Serverbetriebssystemmaschinen auf das Minimum.
- Verwenden Sie Treiberzuordnung zu systemeigenen Treibern.
- Installieren Sie nie ungetestete Druckertreiber in einer Produktionssite.
- Vermeiden Sie Updates von Treibern. Versuchen Sie stets, einen Treiber zu deinstallieren, den Server neu zu starten und dann einen Ersatztreiber zu installieren.
- Deinstallieren Sie nicht verwendete Treiber oder verwenden Sie die Richtlinie Druckertreiberzuordnung und -kompatibilität, um zu verhindern, dass Drucker mit dem Treiber erstellt werden.
- Vermeiden Sie möglichst Kernelmodustreiber der Version 2.
- Wenden Sie sich an den Hersteller oder sehen Sie in der Citrix Ready-Produktdokumentation [www.citrix.com/ready](http://www.citrix.com/ready) nach, ob ein Druckermodell unterstützt wird.

Im Allgemeinen werden alle von Microsoft zur Verfügung gestellten Druckertreiber mit Terminaldiensten getestet und ihre Funktion unter Citrix gewährleistet. Vergewissern Sie sich jedoch vor Einsatz eines Druckertreibers eines Drittanbieters, dass der Treiber von Windows Hardware Quality Labs (WHQL) für Terminaldienste zertifiziert wurde. Citrix vergibt keine Zertifizierung für Druckertreiber.

## Sicherheitsüberlegungen

Citrix Drucklösungen sind inhärent sicher.

- Der Citrix Druckmanagerdienst überwacht und reagiert fortlaufend auf Sitzungsereignisse wie An- und Abmeldung, Trennen, Wiederverbinden und Beenden der Sitzung. Er behandelt Anforderungen, indem er die Identität des Benutzers der aktuellen Sitzung übernimmt.
- Beim Citrix Drucken wird jedem Drucker ein eindeutiger Namespace in einer Sitzung zugewiesen.
- Die Standardsicherheitsbeschreibung für automatisch erstellte Drucker wird eingerichtet, um sicherzustellen, dass die in einer Sitzung automatisch erstellten Clientdrucker für Benutzer in anderen Sitzungen nicht zugänglich sind. Standardmäßig können Administratoren nicht versehentlich auf einem Clientdrucker einer anderen Sitzung drucken, obwohl sie jeden Clientdrucker sehen und die Berechtigungen dafür manuell ändern können.

## Standarddruckvorgänge

Wenn Sie keine Richtlinienregeln konfigurieren, zeigt sich standardmäßig das folgende Druckverhalten:

- Universeller Druckserver ist deaktiviert.
- Alle auf dem Benutzergerät konfigurierten Drucker werden automatisch zu Beginn jeder Sitzung konfiguriert. Dieses Verhalten entspricht der Richtlinieneinstellung Clientdrucker automatisch erstellen mit der Option Alle Clientdrucker automatisch erstellen.
- Das System leitet alle Druckaufträge, die in Warteschlangen für an Benutzergeräte lokal angeschlossene Drucker gestellt wurden, als Clientdruckaufträge weiter (d. h. über den ICA-Kanal und durch das Benutzergerät).

- Das System leitet alle Druckaufträge, die in Warteschlangen von Netzwerkdruckern gestellt wurden, direkt über Serverbetriebssystemmaschinen. Falls die Aufträge vom System nicht über das Netzwerk weitergeleitet werden können, werden sie als umgeleiteter Clientdruckauftrag über das Benutzergerät weitergeleitet.  
Dieses Verhalten entspricht dem Deaktivieren der Citrix Richtlinieneinstellung Direkte Verbindungen zu Druckservern.
- Standardmäßig versucht das System, die Druckeigenschaften (eine Kombination aus den Druckeinstellungen des Benutzers und den gerätespezifischen Druckeinstellungen) auf dem Benutzergerät zu speichern. Wenn der Client diesen Vorgang nicht unterstützt, werden die Druckeigenschaften vom System in Benutzerprofilen auf der Serverbetriebssystemmaschine gespeichert.  
Dieses Verhalten entspricht der Citrix Richtlinieneinstellung Speicherung von Druckereigenschaften mit der Option Nur im Profil speichern, wenn sie nicht auf dem Client gespeichert sind.
- Das System verwendet die Windows-Version des Druckertreibers, falls sie auf der Serverbetriebssystemmaschine verfügbar ist. Ist der Druckertreiber nicht verfügbar, versucht das System, den Treiber vom Windows-Betriebssystem zu installieren. Ist der Treiber in Windows nicht verfügbar, wird ein universeller Citrix Druckertreiber verwendet.  
Dieses Verhalten entspricht dem Aktivieren der Citrix Richtlinieneinstellung Automatische Installation von mitgelieferten Druckertreibern und Konfigurieren der Einstellung Universal Printing mit der Option Universelles Drucken nur verwenden, wenn angeforderter Treiber nicht verfügbar ist.

Das Aktivieren von Automatische Installation von mitgelieferten Druckertreibern kann dazu führen, dass eine große Anzahl systemeigener Druckertreiber installiert wird.

Hinweis: Wenn Sie nicht sicher sind, welche Standardwerte voreingestellt sind, zeigen Sie sie an, indem Sie eine neue Richtlinie erstellen und alle Druckrichtlinienregeln aktivieren. Die angezeigte Option ist die Standardoption.

Immer aktive Protokollierung

XenApp und XenDesktop 7.6 FP3 umfassen eine stets aktivierte Protokollierung für den Druckserver und das Drucksubsystem auf dem VDA.

Zum Sortieren der Protokolle als ZIP-Datei für den E-Mail-Versand bzw. für den automatischen Upload an Citrix Insights Services verwenden Sie das mit dem VDA-Installer in Version 7.6 FP3 gelieferte PowerShell-Cmdlet "Start-TelemetryUpload".

# Druckrichtlinien und Einstellungen

May 10, 2016

Wenn Benutzer von veröffentlichten Anwendungen aus auf Drucker zugreifen, können Sie über Citrix Richtlinien Folgendes konfigurieren:

- Wie das Drucker-Provisioning erfolgt (bzw. wie Drucker zu Sitzungen hinzugefügt werden)
- Wie Druckaufträge weitergeleitet werden
- Wie Druckertreiber verwaltet werden

Sie können verschiedene Druckkonfigurationen für unterschiedliche Benutzergeräte, Benutzer oder beliebige andere Objekte haben, nach denen Richtlinien gefiltert werden.

Die meisten Druckfunktionen werden über die Citrix Druckrichtlinien konfiguriert. Druckeinstellungen folgen dem Standardverhalten für Citrix Richtlinien:

Druckereinstellungen können vom System am Ende einer Sitzung in das Druckerobjekt oder das Clientdruckgerät geschrieben werden, sofern das Netzwerkkonto des Benutzers ausreichende Berechtigungen hat. Standardmäßig verwendet Receiver die Einstellungen, die im Druckerobjekt in der Sitzung gespeichert wurden, bevor an anderen Orten nach Einstellungen gesucht wird.

Standardmäßig werden die Druckereigenschaften auf dem Benutzergerät (falls vom Gerät unterstützt) oder im Benutzerprofil auf der Serverbetriebssystemmaschine gespeichert oder beibehalten. Wenn die Druckereigenschaften während einer Sitzung vom Benutzer geändert werden, werden diese Änderungen im Benutzerprofil auf der Maschine aktualisiert. Wenn sich der Benutzer das nächste Mal anmeldet oder eine neue Verbindung herstellt, übernimmt das Benutzergerät die beibehaltenen Einstellungen. Das heißt, auf dem Benutzergerät geänderte Druckereigenschaften wirken sich nicht auf die aktuelle Sitzung aus bis zum Ab- und erneuten Anmelden des Benutzers.

## Speicherorte für Druckeinstellungen

In Windows-Druckumgebungen können die an den Druckvoreinstellungen vorgenommenen Änderungen auf dem lokalen Computer oder in einem Dokument gespeichert werden. Wenn Benutzer in dieser Umgebung Druckeinstellungen ändern, können diese Änderungen an folgenden Positionen gespeichert werden:

- **Auf dem Benutzergerät:** Windows-Benutzer können Geräteeinstellungen auf dem Benutzergerät ändern, indem sie mit der rechten Maustaste auf die Drucker in der Systemsteuerung klicken und Druckeinstellungen wählen. Wenn beispielsweise Querformat als Seitenausrichtung ausgewählt wird, gilt Querformat als Standard-Seitenausrichtung für diesen Drucker.
- **In einem Dokument:** Bei Textverarbeitungs- und Desktop-Publishing-Programmen werden Dokumenteinstellungen, z. B. die Seitenausrichtung, häufig in Dokumenten gespeichert. Wenn Sie beispielsweise ein zu druckendes Dokument in eine Warteschlange setzen, speichert Microsoft Word die von Ihnen angegebenen Druckvoreinstellungen wie Seitenausrichtung und Druckernamen im Dokument selbst. Diese Einstellungen erscheinen standardmäßig, wenn Sie dieses Dokument das nächste Mal drucken.
- **Benutzerseitige Änderungen in einer Sitzung:** Das System übernimmt Änderungen an den Druckeinstellungen eines automatisch erstellten Druckers nur, wenn diese in der Systemsteuerung der Sitzung vorgenommen wurden, also auf der Serverbetriebssystemmaschine.
- **Auf der Serverbetriebssystemmaschine:** Dies sind die Standardeinstellungen, die einem bestimmten Druckertreiber auf der Maschine zugeordnet sind.

Die in einer Windows-Umgebung gespeicherten Einstellungen sind abhängig von der Stelle, an der die Einstellungen vom

Benutzer vorgenommen wurden. Das bedeutet außerdem, dass die an einer Stelle wie einer Tabellenkalkulation angezeigten Druckeinstellungen sich von den Einstellungen an anderen Stellen, beispielsweise in Dokumenten, unterscheiden können. Die auf einen bestimmten Drucker angewendeten Druckeinstellungen variieren daher innerhalb einer Sitzung.

## Hierarchie der Benutzerdruckeinstellungen

Da die Druckeinstellungen an verschiedenen Stellen gespeichert werden können, verarbeitet das System sie gemäß einer bestimmten Priorität. Sie dürfen auch nicht vergessen, dass Geräteeinstellungen anders behandelt werden als Dokumenteinstellungen und normalerweise Vorrang vor diesen haben.

Standardmäßig wendet das System immer alle Druckeinstellungen an, die ein Benutzer während einer Sitzung geändert hat, d. h. alle beibehaltenen Einstellungen, bevor andere Einstellungen berücksichtigt werden. Wenn der Benutzer druckt, führt das System die auf der Serverbetriebssystemmaschine gespeicherten Standarddruckereinstellungen mit allen beibehaltenen Einstellungen oder Clientdruckereinstellungen zusammen und wendet sie an.

## Speichern der Druckeinstellungen des Benutzers

Citrix empfiehlt, dass Sie den Speicherort der Druckereigenschaften nicht ändern. Am einfachsten können Sie konsistente Druckereigenschaften sicherstellen, indem Sie die Standardeinstellung beibehalten, wonach die Druckereigenschaften auf dem Benutzergerät gespeichert werden. Wenn das System die Eigenschaften auf dem Benutzergerät nicht speichern kann, wird automatisch auf das Benutzerprofil auf der Serverbetriebssystemmaschine zurückgegriffen.

Überprüfen Sie die Richtlinieneinstellung Speicherung von Druckereigenschaften, wenn diese Szenarios zutreffen:

- Verwendung von älteren Plug-Ins, durch die das Speichern der Druckereigenschaften durch die Benutzer auf einem Benutzergerät unterbunden wird
- Verwendung verbindlicher Profile im Windows-Netzwerk, wobei die Druckereigenschaften der Benutzer beibehalten werden sollen

# Bereitstellen von Druckern

May 10, 2016

Es gibt drei Methoden der Druckerbereitstellung:

- [Universeller Citrix Druckserver](#)
- [Automatisch erstellte Clientdrucker](#)
- [Zuweisen von Netzwerkdruckern an Benutzer](#)

## Universeller Citrix Druckserver

Bei der Wahl der besten Drucklösung für Ihre Umgebung sollten Sie Folgendes berücksichtigen:

- Der universelle Druckserver bietet Features, die beim Windows-Druckanbieter nicht verfügbar sind: Zwischenspeichern von Bildern und Schriftarten, erweiterte Komprimierung, Optimierung und Unterstützung für QoS.
- Der universelle Druckertreiber unterstützt die von Microsoft definierten, öffentlichen geräteunabhängigen Einstellungen. Wenn Benutzer Zugriff auf die Geräteeinstellungen des Druckertreibers eines bestimmten Herstellers benötigen, stellt der universelle Druckserver gepaart mit einem Windows-systemeigenen Treiber die beste Lösung dar. In dieser Konfiguration bleiben die Vorteile des universellen Druckservers erhalten und die Benutzer können zugleich auf bestimmte Druckerfunktionen zugreifen. Allerdings ist zu bedenken, dass Windows-systemeigene Treiber wartungsbedürftig sind.
- Der universelle Druckserver von Citrix bietet universelle Druckunterstützung für Netzwerkdrucker. Der universelle Druckserver verwendet den universellen Druckertreiber, einen einzelnen Treiber auf der Serverbetriebssystemmaschine, mit dem von jedem Gerät aus, einschließlich Thin Clients und Tablets, auf lokalen oder Netzwerkdruckern gedruckt werden kann.

Um den universellen Druckserver mit einem Windows-systemeigenen Treiber zu verwenden, aktivieren Sie den universellen Druckserver. Wenn der Windows-systemeigene Treiber verfügbar ist, wird er standardmäßig verwendet. Andernfalls wird der universelle Druckertreiber verwendet. Um dieses Verhalten zu ändern, beispielsweise zur ausschließlichen Verwendung des Windows-systemeigenen Treibers oder des universellen Druckertreibers, müssen Sie die RichtlinienEinstellung Verwendung universeller Druckertreiber aktualisieren.

## Installieren des Citrix universellen Druckservers (UPS)

### Hinweise:

- Das Bereitstellen des universellen Druckservers auf 32-Bit-Betriebssystemen wird nicht unterstützt.
- Die Benutzerauthentifizierung bei Druckvorgängen erfordert, dass der universelle Druckserver in der gleichen Domäne ist wie der Remotedesktopdienste-VDA.
- Die UPClient-Komponente ist in der VDA-Installation enthalten und Sie müssen die Clientkomponente nicht manuell installieren.

### Installieren des Citrix universellen Druckservers

1. Auf einem Druckserver unter Windows 2008 R2 SP1, Windows Server 2012 oder Windows Server 2012 R2 klicken Sie im LTSR-Metainstaller auf "Extend Deployment" > "Universal Print Server" und folgen Sie den Anweisungen auf dem Bildschirm.
2. Starten Sie den Server neu.

## Automatisch erstellte Clientdrucker

Die folgenden universellen Drucklösungen sind für Clientdrucker verfügbar:

- **Citrix Universeller Drucker:** ein generischer Drucker, der zu Beginn einer Sitzung erstellt wird und nicht an ein Druckgerät gebunden ist. Der Citrix Universelle Drucker muss die verfügbaren Clientdrucker bei der Anmeldung nicht auflisten, wodurch sich der Ressourceneinsatz erheblich reduziert und die Anmeldedauer für die Benutzer verringert wird. Mit dem Citrix Universellen Drucker kann auf jedem clientseitigen Druckgerät gedruckt werden.

Der Citrix Universelle Drucker funktioniert allerdings möglicherweise nicht für alle Benutzergeräte oder Receiver in Ihrer Umgebung. Der Citrix Universelle Drucker erfordert eine Windows-Umgebung und unterstützt nicht das Citrix Offline Plug-In oder Anwendungen, die an Clients gestreamt werden. Verwenden Sie für solche Umgebungen automatisch erstellte Drucker und den universellen Druckertreiber.

Wenn Sie eine universelle Drucklösung für Nicht-Windows-Receiver benötigen, verwenden Sie einen der anderen universellen Postscript/PCL-Druckertreiber, die automatisch installiert werden.

- **Citrix Universeller Druckertreiber:** ein geräteunabhängiger Druckertreiber. Wenn Sie einen universellen Citrix Druckertreiber einrichten, verwendet das System standardmäßig den auf EMF basierenden universellen Druckertreiber. Der universelle Citrix Druckertreiber kann auch kleinere Druckaufträge erstellen als ältere oder weniger umfangreiche Druckertreiber. Für Spezialdrucker wird jedoch u. U. ein gerätespezifischer Treiber benötigt, um die Druckaufträge optimal zu verarbeiten.

**Konfigurieren von Universal Printing:** Verwenden Sie die folgenden Citrix Richtlinieneinstellungen zum Konfigurieren von Universal Printing. Weitere Informationen finden Sie in der Onlinehilfe zu Richtlinieneinstellungen.

- Verwenden universeller Druckertreiber: Mit dieser Einstellung legen Sie fest, wann das universelle Drucken verwendet wird.
- Automatisch generischen universellen Drucker erstellen: Mit dieser Einstellung aktivieren oder deaktivieren Sie die automatische Erstellung des generischen universellen Citrix Druckerobjekts für Sitzungen mit einem Benutzergerät, das mit Universal Printing kompatibel ist. Standardmäßig werden generische universelle Drucker nicht automatisch erstellt.
- Priorität universeller Treiber: Mit dieser Einstellung geben Sie an, in welcher Reihenfolge das System die universellen Druckertreiber verwendet, angefangen mit dem ersten Eintrag in der Liste. Sie können Treiber hinzufügen, bearbeiten oder entfernen und die Reihenfolge der Treiber in der Liste ändern.
- Universelles Drucken - Vorschau-einstellung Mit dieser Einstellung geben Sie an, ob die Druckvorschau für automatisch erstellte oder universelle Drucker verwendet werden soll.
- Universelles Drucken - EMF-Verarbeitungsmodus Mit dieser Einstellung steuern Sie die Verarbeitungsmethode für die EMF-Spooldatei auf dem Windows-Benutzergerät. Standardmäßig werden EMF-Datensätze direkt zum Drucker gespoolt. Direktes Spoolen an den Drucker ermöglicht eine schnellere Verarbeitung der Datensätze durch den Spooler und beansprucht weniger CPU-Ressourcen.

Weitere Richtlinien finden Sie unter [Optimieren der Druckleistung](#). Informationen zum Ändern der Standardeinstellungen (Papierformat, Druckqualität, Farbe, Seitenaufdruck und Auflage) finden Sie unter [CTX113148](#).

**Drucker automatisch über das Benutzergerät erstellen:** Zu Beginn einer Sitzung erstellt das System standardmäßig alle Drucker auf dem Benutzergerät automatisch. Sie können steuern, welche Typen der Drucker ggf. den Benutzern bereitgestellt werden und somit ein automatisches Erstellen verhindern.

Verwenden Sie die Citrix Richtlinieneinstellung Clientdrucker automatisch erstellen, um das automatische Erstellen zu steuern. Sie können Folgendes festlegen:

- Alle für das Benutzergerät sichtbaren Drucker, einschließlich der Netzwerkdrucker und der lokal angeschlossenen Drucker, werden zu Beginn einer Sitzung automatisch erstellt (Standardeinstellung)
- Alle lokalen Drucker, die physisch an das Benutzergerät angeschlossen sind, werden automatisch erstellt
- Nur der Standarddrucker für das Benutzergerät wird automatisch erstellt

- Automatische Erstellung ist für alle Clientdrucker deaktiviert

Die Einstellung Clientdrucker automatisch erstellen erfordert, dass für die Einstellung Clientdruckerumleitung Zugelassen (Standardeinstellung) festgelegt ist.

## Zuweisen von Netzwerkdruckern an Benutzer

Standardmäßig werden die Netzwerkdrucker auf dem Benutzergerät automatisch zu Beginn jeder Sitzung konfiguriert. Sie können die Anzahl der aufgelisteten und zugeordneten Netzwerkdrucker reduzieren, indem Sie festlegen, welche Netzwerkdrucker in jeder Sitzung erstellt werden sollen. Diese Drucker werden als Sitzungsdrucker bezeichnet.

Sie können die Sitzungsdruckerrichtlinien nach IP-Adressen filtern, um das Proximitydrucken (auf dem nächstgelegenen Drucker) zu gewährleisten. Das Drucken auf dem nächstgelegenen Drucker ermöglicht den Benutzern innerhalb eines angegebenen IP-Adressbereichs den automatischen Zugriff auf Netzwerkdruckgeräte, die im gleichen Bereich liegen. Proximitydrucken wird von der Funktion Citrix Universeller Druckserver umgesetzt; die hier beschriebene Konfiguration ist dazu nicht erforderlich.

Proximitydrucken kann folgende Szenarios umfassen:

- Das interne Unternehmensnetzwerk nutzt einen DHCP-Server, der automatisch IP-Adressen für Benutzer zuweist.
- Alle Abteilungen im Unternehmen haben eindeutige zugeordnete IP-Adressbereiche.
- In den IP-Adressbereichen jeder Abteilung gibt es Netzwerkdrucker.

Wenn Proximitydrucken konfiguriert ist und ein Mitarbeiter einer Abteilung in eine andere wechselt, ist keine zusätzliche Druckgerätekonfiguration erforderlich. Sobald das Benutzergerät im IP-Adressbereich der neuen Abteilung erkannt wird, erhält es Zugriff auf alle Netzwerkdrucker in diesem Bereich.

**Konfigurieren bestimmter Drucker für die Umleitung in Sitzungen:** Zum Erstellen von Administratoren zugewiesener Drucker konfigurieren Sie die Citrix Richtlinieneinstellung Sitzungsdrucker. Verwenden Sie zum Hinzufügen eines Netzwerkdruckers zu dieser Richtlinie eine der folgenden Methoden:

- Geben Sie den UNC-Pfad im Format `\\servername\printername` ein.
- Navigieren Sie zu einem Drucker im Netzwerk.
- Navigieren Sie zu Druckern auf einem bestimmten Server. Geben Sie den Servernamen im Format `\\servername` an und klicken Sie auf Durchsuchen.

Wichtig: Der Server führt alle aktivierten Einstellungen für Sitzungsdrucker für alle angewendeten Richtlinien zusammen, angefangen von der höchsten bis zur niedrigsten Priorität. Ist ein Drucker in mehreren Richtlinienobjekten konfiguriert, werden angepasste Standardeinstellungen nur aus dem Richtlinienobjekt mit der höchsten Priorität verwendet, in dem der Drucker konfiguriert ist.

Welche Netzwerkdrucker über die Einstellung Sitzungsdrucker erstellt werden, kann je nachdem, wo die Sitzung gestartet wurde, durch Filtern nach Objekten, z. B. Subnetzen, variieren.

**Festlegen eines Standardnetzwerkdruckers für eine Sitzung:** Standardmäßig wird der Hauptdrucker des Benutzers als Standarddrucker für eine Sitzung verwendet. Verwenden Sie die Citrix Richtlinieneinstellung Standarddrucker, um die Auswahl des Standarddruckers auf dem Benutzergerät in einer Sitzung zu ändern.

1. Wählen Sie unter Standarddrucker eine Einstellung für Standarddrucker des Clients wählen:

- Netzwerkdruckername: Drucker, die mit der Richtlinieneinstellung Sitzungsdrucker hinzugefügt wurden, werden in diesem Menü angezeigt. Wählen Sie den als Standard für diese Richtlinie zu verwendenden Netzwerkdrucker aus.
- Standarddrucker des Benutzers nicht anpassen: Verwendet die Einstellung der Termindienste oder des aktuellen Benutzerprofils für den Standarddrucker. Weitere Informationen finden Sie in der Onlinehilfe zu Richtlinieneinstellungen.

2. Wenden Sie die Richtlinie auf die Benutzergruppe (oder andere gefilterte Objekte) an, auf die sich auswirken soll.

**Konfigurieren von Proximitydrucken:** Das Proximitydrucken (Drucken auf dem nächstgelegenen Drucker) wird ebenfalls über den universellen Druckserver von Citrix bereitgestellt. Dieser erfordert nicht die hier beschriebene Konfiguration.

1. Erstellen Sie eine separate Richtlinie für jedes Subnetz (oder entsprechend dem Druckerstandort).
2. Fügen Sie in jeder Richtlinie der Einstellung Sitzungsdrucker die Drucker an dem geografischen Standort des Subnetzes hinzu.
3. Setzen Sie die Einstellung Standarddrucker auf Standarddrucker des Benutzers nicht anpassen.
4. Filtern Sie die Richtlinien nach Client-IP-Adresse. Aktualisieren Sie diese Richtlinien, um Änderungen der DHCP-IP-Adressbereiche zu berücksichtigen.



# Pflege der Druckumgebung

May 10, 2016

Zur Pflege der Druckumgebung gehört Folgendes:

- Verwalten von Druckertreibern
- Optimieren der Druckleistung
- Anzeigen von Druckern und Verwalten von Druckwarteschlangen

## Verwalten von Druckertreibern

Citrix empfiehlt die Verwendung des universellen Citrix Druckertreibers, um den Verwaltungsaufwand und mögliche Probleme mit Druckertreibern gering zu halten.

Wenn die automatische Erstellung fehlschlägt, installiert das System standardmäßig einen bei Windows integrierten systemeigenen Druckertreiber. Falls kein Treiber verfügbar ist, greift das System automatisch auf den universellen Druckertreiber zurück. Weitere Informationen über Druckertreiber-Standardwerte finden Sie unter [Bewährte Methoden, Sicherheitsüberlegungen und Standardvorgänge](#).

Wenn der universelle Druckertreiber von Citrix nicht für alle Szenarios geeignet ist, reduzieren Sie die Anzahl installierter Treiber auf Serverbetriebssystemmaschinen mit von Druckertreiberzuordnungen. Außerdem bietet die Zuordnung von Druckertreibern folgende Optionen:

- Beschränken bestimmter Drucker auf die ausschließliche Verwendung des universellen Citrix Druckertreibers
- Zulassen oder Verhindern der Erstellung von Druckern mit einem bestimmten Treiber
- Ersetzen veralteter oder beschädigter Treiber durch gewünschte Druckertreiber
- Ersetzen von Clienttreibernamen durch einen unter Windows Server verfügbaren Treiber

**Automatische Installation von Druckertreibern verhindern:** Die automatische Installation von Druckertreibern muss deaktiviert sein, damit Konsistenz zwischen Serverbetriebssystemmaschinen gewährleistet ist. Dies kann über Citrix Richtlinien und/oder Microsoft-Richtlinien erreicht werden. Zum Verhindern der automatischen Installation Windows-systemeigener Druckertreiber deaktivieren Sie die Citrix Richtlinieneinstellung Automatische Installation von mitgelieferten Druckertreibern.

**Zuordnen von Clientdruckertreibern:** Jeder Client liefert bei der Anmeldung Informationen zu den clientseitigen Druckern, einschließlich dem Namen des Druckermodells. Bei der automatischen Erstellung der Clientdrucker werden die Namen der Druckertreiber auf dem Windows-Server ausgewählt, die den Namen der Druckermodule entsprechen, die der Client bereitgestellt hat. Beim automatischen Erstellen werden mit diesen identifizierten verfügbaren Druckertreibern umgeleitete Clientdruckwarteschlangen erstellt.

Gehen Sie bei der Erstellung von Regeln für die Treiberersetzung und der Bearbeitung der Druckeinstellungen für zugeordnete Clientdruckertreiber grundsätzlich folgendermaßen vor:

1. Legen Sie die Regeln für die Treiberersetzung für automatisch erstellte Drucker fest, indem Sie die Citrix Richtlinieneinstellung Druckertreiberzuordnung und -kompatibilität konfigurieren. Fügen Sie dabei den Namen des Clientdruckertreibers hinzu und wählen Sie über das Menü Druckertreiber suchen den Servertreiber aus, durch den Sie den Clientdruckertreiber ersetzen möchten. Sie können in dieser Einstellung Platzhalter verwenden. Damit beispielsweise alle HP-Drucker einen bestimmten Treiber verwenden, geben Sie in der Richtlinieneinstellung HP\* an.
2. Zum Ausschließen eines Druckertreibers wählen Sie den Namen des Treibers aus und aktivieren Sie die Einstellung Nicht erstellen.
3. Sie können bei Bedarf eine Treiberzuordnung bearbeiten, eine Zuordnung löschen oder die Reihenfolge der

Treibereinträge in der Liste ändern.

4. Zum Bearbeiten der Druckereinstellungen für zugeordnete Clientdruckertreiber wählen Sie den Druckertreiber aus, klicken Sie auf Einstellungen und geben Sie die Einstellungen wie Druckqualität, Ausrichtung und Farbe an. Wenn Sie eine Druckoption angeben, die der Druckertreiber nicht unterstützt, hat die Option keine Auswirkung. Mit dieser Einstellung werden die gespeicherten Druckereinstellungen überschrieben, die der Benutzer in einer vorherigen Sitzung festgelegt hat.
5. Citrix empfiehlt, das Verhalten der Drucker nach der Zuordnung von Treibern ausführlich zu testen, da einige Druckfunktionen möglicherweise nur über einen bestimmten Treiber zur Verfügung stehen.

Bei der Benutzeranmeldung wird die Clientdruckerkompatibilitätsliste vom System überprüft, bevor die Clientdrucker eingerichtet werden.

## Optimieren der Druckleistung

Verwenden Sie den universellen Druckserver und den universellen Druckertreiber, um die Leistung zu optimieren. Die folgenden Richtlinien steuern die Druckoptimierung und Komprimierung:

- Universelles Drucken - Optimierungsstandards. Gibt die Standardeinstellungen für den universellen Drucker an, wenn er für eine Sitzung erstellt wird:
  - Mit Gewünschte Bildqualität geben Sie das standardmäßige Bildkomprimierungslimit an, das auf universelles Drucken angewendet wird. In der Standardeinstellung ist Standardqualität aktiviert, d. h. Benutzer können Bilder nur mit der Standardqualitäts- oder geringeren Qualitätskomprimierung drucken.
  - Mit Heavyweight-Komprimierung aktivieren oder deaktivieren Sie das Verringern der Bandbreite unter den Komprimierungsgrad, der von "Gewünschte Bildqualität" festgelegt ist; Bildqualität geht nicht verloren. Standardmäßig ist die Heavyweight-Komprimierung deaktiviert.
  - Mit den Einstellungen Zwischenspeichern von Bildern und Schriftarten legen Sie fest, ob Bilder und Schriftarten, die mehrmals im Druckdatenstrom vorhanden sind, zwischengespeichert werden. Sie stellen damit sicher, dass jedes eindeutige Bild oder jede Schriftart nur einmal zum Drucker gesendet wird. Standardmäßig werden eingebettete Bilder und Schriftarten zwischengespeichert.
  - Mit Nicht-Administratoren können diese Einstellungen ändern legen Sie fest, ob Benutzer die Standardeinstellungen für die Druckoptimierung in einer Sitzung ändern können. Standardmäßig können Benutzer die Standardeinstellungen für die Druckoptimierung nicht ändern.
- Universelles Drucken - Bildkomprimierungslimit. Definiert die maximale Qualität und die minimale Komprimierung für Bilder, die mit dem universellen Druckertreiber gedruckt werden. Das Limit für Bildkomprimierung ist standardmäßig auf Beste Qualität (verlustfreie Komprimierung) gesetzt.
- Universelles Drucken - Druckqualitätslimit. Der Höchstwert für Punkte pro Zoll (dpi) zum Erstellen von Ausdrucken in einer Sitzung. In der Standardeinstellung ist kein Limit angegeben.

Standardmäßig werden alle für Netzwerkdrucker bestimmten Druckaufträge von der Serverbetriebssystemmaschine über das Netzwerk direkt an den Druckserver weitergeleitet. Erwägen Sie, Druckaufträge über die ICA-Verbindung zu leiten, wenn das Netzwerk hohe Latenz oder beschränkte Bandbreite aufweist. Deaktivieren Sie hierzu die Citrix RichtlinienEinstellung Direkte Verbindungen zu Druckservern. Bei einer ICA-Verbindung werden die Daten komprimiert gesendet, es wird somit weniger Bandbreite bei der Übertragung der Daten über das WAN gebraucht.

**Verbessern der Sitzungsleistung durch Limitierung der Druckbandbreite:** Beim Drucken von Dateien von Serverbetriebssystemmaschinen auf Benutzerdruckern, können bei anderen virtuellen Kanälen (z. B. Video) aufgrund des Wettbewerbs um die Bandbreite Leistungsverringerungen entstehen, insbesondere dann, wenn Benutzer über langsamere Netze auf Server zugreifen. Um dies zu verhindern, können Sie die für das Drucken verwendete Bandbreite beschränken. Indem Sie die Datenübertragungsrate für den Druck einschränken, stellen Sie im HDX-Datenstrom eine größere Bandbreite

für die Übertragung von Video, Tastatureingaben und Mausdaten zur Verfügung.

Wichtig: Das Druckerbandbreitenlimit wird immer eingehalten, auch wenn keine anderen Kanäle verwendet werden. Verwenden Sie die nachfolgenden Einstellungen der Citrix Richtlinie Bandbreite, um die Druckerbandbreitenlimits für die Sitzung zu beschränken. Führen Sie diese Aufgabe mit Studio aus, um die Limits für die Site festzulegen. Wenn Sie Limits für einzelne Server festlegen möchten, führen Sie diese Aufgabe über die Gruppenrichtlinien-Verwaltungskonsolle in Windows lokal auf jeder Serverbetriebssystemmaschine aus.

- Die Einstellung Bandbreitenlimit für Druckerumleitung dient zur Angabe der zum Drucken verfügbaren Bandbreite in Kilobits pro Sekunde (KBit/s).
  - Die Einstellung Bandbreitenlimit für Druckerumleitung (Prozent) begrenzt die zum Drucken verfügbare Bandbreite auf einen Prozentanteil der insgesamt verfügbaren Bandbreite.
- Hinweis: Zur Verwendung der Einstellung Bandbreitenlimit für Druckerumleitung (Prozent) müssen Sie auch die Einstellung Bandbreitenlimit für Sitzung insgesamt aktivieren.

Wenn Sie Werte für beide Einstellungen eingeben, wird die strengste Einstellung (mit dem niedrigeren Wert) angewendet.

Zum Abrufen von Echtzeitinformationen zur Druckbandbreite verwenden Sie Citrix Director.

## Anzeigen von Druckern und Verwalten der Druckwarteschlangen

In der folgenden Tabelle wird aufgeführt, wo Sie in Ihrer Umgebung Drucker anzeigen und die Druckwarteschlangen verwalten können.

|                                                             | Druckmodell         | UAC aktiviert? | Speicherort                                                                           |
|-------------------------------------------------------------|---------------------|----------------|---------------------------------------------------------------------------------------|
| Clientdrucker (an das Benutzergerät angeschlossene Drucker) | Clientdruckmodell   | On             | Druckverwaltung:<br>Snap-in der<br>Microsoft<br>Management<br>Console                 |
|                                                             |                     | Off            | Vor Windows 8:<br>Systemsteuerung<br>Windows 8:<br>Snap-In<br>Druckverwaltung         |
| Netzwerkdrucker (Drucker auf einem Netzwerkdruckserver)     | Netzwerkdruckmodell | On             | Druckserver ><br>Druckverwaltung<br>Snap-In der<br>Microsoft<br>Management<br>Console |
|                                                             |                     | Off            | Druckserver ><br>Systemsteuerung                                                      |
| Netzwerkdrucker (Drucker auf einem Netzwerkdruckserver)     | Clientdruckmodell   | On             | Druckserver ><br>Druckverwaltung<br>Snap-In der                                       |

|                                                                                                                                 | Druckmodell         | UAC<br>aktiviert? | Microsoft<br>Speicherort<br>Management<br>Console                             |
|---------------------------------------------------------------------------------------------------------------------------------|---------------------|-------------------|-------------------------------------------------------------------------------|
|                                                                                                                                 |                     | Off               | Vor Windows 8:<br>Systemsteuerung<br>Windows 8:<br>Snap-In<br>Druckverwaltung |
| Lokale Netzwerkserverdrucker (Drucker von einem Netzwerkdruckserver, die einer Serverbetriebssystemmaschine hinzugefügt werden) | Netzwerkdruckmodell | On                | Druckserver ><br>Systemsteuerung                                              |
|                                                                                                                                 |                     | Off               | Druckserver ><br>Systemsteuerung                                              |

Hinweis: Druckwarteschlangen für Netzwerkdrucker, die das Netzwerkdruckmodell verwenden, sind privat und können nicht über das System verwaltet werden.

# Lizenzierung

Nov 02, 2016

## Hinweis

Studio und Director unterstützen Citrix Lizenzserver VPX nicht. Weitere Informationen zu Citrix Lizenzserver VPX finden Sie in der Citrix Lizenzierungsdokumentation.

Sie können die Lizenzierung in Studio verwalten und nachverfolgen, wenn der Lizenzserver in derselben Domäne wie Studio oder in einer vertrauenswürdigen Domäne ist. Informationen zu anderen Lizenzierungsaufgaben finden Sie unter

— *Lizenzieren des Produkts*

Sie müssen für die nachfolgend beschriebenen Aufgaben Volladministrator für die Lizenzierung sein, außer um die Lizenzinformationen anzuzeigen. Zum Anzeigen der Lizenzinformationen in Studio muss der Administrator mindestens Lesezugriff als delegierter Administrator für die Lizenzierung haben. Die integrierten Rollen des Volladministrators und des Administrators mit Leserechten haben diese Berechtigung.

In der folgenden Tabelle werden die unterstützten Editionen und Lizenzierungsmodelle aufgeführt:

| Produkte   | Editionen                                                                                                   | Lizenzmodelle                                                                  |
|------------|-------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------|
| XenApp     | <ul style="list-style-type: none"><li>• Platinum</li><li>• Enterprise</li><li>• Erweitert</li></ul>         | CCU                                                                            |
| XenDesktop | <ul style="list-style-type: none"><li>• Platinum</li><li>• Enterprise</li><li>• App</li><li>• VDI</li></ul> | <ul style="list-style-type: none"><li>• Benutzer/Gerät</li><li>• CCU</li></ul> |

Zum Anzeigen von Lizenzinformationen wählen Sie im Studio-Navigationsbereich Konfiguration und dann Lizenzierung. Eine Zusammenfassung der Lizenzverwendung sowie Einstellungen für die Site werden zusammen mit einer Liste aller Lizenzen angezeigt, die aktuell auf dem angegebenen Lizenzserver installiert sind.

Zum Verwalten von Lizenzen wählen Sie im Studio-Navigationsbereich Konfiguration und dann Lizenzierung. Sie können dann folgende Schritte ausführen:

- Herunterladen einer Lizenz von Citrix:
  1. Klicken Sie im Aktionsbereich auf Lizenzen zuteilen.
  2. Geben Sie den Lizenzzugangscode an, der per E-Mail von Citrix bereitgestellt wird.
  3. Wählen Sie ein Produkt und klicken Sie auf Lizenzen zuteilen. Alle Lizenzen, die für das Produkt verfügbar sind, sind zugeteilt und heruntergeladen. Wenn Sie alle Lizenzen für einen bestimmten Lizenzzugangscode zuteilen und herunterladen, können Sie den Lizenzzugangscode nicht erneut verwenden. Zum Durchführen weiterer Transaktionen mit diesem Code melden Sie sich bei My Account an.
- Hinzufügen von Lizenzen, die auf dem lokalen Computer oder im Netzwerk gespeichert sind:

1. Klicken Sie im Aktionsbereich auf Lizenzen hinzufügen.
  2. Navigieren Sie zu einer Lizenzdatei und fügen Sie sie dem Lizenzserver hinzu.
- Ändern des Lizenzservers:
    1. Klicken Sie im Aktionsbereich auf Lizenzserver ändern.
    2. Geben Sie die Adresse des Lizenzservers im Format name:port an ("name" = DNS-, NetBIOS- oder IP-Adresse). Wenn Sie keine Portnummer angeben, wird der Standardport (27000) verwendet.
  - Auswählen des Lizenztyps:
    - Beim Konfigurieren der Site werden Sie nach Angabe des Lizenzservers aufgefordert, den zu verwendenden Lizenztyp auszuwählen. Stehen auf dem Server keine Lizenzen zur Verfügung, wird automatisch die Option zur Verwendung des Produkts während einer 30-tägigen Testphase ohne Lizenz ausgewählt.
    - Stehen auf dem Server Lizenzen zur Verfügung, werden die entsprechenden Informationen angezeigt. Der Benutzer kann dann die gewünschte Lizenz auswählen. Alternativ können Sie dem Server eine Lizenzdatei hinzufügen und diese dann auswählen.
  - Ändern von Produktedition und Lizenzierungsmodell:
    1. Klicken Sie im Aktionsbereich auf Produktedition bearbeiten.
    2. Aktualisieren Sie die entsprechenden Optionen.
  - Zum Öffnen der License Administration Console wählen Sie im Aktionsbereich License Administration Console. Die Konsole wird normalerweise sofort angezeigt. Wenn das Dashboard jedoch mit Kennwortschutz konfiguriert wurde, werden Sie aufgefordert, die Anmeldeinformationen für die License Administration Console einzugeben. Informationen zur Verwendung der Konsole finden Sie unter  
*— Lizenzieren des Produkts*  
 .
  - Hinzufügen eines Lizenzadministrators:
    1. Klicken Sie im mittleren Bereich auf die Registerkarte Lizenzierungsadministratoren.
    2. Wählen Sie im Aktionsbereich Lizenzadministrator hinzufügen aus.
    3. Navigieren Sie zu dem Benutzer, den Sie als Administrator hinzufügen möchten, und wählen Sie die Berechtigungen.
  - Bearbeiten oder Löschen eines Lizenzadministrators: Wenn Sie einen Administrator auswählen, werden die Optionen "Lizenzadministrator bearbeiten" (zum Ändern der Administratorberechtigungen für den Administrator) und "Lizenzadministrator löschen" im Aktionsbereich angezeigt.
    1. Wählen Sie im mittleren Bereich die Registerkarte Lizenzierungsadministratoren und dann den Administrator, den Sie löschen oder bearbeiten möchten.
    2. Klicken Sie im Aktionsbereich auf Lizenzadministrator bearbeiten bzw. auf Lizenzadministrator löschen.
  - Hinzufügen einer Lizenzadministratorgruppe:
    1. Klicken Sie im mittleren Bereich auf die Registerkarte Lizenzierungsadministratoren.
    2. Wählen Sie im Aktionsbereich Lizenzadministratorgruppe hinzufügen aus.
    3. Navigieren Sie zu der Gruppe, deren Mitglieder Sie als Administratoren hinzufügen möchten, und wählen Sie die Berechtigungen. Beim Hinzufügen einer Active Directory-Gruppe werden den Benutzern dieser Gruppe Lizenzierungsadministratorberechtigungen erteilt.
  - Bearbeiten oder Löschen einer Lizenzadministratorgruppe:
    1. Wählen Sie im mittleren Bereich die Registerkarte Lizenzierungsadministratoren und dann die Administratorgruppe, die Sie löschen oder bearbeiten möchten. Wenn Sie eine Lizenzadministratorgruppe auswählen, werden die Optionen "Lizenzadministratorgruppe bearbeiten" (zum Ändern der Administratorberechtigungen für diese Gruppe) und "Lizenzadministratorgruppe löschen" im Aktionsbereich angezeigt.
    2. Klicken Sie im Aktionsbereich auf Lizenzadministratorgruppe bearbeiten bzw. auf Lizenzadministratorgruppe löschen.

# Verbindungen und Ressourcen

May 10, 2016

Sie erstellen die erste Verbindung mit Hosting-Ressourcen, wenn Sie eine Site erstellen. Später können Sie die Verbindung ändern und neue Verbindungen erstellen. Lesezugriffadministratoren können die Verbindung und Ressourcendetails anzeigen. Sie müssen Volladministrator sein, um Verwaltungsaufgaben an Verbindungen und Ressourcen durchzuführen.

## Erstellen einer Verbindung und von Ressourcen

Die Hosting-Ressourcen müssen vor Erstellen einer Verbindung verfügbar sein.

1. Wählen Sie im Studio-Navigationsbereich Konfiguration > Hosting.
2. Wählen Sie im Aktionsbereich Verbindung und Ressourcen hinzufügen.
3. Wählen Sie Neue Verbindung erstellen.
4. Treffen Sie auf der Seite Verbindung folgende Auswahl:
  - Wählen Sie den Verbindungstyp und geben Sie einen Namen für die Verbindung ein. Wählen Sie einen Namen, anhand dessen Administratoren den Hosttyp und die Adresse der Bereitstellung erkennen können. Welche zusätzlichen Informationen erforderlich sind, hängt von dem ausgewählten Verbindungstyp ab.

| Verbindungstyp                                                                                                                     | Erforderliche Informationen                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Citrix XenServer, Microsoft System Center Virtual Machine Manager, VMware vSphere oder Microsoft Configuration Manager Wake-On-LAN | <p>Geben Sie die Verbindungs-URL, den Benutzernamen und das Kennwort ein.</p> <ul style="list-style-type: none"><li>• Für XenServer empfiehlt Citrix die Verwendung von HTTPS zur Sicherung der Kommunikation. Um HTTP zu verwenden, müssen Sie das standardmäßig mit XenServer installierte SSL-Zertifikat durch ein Zertifikat von einer vertrauenswürdigen Zertifizierungsstelle ersetzen; Informationen finden Sie unter <a href="#">CTX128656</a>.</li><li>• Bei XenServer können Sie die neue Verbindung bearbeiten und die Hypervisors für hohe Verfügbarkeit auswählen, sofern hohe Verfügbarkeit für XenServer aktiviert ist.</li></ul>                                                                                                                        |
| Citrix CloudPlatform oder Amazon Web Services (AWS)                                                                                | <p>Geben Sie Verbindungs-URL, API-Schlüssel und geheimen Schlüssel an.</p> <ul style="list-style-type: none"><li>• Sie können über die Schaltfläche "Durchsuchen" eine vom Cloud-Administrator bereitgestellte Schlüsselimportdatei ansteuern, um den API-Schlüssel und den geheimen Schlüssel anzugeben.</li><li>• Die für das AWS-Rootkonto von der AWS-Konsole abgerufene Anmeldeinformationsdatei hat nicht das gleiche Format wie die Anmeldeinformationsdateien, die für Standard-AWS-Benutzer heruntergeladen werden. Deshalb kann diese Datei von Studio nicht zum Auffüllen der Felder "API-Schlüssel" und "Geheimer Schlüssel" verwendet werden. Wenn Sie Studio in einer AWS-Umgebung ausführen, verwenden Sie AWS-IAM-Anmeldeinformationsdateien.</li></ul> |

- Wählen Sie die Tools, die Sie zum Erstellen von virtuellen Maschinen verwenden möchten. Für Hypervisors, die GPU-Ressourcen bereitstellen, wählen Sie Studio-Tools.
5. Wählen Sie auf der Seite Speicher die Speichertypen und Geräte. Wenn Sie Maschinenerstellungsdienste verwenden, wählen Sie das Netzwerk und Speicherressourcen für die neuen virtuellen Maschinen. Wenn Sie einen freigegebenen Speicher über XenServer-Verbindungen verwenden, können Sie IntelliCache aktivieren, um die Last auf dem Speichergerät zu reduzieren. Weitere Informationen zum Verwenden von IntelliCache finden Sie weiter unten.

6. Wenn die Verbindung GPU-Funktionen bietet, wählen Sie die Option zur Verwendung der Grafikvirtualisierung und anschließend einen GPU-Typ und eine GPU-Gruppe.
7. Geben Sie einen Namen für die Ressourcen ein.

### Erstellen von Ressourcen und einer Verbindung aus einer vorhandenen Verbindung

1. Wählen Sie im Studio-Navigationsbereich Konfiguration > Hosting.
2. Wählen Sie im Aktionsbereich Verbindung und Ressourcen hinzufügen.
3. Wählen Sie Vorhandene Verbindung verwenden und dann die gewünschte Verbindung.
4. Wählen Sie die Tools, die Sie zum Erstellen von virtuellen Maschinen verwenden möchten. Für Hypervisors, die GPU-Ressourcen bereitstellen, wählen Sie Studio-Tools. Wenn die Verbindung GPU-Funktionen bietet, wählen Sie die Option zur Verwendung der Grafikvirtualisierung und anschließend einen GPU-Typ und eine GPU-Gruppe.
5. Geben Sie einen Namen für die Ressourcen ein.

### Speicher hinzufügen

1. Wählen Sie im Studio-Navigationsbereich Konfiguration > Hosting.
2. Wählen Sie eine Verbindung und dann im Aktionsbereich Speicher hinzufügen.
3. Wählen Sie den hinzuzufügenden Speicher aus.

### Bearbeiten des Speichers

1. Wählen Sie im Studio-Navigationsbereich Konfiguration > Hosting.
2. Wählen Sie einen Ressourceneintrag unter einer Verbindung und dann im Aktionsbereich "Speicher bearbeiten".
3. Aktivieren oder deaktivieren Sie auf der Seite Standardspeicher die Kontrollkästchen für die Speicherorte, an denen virtuelle Maschinen gespeichert werden. Wenn Sie einen Speicherort deaktivieren, der neue Maschinen angenommen hat, nimmt dieser keine weiteren neuen Maschinen an. Vorhandene Maschinen verwenden weiterhin diesen Speicherort (und schreiben Daten hinein). Daher kann ein Speicherort voll werden, selbst wenn keine neuen Maschinen mehr angenommen werden.  
Bei Verwendung eines PVD-Speichers aktivieren oder deaktivieren Sie die Kontrollkästchen auf der Seite PVD-Speicher ebenfalls.

### Bearbeiten einer Verbindung

Verwenden Sie diese Vorgehensweise nicht, um eine Verbindung umzubenennen oder eine neue Verbindung zu erstellen. Dafür sind andere Schritte erforderlich.

1. Wählen Sie im Studio-Navigationsbereich Konfiguration > Hosting.
2. Wählen Sie die Verbindung und dann im Aktionsbereich Verbindung bearbeiten.
  - Zum Ändern der Verbindungsadresse und -anmeldeinformationen klicken Sie auf der Seite Verbindungseigenschaften auf Einstellungen bearbeiten und geben die neuen Informationen ein.  
Sie können die GPU-Einstellungen für eine Verbindung nicht ändern, da Maschinenkataloge, die auf diese Ressource zugreifen, ein entsprechendes GPU-spezifisches Masterimage verwenden müssen. Erstellen Sie eine neue Verbindung.
  - Zum Angeben der Server mit hoher Verfügbarkeit für eine XenServer-Verbindung klicken Sie auf der Seite Verbindungseigenschaften auf Server mit hoher Verfügbarkeit bearbeiten. Citrix empfiehlt, dass Sie alle Server im Pool auswählen, um die Kommunikation mit XenServer zu ermöglichen, wenn der Poolmaster ausfällt.
  - Für eine Wake-On-LAN-Verbindung unter Microsoft System Center Configuration Manager geben Sie auf der Seite Erweitert ConfigMgr Wake Proxy, Magic Packets und Paketübertragungsinformationen an.
  - Zum Konfigurieren einer Drosselung basierend auf Schwellenwerten gleichzeitiger Aktionen an der Verbindung führen Sie die nachfolgend aufgeführten Schritte durch. (Dies kann nützlich sein, wenn durch die



Energieverwaltungseinstellungen der gleichzeitige Start zu vieler oder zu weniger Maschinen zugelassen wird.)

- Geben Sie auf der Seite Erweitert für Gleichzeitige Aktionen (alle Typen) und Gleichzeitige Updates für Personal vDisk-Inventar zwei Werte ein: die maximale absolute Zahl Aktionen/Updates, die gleichzeitig an dieser Verbindung auftreten darf und den Prozentsatz aller Maschinen, die diese Verbindung verwenden. Sie müssen beide Werte angeben, angewendet wird jedoch der geringere der konfigurierten Werte.  
Beispiel: Wird in einer Bereitstellung mit 34 Maschinen die Einstellung Gleichzeitige Aktionen (alle Typen) auf einen absoluten Wert von 10 und einen Prozentsatz von 10 festgelegt, wird als tatsächliches Limit 3 angewendet (d. h. 10 Prozent von 34 auf die nächste Ganzzahl gerundet – ein kleinerer Wert als die absolute Zahl von 10 Maschinen).
- Sie geben die maximale Anzahl neuer Aktionen pro Minute an. Dies ist eine absolute Zahl.  
Hinweis: Geben Sie die Informationen auf der Seite Erweitert im Feld Verbindungsoptionen nur unter der Anleitung eines Supportmitarbeiters von Citrix ein.

## Aktivieren und Deaktivieren des Wartungsmodus für eine Verbindung

Wenn Sie den Wartungsmodus für eine Verbindung aktivieren, können keine neuen Energieaktionen auf in dieser Verbindung gespeicherten Maschinen stattfinden. Benutzer können keine Verbindung mit einer Maschine herstellen, wenn sie im Wartungsmodus ist. Wenn Benutzer bereits verbunden sind, wird der Wartungsmodus wirksam, sobald sich die Benutzer abmelden.

1. Wählen Sie im Studio-Navigationsbereich Konfiguration > Hosting.
2. Wählen Sie die Verbindung aus. Zum Aktivieren des Wartungsmodus wählen Sie im Aktionsbereich Wartungsmodus einschalten. Zum Deaktivieren des Wartungsmodus wählen Sie Wartungsmodus ausschalten.

Sie können den Wartungsmodus auch für einzelne Maschinen ein- und ausschalten (siehe unten).

## Löschen einer Verbindung

Achtung: Das Löschen einer Verbindung kann zur Folge haben, dass eine große Zahl von Maschinen gelöscht wird, Datenverlust eingeschlossen. Stellen Sie sicher, dass die Benutzerdaten auf den betroffenen Maschinen gesichert wurden oder nicht mehr benötigt werden.

Vor dem Löschen einer Verbindung müssen Sie Folgendes sicherstellen:

- Alle Benutzer sind von den in dieser Verbindung gespeicherten Maschinen abgemeldet.
  - Es werden keine getrennten Benutzersitzungen ausgeführt.
  - Der Wartungsmodus wird für gepoolte und dedizierte Maschinen aktiviert.
  - Alle Maschinen in Maschinenkatalogen sind ausgeschaltet.
1. Wählen Sie im Studio-Navigationsbereich Konfiguration > Hosting.
  2. Wählen Sie die Verbindung und dann im Aktionsbereich Verbindung löschen.
  3. Wenn für die Verbindung Maschinen gespeichert sind, werden Sie gefragt, ob die Maschinen gelöscht werden sollen. Wenn dies der Fall ist, geben Sie an, was mit dem zugewiesenen Active Directory-Computerkonten passieren soll. Ein Maschinenkatalog kann nicht mehr verwendet werden, wenn Sie eine Verbindung löschen, auf die dieser Katalog verweist. Verweist ein Katalog auf diese Verbindung, haben Sie die Option zum Löschen des Katalogs. Stellen Sie vor dem Löschen eines Katalogs sicher, dass er nicht von anderen Verbindungen verwendet wird.

## Umbenennen einer Verbindung

1. Wählen Sie im Studio-Navigationsbereich Konfiguration > Hosting.
2. Wählen Sie die Verbindung und dann im Aktionsbereich Verbindung umbenennen.

## Anzeigen von Maschinendetails für eine Verbindung

1. Wählen Sie im Studio-Navigationsbereich Konfiguration > Hosting.
2. Wählen Sie die Verbindung aus und dann im Aktionsbereich Maschinen anzeigen.

Im oberen Bereich werden die Maschinen angezeigt, auf die über die Verbindung zugegriffen wird. Wählen Sie eine Maschine aus, um die Details im unteren Bereich anzuzeigen. Für geöffnete Sitzungen werden auch Sitzungsdetails angezeigt.

Sie können das Suchfeature verwenden, um Maschinen schnell aufzufinden. Wählen Sie entweder eine gespeicherte Suche aus der Liste im oberen Bereich des Bildschirms aus oder erstellen Sie eine neue Suche. Sie können nach dem Maschinennamen suchen, indem Sie den ganzen Namen oder einen Teil des Namens eingeben. Alternativ können Sie auch einen Ausdruck für eine erweiterte Suche erstellen. Klicken Sie auf die Erweiterungsschaltfläche, um einen Ausdruck zu erstellen, und wählen Sie dann aus den angezeigten Listen Eigenschaften und Operatoren aus.

## Verwalten von Maschinen einer Verbindung

1. Wählen Sie im Studio-Navigationsbereich Konfiguration > Hosting.
2. Wählen Sie eine Verbindung und dann im Aktionsbereich Maschinen anzeigen.
3. Wählen Sie eine der folgenden Aktionen im Aktionsbereich. Abhängig vom Maschinenzustand und dem Verbindungstyp sind einige Aktionen möglicherweise nicht verfügbar.
  - Starten: startet die Maschine, wenn diese ausgeschaltet oder angehalten ist.
  - Anhalten: hält die Maschine an, ohne sie herunterzufahren, und aktualisiert die Liste der Maschinen.
  - Herunterfahren: Das Betriebssystem wird aufgefordert, herunterzufahren.
  - Herunterfahren erzwingen: Die Maschine wird zwingend abgeschaltet und die Liste der Maschinen wird aktualisiert.
  - Neu starten: Das Betriebssystem wird heruntergefahren und die Maschine wird dann neu gestartet. Wenn das Betriebssystem diese Aufgaben nicht ausführen kann, bleibt der Desktop im aktuellen Zustand.
  - Wartungsmodus aktivieren: Sollen vorübergehend keine Verbindungen mit einer Maschine hergestellt werden, versetzen Sie die Maschine in den Wartungsmodus. Benutzer können keine Verbindung mit einer Maschine in diesem Zustand herstellen. Wenn Benutzer verbunden sind, wird der Wartungsmodus wirksam, sobald sich die Benutzer abmelden.

Anweisungen zum Aktivieren bzw. Deaktivieren des Wartungsmodus für alle Maschinen, auf die über eine Verbindung zugegriffen wird, finden Sie oben.
  - Aus Bereitstellungsgruppe entfernen: Beim Entfernen einer Maschine aus einer Bereitstellungsgruppe wird sie nicht aus dem von der Bereitstellungsgruppe verwendeten Maschinenkatalog gelöscht. Sie können eine Maschine nur entfernen, wenn keine Benutzer mit ihr verbunden sind (aktivieren Sie den Wartungsmodus, um zu verhindern, dass Benutzer eine Verbindung herstellen, während Sie die Maschine entfernen).
  - Löschen: Wenn Sie eine Maschine löschen, können Benutzer nicht mehr darauf zugreifen und die Maschine wird aus dem Maschinenkatalog gelöscht. Stellen Sie vor dem Löschen einer Maschine sicher, dass alle Benutzerdaten gesichert wurden oder nicht mehr benötigt werden. Sie können eine Maschine nur löschen, wenn keine Benutzer mit ihr verbunden sind (aktivieren Sie den Wartungsmodus, um zu verhindern, dass Benutzer eine Verbindung herstellen, während Sie die Maschine löschen).

Bei Aktionen, bei denen eine Maschine heruntergefahren wird, wird diese ausgeschaltet, wenn das Herunterfahren nicht innerhalb von 10 Minuten erfolgt. Wenn Windows versucht, während des Herunterfahrens Updates zu installieren, besteht die Gefahr, dass die Maschine ausgeschaltet wird, bevor die Updates abgeschlossen sind.

## Löschen, Umbenennen oder Testen von Ressourcen

1. Wählen Sie im Studio-Navigationsbereich Konfiguration > Hosting.
2. Wählen Sie die Ressource aus und wählen Sie dann den entsprechenden Eintrag im Aktionsbereich: Ressourcen löschen,

Ressourcen umbenennen oder Ressourcen testen.

## Verwenden von IntelliCache für XenServer-Verbindungen

Durch den Einsatz von IntelliCache werden gehostete VDI-Bereitstellungen kostengünstiger, da eine Kombination aus freigegebenem und lokalem Speicher verwendet werden kann. Dies verbessert die Leistung und reduziert den Datenverkehr im Netzwerk. Das Masterimage aus dem freigegebenen Speicher wird im lokalen Speicher zwischengespeichert, wodurch die Anzahl der Lesevorgänge im freigegebenen Speicher reduziert wird. Bei gemeinsam genutzten Desktops werden Schreibvorgänge auf den differenzierenden Festplatten in den lokalen Speicher auf dem Host und nicht in den gemeinsam genutzten Speicher geschrieben.

- Der freigegebene Speicher muss NFS sein, wenn Sie IntelliCache verwenden.
- Citrix empfiehlt die Verwendung eines lokalen Speichergeräts mit hoher Leistung, um eine schnellstmögliche Datenübertragung zu gewährleisten.

Um IntelliCache zu verwenden, müssen Sie es in diesem Produkt und XenServer aktivieren.

- Bei der Installation von XenServer, wählen Sie **Enable thin provisioning (Optimized storage for XenDesktop)**. Citrix bietet keine Unterstützung für gemischte Serverpools, auf denen IntelliCache auf manchen Servern aktiviert ist und auf anderen nicht. Weitere Informationen finden Sie in der Dokumentation für XenServer.
- In XenApp und XenDesktop ist IntelliCache standardmäßig deaktiviert. Sie können die Einstellung nur beim Erstellen einer XenServer-Verbindung ändern, IntelliCache kann später nicht deaktiviert werden. Gehen Sie folgendermaßen vor, wenn Sie eine XenServer-Verbindung von Studio aus hinzufügen:
  - Wählen Sie als Speichertyp **Freigegeben** aus.
  - Aktivieren Sie das Kontrollkästchen **IntelliCache verwenden**.

## Verbindungstimer

Sie können mit Richtlinieneinstellungen drei Verbindungstimer konfigurieren:

- **Timer für längste Verbindung:** Diese Einstellung legt die Höchstdauer einer ununterbrochenen Verbindung zwischen einem Benutzergerät und einem Desktop fest. Verwenden Sie die Richtlinieneinstellungen **"Sitzungsverbindungstimer"** und **"Sitzungsverbindung - Timerintervall"**.
- **Timer für inaktive Verbindungen:** Diese Einstellung legt fest, wie lange eine ununterbrochene Verbindung zwischen einem Benutzergerät und einem virtuellen Desktop erhalten wird, wenn keine Eingabe vom Benutzer erfolgt. Verwenden Sie die Richtlinieneinstellungen **"Sitzungsleerlaufstimer"** und **"Sitzungsleerlauf - Timerintervall"**.
- **Timer für getrennte Verbindungen:** Diese Einstellung legt fest, wie lange ein getrennter, gesperrter virtueller Desktop gesperrt bleibt, bis die Sitzung abgemeldet wird. Verwenden Sie die Richtlinieneinstellungen **"Timer für getrennte Sitzung"** und **"Getrennte Sitzungen - Timerintervall"**.

Wenn Sie eine dieser Einstellungen aktualisieren, achten Sie darauf, dass sie in der ganzen Bereitstellung konsistent sind.

# Verbindungsleasing

May 10, 2016

Um sicherzustellen, dass die Sitedatenbank immer verfügbar ist, empfiehlt Citrix, unter Befolgung der bewährten Methoden zur hohen Verfügbarkeit von Microsoft mit einer fehlertoleranten SQL Server-Bereitstellung zu beginnen. Delivery Controller können jedoch aufgrund von Netzwerkproblemen und Unterbrechungen möglicherweise nicht auf die Datenbank zugreifen, was dazu führt, dass Benutzer keine Verbindung mit ihren Anwendungen oder Desktops herstellen können.

Das Feature für Verbindungsleasing ergänzt die bewährten Methoden zur hohen Verfügbarkeit bei SQL Server, da es Benutzern die Wiederverbindung mit den zuletzt verwendeten Anwendungen und Desktops ermöglicht, selbst wenn die Sitedatenbank nicht verfügbar ist.

Benutzer verfügen zwar über eine große Anzahl veröffentlichter Ressourcen, verwenden jedoch oft nur wenige davon regelmäßig. Wenn Sie das Verbindungsleasing aktivieren, werden die Benutzerverbindungen mit den zuletzt verwendeten Anwendungen und Desktops im Normalbetrieb (wenn die Datenbank verfügbar ist) von jedem Controller zwischengespeichert.

Die auf jedem Controller generierten Leases werden in die Sitedatenbank für eine regelmäßige Synchronisierung mit den anderen Controllern der Site hochgeladen. Neben den Leases werden in jedem Controllercache Informationen zu Anwendung, Desktop, Symbol und Worker gespeichert. Lease und zugehörige Informationen werden auf dem lokalen Datenträger eines jeden Controllers gespeichert. Wenn die Datenbank nicht mehr verfügbar ist, wechselt der Controller in den Leasingverbindungsmodus und beim Versuch der Herstellung oder des Neuaufbaus einer Verbindung mit einer kürzlich verwendeten Anwendung oder einem kürzlich verwendeten Desktop von StoreFront aus werden die zwischengespeicherten Vorgänge "abgespielt".

Die Verbindungen werden für eine Leasedauer von zwei Wochen zwischengespeichert. Wenn die Datenbank nicht verfügbar ist, besteht daher über StoreFront Zugriff auf die Desktops und Anwendungen, die in den vorherigen zwei Wochen gestartet wurden. Auf Desktops und Anwendungen, die nicht während der zweiwöchigen Leasedauer gestartet wurden, kann jedoch nicht zugegriffen werden, wenn die Datenbank nicht verfügbar ist. Wenn eine Anwendung z. B. vor drei Wochen gestartet wurde, ist ihre Lease abgelaufen und sie kann bei Nichtverfügbarkeit der Datenbank nicht gestartet werden. Leases für langwährende aktive oder getrennte Anwendungen oder Desktopsitzungen werden verlängert und somit nicht als abgelaufen behandelt.

Standardmäßig gilt das Verbindungsleasing für die komplette Site. Sie können jedoch alle Leases für bestimmte Benutzer widerrufen, wodurch verhindert wird, dass diese auf Anwendungen oder Desktops zugreifen können, wenn der Controller im Leasingverbindungsmodus ausgeführt wird. Mehrere weitere Registrierungseinstellungen gelten pro Controller.

## Überlegungen und Einschränkungen

Das Verbindungsleasing kann zwar die Verbindungsresilienz und die Produktivität der Benutzer verbessern, es sind jedoch Überlegungen im Hinblick auf Verfügbarkeit, Betrieb und Leistung anderer Funktionen anzustellen.

Das Verbindungsleasing wird für servergehostete Anwendungen und Desktops und statische (zugewiesene) Desktops unterstützt. Es wird nicht unterstützt für gepoolte VDI-Desktops oder Benutzer, denen kein Desktop zugewiesen wurde, wenn die Datenbank betriebsunfähig wird.

Wenn der Controller im Leasingverbindungsmodus ist, gilt Folgendes:

- Administratoren können Studio, Director und die PowerShell-Konsole nicht verwenden.
- Workspace Control ist nicht verfügbar. Wenn sich ein Benutzer bei Receiver anmeldet, werden Sitzungen nicht

automatisch wieder verbunden. Der Benutzer muss die Anwendung neu starten.

- Wenn eine neue Lease unmittelbar vor dem Nichtverfügbarwerden der Datenbank erstellt wurde, die Leaseinformationen jedoch noch nicht auf allen Controllern synchronisiert wurden, können Benutzer möglicherweise eine betroffene Ressource nicht starten.
- Benutzer servergehosteter Anwendungen und Desktops können möglicherweise mehr Sitzungen verwenden als das für sie konfigurierte Sitzungslimit zulässt. Beispiel:
  - Ein Benutzer stellt eine Verbindung von einem Gerät (extern über NetScaler Gateway) her, während der Controller im normalen Verbindungsmodus ist. Später stellt der Benutzer von einem anderen Gerät im LAN eine Verbindung her, während Controller im Leasingverbindungsmodus ist. In diesem Fall findet möglicherweise kein Roaming der Sitzung statt.
  - Die Wiederverbindung einer Sitzung kann fehlschlagen, wenn eine Anwendung startet, kurz bevor die Datenbank betriebsunfähig wird. In diesem Fall werden eine neue Sitzung und Anwendungsinstanz gestartet.
- Bei statischen (zugewiesenen) Desktops findet keine Energieverwaltung statt. VDAs, die beim Umschalten des Controllers in den Leasingverbindungsmodus ausgeschaltet werden, stehen erst wieder zur Verfügung, wenn die Datenbankverbindung wiederhergestellt ist, es sei denn, der Administrator schaltet sie manuell ein.
- Sind Sitzungsvorabstart und Sitzungsfortbestehen aktiviert, werden neue Vorabstartsitzungen nicht gestartet. Vorab gestartete und fortbestehende Sitzungen werden während der Nichtverfügbarkeit der Datenbank nicht entsprechend den konfigurierten Schwellenwerten beendet.
- Die Lastverwaltung der Site kann beeinträchtigt werden. Serverbasierte Verbindungen werden an den zuletzt verwendeten VDA geleitet. Lastauswertungsprogramme (und insbesondere Sitzungszahlregeln) werden möglicherweise überschritten.
- Der Controller wechselt nicht in den Leasingverbindungsmodus, wenn Sie die Datenbank mit SQL Server Management Studio offline nehmen. Verwenden Sie stattdessen eine der folgenden Transact-SQL-Anweisungen:
  - ALTER DATABASE <Datenbankname> SET OFFLINE WITH ROLLBACK IMMEDIATE
  - ALTER DATABASE <Datenbankname> SET OFFLINE WITH ROLLBACK AFTER <Sekunden>Beide Anweisungen beenden ausstehende Transaktionen und trennen die Verbindung des Controllers mit der Datenbank. Der Controller wechselt dann in den Leasingverbindungsmodus.

Beim Verbindungsleasing gibt es zwei kurze Intervalle, während derer Benutzer keine Verbindung (wieder)herstellen können: (1) zwischen dem Zeitpunkt, zu dem die Datenbank betriebsunfähig wird bis zum Umschalten des Controllers in den Leasingverbindungsmodus und (2) vom Zeitpunkt des Umschaltens des Controllers vom Leasingverbindungsmodus bis der Datenbankzugriff vollständig wiederhergestellt ist und die VDAs erneut registriert sind.

Weitere Überlegungen finden Sie unter [XenDesktop 7.6 Connection Leasing Design Considerations](#).

## Konfigurieren und Bereitstellen

Beim Konfigurieren der Bereitstellung zum Ermöglichen des Verbindungsleasings ist Folgendes zu beachten:

- VDAs müssen mindestens in Version 7.6 und die Maschinenkataloge und Bereitstellungsgruppen, die diese Maschinen verwenden, müssen in der entsprechenden Mindestversion vorliegen.
- Die Speicherplatzanforderungen der Sitedatenbank sind größer.
- Jeder Controller erfordert zusätzlichen Speicherplatz für die zwischengespeicherten Leasedateien.

Das Verbindungsleasing ist standardmäßig aktiviert.

Sie können das Verbindungsleasing im PowerShell-SDK oder in der Windows-Registrierung deaktivieren und aktivieren. Im PowerShell-SDK können Sie auch aktuelle Leases entfernen. Die folgenden PowerShell-Cmdlets wirken sich auf das Verbindungsleasing aus. Informationen hierzu finden Sie in der Cmdlet-Hilfe.

- Set-BrokerSite -ConnectionLeasingEnabled \$true | \$false: Aktiviert bzw. deaktiviert das Verbindungsleasing. Standard =

\$true

- Get-BrokerServiceAddedCapability: Gibt "ConnectionLeasing" für den lokalen Controller aus.
- Get-BrokerLease: Ruft alle Leases oder einen gefilterten Satz aktueller Leases auf.
- Remove-BrokerLease: Markiert alle Leases oder einen gefilterten Satz Leases zum Löschen.
- Update-BrokerLocalLeaseCache: aktualisiert den Cache für das Verbindungsleasing auf dem lokalen Controller. Bei der nächsten Synchronisierung werden die Daten neu synchronisiert.

# Virtuelle IP-Adressen und virtuelles Loopback

May 10, 2016

Hinweis: Diese Features gelten nur für Maschinen unter Windows Server 2008 R2 und Windows Server 2012 R2. Sie gelten nicht für Windows-Desktopbetriebssystemmaschinen.

Die Microsoft virtuelle IP-Adresse stellt einer veröffentlichten Anwendung eine eindeutige dynamisch zugeordnete IP-Adresse für jede Sitzung bereit. Mit dem Citrix Feature des virtuellen Loopbacks können Sie Anwendungen, die mit dem lokalen Host (localhost) kommunizieren (normalerweise 127.0.0.1), so konfigurieren, dass sie eine eindeutige virtuelle Loopbackadresse im Bereich des lokalen Hosts verwenden (127.\*).

Einige Anwendungen, z. B. CRM oder CTI, verwenden eine IP-Adresse für die Adressierung, Lizenzierung, Identifizierung und andere Zwecke und erfordern daher eine eindeutige IP-Adresse oder eine Loopbackadresse in Sitzungen. Andere Anwendungen binden sich möglicherweise an einen statischen Port an, sodass das Starten weiterer Instanzen einer Anwendung in Mehrbenutzerumgebungen fehlschlägt, da der Port bereits verwendet wird. Damit solche Anwendungen in einer XenApp-Umgebung richtig ausgeführt werden können, benötigen Sie für jedes Gerät eine eindeutige IP-Adresse.

Virtuelle IP-Adressen und virtuelles Loopback sind unabhängige Features. Sie können ein Feature oder beide wählen.

Zusammenfassung der Administratoraktion:

- Zur Verwendung von Microsoft virtuellen IPs aktivieren und konfigurieren Sie die Funktion auf dem Windows-Server.
- Für die Verwendung von virtuellem Loopback von Citrix konfigurieren Sie zwei Einstellungen in einer Citrix Richtlinie.

## Virtuelle IP

Wenn die virtuelle IP aktiviert und auf dem Windows-Server konfiguriert ist, scheint jede konfigurierte Anwendung, die in einer Sitzung ausgeführt wird, eine eindeutige Adresse zu haben. Benutzer greifen auf diese Anwendungen auf einem XenApp-Server genauso wie auf andere veröffentlichte Anwendungen zu. Ein Prozess erfordert die virtuelle IP in den folgenden Fällen:

- Der Prozess verwendet eine hartcodierte TCP-Portnummer
- Der Prozess verwendet Windows Sockets und benötigt eine eindeutige IP-Adresse oder eine angegebene TCP-Portnummer

So ermitteln Sie, ob eine Anwendung virtuelle IP-Adressen verwenden muss

1. Beziehen Sie das TCPView-Tool von Microsoft. Das Programm zeigt alle Anwendungen an, die an spezifische IP-Adressen und Ports binden.
2. Deaktivieren Sie das Auflösen von IP-Adressen, sodass statt der Adressen die Hostnamen angezeigt werden.
3. Starten Sie die Anwendung und ermitteln Sie mit TCPView, welche IP-Adressen und Ports von der Anwendung geöffnet werden und welche Prozesse diese Ports öffnen.
4. Konfigurieren Sie alle Prozesse, die die IP-Adresse des Servers, 0.0.0.0 oder 127.0.0.1, öffnen.
5. Starten Sie eine zusätzliche Instanz der Anwendung, um sicherzustellen, dass sie nicht dieselbe IP-Adresse auf einem anderen Port öffnet.

## Funktionsweise von Microsoft Remote Desktop-Virtualisierung

- Die virtuelle IP-Adressierung muss auf dem Microsoft Server aktiviert sein.  
Beispiel: In einer Umgebung mit Windows Server 2008 R2 erweitern Sie im Server-Manager "Remotedesktopdienste > Remotedesktop-Sitzungshostverbindungen", um das Remotedesktop-IP-Virtualisierungsfeature zu aktivieren, und konfigurieren Sie die Einstellungen so, dass IP-Adressen dynamisch mit dem DHCP-Server pro Sitzung oder pro Programm zugewiesen werden. Weitere Informationen finden Sie in der Microsoft Dokumentation.



- Nach der Aktivierung des Features fordert der Server beim Sitzungsstart dynamisch zugewiesene IP-Adressen vom DHCP-Server an.
- Das Remotedesktop-IP-Virtualisierungsfeature weist den Remotedesktopverbindungen die IP-Adressen pro Sitzung oder pro Programm zu. Wenn Sie IP-Adressen für mehrere Programme zuweisen, verwenden sie eine gemeinsame IP-Adresse pro Sitzung.
- Wenn einer Sitzung eine Adresse zugewiesen wurde, verwendet die Sitzung die virtuelle Adresse statt der primären IP-Adresse des Systems bei den folgenden Aufrufen: Bind, Closesocket, Connect, WSACconnect, WSAAccept, getpeername, getsockname, sendto, WSASendTo, WSASocketW, gethostbyaddr, getnameinfo, getaddrinfo

Wenn das IP-Virtualisierungsfeature von Microsoft in der Hostingkonfiguration der Remotedesktopsitzung verwendet wird, sind Anwendungen an bestimmte IP-Adressen gebunden, indem eine Filterkomponente zwischen die Anwendung und den Winsock-Funktionsaufrufen eingefügt wird. Die Anwendung erkennt dann nur die IP-Adresse, die sie verwenden soll. Sollte die Anwendung versuchen, TCP- oder UDP-Kommunikation abzuhehren, wird sie automatisch an die zugewiesene virtuelle IP-Adresse (oder Loopbackadresse) gebunden und alle von der Anwendung geöffneten Ausgangsverbindungen gehen von der an die Anwendung gebundene IP-Adresse aus.

In Funktionen, die eine Adresse ausgeben, wie z. B. GetAddrInfo() (über eine Windows-Richtlinie gesteuert), untersucht die virtuelle IP beim Abrufen der IP-Adresse des lokalen Hosts die zurückgegebene IP-Adresse und ändert sie in die virtuelle IP-Adresse der Sitzung. Anwendungen, die mit solchen Namensfunktionen versuchen, die IP-Adresse des lokalen Servers zu ermitteln, erhalten nur die eindeutige virtuelle IP-Adresse, die der Sitzung zugeordnet wurde. Diese IP-Adresse wird oft in späteren Socket-Aufrufen, wie "Bind" oder "Connect", verwendet.

Oft fordern Anwendungen eine Bindung an einen Port zum Abhören der Adresse 0.0.0.0. Wenn eine Anwendung dies versucht und einen statischen Port verwendet, können Sie höchstens eine Instanz der Anwendung starten. Das virtuelle IP-Adressfeature sucht in diesen Aufrufen nach 0.0.0.0 und ändert den Abruf so, dass die angegebene virtuelle IP-Adresse abgehört wird. Dies ermöglicht, dass mehrere Anwendungen denselben Port auf demselben Computer abhören, da sie auf verschiedenen Adressen abhören. Der Aufruf wird nur geändert, wenn er in einer ICA-Sitzungen erfolgt und virtuelle IP-Adressen aktiviert sind. Beispiel: Wenn zwei Instanzen einer Anwendung, die in unterschiedlichen Sitzungen ausgeführt werden, eine Bindung mit allen Schnittstellen (0.0.0.0) und einen bestimmten Port (z. B. 9000) versuchen, werden sie an VIPAddress1:9000 und VIPAddress2:9000 gebunden und es gibt keinen Konflikt.

## Virtuelles Loopback

Bei Aktivierung der Citrix Richtlinieneinstellungen für virtuelles Loopback kann jede Sitzung eine eigene Loopbackadresse für die Kommunikation haben. Verwendet eine Anwendung in einem Winsock-Aufruf die Adresse des lokalen Hosts (Standard ist 127.0.0.1), ersetzt das virtuelle Loopback die Adresse 127.0.0.1 einfach durch 127.X.X.X ersetzt, wobei X.X.X die Sitzungs-ID + 1 ist. Wenn die Sitzungs-ID zum Beispiel 7 ist, ist die Adresse 127.0.0.8. Im unwahrscheinlichen Fall, dass die Sitzungs-ID größer ist, als im vierten Oktett zulässig (mehr als 255), wird beim nächsten Oktett weitergemacht (127.0.1.0) bis zum Maximum von 127.255.255.255.

Ein Prozess erfordert das virtuelle Loopback in den folgenden Fällen:

- Der Prozess verwendet die Windows-Sockets-Loopbackadresse (localhost) (127.0.0.1)
- Der Prozess verwendet eine hartcodierte TCP-Portnummer

Verwenden Sie die [Richtlinieneinstellungen für virtuelles Loopback](#) für Anwendungen, die eine Loopbackadresse für prozessübergreifende Kommunikation verwenden. Eine zusätzliche Konfiguration ist nicht erforderlich. Virtuelles Loopback ist nicht von virtueller IP abhängig, sodass der Microsoft-Server nicht konfiguriert werden muss.

- Virtuelle IP - Loopbackunterstützung: Wenn diese Richtlinieneinstellung aktiviert ist, kann jede Sitzung eine eigene virtuelle Loopbackadresse haben. Diese Einstellung ist standardmäßig aktiviert. Das Feature gilt nur für Anwendungen, die



mit der Richtlinieneinstellung Virtuelle IP - Programme für virtuelles Loopback angegeben wurden.

- Virtuelle IP - Programme für virtuelles Loopback: Mit dieser Richtlinieneinstellung geben Sie die Anwendung an, die das Feature "Virtuelles IP-Loopback" verwenden. Diese Einstellung gilt nur, wenn die Richtlinieneinstellung Virtuelle IP - Loopbackunterstützung aktiviert ist.

## Verwandtes Feature

Mit den folgenden Registrierungseinstellungen stellen Sie sicher, dass virtuelles Loopback den Vorrang vor virtuelle IP erhält; dies wird als bevorzugtes Loopback bezeichnet. Achten Sie jedoch auf Folgendes:

- Bevorzugtes Loopback wird nur unter Windows 2008 R2 unterstützt.
- Verwenden Sie bevorzugtes Loopback nur, wenn virtuellen IP-Adressen und das virtuelle Loopback aktiviert sind, sonst erhalten Sie u. U. unerwartete Ergebnisse.
- Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

Führen Sie regedit auf den Servern aus, auf dem die Anwendungen installiert sind.

- HKEY\_LOCAL\_MACHINE\SOFTWARE\Wow6432Node\Citrix\VIP (HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\VIP für 32-Bit-Maschinen)
- Name: PreferLoopback, Typ: REG\_DWORD, Wert: 1
- Name: PreferLoopbackProcesses, Typ: REG\_MULTI\_SZ Wert:

# Sekundärer Datenbankspeicherort

May 10, 2016

Standardmäßig befinden sich die Konfigurationsprotokollierungsdatenbank und die Überwachungsdatenbank (die sekundären Datenbanken) auf demselben Server wie die Sitekonfigurationsdatenbank. Zunächst haben alle drei Datenbanken denselben Namen. Citrix empfiehlt, den Speicherort der sekundären Datenbanken nach dem Erstellen einer Site zu ändern. Sie können die Konfigurationsprotokollierungs- und die Überwachungsdatenbanken auf einem Server oder auf unterschiedlichen Servern hosten. Die Backupstrategie kann für jede Datenbank anders sein.

Berücksichtigen Sie Folgendes, wenn Sie den Speicherort der Konfigurationsprotokollierungs- oder Überwachungsdatenbank ändern:

- Die Daten werden nicht aus der bestehenden Datenbank in die neue Datenbank importiert.
- Die Protokolle beider Datenbanken können beim Abrufen von Protokollen nicht aggregiert werden.
- Der erste Protokolleintrag in der neuen Datenbank gibt an, dass eine Datenbankänderung stattfand, die vorherige Datenbank wird jedoch nicht angegeben.

Bevor Sie den Speicherort der Konfigurationsprotokollierungs- oder der Überwachungsdatenbank ändern, müssen Sie eine unterstützte Version von Microsoft SQL Server auf dem Server installieren, auf dem die Datenbank gespeichert wird. Richten Sie je nach Bedarf Spiegel, Cluster oder andere unterstützte Redundanzinfrastrukturen ein.

Sie können den Speicherort der Konfigurationsprotokollierungsdatenbank nicht ändern, wenn die verbindliche Protokollierung aktiviert ist.

Hinweis: Sie können mit dieser Methode den Speicherort der Sitekonfigurationsdatenbank nicht ändern.

1. Wählen Sie im Studio-Navigationsbereich Konfiguration aus. Die Namen und Adressen der drei Datenbanken werden aufgelistet sowie die Spiegelserveradressen, falls konfiguriert.
2. Wählen Sie die Datenbank aus, für die Sie einen neuen Speicherort angeben möchten, und wählen Sie dann im Aktionsbereich Datenbank ändern aus.
3. Geben Sie mit einem der Formulare aus der folgenden Tabelle den Speicherort der neuen SQL Server-Installation und den Namen der Datenbank an.

| Datenbanktyp                 | Eingabe                     | Auswirkungen dieser Datenbankkonfiguration                                                                                |
|------------------------------|-----------------------------|---------------------------------------------------------------------------------------------------------------------------|
| Eigenständig oder gespiegelt | Servername                  | Die Standardinstanz wird verwendet und SQL Server verwendet den Standardport.                                             |
|                              | Servername\Instanzname      | Eine benannte Instanz wird verwendet und SQL Server verwendet den Standardport.                                           |
|                              | servername,Portnummer       | Die Standardinstanz wird verwendet und SQL Server verwendet einen benutzerdefinierten Port. (Das Komma ist erforderlich.) |
| Sonstiges                    | cluster-name                | Eine geclusterte Datenbank.                                                                                               |
|                              | availability-group-listener | Eine stets aktivierte Datenbank.                                                                                          |

4. Wenn Sie die Datenbank von Studio erstellen lassen möchten, klicken Sie auf OK. Wenn Sie dazu aufgefordert werden,

klicken Sie auf OK und Studio erstellt die Datenbank automatisch. Studio versucht, mit den Anmeldeinformationen des aktuellen Studio-Benutzers auf die Datenbank zuzugreifen; wenn dies fehlschlägt, werden Sie zur Eingabe der Anmeldeinformationen des Datenbankbenutzers aufgefordert. Das Datenbankschema wird dann von Studio in die Datenbank hochgeladen. (Die Anmeldeinformationen werden nur für den Zeitraum der Datenbankerstellung gespeichert.)

5. Wenn Sie die Datenbank manuell erstellen möchten, klicken Sie auf Skript erstellen. Die generierten Skripts enthalten Anweisungen, wie Sie die Datenbank und ggf. die Spiegeldatenbank manuell erstellen. Stellen Sie vor dem Hochladen des Schemas sicher, dass die Datenbank leer ist und dass mindestens ein Benutzer Zugriffs- bzw. Änderungsberechtigung für die Datenbank hat.

# Delivery Controller-Umgebung

May 10, 2016

In einer Bereitstellung ist der Delivery Controller die serverseitige Komponente, die für die Verwaltung des Benutzerzugriffs sowie das Brokering und Optimieren von Verbindungen zuständig ist. Controller stellen auch die Maschinenerstellungsdienste zur Erstellung von Desktop- und Serverimages bereit.

Eine Site muss mindestens über einen Delivery Controller verfügen. Nach der Installation des ersten Controllers und der Erstellung einer Site können Sie weitere Controller hinzufügen. Es gibt zwei Hauptvorteile, mehr als einen Controller in einer Site zu haben.

- Redundanz: Als bewährte Methode sollte eine Produktionssite immer mindestens zwei Controller auf unterschiedlichen physischen Servern haben. Wenn ein Controller ausfällt, können die anderen die Verwaltung der Verbindungen und der Site übernehmen.
- Skalierbarkeit: Je intensiver die Aktivität einer Site, umso mehr nehmen CPU-Auslastung auf dem Controller und SQL Server-Datenbankaktivität zu. Zusätzliche Controller bieten die Möglichkeit, mehr Benutzer, Anwendungen und Desktopanforderungen zu verarbeiten und die Reaktionszeit insgesamt zu verbessern.

## So erkennen Virtual Delivery Agents (VDAs) die Controller

VDAs können erst verwendet werden, wenn sie bei einem Controller der Site registriert wurden (Herstellen der Kommunikation). Zur Suche eines Controllers überprüft der VDA die Controllerliste ListOfDDCs. Die Liste "ListOfDDCs" enthält einen oder mehrere DNS-Einträge oder IP-Adressen, die den VDA an die Controller der Site verweisen. Um einen Lastausgleich zu erzielen, verteilt der VDA die Verbindungen automatisch über alle Controller in der Liste.

Neben ListOfDDCs gibt die Liste ListOfSIDs an, über welche Maschinensicherheits-IDs (SIDs) der VDA als Controller angesprochen werden kann. Die Liste "ListOfSIDs" kann verwendet werden, um die Last auf Active Directory zu verringern oder um Sicherheitsbedrohungen durch einen nicht sicheren DNS-Server zu vermeiden.

Die Listen "ListOfDDCs" und "ListOfSIDs" müssen auf allen VDAs auf dem neuesten Stand sein, wenn Controller in der Site hinzugefügt und entfernt werden. Sind die Listen nicht aktuell, werden Sitzungsstarts, die durch einen nicht aufgeführten Controller vermittelt werden, vom VDA möglicherweise abgelehnt. Ungültige Einträge in der Liste können den Start der Systemsoftware des virtuellen Desktops verzögern. Halten Sie die Listen wie folgt auf dem neuesten Stand:

- Verwenden Sie automatische Updates zur automatischen Aktualisierung von ListOfDDCs und ListOfSIDs, wenn Controller hinzugefügt oder entfernt werden. Standardmäßig sind automatische Updates aktiviert.
- Eigenverwaltung: Aktualisieren Sie die Richtlinien- oder Registrierungseinstellungen für die Controller manuell.

Die Informationen in den Listen "ListOfDDCs" und "ListOfSIDs" können aus verschiedenen Orten der Bereitstellung stammen. Folgende Speicherorte werden vom VDA nacheinander überprüft, bis die Listen gefunden werden:

1. Ein Speicherort für den persistenten Speicher für automatische Updates. Dieser Speicherort enthält die Controllerinformationen, wenn automatische Updates aktiviert sind und nachdem der VDA erfolgreich zum ersten Mal nach der Installation registriert wurde. (Hier werden auch Maschinenrichtlinieninformationen gespeichert, sodass Richtlinieneinstellungen bei Neustarts beibehalten werden.)  
Bei der ersten Registrierung nach der Installation oder bei deaktivierten automatischen Updates werden folgende Speicherorte vom VDA geprüft:
2. Richtlinieneinstellungen (Controller, Controller-SIDs).
3. Controllerinformationen des Virtual Desktop Agent-Schlüssels in der Registrierung. Diese Werte stammen ursprünglich vom VDA-Installationsprogramm, basierend auf den Controllerinformationen, die Sie bei der Installation des VDAs

eingeben.

4. Auf Organisationseinheiten basierende Controller-Discovery. Dies ist eine ältere Methode die zugunsten der Abwärtskompatibilität beibehalten wird.
5. Die durch die Maschinenerstellungsdienste erstellte Datei Personality.ini.

Wenn in ListOfDDCs mehrere Controller angegeben sind, erfolgt die Verbindung mit ihnen durch den VDA in einer zufälligen Reihenfolge. Die Liste "ListOfDDCs" kann auch Controllergruppen enthalten, die an den Klammern um zwei oder mehr Controllereinträge zu erkennen sind. Der VDA versucht, eine Verbindung mit jedem Controller in einer Gruppe herzustellen, bevor er weitere Einträge in der Liste "ListOfDDCs" versucht.

Für XenDesktop-Benutzer, die ein Upgrade einer Version vor 7.0 durchgeführt haben, ersetzt das Feature für automatische Updates die CNAME-Funktion der Vorversion. Sie können die CNAME-Funktion ggf. wieder aktivieren, jedoch funktioniert das DNS-Aliasing nur konsistent, wenn Sie das AutoUpdate-Feature und die CNAME-Funktion nicht zusammen verwenden. Weitere Informationen zum erneuten Aktivieren der CNAME-Funktion finden Sie unter [CTX137960](#).

## Überlegungen zur Verwendung automatischer Updates oder der Eigenverwaltung

Die Richtlinieneinstellung zur Aktivierung/Deaktivierung der automatischen Updates ist standardmäßig aktiviert.

Bei folgenden Bereitstellungstypen sind keine automatischen Updates möglich, stattdessen muss die Eigenverwaltung verwendet werden:

- Bereitstellungen mit Controllergruppen
- Bereitstellungen, die aus Sicherheitsgründen ListOfSIDs verwenden (Bei Bereitstellungen mit ListOfSIDs zur Verringerung der Active Directory-Last können automatische Updates verwendet werden)
- Bereitstellungen, die Provisioning Services ohne Datenträger für Zurückschreiben verwenden
- Bereitstellungen, die die Controller- oder Controller-SIDs-Richtlinieneinstellung verwenden

## Verwenden automatischer Updates

Die Richtlinieneinstellung für automatische Updates ist in der Kategorie "Virtual Delivery Agent".

- Zur Verwendung automatischer Updates aktivieren Sie die Richtlinieneinstellung Automatische Controllerupdates aktivieren. Diese Einstellung ist standardmäßig aktiviert.
- Deaktivieren Sie die Richtlinieneinstellung Automatische Controllerupdates aktivieren, um automatische Updates zu deaktivieren.

Wird bei aktivierten automatischen Updates ein VDA installiert, versucht der VDA, sich mit einem der Controller-Werte zu registrieren, die bei der Installation des VDAs von Ihnen eingegeben wurden. Das Installationsprogramm schreibt die Controllerinformationen, die Sie bei der Installation des VDAs angegeben haben, in den Registrierungswert "ListOfDDCs".

Nach der VDA-Registrierung sendet der Controller, bei dem die Registrierung erfolgte, eine Liste der vollständig qualifizierten Domännennamen (FQDNs) und Sicherheits-IDs (SIDs) an den VDA. Der VDA schreibt diese Liste in den persistenten Speicher für automatische Updates. Die Sitekonfigurationsdatenbank wird alle 90 Minuten von jedem Controller auf Controllerinformationen überprüft: Wenn seit der letzten Überprüfung ein Controller hinzugefügt oder entfernt wurde oder wenn eine Richtlinie geändert wurde, wird die aktualisierte Liste vom Controller an die registrierten VDAs gesendet. Der VDA nimmt alle Verbindungen von allen Controllern in der aktuellen Liste an.

Geht eine Liste ein, die den Controller, bei dem der VDA registriert ist, nicht enthält (d. h. der Controller wurde aus der Site entfernt), nimmt der VDA eine neue Registrierung bei einem der Controller aus der Liste vor. Nach der Registrierung bzw. Neuregistrierung erhält der VDA eine aktualisierte Liste.

Beispiel:

1. Eine Bereitstellung hat drei Controller: A, B und C. Ein VDA ist installiert und wird bei Controller B registriert (dies wurde bei der Installation des VDAs festgelegt).
2. Zwei Controller (D und E) werden der Site hinzugefügt. Innerhalb von 90 Minuten erhalten die VDAs aktualisierte Listen und können Verbindungen von Controllern A, B, C, D und E annehmen. (Die Last wird erst nach Neustart der VDAs gleichmäßig auf alle Controller verteilt.)
3. Controller B wird aus der Site entfernt. Innerhalb von 90 Minuten erhalten die VDAs aktualisierte Listen, da seit der letzten Überprüfung eine Controlleränderung stattfand. Der in Schritt 1 installierte VDA ist bei Controller B registriert, der jedoch in der Liste nicht mehr enthalten ist. Somit wird der VDA bei einem der anderen Controller der Liste (A, C, D und E) neu registriert.

## Eigenverwaltung

Wenn Sie keine automatischen Updates verwenden, müssen Sie die Citrix Richtlinieneinstellung oder die Registrierungswerte für jeden Virtual Delivery Agent (VDA) der Site (oder das VDA-Image) aktualisieren, wenn Delivery Controller in der Site hinzugefügt, verschoben oder entfernt wurden. Registrierungsänderungen können auch mit dem Gruppenrichtlinienobjekt aktualisiert werden.

So verwenden Sie die Eigenverwaltung mit Citrix Richtlinieneinstellungen

1. Aktualisieren Sie die Werte der vollqualifizierten Domännennamen in der Richtlinieneinstellung Controller. Diese Richtlinieneinstellung ist in der Kategorie "Virtual Delivery Agent".
2. Wenn Sie in Ihrer Bereitstellung auch ListOfSIDs-Listen verwenden, aktualisieren Sie die SID-Werte der Richtlinieneinstellung Controller-SIDs.

So verwenden Sie die Eigenverwaltung mit der Registrierung

Achtung: Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

1. Aktualisieren Sie den Registrierungsschlüssel "ListOfDDCs", der die vollqualifizierten Domännennamen aller Controller in der Site enthält. (Dieser Schlüssel entspricht der Active Directory-Site-Organisationseinheit.) Trennen Sie mehrere Werte mit Leerzeichen. Umschließen Sie Controllergruppen mit Klammern.

HKEY\_LOCAL\_MACHINE\Software\Citrix\VirtualDesktopAgent\ListOfDDCs (REG\_SZ)

Wenn die Registrierungsstruktur HKEY\_LOCAL\_MACHINE\Software\Citrix\VirtualDesktopAgent sowohl die Registrierungsschlüssel "ListOfDDCs" und "FarmGUID" enthält, wird ListOfDDCs für Controller-Discovery verwendet; FarmGUID ist vorhanden, wenn die Organisationseinheit der Site bei der Installation des VDAs angegeben wurde.

2. Aktualisieren Sie wahlweise den Registrierungsschlüssel "ListOfSIDs":

HKEY\_LOCAL\_MACHINE\Software\Citrix\VirtualDesktopAgent\ListOfSIDs (REG\_SZ)

# Hinzufügen, Entfernen, Verschieben von Controllern und Verschieben eines VDAs

May 10, 2016

Zum Hinzufügen, Entfernen oder Verschieben eines Delivery Controllers benötigen Sie die folgenden Rollen oder Berechtigungen:

| Vorgang                                   | Zweck                                                                                                                            | Serverrolle        | Datenbankrolle |
|-------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------|--------------------|----------------|
| Datenbank erstellen                       | Geeignete, leere Datenbank erstellen                                                                                             | dbcreator          |                |
| Schema erstellen                          | Erstellen aller dienstspezifischen Schemas und Hinzufügen des ersten Controllers zur Site                                        | securityadmin<br>* | db_owner       |
| Controller hinzufügen                     | Hinzufügen eines Controller (ein anderer als der Erste) zur Site                                                                 | securityadmin<br>* | db_owner       |
| Controller hinzufügen (Spiegelungsserver) | Hinzufügen einer Controller-Anmeldung zu dem Datenbankserver, der derzeit die gespiegelte Rolle einer gespiegelten Datenbank hat | securityadmin<br>* |                |
| Controller entfernen                      | Entfernen eines Controllers von der Site                                                                                         |                    | db_owner       |
| Schema aktualisieren                      | Anwenden von Aktualisierungen oder Hotfixes auf das Schema                                                                       |                    | db_owner       |

\* Zwar ist die securityadmin-Serverrolle technisch restriktiver als die sysadmin-Serverrolle, aber in der Praxis ist sie als gleichwertig anzusehen.

Wenn Sie mit Studio diese Vorgänge ausführen, muss das Benutzerkonto explizit Mitglied der sysadmin-Serverrolle sein.

Wenn in der Bereitstellung Datenbankspiegelung verwendet wird, gilt Folgendes:

- Vor dem Hinzufügen, Entfernen oder Verschieben von Controllern müssen Sie sicherstellen, dass sowohl die gespiegelte als auch die Hauptdatenbank ausgeführt werden. Wenn Sie mit Skripts für SQL Server Management Studio arbeiten, müssen Sie den SQLCMD-Modus vor dem Ausführen des Skripts aktivieren.
- Zum Prüfen der Spiegelung nach dem Hinzufügen, Entfernen oder Verschieben des Controllers führen Sie das PowerShell-Cmdlet `get-configdbconnection` aus, um sicherzustellen, dass der Failoverpartner in der Verbindungszeichenfolge für die Spiegelung eingerichtet wurde.

Gehen Sie nach dem Hinzufügen, Entfernen oder Verschieben eines Controllers wie folgt vor:

- Wenn das automatische Update aktiviert ist, erhalten die Virtual Delivery Agents (VDAs) eine aktualisierte Liste der Controller innerhalb von 90 Minuten.
- Ist das automatische Update nicht aktiviert, müssen Sie sicherstellen, dass die Controllerrichtlinieneinstellung oder der Registrierungsschlüssel "ListOfDDCs" für alle VDAs aktualisiert wird. Nachdem Sie einen Controller in eine andere Site verschoben haben, müssen Sie die Richtlinieneinstellung oder den Registrierungsschlüssel in beiden Sites aktualisieren.

## Hinzufügen eines Controllers

Sie können keine Server, die mit einer früheren Version dieser Software installiert wurden, in einer Site hinzufügen, die mit dieser Version erstellt wurde.

1. Führen Sie auf dem Server, den Sie hinzufügen möchten, das Installationsprogramm aus und wählen Sie die Kernkomponenten aus, die Sie installieren möchten.
2. Klicken Sie in Studio auf Vorhandener Bereitstellung beitreten und geben Sie die Siteadresse ein.

## Entfernen eines Controllers

Beim Entfernen eines Controllers werden die Citrix Software und andere Komponenten nicht deinstalliert. Es wird lediglich der Controller aus der Datenbank entfernt, der damit für das Brokering von Verbindungen oder sonstige Aufgaben nicht mehr verfügbar ist. Wenn Sie einen Controller entfernen, können Sie diesen zu einem späteren Zeitpunkt der gleichen oder einer anderen Site wieder hinzufügen. Eine Site benötigt mindestens einen Controller. Aus diesem Grund können Sie den letzten in Studio aufgelisteten Controller nicht entfernen.

Hinweis: Stellen Sie sicher, dass der Controller eingeschaltet ist, sodass Studio in weniger als einer Stunde geladen wird. Wenn Studio den Controller lädt, den Sie entfernen möchten, schalten Sie den Controller aus, wenn Sie dazu aufgefordert werden.

Wenn Sie einen Controller von einer Site entfernen, wird die Controller-Anmeldung für den Datenbankserver nicht entfernt. Auf diese Weise wird vermieden, dass eine Anmeldung entfernt wird, die von den Diensten anderer Produkte auf demselben Computer verwendet wird. Die Anmeldung muss manuell entfernt werden, wenn sie nicht mehr erforderlich ist. Dazu benötigen Sie die Berechtigungen der securityadmin-Serverrolle.

**Wichtig:** Entfernen Sie den Controller *erst dann* aus Active Directory, wenn Sie ihn aus der Site entfernt haben.

1. Wählen Sie im Studio-Navigationsbereich Konfiguration > Controller und wählen Sie anschließend den Controller aus, den Sie entfernen möchten.
2. Wählen Sie im Aktionsbereich Controller entfernen. Wenn Sie nicht über die erforderlichen Datenbankrollen und Berechtigungen verfügen, können Sie ein Skript erstellen, mit dem der Datenbankadministrator den Controller für Sie entfernen kann.
3. Möglicherweise müssen Sie das Maschinenkonto des Controllers auf dem Datenbankserver entfernen. Bevor Sie dies durchführen, überprüfen Sie, ob das Konto von einem anderen Dienst verwendet wird.

Nachdem Sie mit Studio einen Controller entfernt haben, besteht ggf. kurze Zeit weiter Datenverkehr zu diesem Controller, um sicherzustellen, dass die aktuellen Tasks einwandfrei abgeschlossen werden. Wenn Sie das Entfernen eines Controllers in sehr kurzer Zeit erzwingen möchten, empfiehlt Citrix, den Server, auf dem er installiert war, herunterzufahren oder aus Active Directory zu entfernen. Starten Sie dann die anderen Controller in der Site neu, um sicherzustellen, dass keine weitere Kommunikation mit dem entfernten Controller stattfindet.

## Verschieben eines Controllers in eine andere Site

Controller können nicht in eine Site verschoben werden, die mit einer früheren Version dieser Software erstellt wurde.

1. Wählen Sie in der derzeitigen Site des Controllers (der alten Site) im Studio-Navigationsbereich Konfiguration > Controller und wählen Sie anschließend den Controller aus, den Sie verschieben möchten.
2. Wählen Sie im Aktionsbereich Controller entfernen. Wenn Sie nicht über die erforderlichen Datenbankrollen und Berechtigungen verfügen, können Sie ein Skript erstellen, mit dem der Datenbankadministrator den Controller für Sie entfernen kann. Eine Site benötigt mindestens einen Controller. Aus diesem Grund können Sie den letzten in Studio aufgelisteten Controller nicht entfernen.
3. Öffnen Sie Studio auf dem zu verschiebenden Controller, setzen Sie bei entsprechender Aufforderung die Dienste



zurück, wählen Sie Vorhandener Site beitreten und geben Sie die Adresse der neuen Site ein.

## Verschieben eines VDA in eine andere Site

Wenn ein VDA mit Provisioning Services bereitgestellt wurde oder wenn er ein vorhandenes Image ist, können Sie einen VDA in eine andere Site (von Site 1 nach Site 2) verschieben, wenn Sie ein Upgrade vornehmen oder wenn Sie ein in einer Testsite erstelltes VDA-Image in eine Produktionssite verschieben. Mit Maschinenerstellungsdienste (MCS) bereitgestellte VDAs können nicht zwischen Sites verschoben werden, da MCS das Ändern der ListOfDDCs nicht unterstützt, die ein VDA zum Registrieren mit einem Controller prüft. Mit Maschinenerstellungsdienste bereitgestellte VDAs prüfen immer die ListOfDDCs, die der Site zugeordnet ist, in der sie erstellt wurden.

Es gibt zwei Möglichkeiten, einen VDA in eine andere Site zu verschieben: mit dem Installationsprogramm oder mit Citrix Richtlinien.

- Installationsprogramm: Führen Sie das Installationsprogramm aus und fügen Sie einen Controller hinzu, wobei Sie in Site 2 einen vollqualifizierten Domännennamen (DNS-Eintrag) eines Controllers angeben.  
Wichtig: Geben Sie Controller im Installationsprogramm nur dann an, wenn die Richtlinieneinstellung "Controller" nicht verwendet wird.
- Gruppenrichtlinienobjekt-Editor: Im folgenden Beispiel werden mehrere VDAs verschoben.
  1. Erstellen Sie eine Richtlinie in Site 1 mit den nachfolgenden Einstellungen und filtern Sie die Richtlinie auf Bereitstellungsebene, um eine mehrstufige VDA-Migration zwischen den Sites zu erzielen.
    - Controller: mit vollqualifizierten Domännennamen (DNS-Einträgen) von einem oder mehreren Controllern der Site 2.
    - Automatische Controllerupdates aktivieren: auf "Deaktiviert" gesetzt.
  2. Jeder VDA in der Bereitstellungsgruppe wird innerhalb von 90 Minuten auf die neue Richtlinie hingewiesen. Der VDA ignoriert die eingegangene Liste der Controller (da automatische Updates deaktiviert sind) und wählt einen der in der Richtlinie angegebenen Controller, d. h. einen der Controller in Site 2.
  3. Wenn der VDA erfolgreich bei einem Controller der Site 2 registriert wurde, empfängt er die Liste "ListOfDDCs" und die Richtlinieninformationen von Site 2, für die automatische Updates standardmäßig aktiviert sind. Da der Controller, bei dem der VDA in Site 1 registriert war, nicht in der vom Controller in Site 2 gesendeten Liste ist, erfolgt eine erneute Registrierung des VDAs unter Auswahl eines Controllers der Liste von Site 2. Ab sofort wird der VDA automatisch mit Informationen von Site 2 aktualisiert.

# Auf Organisationseinheiten von Active Directory-basierte Controller-Discovery

Nov 02, 2016

Diese Delivery Controller-Erkennungsmethode wird vor allem unterstützt, um Abwärtskompatibilität zu gewährleisten. Sie gilt nur für Virtual Delivery Agents (VDAs) für Windows-Desktopbetriebssysteme, aber nicht für VDAs für Windows-Serverbetriebssysteme. Bei der Active Directory-basierten Discovery müssen alle Computer in einer Site Mitglied einer Domäne sein und zwischen der Domäne, die vom Controller verwendet wird, und den von den Desktops verwendeten Domänen muss eine Vertrauensstellung bestehen. Wenn Sie diese Methode verwenden, müssen Sie die GUID der Organisationseinheit in jeder Desktopregistrierung konfigurieren.

Führen Sie für eine auf Organisationseinheiten basierende Controller-Discovery das PowerShell-Skript `Set-ADControllerDiscovery.ps1` auf dem Controller aus (das Skript ist auf jedem Controller im Ordner `$Env:ProgramFiles\Citrix\Broker\Service\Setup Scripts`). Um das Skript auszuführen, müssen Sie über `CreateChild-Berechtigungen` für eine übergeordnete Organisationseinheit verfügen sowie über vollständige Administratorrechte.

Wenn Sie eine Site erstellen, müssen Sie eine entsprechende Organisationseinheit in Active Directory erstellen, wenn Desktops die Controller in der Site mit Active Directory ermitteln sollen. Die Organisationseinheit kann in jeder Domäne in der Gesamtstruktur erstellt werden, die Ihre Computer enthält. Optimalerweise sollte die Organisationseinheit auch die Controller der Site enthalten; dies wird jedoch nicht erzwungen bzw. ist nicht erforderlich. Ein Domänenadministrator mit den entsprechenden Privilegien kann die Organisationseinheit als leeren Container erstellen und dann Administrationsrechte an der Organisationseinheit an einen Citrix Administrator delegieren.

Das Skript erstellt einige wichtige Objekte. Nur Active Directory-Standardobjekte werden erstellt und verwendet. Es ist nicht erforderlich, das Schema zu erweitern.

- Eine Controller-Sicherheitsgruppe. Das Computerkonto aller Controller in der Site muss ein Mitglied dieser Sicherheitsgruppe sein. Desktops in einer Site nehmen nur Daten von Controllern an, die Mitglied dieser Sicherheitsgruppe sind.  
Stellen Sie sicher, dass alle Controller auf allen virtuellen Desktops, auf denen der VDA ausgeführt wird, die Berechtigung "Auf diesen Computer vom Netzwerk aus zugreifen" haben. Geben Sie hierfür der Controller-Sicherheitsgruppe dieses Privileg. Wenn Controller nicht über diese Berechtigung verfügen, werden VDAs nicht registriert.
- Ein Dienstverbindungspunkt-Objekt, das Informationen zur Site enthält, wie den Namen der Site. Wenn Sie eine Organisationseinheit der Site mit dem Verwaltungstool "Active Directory-Benutzer und -Computer" prüfen, müssen Sie ggf. "Erweiterte Funktionen" im Menü "Ansicht" aktivieren, um Dienstverbindungspunkt-Objekte anzuzeigen.
- Einen "RegistrationServices" genannten Container, der in der Organisationseinheit der Site erstellt wird. Dieser Container enthält ein Dienstverbindungspunkt-Objekt für jeden Controller in der Site. Bei jedem Start des Controllers wird der Inhalt des Dienstverbindungspunkts geprüft und ggf. aktualisiert.

Wenn nach dem Abschluss der Erstinstallation mehrere Administratoren Controller hinzufügen oder entfernen, benötigen sie Berechtigungen zum Erstellen und Löschen von untergeordneten Objekten für den RegistrationServices-Container und zum Schreiben der Eigenschaften der Controller-Sicherheitsgruppe (diese Berechtigungen werden automatisch dem Administrator gewährt, der das Skript "Set-ADControllerDiscovery.ps1" ausführt). Der Domänenadministrator oder der Administrator der Originalinstallation kann diese Berechtigungen zuweisen; Citrix empfiehlt, dass Sie eine Sicherheitsgruppe für diesen Zweck einrichten.

Wenn Sie eine Organisationseinheit der Site verwenden:

- Informationen werden nur bei der Installation oder Deinstallation dieser Software in Active Directory geschrieben oder wenn beim Start eines Controllers die Informationen im Dienstverbindungspunkt aktualisiert werden müssen (der Controller wurde beispielsweise umbenannt oder der Kommunikationsport wurde geändert). In der Standardeinstellung erstellt das Skript "Set-ADControllerDiscovery.ps1" die richtigen Berechtigungen für die Organisationseinheit der Site und gibt jedem Controller Schreibrechte für den Dienstverbindungspunkt. Mit dem Inhalt der Objekte in der Organisationseinheit der Site wird eine Vertrauensstellung zwischen Desktops und Controllern erstellt. Stellen Sie Folgendes sicher:
  - Nur autorisierte Administratoren können der Controller-Sicherheitsgruppe über die Zugriffssteuerungsliste der Sicherheitsgruppe Computer hinzufügen oder Computer aus der Gruppe entfernen.
  - Nur autorisierte Administratoren und der relevante Controller können Informationen im Dienstverbindungspunkt des Controllers ändern.
- Bei einer Bereitstellung mit Replikation entstehen möglicherweise Verzögerungen; Weiteres hierzu finden Sie in der Microsoft Dokumentation. Dies ist besonders wichtig, wenn Sie die Organisationseinheit der Site in einer Domäne erstellen, die Domänencontroller in mehreren Active Directory-Sites enthält. Abhängig vom Standort der Desktops, der Controller und der Domänencontroller sind Änderungen, die Sie an Active Directory beim erstmaligen Erstellen der Organisationseinheit der Site, beim Installieren oder Deinstallieren von Controllern oder beim Ändern von Controllernamen oder Kommunikationsports vornehmen, erst nach der Replikation auf den entsprechenden Domänencontroller für Desktops erkennbar. Symptome einer solchen Replikationsverzögerung sind u. a. Desktops, die keine Verbindung mit Controllern herstellen können und daher nicht für Benutzerverbindungen verfügbar sind.
- Diese Software verwendet mehrere standardmäßige Computerobjektattribute in Active Directory zum Verwalten von Desktops. Je nach Bereitstellung kann der vollqualifizierte Domänenname des Maschinenobjekts so wie er im Active Directory-Datensatz des Desktops gespeichert ist in den Verbindungseinstellungen eingeschlossen werden, die an den Benutzer zum Herstellen einer Verbindung zurückgegeben werden. Stellen Sie sicher, dass diese Informationen mit den Informationen in der DNS-Umgebung übereinstimmen.

## Berechtigungen

Der Citrix Administrator, der das Skript zum Erstellen einer Site ausführt, muss für die Site-Organisationseinheit (OU) Berechtigungen zum Erstellen von Objekten (SCP, Container und Sicherheitsgruppen) haben.

(Wenn keine Organisationseinheit für die Site vorhanden ist, benötigt der Administrator die Rechte, eine zu erstellen. Citrix empfiehlt, dass der AD-Domänenadministrator vorab eine OU erstellt und dem Citrix Siteadministrator die Berechtigungen überträgt. Das Skript kann auch die Organisationseinheit für die Site erstellen. Dazu benötigt der Administrator die Berechtigung zum Erstellen von Organisationseinheiten für die übergeordnete Organisationseinheit der neuen Organisationseinheit. Wie erwähnt empfiehlt Citrix dies jedoch nicht.

Zum späteren Hinzufügen oder Entfernen eines Controllers muss der Citrix Administrator die Berechtigungen zum Hinzufügen/Entfernen einer Maschine zur/aus der Sicherheitsgruppe und zum Erstellen/Löschen eines SCPs haben.

Im normalen Betrieb benötigen Controller und VDAs Berechtigungen für alle Objekte in der Organisationseinheit und darunter. VDAs greifen auf die Organisationseinheit unter ihrer eigenen Maschinenidentität zu. Diese muss mindestens Leseberechtigung in der Organisationseinheit haben, damit Controller gesucht werden können. Controller müssen die Berechtigung zum Festlegen der Eigenschaften ihres eigenen SCP-Objekts im Container haben.

Durch Zuweisung der vollständigen Berechtigungen für den Citrix Administrator werden alle diese Aktionen zugelassen. Wenn Ihre Bereitstellung jedoch strengere Sicherheitsanforderungen hat (z. B. eine Festlegung, wer das Skript für welche

Aktion verwenden darf), können Sie mit dem Assistenten zum Zuweisen der Objektverwaltung spezifische Berechtigungen festlegen. Durch das nachfolgende Beispielfahren werden Berechtigungen zum Erstellen der Site gewährt.

1. Erstellen Sie eine Organisationseinheit (OU) für die untergeordneten Objekte: Dienstverbindungspunkt (SCP), Container und Sicherheitsgruppe.
2. Wählen Sie die Organisationseinheit aus, klicken Sie mit der rechten Maustaste und wählen Sie **Objektverwaltung zuweisen**.
3. Geben Sie im Assistenten zum Zuweisen der Objektverwaltung den Domänenbenutzer für die Zuweisung der Objektverwaltung für die OU an.
4. Wählen Sie auf der Seite **Zuzuweisende Aufgaben** die Option **Benutzerdefinierte Tasks zum Zuweisen erstellen**.
5. Übernehmen Sie auf der Seite **Active Directory-Objekttyp** den Standardwert **Diesem Ordner, bestehenden Objekten in diesem Ordner und neuen Objekten in diesem Ordner**.
6. Klicken Sie auf der Seite **Berechtigungen** auf **Alle untergeordneten Objekte erstellen**.
7. Schließen Sie den Assistenten zum Bestätigen der Berechtigungen ab.

Verschieben eines Controllers mit auf Organisationseinheiten basierender Controller-Discovery in eine andere Site

Folgen Sie den Anweisungen unter [Verschieben eines Controllers in eine andere Site](#). Nach dem Entfernen des Controllers aus der alten Site (Schritt 2) führen Sie das PowerShell-Skript "Set-ADControllerDiscovery –sync" aus.

Das Skript synchronisiert die Organisationseinheit mit dem aktuellen Satz an Controllern. Nach dem Beitritt zur vorhandenen Site (Schritt 3), führen Sie das gleiche Skript auf einem Controller in der neuen Site aus.

# Sitzungsverwaltung

May 10, 2016

Das Erhalten der Sitzungsaktivität ist wichtig, um eine optimale Benutzererfahrung zu gewährleisten. Eine Unterbrechung der Verbindung aufgrund von unzuverlässigen Netzwerken, stark variierender Netzwerklatenz oder Bereichseinschränkungen von drahtlosen Geräten kann zu Frustrationen bei den Benutzern führen. Ein schneller Wechsel zwischen Arbeitsstationen und Zugriff auf dieselben Anwendungen bei jeder Anmeldung ist eine Priorität für viele mobile Mitarbeiter, z. B. von Mitarbeitern in einem Krankenhaus.

Die folgenden Features dienen dazu, die Sitzungszuverlässigkeit zu optimieren, Unannehmlichkeiten, Ausfallzeiten und Produktivitätsverluste zu reduzieren, und mobilen Benutzern einen schnellen und einfachen Wechsel zwischen Geräten zu ermöglichen.

- Sitzungszuverlässigkeit
- Automatische Wiederverbindung von Clients
- ICA-Keep-Alive
- Workspace Control

## Sitzungszuverlässigkeit

Durch die Sitzungszuverlässigkeit bleiben Sitzungen aktiv und auf dem Bildschirm des Benutzers, wenn die Netzwerkkonnektivität unterbrochen wird. Die Benutzer sehen so lange weiterhin die Anwendung, die sie verwenden, bis die Netzwerkkonnektivität wiederhergestellt ist.

Diese Funktion ist besonders für mobile Benutzer mit drahtlosen Verbindungen geeignet. Ein Benutzer mit einer drahtlosen Verbindung fährt z. B. in einen Tunnel und die Verbindung wird vorübergehend unterbrochen. Normalerweise würde die Sitzung getrennt und nicht mehr auf dem Bildschirm angezeigt. Der Benutzer müsste sich neu mit der getrennten Sitzung verbinden. Mit der Sitzungszuverlässigkeit bleibt die Sitzung auf der Maschine aktiv. Der Verlust der Verbindung ist zu erkennen, da auf dem Client der Bildschirm fixiert und der Mauszeiger als sich drehende Sanduhr angezeigt wird, bis die Verbindung am Ende des Tunnels wiederhergestellt ist. Der Benutzer kann während der Unterbrechung weiterhin auf die Anzeige zugreifen und mit der Anwendung weiterarbeiten, wenn die Netzwerkverbindung wiederhergestellt ist. Die Sitzungszuverlässigkeit verbindet Benutzer ohne Neuauthentifizierung wieder.

Citrix Receiver-Benutzer können die Controllereinstellung nicht überschreiben.

Sie können die Sitzungszuverlässigkeit mit SSL (Secure Sockets Layer) verwenden. Mit SSL werden nur die Daten verschlüsselt, die zwischen dem Benutzergerät und NetScaler Gateway gesendet werden.

Sie aktivieren und konfigurieren die Sitzungszuverlässigkeit mit den folgenden Einstellungen:

- Mit der Richtlinieneinstellung Sitzungszuverlässigkeit - Verbindungen können Sie die Sitzungszuverlässigkeit aktivieren oder deaktivieren.
- Der Standardwert für die Einstellung Sitzungszuverlässigkeit - Timeout ist 180 Sekunden (3 Minuten). Obwohl Sie den Zeitraum vergrößern können, den die Sitzungszuverlässigkeit eine Sitzung offen lässt, sollten Sie dabei berücksichtigen, dass diese Funktion den Benutzer nicht zu einer Neuauthentifizierung auffordert, um den Bedienungskomfort zu erhöhen. Je länger eine Sitzung offen gelassen wird, desto höher ist das Risiko, dass der Benutzer abgelenkt wird und das Benutzergerät verlässt. Benutzer ohne Berechtigung hätten in dem Fall möglicherweise Zugriff auf die Sitzung.
- Eingehende Sitzungszuverlässigkeitsverbindungen verwenden Port 2598, es sei denn, die Portnummer wurde unter Sitzungszuverlässigkeit - Portnummer geändert.
- Verwenden Sie die Funktion zur automatischen Wiederverbindung von Clients, wenn Sie möchten, dass Benutzer eine

Verbindung mit unterbrochenen Sitzungen nur mit einer Neuauthentifizierung wiederherstellen können. Sie können die Einstellung für die Citrix Richtlinie Authentifizierung bei automatischer Wiederverbindung von Clients so konfigurieren, dass Benutzer aufgefordert werden, sich neu zu authentifizieren, wenn sie sich mit einer unterbrochenen Sitzung wiederverbinden.

Wenn Sie sowohl Sitzungszuverlässigkeit als auch die Funktion zur automatischen Wiederverbindung verwenden, werden beide Funktionen nacheinander ausgeführt. Die Sitzungszuverlässigkeit beendet oder trennt die Benutzersitzung, nachdem der mit der Option Sitzungszuverlässigkeit - Timeout festgelegte Zeitraum abgelaufen ist. Anschließend werden die Richtlinieneinstellungen für die automatische Wiederverbindung von Clients wirksam und es wird versucht, eine Verbindung mit der unterbrochenen Sitzung wiederherzustellen.

## Automatische Wiederverbindung von Clients

Mit der automatischen Wiederverbindung von Clients kann Receiver unabsichtlich getrennte ICA-Sitzungen erkennen und die Benutzer automatisch wieder mit den betroffenen Sitzungen verbinden. Wenn diese Funktion auf dem Server aktiviert ist, müssen Benutzer nicht manuell eine neue Verbindung herstellen, um mit ihrer Arbeit fortfahren zu können.

Bei Anwendungssitzungen versucht Receiver, die Verbindung mit der Sitzung wiederherzustellen, bis die Wiederverbindung erfolgreich war oder der Benutzer die Wiederverbindung abbricht.

Bei Desktopsitzungen versucht Receiver eine festgelegte Zeit lang, die Verbindung mit der Sitzung wiederherzustellen, bis die Wiederverbindung erfolgreich war oder der Benutzer die Wiederverbindung abbricht. Der Standardwert für diese Zeit ist fünf Minuten. Um die Zeit zu ändern, bearbeiten Sie die Registrierung auf dem Benutzergerät:

`HKLM\Software\Citrix\ICA Client\TransportReconnectRetryMaxTimeSeconds; DWORD;<Sekunden>`

wobei <Sekunden> die Zeit in Sekunden ist, nach deren Ablauf keine weiteren Versuche der Sitzungswiederherstellung erfolgen.

Sie aktivieren und konfigurieren die automatische Wiederverbindung von Clients mit den folgenden Einstellungen:

- Automatische Wiederverbindung von Clients: Aktiviert oder deaktiviert die automatische Wiederverbindung desselben Clients, nach dem die Verbindung unterbrochen wurde.
- Authentifizierung bei automatischer Wiederverbindung von Clients: Aktiviert oder deaktiviert die erforderliche Benutzerauthentifizierung bei der automatischen Wiederverbindung
- Protokollierung der automatischen Wiederverbindung von Clients: Aktiviert oder deaktiviert die Protokollierung von Wiederverbindungsereignissen im Ereignisprotokoll. Die Protokollierung ist standardmäßig deaktiviert. Wenn diese Einstellung aktiviert ist, werden Informationen zu erfolgreichen oder fehlgeschlagenen automatischen Wiederverbindungsereignissen im Systemprotokoll des Servers aufgezeichnet. Jeder Server speichert die Informationen zu Wiederverbindungsereignissen im eigenen Systemprotokoll; die Site stellt kein Protokoll aller Wiederverbindungsereignisse aller Server bereit.

Bei der automatischen Wiederverbindung von Clients findet eine Authentifizierung mit verschlüsselten Anmeldeinformationen statt. Wenn sich ein Benutzer anmeldet, verschlüsselt und speichert der Server die Anmeldeinformationen und erstellt und sendet ein Cookie mit einem Verschlüsselungsschlüssel an Receiver. Receiver übergibt den Schlüssel zum Wiederverbinden an den Server. Der Server entschlüsselt die Anmeldeinformationen und gibt sie an die Windows-Anmeldung für eine Authentifizierung weiter. Benutzer müssen sich beim Ablaufen von Cookies neu authentifizieren, um Sitzungen wiederherzustellen.

Cookies werden nicht verwendet, wenn Sie die Einstellung Authentifizierung bei automatischer Wiederverbindung von Clients aktivieren. Stattdessen wird der Benutzer in einem Dialogfeld zur Eingabe der Anmeldeinformationen aufgefordert, wenn Receiver versucht, die Verbindung automatisch wiederherzustellen.

Für einen maximalen Schutz der Anmeldeinformationen von Benutzern und von Sitzungen sollten Sie die SSL-Verschlüsselung für die gesamte Kommunikation zwischen Clients und Site verwenden.

Sie deaktivieren die automatische Wiederverbindung in Citrix Receiver für Windows mit der Datei `icaclient.adm`. Diese Datei liegt derzeit nur in Englisch vor. Weitere Informationen finden Sie in der Dokumentation zu Ihrer Version von Receiver für Windows.

Einstellungen für Verbindungen wirken sich auch auf die automatische Wiederverbindung von Clients aus:

- In der Standardeinstellung wird die automatische Wiederverbindung von Clients durch Richtlinieneinstellungen auf der Siteebene aktiviert (siehe oben). Der Benutzer muss sich nicht authentifizieren. Wenn jedoch die ICA-TCP-Verbindung eines Servers so konfiguriert wurde, dass Sitzungen mit einer unterbrochenen Kommunikationsverbindung zurückgesetzt werden, findet die automatische Wiederverbindung nicht statt. Die automatische Wiederverbindung von Clients funktioniert nur, wenn der Server Sitzungen trennt, wenn eine unterbrochene Verbindung oder eine Verbindungszeitout vorliegt. In diesem Zusammenhang bezieht sich die ICA-TCP-Verbindung auf den virtuellen Serverport (nicht auf eine tatsächliche Netzwerkverbindung), der für Sitzungen in TCP/IP-Netzwerken verwendet wird.
- Standardmäßig ist die ICA-TCP-Verbindung auf einem Server so eingestellt, dass Sitzungen mit unterbrochenen Verbindungen oder Verbindungen, die das Zeitlimit überschritten haben, getrennt werden. Getrennte Sitzungen bleiben im Systemspeicher intakt und stehen für eine Wiederverbindung durch Receiver zur Verfügung.
- Die Verbindung kann so konfiguriert werden, dass Sitzungen mit unterbrochenen Verbindungen oder Verbindungen mit Timeouts zurückgesetzt oder abgemeldet werden. Wenn eine Sitzung zurückgesetzt wird, wird bei einem Wiederverbindungsversuch eine neue Sitzung eingeleitet; die Umgebung des Benutzers wird in der verwendeten Anwendung nicht wiederhergestellt, sondern die Anwendung wird neu gestartet.
- Wenn der Server für das Zurücksetzen von Sitzungen konfiguriert ist, erstellt die automatische Wiederverbindung von Clients eine neue Sitzung. Benutzer müssen dann ihre Anmeldeinformationen eingeben, um sich am Server anzumelden.
- Die automatische Wiederverbindung kann fehlschlagen, wenn Receiver oder das Plug-In falsche Authentifizierungsinformationen übergeben (dies kann während eines Angriffs passieren), oder wenn der Server feststellt, dass zu viel Zeit seit dem Erkennen der unterbrochenen Verbindung verstrichen ist.

## ICA-Keep-Alive

ICA-Keep-Alive verhindert, dass Sitzungen durch unterbrochene Verbindungen getrennt werden. Wenn der Server keine Aktivität erkennt (z. B. keine Zeitänderungen, Mausbewegungen oder Bildschirmaktualisierungen) wird verhindert, dass die Sitzung durch die Remotedesktopdienste getrennt wird. Der Server sendet alle paar Sekunden Keep-Alive-Pakete, um zu erkennen, ob die Sitzung aktiv ist. Wenn die Sitzung nicht mehr aktiv ist, wird die Sitzung vom Server als "Getrennt" gekennzeichnet.

Hinweis: ICA-Keep-Alive funktioniert nur, wenn Sie die Sitzungszuverlässigkeit nicht verwenden. Die Sitzungszuverlässigkeit hat eigene Mechanismen für das Aufrechterhalten von Verbindungen. Konfigurieren Sie ICA-Keep-Alive nur für Verbindungen, die keine Sitzungszuverlässigkeit verwenden.

ICA-Keep-Alive-Einstellungen überschreiben Keep-Alive-Einstellungen, die für Microsoft Windows-Gruppenrichtlinien konfiguriert wurden.

Sie aktivieren und konfigurieren ICA-Keep-Alive mit den folgenden Einstellungen:

- ICA-Keep-Alive - Timeout: Gibt das Intervall (1-3600 Sekunden) für das Senden von ICA-Keep-Alive-Meldungen an. Konfigurieren Sie diese Option nicht, wenn die Netzwerksoftware inaktive Sitzungen schließen soll und unterbrochene Verbindungen in der Umgebung so selten sind, dass die Wiederverbindung mit Sitzungen nicht wichtig ist. Die Standardeinstellung von 60 Sekunden bedeutet, dass alle 60 Sekunden ICA-Keep-Alive-Pakete an Benutzergeräte gesendet werden. Antwortet ein Benutzergerät nicht in 60 Sekunden, wird der Status der ICA-Verbindung auf "Getrennt" gesetzt.



- ICA-Keep-Alives: Sendet oder verhindert das Senden von ICA-Keep-Alive-Meldungen.

## Workspace Control

Mit Workspace Control können Desktops und Anwendungen einem Benutzer von einem Gerät zum anderen folgen. Diese Roamingfähigkeit ermöglicht Benutzern den Zugriff auf alle Desktops oder offene Anwendungen von einem beliebigen Ort aus, ohne Neustart des Desktops oder der Anwendungen auf jedem einzelnen Gerät. Sie müssen sich lediglich anmelden. Mit Workspace Control kann das Pflegepersonal in einem Krankenhaus beispielsweise schnell an eine andere Arbeitsstation wechseln und nach der Anmeldung auf dieselben Anwendungen zugreifen. Bei entsprechender Konfiguration von Workspace Control können die Mitarbeiter die Verbindung zu mehreren Anwendungen auf einem Clientgerät trennen und die Verbindung zu denselben Anwendungen auf einem anderen Clientgerät wiederherstellen.

Workspace Control wirkt sich auf die folgenden Aktivitäten aus:

- **Anmelden:** Standardmäßig ermöglicht Workspace Control den Benutzern die Verbindung zu allen ausgeführten Desktops und Anwendungen bei der Anmeldung automatisch wiederherzustellen, ohne sie erneut manuell zu öffnen. Mit Workspace Control können Benutzer getrennte Desktops oder Anwendungen öffnen sowie alle, die auf einem anderen Clientgerät aktiv sind. Beim Trennen der Verbindung mit einem Desktop bzw. einer Anwendung wird das Desktop bzw. die Anwendung weiterhin auf dem Server ausgeführt. Bei Benutzern im Roamingbetrieb, die einige Desktops oder Anwendungen auf einem Clientgerät ausführen müssen, während sie auf einem anderen Clientgerät eine Wiederverbindung zu einem Teil ihres Desktops bzw. ihrer Anwendungen durchführen möchten, können Sie das Wiederverbindungsverhalten bei der Anmeldung so konfigurieren, dass nur die Desktops bzw. Anwendungen geöffnet werden, die zuvor getrennt wurden.
- **Wiederverbinden:** Nach der Anmeldung am Server können Benutzer eine Verbindung zu allen ihren Desktops oder Anwendungen jederzeit wiederherstellen, indem Sie auf "Wiederverbinden" klicken. Beim Wiederverbinden werden standardmäßig sowohl getrennte Desktops oder Anwendungen geöffnet als auch alle aktiven Anwendungen, die derzeit auf einem anderen Clientgerät ausgeführt werden. Sie können die Wiederverbindung so konfigurieren, dass nur die Desktops oder Anwendungen geöffnet werden, deren Verbindung der Benutzer zuvor getrennt hat.
- **Abmelden:** Bei Benutzern, die Desktops oder Anwendungen über StoreFront öffnen, können Sie den Abmeldebefehl so konfigurieren, dass Benutzer entweder von StoreFront und allen aktiven Sitzungen gleichzeitig oder nur von StoreFront abgemeldet werden.
- **Verbindung wird getrennt:** Benutzer können die Verbindung mit allen ausgeführten Desktops und Anwendungen gleichzeitig trennen.

Workspace Control ist nur für Benutzer von Receiver verfügbar, die über eine Citrix StoreFront-Verbindung auf Desktops und Anwendungen zugreifen. Workspace Control ist standardmäßig für virtuelle Desktopsitzungen deaktiviert, für gehostete Anwendungen aber aktiviert. Die Sitzungs freigabe zwischen veröffentlichten Desktops und veröffentlichten Anwendungen in diesen Desktops erfolgt nicht standardmäßig.

Benutzerrichtlinien, Clientlaufwerkzuordnungen und Druckerkonfigurationen ändern sich entsprechend, wenn ein Benutzer ein neues Clientgerät verwendet. Diese Richtlinien und Zuordnungen werden auf dem Clientgerät angewendet, auf dem der Client derzeit bei der Sitzung angemeldet ist. Wenn sich Pflegepersonal z. B. von einem Clientgerät in der Notaufnahme des Krankenhauses abmeldet und dann an einer Arbeitsstation in der Röntgenabteilung anmeldet, gelten für die Sitzung die Richtlinien, Druckerzuordnungen und Clientlaufwerkzuordnungen der Röntgenabteilung, solange die Sitzung gestartet wird.

Sie können die den Benutzern angezeigten Drucker je nach Standort anpassen. Außerdem können Sie steuern, ob Benutzer auf lokalen Druckern drucken können, wie viel Bandbreite bei einer Remoteverbindung verwendet wird sowie andere Aspekte des Druckens.

Weitere Informationen zur Aktivierung und Konfiguration von Workspace Control für Benutzer finden Sie in der StoreFront-





# Durchführung einer Suche in Studio

May 10, 2016

Verwenden Sie die Suchfunktion, um Informationen zu spezifischen Maschinen, Sitzungen, Maschinenkatalogen, Anwendungen oder Bereitstellungsgruppen zu finden.

1. Wählen Sie im Studio-Navigationsbereich Suchen aus.

Hinweis: Sie können mit dem Suchfeld der Registerkarten "Maschinenkataloge" oder "Bereitstellungsgruppen" keine Suche durchführen. Verwenden Sie den Suchknoten im Navigationsbereich.

Zum Anzeigen weiterer Suchkriterien klicken Sie auf das Pluszeichen neben den Dropdownlisten. Klicken Sie zum Entfernen von Suchkriterien auf das Minuszeichen.

2. Geben Sie den Namen ein oder verwenden Sie die Dropdownliste, um eine andere Suchoption für das gesuchte Element auszuwählen.
3. Speichern Sie Ihre Suche, falls gewünscht, durch Wählen von Speichern unter. Die Suche wird in der Liste "Gespeicherte Suchvorgänge" angezeigt.

Sie können auch auf das Symbol "Suche erweitern" (doppelte nach unten gerichtete spitze Klammern) klicken, um eine Dropdownliste mit Sucheigenschaften anzuzeigen. Führen Sie eine erweiterte Suche durch, indem Sie einen Ausdruck anhand der Eigenschaften in der Dropdownliste zusammenstellen.

Tipps zur Verbesserung der Suche:

- Um zusätzliche Eigenschaften in die Anzeige zu integrieren, anhand derer Sie dann suchen und sortieren können, klicken Sie mit der rechten Maustaste auf eine Spalte und wählen Sie Spalten auswählen.
- Wählen Sie zum Suchen eines mit einer Maschine verbundenen Benutzergeräts "Client (IP)" und "ist", und geben Sie die IP-Adresse des Geräts ein.
- Wenn Sie aktive Sitzungen suchen, verwenden Sie "Sitzungszustand", "ist" und "Verbunden".
- Um alle Maschinen in einer Bereitstellungsgruppe aufzuführen, wählen Sie im Navigationsbereich Bereitstellungsgruppen, wählen Sie die Gruppe aus und wählen Sie dann im Aktionsbereich Maschinen anzeigen.

# Unterstützung für IPv4/IPv6

May 10, 2016

Dieses Release unterstützt reines IPv4, reines IPv6 und Bereitstellungen mit dualtem Stapel, bei denen überlappende IPv4- und IPv6-Netzwerke verwendet werden.

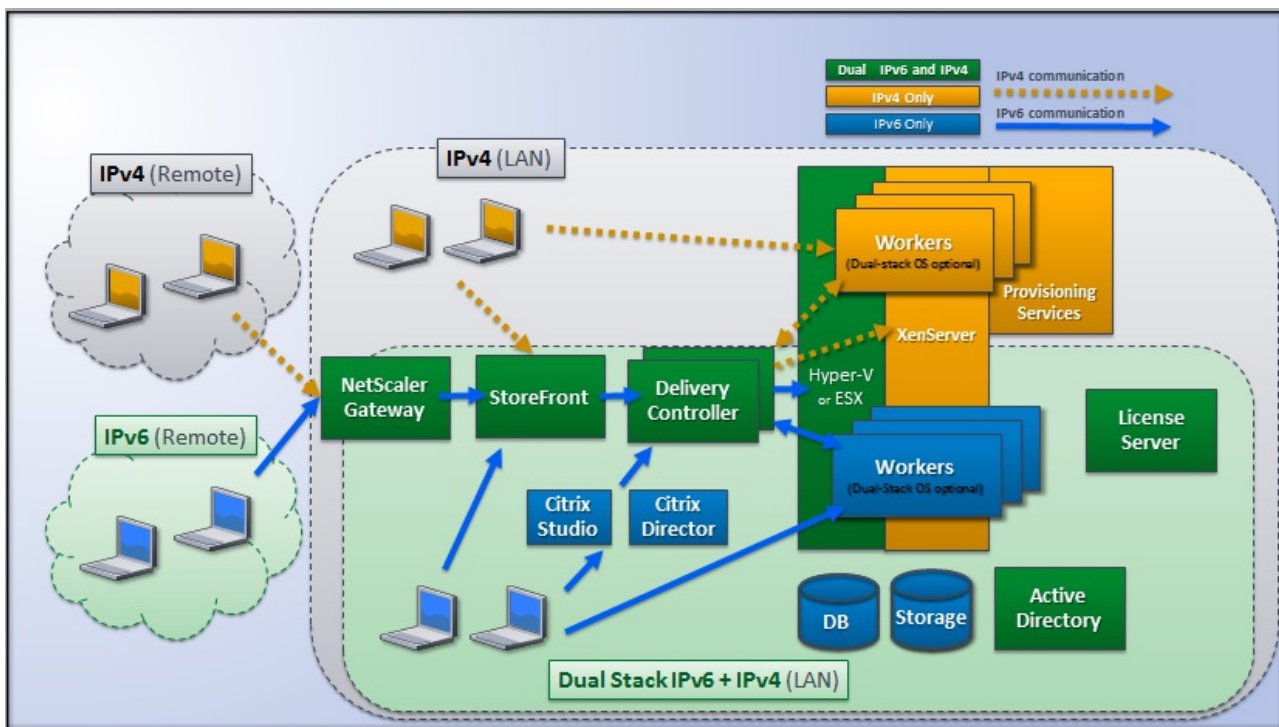
Die IPv6-Kommunikation wird mit zwei verbindungspezifischen Citrix Richtlinieneinstellungen für Virtual Delivery Agent (VDA) gesteuert:

- Die primäre Einstellung erzwingt die Verwendung von IPv6: Nur IPv6-Controllerregistrierung verwenden.
- Die abhängige Einstellung definiert eine IPv6-Netzwerkmaske: IPv6-Netzwerkmaske für Controllerregistrierung.

Wenn Nur IPv6-Controllerregistrierung verwenden aktiviert ist, erfolgt die VDA-Registrierung bei einem Delivery Controller für eingehende Verbindungen über eine IPv6-Adresse.

## IPv4-/IPv6-Bereitstellung mit dualtem Stapel

Die folgende Abbildung zeigt eine IPv4-/IPv6-Bereitstellung mit dualtem Stapel. In diesem Szenario ist ein Worker ein auf einem Hypervisor oder auf einer physischen Maschine installierter VDA, der primär zum Aktivieren von Verbindungen für Anwendungen und Desktops verwendet wird. Komponenten, die für den Parallelbetrieb von IPv6 und IPv4 ausgelegt sind, werden auf Betriebssystemen ausgeführt, die Tunneling oder Dual Protocol-Software nutzen.



Diese Citrix Produkte, Komponenten und Features unterstützen nur IPv4:

- Provisioning Services
- XenServer Version 6.x
- VDAs, die nicht mit der Richtlinieneinstellung Nur IPv6-Controllerregistrierung verwenden gesteuert werden
- XenApp-Versionen vor 7.5, XenDesktop-Versionen vor 7 und EdgeSight

In dieser Bereitstellung gilt:

- Wenn ein Team häufig ein IPv6-Netzwerk verwendet und der Administrator die Nutzung von IPv6-Datenverkehr

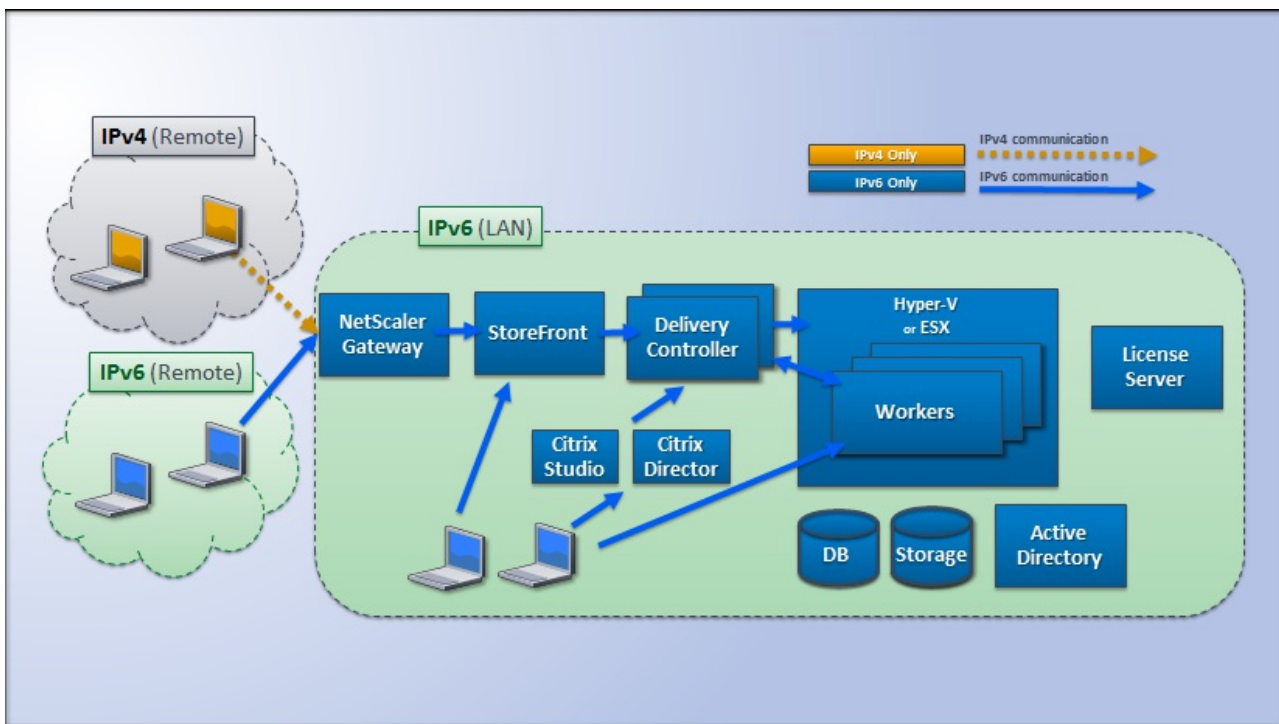
wünscht, veröffentlicht der Administrator IPv6-Desktops und -Anwendungen für die Benutzer auf der Basis eines Workerimages oder einer Organisationseinheit, für das bzw. die die primäre IPv6-Richtlinieneinstellung (Nur IPv6-Controllerregistrierung verwenden) aktiviert ist.

- Wenn ein Team häufig ein IPv4-Netzwerk verwendet, veröffentlicht der Administrator IPv4-Desktops und -Anwendungen für die Benutzer auf der Basis eines Workerimages oder einer Organisationseinheit, für das bzw. die die primäre IPv6-Richtlinieneinstellung deaktiviert ist (d. h. Nur IPv6-Controllerregistrierung verwenden ist deaktiviert = Standardeinstellung).

## Reine IPv6-Bereitstellung

Die folgende Abbildung zeigt eine reine IPv6-Bereitstellung. Für dieses Szenario gilt:

- Die Komponenten werden auf Betriebssystemen ausgeführt, die ein IPv6-Netzwerk unterstützen.
- Die primäre Citrix Richtlinieneinstellung (Nur IPv6-Controllerregistrierung verwenden) ist für alle VDAs aktiviert; sie müssen sich beim Controller mit einer IPv6-Adresse registrieren.



## Richtlinieneinstellungen für IPv6

Bei reiner IPv6-Implementierung bzw. Implementierung von IPv4/IPv6 mit dualem Stapel sind zwei Citrix Richtlinieneinstellungen relevant. Konfigurieren Sie die folgenden verbindungsbezogenen Einstellungen:

- Nur IPv6 Controllerregistrierung verwenden: Steuert das Adressformat, mit dem der Virtual Delivery Agent (VDA) beim Delivery Controller registriert wird. Standard = deaktiviert
  - Wenn der VDA mit dem Controller kommuniziert, wird eine IPv6-Adresse verwendet, deren Auswahl folgender Reihenfolge unterliegt: globale IP-Adresse, Unique Local Address (ULA), Link-Local-Adresse (nur wenn keine anderen IPv6-Adressen verfügbar sind).
  - Bei deaktivierter Einstellung wird der VDA mit der IPv4-Adresse der Maschine für die Kommunikation beim Controller registriert.
- IPv6-Netzwerkmaske für Controllerregistrierung: Eine Maschine kann über mehrere IPv6-Adressen verfügen. Mit dieser Richtlinieneinstellung können Administratoren den VDA auf ein bevorzugtes Subnetz beschränken (anstelle einer globalen IP, sofern registriert). Mit dieser Einstellung wird das Netzwerk festgelegt, in dem der VDA registriert wird: Der VDA wird nur bei der ersten Adresse registriert, die mit der angegebenen Netzwerkmaske übereinstimmt. Diese Einstellung ist nur

gültig, wenn die Richtlinieneinstellung Nur IPv6-Controllerregistrierung verwenden aktiviert ist. Standard = leere Zeichenfolge

Wichtig: Die Verwendung von IPv4 oder IPv6 durch einen VDA wird ausschließlich über diese Richtlinieneinstellungen gesteuert. Das bedeutet, um die IPv6-Adressierung nutzen zu können, muss der VDA von einer Citrix Richtlinie gesteuert werden, bei der die Einstellung Nur IPv6-Controllerregistrierung verwenden aktiviert ist.

#### Überlegungen zur Bereitstellung

Wenn Ihre Umgebung sowohl IPv4- und IPv6-Netzwerke umfasst, benötigen Sie separate Bereitstellungsgruppenkonfigurationen für IPv4-exklusive Clients und für die Clients, die Zugriff auf das IPv6-Netzwerk haben. Verwenden Sie ggf. Namen, die manuelle Active Directory-Gruppenzuweisung oder SmartAccess-Filter zur Unterscheidung der Benutzer.

Die Wiederverbindung mit einer Sitzung kann fehlschlagen, wenn die Verbindung auf einem IPv6-Netzwerk initiiert wird und dann Wiederverbindungsversuche von einem internen Client erfolgen, der nur über IPv4-Zugriff verfügt.

# Clientordnerumleitung

May 10, 2016

Durch die Clientordnerumleitung ändert sich der Zugriff auf clientseitige Dateien bei der hostseitigen Sitzung. Wird auf dem Server nur die Hostlaufwerkzuordnung aktiviert, werden die clientseitigen vollständigen Volumes den Sitzungen automatisch als UNC-Link (Universal Naming Convention) zugeordnet. Wenn Sie die Hostordnerumleitung auf dem Server aktivieren und der Benutzer sie auf dem Benutzergerät konfiguriert, wird der Teil des vom lokalen Benutzer angegebenen lokalen Volumes umgeleitet.

Nur die vom Benutzer angegebenen Ordner statt des kompletten Dateisystems auf dem Benutzergerät werden als UNC-Links in den Sitzungen angezeigt. Wenn Sie UNC-Links durch die Registrierung deaktivieren, werden Clientordner als zugeordnete Laufwerke in der Sitzung angezeigt.

Die Clientordnerumleitung wird nur auf Windows-Desktopbetriebssystemmaschinen unterstützt.

Die Clientordnerumleitung für ein externes USB-Laufwerk wird beim Trennen und Wiederverbinden des Geräts nicht gespeichert.

Aktivieren Sie die Hostordnerumleitung auf dem Server. Geben Sie dann auf dem Clientgerät an, welche Ordner umgeleitet werden sollen (die Anwendung, die Sie zur Angabe der Clientordneroptionen verwenden, ist in diesem Release von Citrix Receiver enthalten).

Achtung: Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

## 1. Auf der Hostseite:

1. Erstellen Sie einen Schlüssel: HKEY\_LOCAL\_MACHINE\SOFTWARE\Wow6432Node\Citrix\Client Folder Redirection.
2. Erstellen Sie einen Wert für REG\_DWORD.
  - Name: CFROnlyModeAvailable
  - Typ: REG\_DWORD
  - Wert: Stellen Sie 1 ein.

## 2. Auf dem Benutzergerät:

1. Stellen Sie sicher, dass die neueste Version von Receiver installiert ist.
2. Starten Sie vom Installationsverzeichnis von Receiver aus CtxCFRUI.exe.
3. Wählen Sie das Optionsfeld Benutzerdefiniert und fügen Sie Ordner hinzu oder bearbeiten oder entfernen Sie Ordner.
4. Trennen Sie die Sitzungen und stellen Sie dann neue Verbindungen her, damit die Einstellung wirksam wird.

# Personal vDisk 7.x

May 10, 2016

Das Personal vDisk-Feature bietet die Einzelimageverwaltung für gepoolte und gestreamte Desktops, während Benutzer Anwendungen installieren und ihre Desktopeinstellungen anpassen können. Im Gegensatz zu traditionellen Virtual Desktop Infrastructure (VDI)-Bereitstellungen mit gepoolten Desktops, bei denen Benutzer Anpassungen und eigene Anwendungen verlieren, wenn der Administrator das Masterimage ändert, werden in Bereitstellungen mit persönlichen vDisks diese Änderungen beibehalten. Dies bedeutet, dass Administratoren die Masterimages auf einfache Weise zentral verwalten können, während Benutzer gleichzeitig von einer individuell angepassten Desktoperfahrung profitieren.

Persönliche vDisks erreichen diese Trennung, indem Sie alle auf der VM des Benutzers vorgenommenen Änderungen an einen separaten Datenträger, die persönliche vDisk, weiterleiten, die mit der VM des Benutzers verknüpft ist. Der Inhalt der persönlichen vDisk wird zur Laufzeit mit dem Inhalt des Masterimages zusammengeführt, um eine einheitliche Erfahrung zu gewährleisten. Auf diese Weise können Benutzer immer noch auf die Anwendungen zugreifen, die der Administrator auf dem Masterimage zur Verfügung gestellt hat.

Persönliche vDisks bestehen aus zwei Teilen, die unterschiedliche Laufwerksbuchstaben verwenden und ungefähr gleich groß sind:

- Benutzerprofil: Dieser Teil enthält Benutzerdaten, Dokumente und das Benutzerprofil. Standardmäßig wird hierfür Laufwerk P: verwendet, Sie können aber einen anderen Laufwerksbuchstaben wählen, wenn Sie einen Maschinenkatalog mit Maschinen erstellen, die persönliche vDisks verwenden. Das verwendete Laufwerk hängt auch von der Einstellung für EnableUserProfileRedirection ab.
- Virtual Hard Disk-Datei (.vhd): Dieser Teil enthält alle anderen Objekte, z. B. Anwendungen, die unter C:\Programme installiert sind. Dieser Teil wird nicht in Windows Explorer angezeigt und benötigt seit Version 5.6.7 keinen Laufwerksbuchstaben.

Mit persönlichen vDisks können Anwendungen auf Abteilungsebene bereitgestellt werden und sie unterstützen auch Anwendungen, die von Benutzern heruntergeladen und installiert wurden, einschließlich solcher, für die Treiber (außer Phase-1-Treiber), Datenbanken und Maschinenverwaltungssoftware erforderlich ist. Wenn die Änderung eines Benutzers mit der Änderung eines Administrators kollidiert, können diese Änderungen mit einer persönlichen vDisk leicht und automatisch abgestimmt werden.

Außerdem können lokal verwaltete Anwendungen (z. B. solche, die von lokalen IT-Abteilungen bereitgestellt werden) auch in der Umgebung des Benutzers bereitgestellt werden. Der Benutzer bemerkt keine Unterschiede bei der Verwendung; mit persönlichen vDisks wird sichergestellt, dass alle Änderungen und alle installierten Anwendungen auf der vDisk gespeichert werden. In Fällen, bei denen eine Anwendung auf einer persönlichen vDisk genau mit einer Anwendung auf einem Masterimage übereinstimmt, wird die Version auf der persönlichen vDisk aus Platzspargründen verworfen, ohne dass der Benutzer den Zugriff auf die Anwendung verliert.

Persönliche vDisks werden physisch auf dem Hypervisor gespeichert, sie müssen sich aber nicht am gleichen Speicherort befinden wie andere auf dem virtuellen Desktop bereitgestellte Datenträger. Dadurch können sich die Kosten für persönlichen vDisk-Speicher verringern.

Wenn Sie während der Site-Erstellung eine Verbindung erstellen, legen Sie Speicherorte für die Datenträger fest, die von den virtuellen Maschinen verwendet werden. Sie können die persönlichen vDisks von den Datenträgern für das Betriebssystem trennen. Jede VM muss Zugriff auf den Speicherort der beiden Datenträger haben. Wenn Sie für beide lokalen Speicher verwenden, muss der Hypervisor in der Lage sein, auf beide zuzugreifen. Um dies zu gewährleisten, bietet Studio nur

kompatible Speicherorte an. Später können Sie auch über "Konfiguration > Hosting" in Studio persönliche vDisks und Speicher dafür vorhandenen Hosts (aber nicht Maschinenkatalogen) hinzufügen.

Erstellen Sie regelmäßig ein Backup der persönlichen vDisks mit der bevorzugten Methode. vDisks sind standardmäßige Volumes in der Speicherebene eines Hypervisors. Sie können genauso wie andere Volumes gesichert werden.

## Neue Features in Personal vDisk 7.6.1

Dieses Release enthält folgende Verbesserungen:

- Diese Version von Personal vDisk enthält Leistungsverbesserungen, durch die die zum Anwenden eines Imageupdates auf einen Personal vDisk-Katalog benötigte Zeit verkürzt wird.

Die folgenden Probleme wurden in diesem Release behoben:

- Bei dem Versuch, eine Basis-VM mit einem direkten Upgrade von Microsoft Office 2010 auf Microsoft Office 2013 zu aktualisieren, wurde dem Benutzer ein Konfigurationsfenster gefolgt von der folgenden Fehlermeldung angezeigt: "Error 25004. The product key you entered cannot be used on this machine." Zuvor wurde in diesem Fall empfohlen, Office 2010 auf der Basis-VM zu deinstallieren und danach Office 2013 zu installieren. Jetzt ist es nicht mehr nötig, Office 2010 beim Durchführen eines direkten Upgrades auf der Basis-VM zu deinstallieren (#391225).
- Wenn während eines Imageupdates eine höhere Version von Microsoft .Net auf der persönlichen vDisk des Benutzers gefunden wurde, wurde diese vom Basisimage mit einer niedrigeren Version überschrieben. Dies führte für Benutzer zu Problemen beim Ausführen bestimmter, auf ihrer persönlichen vDisk installierter Anwendungen, die eine höhere Version, z. B. Visual Studio, benötigten (#439009).
- Ein mit Provisioning Services bereitgestellter Datenträger, auf dem Personal vDisk installiert und aktiviert ist, kann nicht zum Erstellen eines nicht-Personal vDisk-Maschinenkatalogs verwendet werden. Diese Einschränkung wurde entfernt (#485189).

## Info zu Personal vDisk 7.6

Neue Features in Version 7.6:

- Verbesserte Fehlerbehandlung und Berichterstellung für Personal vDisk. Wenn Sie in Studio PvD-aktivierte Maschinen in einem Katalog anzeigen, können Sie auf der Registerkarte "PvD" den Überwachungsstatus während Imageupdates sowie die geschätzte Abschlusszeit und den Fortschritt verfolgen. Außerdem wurden die Zustandsanzeigen verbessert.
- Ein Überwachungstool für PvD-Imageupdates ist für frühere Versionen auf dem ISO-Medium verfügbar (ISO\Support\Tools\Scripts\PvdTool). Überwachungsfunktionen wurden auch in früheren Releases unterstützt, jedoch sind die Berichterstellungsfunktionen nicht so robust wie im aktuellen Release.
- Im Provisioning Services-Testmodus können Sie Maschinen mit einem aktualisierten Image in einem Testkatalog starten. Nachdem Sie die Stabilität überprüft haben, können Sie die Testversion der persönlichen vDisk auf "Production" hochstufen.
- Mit einer neuen Funktion können Sie das Delta (die Differenz) zwischen zwei Beständen während einer Bestandsaktualisierung berechnen statt sie für jeden PvD-Desktop einzeln berechnen zu müssen. Es sind neue Befehle zum Exportieren und Importieren eines vorherigen Bestands für MCS-Kataloge verfügbar. (Provisioning Services-Master-vDisks haben bereits den vorherigen Bestand.)

Bekannte Probleme aus Version 7.1.3, die in Version 7.6 behoben wurden:

- Unterbrechen des Upgrades einer Personal vDisk-Installation kann dazu führen, dass eine vorhandene Personal vDisk-Installation beschädigt wird. [#424878]
- Ein virtueller Desktop reagiert möglicherweise nicht mehr, wenn die persönliche vDisk über einen langen Zeitraum ausgeführt wird und es zu einem Speicherverlust im nicht-ausgelagerten Speicher kommt. [#473170]



Neue bekannten Probleme in Version 7.6:

- Das Vorhandensein von Antivirenprodukten kann sich darauf auswirken, wie lange das Durchführen einer Bestandsaktualisierung oder eines Updates dauert. Die Leistung kann verbessert werden, wenn Sie CtxPvD.exe und CtxPvDSvc.exe der PROCESS-Ausschlussliste des Antivirenprodukts hinzufügen. Diese Dateien befinden sich im Ordner C:\Programme\Citrix\personal vdisk\bin. [#326735]
- Feste Links zwischen Dateien, die vom Masterimage geerbt wurden, bleiben in persönlichen vDisk-Katalogen nicht erhalten. [#368678]
- Nach dem Upgrade von Office 2010 auf 2013 auf dem Personal vDisk-Masterimage kann Office nicht auf virtuellen Maschinen gestartet werden, da der KMS-Lizenzierungsproduktschlüssel von Office beim Upgrade entfernt wurde. Als Workaround können Sie Office 2010 deinstallieren und Office 2013 auf dem Masterimage neu installieren. [#391225]
- Personal vDisk-Kataloge unterstützen keine VMware Paravirtual SCSI (PVSCSI)-Controller. Verwenden Sie den Standardcontroller, um dieses Problem zu vermeiden. [#394039]
- Für virtuelle Desktops, die mit Personal vDisk Version 5.6.0 erstellt wurden und auf 7 aktualisiert werden, finden Benutzer, die sich an der Master-VM anmelden, nicht alle Dateien in der gepoolten VM. Dieses Problem tritt auf, da ein neues Benutzerprofil erstellt wird, wenn sich Benutzer an der gepoolten VM anmelden. Für dieses Problem gibt es keinen Workaround. [#392459]
- Personal vDisks, die unter Windows 7 ausgeführt werden, können das Feature "Sichern und Wiederherstellen" nicht verwenden, wenn der Windows-Systemschutz aktiviert ist. Wenn der Systemschutz deaktiviert ist, wird ein Backup des Benutzerprofils aber nicht der Datei userdata.v2.vhd erstellt. Citrix empfiehlt, den Systemschutz zu deaktivieren und ein Backup des Benutzerprofils mit dem Feature "Sichern und Wiederherstellen" zu erstellen. [#360582]
- Beim Erstellen einer VHD-Datei auf der Basis-VM mit dem Datenträgerverwaltungstool können Sie die virtuelle Festplatte (VHD) möglicherweise nicht bereitstellen. Um dieses Problem zu umgehen, kopieren Sie die VHD auf das PvD-Volume. [#355576]
- Office 2010-Verknüpfungen bleiben auf virtuellen Desktops erhalten, nachdem diese Software entfernt wurde. Um dieses Problem zu umgehen, löschen Sie die Verknüpfungen. [#402889]
- Wenn Sie Microsoft Hyper-V verwenden, können Sie mit Maschinen mit persönlichen vDisks keinen Katalog erstellen, wenn die Maschinen lokal gespeichert sind und die vDisks auf freigegebenen Clustervolumes (CSVs) gespeichert sind. Die Katalogerstellung schlägt fehl und ein Fehler wird angezeigt. Um dieses Problem zu umgehen, verwenden Sie einen anderen Speicherort für die vDisks. [#423969]
- Wenn Sie sich das erste Mal bei einem virtuellen Desktop anmelden, der von einem Provisioning Services-Katalog erstellt wurde, werden Sie aufgefordert, den Desktop neu zu starten, wenn die persönliche vDisk zurückgesetzt wurde (mit dem Befehl "ctxpvd.exe -s reset"). Um dieses Problem zu lösen, starten Sie den Desktop neu, wenn Sie dazu aufgefordert werden. Es handelt sich hier um eine einmalige Zurücksetzung, die bei erneuter Anmeldung nicht erforderlich ist. [#340186]
- Wenn Sie .NET 4.5 auf einer persönlichen vDisk installieren und .NET 4.0 bei einem späteren Update des Images installiert oder bearbeitet wird, schlagen Anwendungen, die von .NET 4.5 abhängig sind, fehl. Um dieses Problem zu umgehen, stellen Sie .NET 4.5 vom Basisimage aus als Imageupdate bereit.
- Weitere Informationen finden Sie unter  
— *Bekannte Probleme*  
in der Dokumentation zu XenApp und XenDesktop 7.6.

### Info zu Personal vDisk 7.1.3

Bekannte Probleme aus Version 7.1.1, die in Version 7.1.3 behoben wurden:

- Direkte Upgrades von Personal vDisk 5.6.0 auf Personal vDisk 7.x können zu Fehlern bei Personal vDisk führen. [#432992]
- Benutzer können möglicherweise nur zeitweilige Verbindungen zu virtuellen Desktops mit persönlichen vDisks herstellen. [#437203]

- Wenn das Update eines Personal vDisk-Images während der Aktualisierung von Personal vDisk 5.6.5 oder höher auf Personal vDisk 7.0 oder höher unterbrochen wird, können spätere Updates fehlschlagen. [#436145]

### Info zu Personal vDisk 7.1.1

Bekannte Probleme aus Version 7.1, die in Version 7.1.1 behoben wurden:

- Nach der Aktualisierung auf Symantec Endpoint Protection 12.1.3 über ein Imageupdate meldet symhelp.exe beschädigte Antivirus-Definitionen. [#423429]
- Personal vDisk kann den Neustart gepoolter Desktops verursachen, wenn der Dienststeuerungs-Manager (services.exe) abstürzt. [#0365351]

Neue bekannten Probleme in Version 7.1.1: Keine

### Info zu Personal vDisk 7.1

Neue Features in Version 7.1:

- Sie können Personal vDisk nun auch auf Desktops unter Windows 8.1 verwenden und die Ereignisprotokollierung wurde verbessert.
- Copy-on-Write (CoW) wird in diesem Release nicht mehr unterstützt. Beim Upgrade von Personal vDisk Version 7.0 auf 7.1 gehen alle von CoW verwalteten Datenänderungen verloren. CoW war ein experimentelles Feature in XenDesktop 7 und es war standardmäßig deaktiviert. Wenn Sie es nicht aktiviert haben, sind Sie nicht betroffen.

Bekannte Probleme aus Version 7.0.1, die in Version 7.1 behoben wurden:

- Wenn der Wert des Personal vDisk-Registrierungsschlüssels EnableProfileRedirection auf 1 oder ON eingestellt ist, und Sie den Wert beim Aktualisieren des Images in 0 oder OFF ändern, wird der ganze Personal vDisk-Speicherplatz u. U. den vom Benutzer installierten Anwendungen zugeordnet, sodass kein Platz für Benutzerprofile vorhanden ist, die auf der vDisk bleiben. Wenn die Profillumleitung für einen Katalog deaktiviert ist und Sie die Profillumleitung während eines Imageupdates aktivieren, können sich Benutzer u. U. nicht bei ihren virtuellen Desktops anmelden. [#381921]
- Der Desktopdienst protokolliert nicht den richtigen Fehler in der Ereignisanzeige, wenn das Update des Personal vDisk-Bestands fehlschlägt. [#383331]
- Beim Upgrade auf Personal vDisk 7.x bleiben geänderte Regeln nicht erhalten. Dieses Problem wurde für das Upgrade von Version 7.0 auf Version 7.1 behoben. Beim Upgrade von Version 5.6.5 auf Version 7.1 müssen Sie die Regeldatei zuerst speichern und die Regeln nach dem Upgrade erneut anwenden. [#388664]
- Personal vDisks, die unter Windows 8 ausgeführt werden, können keine Anwendungen vom Windows Store installieren. Eine Fehlermeldung gibt an, dass der Kauf nicht abgeschlossen werden konnte. Wenn Sie den Windows Update-Dienst aktivieren, wird das Problem nun behoben. Von Benutzern installierte Anwendungen müssen jedoch nach dem Neustart des Systems neu installiert werden. [#361513]
- Einige symbolische Verknüpfungen fehlen auf gepoolten Windows 7-Desktops mit persönlichen vDisks. Daher werden Symbole, die von Anwendungen unter C:\Benutzer\Alle Benutzer gespeichert werden, nicht im Startmenü angezeigt. [#418710]
- Der Start einer persönlichen vDisk schlägt fehl, wenn ein USN-Journalüberlauf aufgrund zahlreicher Systemänderungen nach einer Bestandsaktualisierung auftritt. [#369846]
- Der Start einer persönlichen vDisk schlägt fehl und der Statuscode 0x20 sowie Fehlercode 0x20000028 werden angezeigt. [#393627]
- Symantec Endpoint Protection 12.1.3 zeigt die Fehlermeldung "Proactive Threat Protection is malfunctioning" an und der Liveupdatestatus dieser Komponente ist nicht verfügbar. [#390204]

Neue bekannte Probleme in Version 7.1: Weitere Informationen finden Sie unter

— *Bekannte Probleme*

in der Dokumentation zu XenDesktop 7.1.

## Info über Personal vDisk 7.0.1

Neu in Version 7.0.1: Personal vDisk ist jetzt robuster bei Änderungen der Umgebung. Virtuelle Desktops mit persönlichen vDisks werden jetzt beim Delivery Controller registriert, selbst wenn das Imageupdate fehlschlägt, und das Herunterfahren von unsicheren Systemen setzt die vDisks in einen permanent deaktivierten Zustand. Darüber hinaus können Sie nun während einer Bereitstellung Dateien und Ordner mithilfe von Regeldateien von vDisks ausschließen.

Bekannte Probleme aus Version 5.6.13, die in Version 7.0.1 behoben wurden:

- Änderungen an der Mitgliedschaft einer Gruppe, die von einem Benutzer an einem gepoolten virtuellen Desktop vorgenommen wurden, gehen u. U. nach einem Imageupdate verloren. [#286227]
- Imageupdates schlagen u. U. mit einer Fehlermeldung, dass der Speicherplatz auf dem Datenträger niedrig ist, fehl, selbst wenn die persönliche vDisk genug Speicherplatz hat. [#325125]
- Die Installation einiger Anwendungen schlägt auf virtuellen Desktops mit einer persönlichen vDisk fehl, und eine Meldung weist auf einen erforderlichen Neustart hin. Grund ist eine ausstehende Umbenennung. [#351520]
- Symbolische Links, die im Masterimage erstellt wurden, funktionieren nicht auf virtuellen Desktops mit persönlichen vDisks. [#352585]
- In Umgebungen mit der Citrix Profilverwaltung und Personal vDisk funktionieren Anwendungen, die Benutzerprofile auf einem Systemvolume überprüfen u. U. nicht richtig, wenn die Profilverwaltung aktiviert ist. [#353661]
- Die Bestandsaktualisierung schlägt auf Masterimages fehl, wenn der Bestand größer als 2 GB ist. [#359768]
- Imageupdates schlagen mit einem Fehlercode 112 fehl und persönliche vDisks sind beschädigt, selbst wenn die vDisks ausreichend Speicherplatz für das Update haben. [#363003]
- Das Skript für das Ändern der Größe schlägt für Kataloge fehl, die mehr als 250 Desktops haben. [#363365]
- Benutzerseitige Änderungen einer Umgebungsvariablen gehen bei einer Aktualisierung des Images verloren. [#372295]
- Lokale Benutzer, die auf einem virtuellen Desktop mit einer persönlichen vDisk erstellt wurden, gehen bei einer Aktualisierung des Images verloren. [#377964]
- Der Start einer persönlichen vDisk schlägt u. U. fehl, wenn ein USN-Journalüberlauf aufgrund zahlreicher Systemänderungen nach einer Bestandsaktualisierung auftritt. Erhöhen Sie die USN-Journalgröße im Masterimage auf mindestens 32 MB und aktualisieren Sie das Image, um dieses Problem zu vermeiden. [#369846]
- Ein Problem wurde mit Personal vDisk festgestellt, das ein richtiges Funktionieren der Registrierungsstrukturaktionen des AppSense-Umgebungs-Managers verhindert, wenn AppSense im Ersetzenmodus verwendet wird. Citrix und AppSense arbeiten an der Behebung des Problems, das mit dem Verhalten der RegRestoreKey-API zusammenhängt, wenn Personal vDisk installiert ist. [#0353936]

## Release-unabhängige bekannte Probleme

- Wenn eine Anwendung auf einer persönlichen vDisk (PvD) mit einer auf dem Masterimage installierten anderen Anwendung derselben Version verbunden ist, funktioniert die Anwendung auf der PvD nach einem Imageupdate möglicherweise nicht mehr. Dieses Problem tritt auf, wenn die Anwendung vom Masterimage deinstalliert oder auf eine neuere Version aktualisiert wird, da durch diese Aktion die Dateien entfernt werden, die die Anwendung auf der PvD vom Masterimage benötigt. Um dies zu verhindern, lassen Sie die Anwendung mit den Dateien, die von der Anwendung auf der PvD benötigt werden, auf dem Masterimage.  
Beispiel: Das Masterimage enthält Office 2007 und ein Benutzer installiert Visio 2007 auf der PvD. Die Office-Anwendungen und Visio funktionieren einwandfrei. Später ersetzt der Administrator Office 2007 durch Office 2010 auf dem Masterimage und aktualisiert anschließend alle betroffenen Maschinen mit dem aktualisierten Image. Visio 2007 funktioniert nicht mehr. Um dies zu verhindern, lassen Sie Office 2007 auf dem Masterimage. [#320915]
- Wenn Personal vDisk verwendet wird, verwenden Sie bei der Bereitstellung von McAfee Virus Scan Enterprise (VSE) die

Version 8.8 Patch 4 oder höher auf einem Masterimage. [#303472]

- Wenn eine Verknüpfung für eine Datei auf dem Masterimage nicht mehr funktioniert, weil das Verknüpfungsziel in PvD umbenannt wurde, erstellen Sie die Verknüpfung neu. [#367602]
- Verwenden Sie keine absoluten bzw. festen Links auf einem Masterimage. [#368678]
- Das mit Windows 7 verfügbare Feature "Sichern und Wiederherstellen" wird auf der persönlichen vDisk nicht unterstützt. [#360582]
- Nach der Anwendung eines aktualisierten Masterimages ist kein Zugriff auf die Konsole für lokale Benutzer und Gruppen möglich oder sie zeigt inkonsistente Daten an. Um das Problem zu lösen, setzen Sie die Benutzerkonten auf der VM zurück. Dazu muss die Sicherheitsstruktur zurückgesetzt werden. Dieses Problem wurde im Release 7.1.2 behoben und löst auch das Problem für VMs in späteren Releases, aber nicht für VMs, die mit einer früheren Version erstellt und dann aktualisiert wurden. [#488044]
- Bei der Verwendung einer gepoolten VM in einer ESX Hypervisor-Umgebung wird Benutzern eine Neustartaufforderung angezeigt, wenn der ausgewählte SCSI-Controllertyp ein "VMware Paravirtual" ist. Um dieses Problem zu umgehen, verwenden Sie den Controllertyp "LSI SCSI". [#394039]
- Nach dem Zurücksetzen von PvD auf einem mit Provisioning Services erstellten Desktop wird Benutzern nach der Anmeldung an der VM u. U. eine Neustartaufforderung angezeigt. Um dieses Problem zu umgehen, starten Sie den Desktop neu. [#340186]
- Benutzer von Windows 8.1-Desktops können sich u. U. nicht an ihren PvDs anmelden. Einem Administrator wird möglicherweise die Meldung "PvD was disabled due to unsafe shutdown" angezeigt und das PvDActivation-Protokoll enthält u. U. die Meldung "Failed to load reg hive [\Device\IvmVhdDisk00000001\CitrixPvD\Settings\RingCube.dat]". Dieses Problem tritt auf, wenn die VM nicht sicher heruntergefahren wird. Sie umgehen das Problem, indem Sie die persönliche vDisk zurücksetzen. [#474071]

# Installation und Upgrade

Nov 02, 2016

Personal vDisk 7.x wird seit XenDesktop 5.6 bis zur aktuellen Version unterstützt. Für jede XenDesktop-Version werden in der Dokumentation unter "Systemanforderungen" die unterstützten Betriebssysteme für Virtual Delivery Agents (VDAs) sowie die unterstützten Versionen von Hosts (Virtualisierungsressourcen) und Provisioning Services aufgelistet. Weitere Informationen zu Provisioning Services-Aufgaben finden Sie in der Dokumentation zu Provisioning Services.

## Installieren und Aktivieren von PvD

PvD wird automatisch installiert, wenn Sie einen VDA für Desktopbetriebssysteme auf einer Maschine installieren oder aktualisieren. Wenn Sie die PvD-Software nach der Installation des VDAs aktualisieren, verwenden Sie die [hier](#) verfügbare PvD-MSI (Citrix Anmeldeinformationen erforderlich).

Aktivieren von PvD:

- PvD wird automatisch aktiviert, wenn Sie mit den Maschinenerstellungsdiensten (MCS) einen Maschinenkatalog erstellen, dessen Desktopbetriebssystemmaschinen eine persönliche vDisk verwenden.
- Wenn Sie Provisioning Services (PVS) verwenden, wird PvD automatisch aktiviert, wenn Sie während der Erstellung des Masterimages den Bestand aufnehmen oder wenn bei einem automatischen Update der Bestand aktualisiert wird.

Während der VDA-Installation haben Sie die Möglichkeit, PvD zu aktivieren, indem Sie das Kontrollkästchen "Personal vDisk" auf der grafischen Benutzeroberfläche auswählen oder über die Befehlszeilenschnittstelle die Option "/baseimage" angeben. Auch wenn Sie diese Aktion bei der Installation des VDAs (Standard) nicht ausführen, können Sie mit dem gleichen Image PvD-Desktops und Desktops ohne PvD erstellen, da PvD während der Katalogerstellung aktiviert wird.

## Hinzufügen von persönlichen vDisks

Sie fügen Hosts persönliche vDisks hinzu, wenn Sie eine Site konfigurieren. Sie können denselben Speicher auf dem Host für VMs und persönliche vDisks verwenden oder Sie können einen anderen Speicher für persönliche vDisks auswählen.

Später können Sie auch persönliche vDisks und ihren Speicher vorhandenen Hosts (Verbindungen), jedoch nicht Maschinenkatalogen, hinzufügen.

1. Wählen Sie im Studio-Navigationsbereich Konfiguration > Hosting.
2. Wählen Sie im Aktionsbereich Persönlichen vDisk-Speicher hinzufügen und geben Sie einen Speicherort an.

## Aktualisieren von PvD

Die einfachste Methode, Personal vDisk von einer früheren 7.x-Version zu aktualisieren, ist das Aktualisieren der VDAs für Desktopbetriebssysteme mit der aktuellen XenDesktop-Version. Nehmen Sie anschließend den PvD-Bestand auf.

Sie können auch nur PvD mit der [hier](#) verfügbaren PvD-MSI aktualisieren.

## Deinstallieren von PvD

Sie können die PvD-Software mit einer der folgenden beiden Methoden deinstallieren:

- Deinstallieren Sie den VDA. Dabei wird die PvD-Software ebenfalls entfernt.
- Wenn Sie PvD mit der PvD-MSI aktualisiert haben, können Sie es über die Liste der Programme deinstallieren.

Wenn Sie PvD deinstallieren und dieselbe oder eine neuere Version neu installieren, erstellen Sie eine Sicherungskopie des Registrierungsschlüssels HKLM\Software\Citrix\personal vDisk\config, der Konfigurationseinstellungen für die Umgebung

enthält, die sich geändert haben können. Nach der Installation von PVD können Sie die Registrierungswerte, die sich geändert haben, durch einen Vergleich mit der Sicherungskopie zurücksetzen.

# Konfiguration und Verwaltung

May 10, 2016

In diesem Abschnitt werden Themen erläutert, die beim Konfigurieren und Verwalten einer Personal vDisk (PvD)-Umgebung zu berücksichtigen sind. Darüber hinaus werden Best Practices und Aufgabenbeschreibungen behandelt.

Beachten Sie Folgendes beim Arbeiten in der Windows-Registrierung:

Achtung: Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab. Überlegungen zur Größe einer persönlichen vDisk

Die folgenden Faktoren beeinflussen die Größe des PvD-Hauptvolumes:

- **Größe der Anwendungen, die Benutzer auf ihren PvDs installieren**

Bei Neustarts bestimmt PvD den im Anwendungsbereich verbleibenden freien Speicherplatz (UserData.v2.vhd). Wenn dieser Wert unter 10 % fällt, wird der Anwendungsbereich erweitert, indem ungenutzter Speicherplatz des Profilbereichs (standardmäßig der freie Speicherplatz auf dem Laufwerk P:) genutzt wird. Der zum Anwendungsbereich hinzugefügte Speicherplatz beträgt ungefähr 50 % des zusammengefassten freien Speicherplatzes von Anwendungs- und Profilbereich.

Beispiel: Wenn der Anwendungsbereich auf einer 10 GB PvD, der standardmäßig eine Größe von 5 GB hat, den Wert 4,7 GB erreicht und der Profilbereich über 3 GB freien Speicherplatz verfügt, wird der zusätzliche Speicherplatz wie folgt berechnet:

erweiterter Speicherplatz =  $(5,0 - 4,7) : 2 + 3,0 : 2 = 1,65$  GB

Der Wert für den zusätzlichen Speicherplatz kann nur ungefähr angegeben werden, da Abstriche für das Speichern von Protokollen und für Mehraufwand gemacht werden müssen. Die Berechnung und mögliche Größenänderung wird bei jedem Neustart ausgeführt.

- **Größe der Benutzerprofile (wenn keine separate Profilverwaltungslösung verwendet wird)**

Zusätzlich zu dem für Anwendungen erforderlichen Speicherplatz muss auf persönlichen vDisks auch genügend Speicherplatz für das Speichern von Benutzerprofilen vorhanden sein. Schließen Sie alle nicht umgeleiteten speziellen Ordner (z. B. Dokumente und Musik) in Ihre Speicherplatzberechnungen ein. Vorhandene Profilgrößen sind in der Systemsteuerung (sysdm.cpl) verfügbar.

Einige Profilumleitungslösungen speichern Stubdateien (Sentineldateien) anstelle echter Profildaten. Zu Beginn kann es aussehen, als würden die Profillösungen keine Daten speichern. Sie nutzen jedoch einen Dateiverzeichniseintrag pro Stubdatei im Dateisystem, was ungefähr 4 KB pro Datei ausmacht. Wenn Sie eine solche Lösung verwenden, schätzen Sie die Größe anhand der echten Profildaten und nicht basierend auf den Stubdateien.

Unternehmensanwendungen zur Dateifreigabe, wie ShareFile und Dropbox, synchronisieren oder laden Daten möglicherweise auf die Profildatenbereiche der persönlichen vDisks von Benutzern herunter. Wenn Sie derartige Anwendungen verwenden, veranschlagen Sie genug Speicherplatz für diese Daten.

- **Mehraufwand durch die virtuelle Festplattenvorlage, die den PvD-Bestand enthält**

Die virtuelle Festplattenvorlage enthält die PvD-Bestandsdaten (Sentineldateien, die zum Inhalt des Masterimages

gehören). Der PvD-Anwendungsbereich wird von dieser virtuellen Festplatte erstellt. Da jede Sentineldatei bzw. jeder Sentinelordner einen Dateiverzeichniseintrag im Dateisystem enthält, nimmt der Inhalt der virtuellen Festplattenvorlage PvD-Anwendungsspeicherplatz in Anspruch, bevor der Endbenutzer überhaupt eine Anwendung installiert hat. Sie können die Größe der virtuellen Festplattenvorlage bestimmen, indem Sie zum Masterimage navigieren, nachdem der Bestand aufgenommen wurde. Sie können zur ungefähren Berechnung auch die folgende Formel verwenden:

Größe der virtuellen Festplattenvorlage = (Anzahl der Dateien des Basisimage) x 4 KB

Sie ermitteln die Anzahl der Dateien und Ordner, indem Sie mit der rechten Maustaste auf das Laufwerk C: des Images der Basis-VM klicken und Eigenschaften auswählen. Beispiel: Ein Image mit 250.000 Dateien ergibt eine virtuelle Festplattenvorlage von ungefähr 1.024.000.000 Bytes (nicht ganz 1 GB). Dieser Speicherplatz ist nicht für Anwendungsinstallationen im PvD-Anwendungsbereich verfügbar.

- **Mehraufwand für PvD-Imageupdatevorgänge**

Während PvD-Imageupdatevorgänge ausgeführt werden, muss im Stammverzeichnis der PvD (standardmäßig P:) genug Speicherplatz vorhanden sein, um die Änderungen der zwei Imageversionen mit den Änderungen, die der Benutzer an der PvD vorgenommen hat, zusammenzuführen. Normalerweise reserviert die PvD einige Hundert MB für diesen Zweck. Durch Extradaten, die im Laufwerk P: gespeichert wurden, ist jedoch möglicherweise nicht genug Speicherplatz zum Durchführen des Imageupdates vorhanden. Mit dem PvD-Poolstatistikscript (auf dem XenDesktop-Installationsmedium im Ordner "Support/Tools/Scripts") oder mit dem Überwachungstool für PvD-Imageupdates (im Ordner "Support/Tools/PvdTool") können Sie die PvD-Datenträger in einem Katalog identifizieren, für die ein Update geplant ist und die fast voll sind.

Das Vorhandensein von Antivirenprodukten kann sich darauf auswirken, wie lange das Durchführen einer Bestandsaktualisierung oder eines Updates dauert. Die Leistung kann verbessert werden, wenn Sie CtxPvD.exe und CtxPvDSvc.exe der Ausschlussliste des Antivirenprodukts hinzufügen. Diese Dateien befinden sich im Ordner C:\Programme\Citrix\personal vdisk\bin. Durch das Ausschließen dieser ausführbaren Dateien vom Antivirenskan kann die Bestandsaktualisierungs- und Imageupdateleistung um das Zehnfache beschleunigt werden.

- **Mehraufwand für unerwartete Zunahme (unerwartete Anwendungsinstallationen usw.)**

Kalkulieren Sie zusätzlichen Speicherplatz ein (entweder eine feste Menge oder einen Prozentsatz der vDisk-Größe), um auf unerwartete Anwendungsinstallationen von Benutzern während der Bereitstellung vorbereitet zu sein.

So konfigurieren Sie die Größe und Zuordnung der persönlichen vDisk

Sie können den Algorithmus für die automatische Größenänderung, der die Größe der virtuellen Festplatte relativ zum Laufwerk P: festlegt, manuell anpassen, indem Sie die Ausgangsgröße der virtuellen Festplatte festlegen. Dies ist nützlich, wenn Sie beispielsweise wissen, dass Benutzer zahlreiche Anwendungen installieren werden, die nicht alle auf die virtuelle Festplatte passen werden, selbst wenn die Größe mit dem Algorithmus angepasst wurde. In dieser Situation können Sie die Ausgangsgröße des Anwendungsspeicherplatzes erhöhen, damit die vom Benutzer installierten Anwendungen genug Platz haben.

Passen Sie die Ausgangsgröße der virtuellen Festplatte am besten auf einem Masterimage an. Sie können die Größe der virtuellen Festplatte auch auf einem virtuellen Desktop anpassen, wenn ein Benutzer nicht genügend Speicherplatz für die Installation einer Anwendung hat. Sie müssen diesen Vorgang jedoch auf allen betroffenen virtuellen Desktops wiederholen, da Sie die Ausgangsgröße der virtuellen Festplatte nicht in einem bereits erstellten Katalog ändern können.

Stellen Sie sicher, dass die virtuelle Festplatte groß genug für das Speichern von Definitionsdateien von Antivirensoftware ist, da diese Dateien normalerweise sehr umfangreich sind.



Legen Sie die folgenden Registrierungsschlüssel in HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\personal vDisk\Config fest. (Ändern Sie keine anderen Einstellungen in diesem Registrierungsschlüssel.) Alle Einstellungen müssen auf dem Masterimage angegeben werden (außer für MinimumVHDSIZEinMB, was auf einer individuellen Maschine geändert werden kann). Auf dem Masterimage angegebene Einstellungen werden bei der nächsten Aktualisierung des Images angewendet.

- **MinimumVHDSIZEinMB**

Gibt die Mindestgröße (in Megabyte) des Anwendungsteils (C:) der persönlichen vDisk an. Die neue Größe muss größer als die vorhandene Größe und kleiner als die Größe des Datenträgers abzüglich PvDReservedSpaceMB sein.

Durch das Erhöhen dieses Werts wird freier Speicherplatz vom Profiteil der vDisk dem Laufwerk C: zugeteilt. Diese Einstellung wird ignoriert, wenn ein geringerer Wert als die aktuelle Größe des Laufwerks C: verwendet wird oder wenn EnableDynamicResizeOfAppContainer auf 0 eingestellt ist.

Standard = 2048

- **EnableDynamicResizeOfAppContainer**

Aktiviert oder deaktiviert den Algorithmus zur dynamischen Größenänderung.

- Bei der Einstellung auf 1 wird der Anwendungsspeicherplatz auf C: automatisch geändert, wenn der freie Speicherplatz auf C: unter 10 % fällt. Zulässige Werte sind 1 und 0. Ein Neustart muss durchgeführt werden, damit die Änderungen wirksam werden.
- Bei der Einstellung auf 0 wird die Größe der virtuellen Festplatte basierend auf der Methode ermittelt, die in XenDesktop-Versionen vor 7.x verwendet wurde.

Standard = 1

- **EnableUserProfileRedirection**

Aktiviert oder deaktiviert die Umleitung des Benutzerprofils auf die vDisk.

- Bei der Einstellung auf 1 leitet PvD das Benutzerprofil auf die vDisk um (standardmäßig auf P:). Profile werden entsprechend einem Windows-Standardprofil im Allgemeinen nach P:\Users umgeleitet. Durch die Umleitung werden die Profile beibehalten, falls der PvD-Desktop zurückgesetzt werden muss.
- Bei der Einstellung auf 0 ist der gesamte Speicherplatz auf der vDisk, abzüglich PvDReservedSpaceMB, dem Laufwerk C: (dem Anwendungsteil der vDisk) zugewiesen, und das vDisk-Laufwerk (P:) ist im Windows Explorer ausgeblendet. Citrix empfiehlt das Deaktivieren der Umleitung durch die Einstellung auf 0, wenn Sie Citrix Profilverwaltung oder eine andere Roamingprofillösung verwenden.

Durch diese Einstellung bleiben die Profile in C:\Users erhalten statt auf die vDisk umgeleitet zu werden, und sie ermöglicht der Roamingprofillösung das Verwalten der Profile.

Dieser Wert stellt sicher, dass der ganze Speicherplatz auf P: Anwendungen zugewiesen ist.

Bei einer Einstellung dieses Werts auf 0 wird davon ausgegangen, dass eine Profilverwaltungslösung eingesetzt wird. Das Deaktivieren der Profulumleitung ohne eine Roamingprofillösung einzusetzen wird nicht empfohlen, da die Profile sonst bei nachfolgenden PvD-Zurücksetzungsvorgängen gelöscht werden.

Ändern Sie diese Einstellung nicht, wenn das Image aktualisiert wird, da sie zwar nicht den Speicherort vorhandener Profile ändert, jedoch den gesamten Speicherplatz auf der PvD dem Laufwerk C: zuweist und die PvD ausblendet.

Konfigurieren Sie diesen Wert vor dem Bereitstellen eines Katalogs. Nach dem Bereitstellen eines Katalogs können Sie diesen Wert nicht mehr ändern.

Wichtig: Ab XenDesktop 7.1 werden Änderungen an diesem Wert nicht angewendet, wenn Sie ein Imageupdate ausführen. Legen Sie den Wert des Schlüssels fest, wenn Sie die Kataloge erstellen, von denen die Profile erstellt werden. Sie können das Umleitungsverhalten später nicht mehr ändern.

Standard = 1

- **PercentOfPvDForApps**

Legt die Teilung zwischen dem Anwendungsteil (C:) und dem Profiltail der vDisk fest. Dieser Wert wird beim Erstellen neuer VMs verwendet sowie während Imageupdates, wenn EnableDynamicResizeOfAppContainer auf 0 eingestellt ist.

Das Ändern der Einstellung PercentOfPvDForApps macht nur einen Unterschied, wenn die Einstellung EnableDynamicResizeOfAppContainer auf 0 festgelegt ist. Standardmäßig ist EnableDynamicResizeOfAppContainer auf 1 (aktiviert) festgelegt, d. h. dass der AppContainer (Laufwerk C:) nur erweitert wird, wenn er fast voll ist (d. h. dynamisch), und zwar wenn weniger als 10 % freier Speicherplatz vorhanden ist.

Durch das Erhöhen von PercentOfPvDForApps wird nur der Gesamtspeicherplatz für die Apps erhöht. Der Speicherplatz wird jedoch nicht sofort verfügbar gemacht. Sie müssen zudem die Zuordnungsteilung im Masterimage konfigurieren, die dann beim nächsten Imageupdate angewendet wird.

Wenn Sie bereits einen Katalog mit Maschinen mit der Einstellung 1 für EnableDynamicResizeOfAppContainer generiert haben, ändern Sie die Einstellung im Masterimage für das nächste Update auf 0 und konfigurieren Sie die entsprechende Zuordnungsteilung. Die angeforderte Teilungsgröße wird eingehalten, solange sie größer als die aktuell zugeteilte Größe des Laufwerks C ist.

Wenn Sie komplette Kontrolle über die Speicherplatzteilung haben möchten, legen Sie den Wert auf 0 fest. Damit haben Sie die komplette Kontrolle über die Größe des Laufwerks C und die Erweiterung des Laufwerks ist nicht davon abhängig, dass ein Benutzer eine bestimmte Menge Speicherplatz unter dem Schwellenwert in Anspruch nimmt.

Standard = 50 % (beiden Teilen wird der gleiche Speicherplatz zugeteilt)

- **PvDReservedSpaceMB**

Legt die Größe des reservierten Speicherplatzes (in MB) auf der vDisk für das Speichern von Personal vDisk-Protokollen und anderen Daten fest.

Wenn in Ihrer Bereitstellung XenApp 6.5 (oder eine frühere Version) und Anwendungsstreaming verwendet wird, erhöhen Sie diesen Wert um die Größe des RadeCache.

Standard = 512

- **PvDResetUserGroup**

Gilt nur für XenDesktop 5.6: Ermöglicht der angegebenen Gruppe von Benutzern, eine persönliche vDisk zurücksetzen. Höhere XenDesktop-Releases verwenden dafür die delegierte Administration.

Weitere Einstellungen:

- **Windows Update Service:** Stellen Sie sicher, dass im Masterimage für Windows Updates die Einstellung Never Check for Update und für den Windows Update Service die Einstellung Disabled ausgewählt ist. Falls der Windows Update Service auf der PvD ausgeführt werden muss, wird durch die Einstellung Never Check for Update verhindert, dass die Updates auf den zugeordneten Maschinen installiert werden.  
Dieser Dienst wird von Windows 8 Store benötigt, um Modern UI-Apps installieren zu können.
- **Windows Updates:** Diese schließen Updates für Internet Explorer ein und müssen auf das Masterimage angewendet werden.
- **Updates requiring restarts:** Bei auf das Masterimage angewendeten Windows Updates sind, abhängig von den im Update enthaltenen Patches, möglicherweise mehrere Neustarts erforderlich, damit die Installation vollständig ausgeführt wird. Stellen Sie vor der PvD-Bestandsaktualisierung sicher, dass Sie das Masterimage richtig neu starten,

damit die Installation von ausgeführten Windows Updates vollständig abgeschlossen wird.

- **Application updates:** Aktualisieren Sie Anwendungen, die auf dem Masterimage installiert sind, um Speicherplatz auf den vDisks von Benutzern zu sparen. Auf diese Weise vermeiden Sie den doppelten Aufwand, der durch das Durchführen von Updates auf den vDisks einzelner Benutzer entstehen würde.

## Überlegungen zu Anwendungen auf dem Masterimage

Es kann zu Konflikten zwischen Software und der von PvD erstellten Benutzerumgebung kommen. Zum Vermeiden von Konflikten müssen Sie die Software auf dem Masterimage (nicht auf den individuellen Maschinen) installieren. Auch wenn es keine Konflikte zwischen der Software und der Ausführung von PvD gibt, empfiehlt Citrix, sie auf dem Masterimage zu installieren.

Folgende Anwendungen müssen auf dem Masterimage installiert werden:

- Agents und Clients (z. B. System Center Configuration Manager-Agent, App-V-Client, Citrix Receiver)
- Anwendungen, die vorrangige Starttreiber installieren oder ändern
- Anwendungen, die Drucker- oder Scannersoftware bzw. -treiber installieren
- Anwendungen, die den Windows-Netzwerkstapel ändern
- VM-Tools wie VMware Tools und XenServer Tools

Folgende Anwendungen sollten auf dem Masterimage installiert werden:

- Anwendungen, die einer großen Anzahl an Benutzern zur Verfügung gestellt werden. Deaktivieren Sie für die folgenden Fälle vor der Bereitstellung die Anwendungsupdates:
  - Unternehmensanwendungen, die Volumenlizenzierung verwenden, z. B. Microsoft Office und Microsoft SQL Server
  - Gängige Anwendungen wie Adobe Reader, Firefox und Chrome
- Große Anwendungen, z. B. SQL Server, Visual Studio und Anwendungsframeworks wie .NET

Die folgenden Empfehlungen und Einschränkungen gelten für Anwendungen, die Benutzer auf Maschinen mit persönlichen vDisks installiert haben. Einige dieser Empfehlungen können nicht durchgesetzt werden, wenn Benutzer Administratorrechte haben:

- Benutzer sollten Anwendungen nicht vom Masterimage deinstallieren und dann auf ihrer persönlichen vDisk neu installieren.
- Vorsicht beim Aktualisieren oder Deinstallieren von Anwendungen auf dem Masterimage. Nachdem Sie die Version einer Anwendung auf dem Image installiert haben, installiert ein Benutzer möglicherweise eine Add-On-Anwendung (z. B. ein Plug-In), das diese Version erfordert. Im Fall einer solchen Abhängigkeit kann es nach dem Aktualisieren oder Deinstallieren der Anwendung auf dem Image beim Add-On zu einer Fehlfunktion kommen. Beispiel: Microsoft Office 2010 ist auf einem Masterimage installiert und ein Benutzer installiert Visio 2010 auf der persönlichen vDisk. Bei einem Upgrade von Office auf dem Masterimage kann die lokal installierte Visio-Anwendung unbrauchbar werden.
- Software mit Lizenzen, die von Hardware abhängig sind (entweder durch ein Dongle oder signaturbasierte Hardware), wird nicht unterstützt.

## Überlegungen zu Provisioning Services

Wenn Sie Provisioning Services mit PvD verwenden:

- Das SOAP-Dienstkonto muss zum Administratorknoten von Studio hinzugefügt werden und es muss über die Rolle "Maschinenadministrator" oder eine höhere Rolle verfügen. Dadurch wird sichergestellt, dass die PvD-Desktops in den Status "Preparing" versetzt werden, wenn die Provisioning Services (PVS) vDisk auf "Production" hochgestuft wird.
- Das Versionsverwaltungsfeature von Provisioning Services muss zum Aktualisieren der persönlichen vDisk verwendet werden. Wenn die Version auf "Production" hochgestuft wird, versetzt der SOAP-Dienst die PvD-Desktops in den Status "Preparing".

- Die Größe der persönlichen vDisk sollte immer größer als der Provisioning Services-Schreibcachedatenträger sein, da Provisioning Services sonst fälschlicherweise die persönliche vDisk als Schreibcache nehmen könnte.
- Nach dem Erstellen einer Bereitstellungsgruppe können Sie die persönliche vDisk mit dem [Überwachungstool für PvD-Imageupdates](#) oder den [Größenänderungs- und poolstats-Skripts](#) (personal-vdisk-poolstats.ps1) überwachen.

Veranschlagen Sie genug Schreibcache. Bei normalem Betrieb werden die meisten Benutzerschreibvorgänge (Änderungen) von PvD erfasst und an die persönliche vDisk umgeleitet. Daher könnten Sie theoretisch die Größe des Provisioning Services-Schreibcache verringern. Wenn PvD jedoch nicht aktiv ist, z. B. während Imageupdatevorgängen, kann ein kleines Provisioning Services-Schreibcache schnell voll sein und den Absturz von Maschinen verursachen.

Citrix empfiehlt, dass Sie die Größe des Provisioning Services-Schreibcache entsprechend den Empfehlungen für Provisioning Services veranschlagen und dann Speicherplatz hinzufügen, der dem Doppelten der virtuellen Festplattenvorlage auf dem Masterimage entspricht (für Zusammenführungen). Es ist unwahrscheinlich, dass ein Zusammenführungsvorgang so viel Speicherplatz beansprucht, aber es ist möglich.

Wenn Sie mit Provisioning Services einen Katalog mit PvD-aktivierten Maschinen bereitstellen:

- Folgen Sie den Anweisungen in der Provisioning Services-Dokumentation.
- Sie können die Drosselungseinstellungen für Energieaktionen ändern, indem Sie die Verbindung in Studio bearbeiten.
- Zum Aktualisieren der Provisioning Services-vDisk installieren bzw. aktualisieren Sie Anwendungen und Software und starten Sie die vDisk neu. Aktualisieren Sie dann den PvD-Bestand und fahren Sie die VM herunter. Stufen Sie anschließend die neue Version auf "Production" hoch. Die PvD-Desktops im Katalog sollten automatisch in den Zustand "Preparing" versetzt werden. Wenn dies nicht der Fall ist, prüfen Sie, ob das SOAP-Dienstkonto Maschinenadministrator- oder höhere Privilegien auf dem Controller hat.

Mit dem Testmodusfeature von Provisioning Services können Sie einen Testkatalog mit Maschinen erstellen, die ein aktualisiertes Masterimage verwenden. Wenn die Tests die Funktionsfähigkeit des Katalogs bestätigen, stufen Sie ihn auf "Production" hoch.

## Überlegungen zu den Maschinenerstellungsdiensten

Wenn Sie mit den Maschinenerstellungsdiensten (MCS) einen Katalog mit PvD-aktivierten Maschinen bereitstellen:

- Folgen Sie den Anweisungen in der XenDesktop-Dokumentation.
- Aktualisieren Sie nach dem Erstellen des Masterimages den PvD-Bestand und schalten Sie dann die VM aus (PvD funktioniert nicht ordnungsgemäß, wenn Sie die VM nicht ausschalten). Erstellen Sie einen Snapshot vom Masterimage.
- Geben Sie im Assistenten zum Erstellen von Maschinenkatalogen die Größe und den Laufwerksbuchstaben der persönlichen vDisk an.
- Nach dem Erstellen einer Bereitstellungsgruppe können Sie die persönliche vDisk mit dem [Überwachungstool für PvD-Imageupdates](#) oder den [Größenänderungs- und poolstats-Skripts](#) (personal-vdisk-poolstats.ps1) überwachen.
- Sie können die Drosselungseinstellungen für Energieaktionen ändern, indem Sie die Verbindung in Studio bearbeiten.
- Wenn Sie das Masterimage aktualisieren, nehmen Sie nach dem Aktualisieren der Anwendungen und der anderen Software auf dem Image den PvD-Bestand auf und schalten Sie anschließend die VM aus. Erstellen Sie einen Snapshot vom Masterimage.
- Überprüfen Sie mit dem Überwachungstool für PvD-Imageupdates oder mit dem Skript "personal-vdisk-poolstats.ps1", ob genügend Speicherplatz auf jeder PvD-aktivierten VM vorhanden ist, die das aktualisierte Masterimage verwendet.
- Nach dem Aktualisieren des Maschinenkatalogs werden die PvD-Desktops in den Zustand "Preparing" versetzt, während sie die Änderungen am neuen Masterimage verarbeiten. Die Desktops werden entsprechend der während des Maschinenupdates festgelegten Rolloutstrategie aktualisiert.
- Während die PvD im Zustand "Preparing" ist, überwachen Sie sie mit dem Überwachungstool für PvD-Imageupdates oder mit dem Skript "personal-vdisk-poolstats.ps1".

## So schließen Sie Dateien und Ordner von vDisks aus

Mit Regeldateien schließen Sie Dateien und Ordner von vDisks aus. Dies ist möglich, während die persönlichen vDisks bereitgestellt werden. Die Regeldateien werden `custom_*_rules.template.txt` genannt und befinden sich im Ordner `\config`. Anmerkungen in den einzelnen Dateien erhalten zusätzliche Informationen.

## So führen Sie beim Aktualisieren eines Masterimages eine Bestandsaktualisierung durch

Wenn Sie PvD nach einer Aktualisierung des Masterimages aktivieren, muss der Bestand des Datenträgers aktualisiert und ein neuer Snapshot erstellt werden.

Wenn Sie eine Anwendung installieren, die Binärdateien im Benutzerprofil des Administrators ablegt, ist die Anwendung nicht für Benutzer freigegebener virtueller Desktops (einschließlich solcher, die auf gepoolten Maschinenkatalogen basieren und mit PvD-Maschinenkatalogen gepoolt sind) verfügbar, da Masterimages nicht von Benutzern sondern von Administratoren verwaltet werden. Benutzer müssen solche Anwendungen selber installieren.

Es ist ratsam, nach jedem Schritt in diesem Verfahren einen Snapshot des Images zu erstellen:

1. Aktualisieren Sie das Masterimage, indem Sie auf der Maschine Anwendungen oder Betriebssystemupdates installieren und das System konfigurieren.  
Bei Masterimages, die auf Windows XP basieren und die Sie mit persönlichen vDisks bereitstellen möchten, müssen Sie sicherstellen, dass keine Dialogfelder offen sind (z. B. Meldungen zur Bestätigung von Softwareinstallationen oder Aufforderungen, nicht signierte Treiber zu verwenden). Offene Dialogfelder auf Masterimages in dieser Umgebung verhindern, dass sich der VDA beim Delivery Controller registriert. Sie können Aufforderungen für nicht signierte Treiber in der Systemsteuerung verhindern. Navigieren Sie zu "System > Hardware > Treibersignierung" und wählen Sie die Option zum Ignorieren von Warnungen.
2. Fahren Sie die Maschine herunter. Klicken Sie bei Windows 7-Maschinen auf Abbrechen, wenn Citrix Personal vDisk das Herunterfahren behindert.
3. Klicken Sie im Citrix Personal vDisk-Dialogfeld auf Bestand aktualisieren. Dieser Vorgang kann einige Minuten dauern.  
Wichtig: Wenn Sie das anschließende Herunterfahren unterbrechen (selbst wenn Sie nur eine kleine Änderung am Image vornehmen), stimmt der Bestand der persönlichen vDisk nicht mehr mit dem Masterimage überein. Dies führt dazu, dass das Personal vDisk-Feature nicht mehr funktioniert. Wenn Sie das Herunterfahren unterbrechen, müssen Sie die Maschine neu starten, herunterfahren und auf "Bestand aktualisieren" klicken, wenn Sie dazu aufgefordert werden.
4. Erstellen Sie, nachdem der Bestandvorgang die Maschine heruntergefahren hat, einen Snapshot des Masterimages.

Sie können einen Bestand in eine Netzwerkfreigabe exportieren und ihn anschließend in ein Masterimage importieren. Weitere Informationen finden Sie unter [Exportieren und Importieren eines PvD-Bestands](#).

## So konfigurieren Sie Drosselungseinstellungen für Verbindungen

Der Citrix Brokerdienst steuert den Energiezustand der Maschinen, die Desktops und Anwendungen bereitstellen. Der Brokerdienst kann mehrere Hypervisoren über einen Delivery Controller steuern. Die Interaktion zwischen einem Controller und dem Hypervisor wird durch Broker-Energieaktionen gesteuert. Aktionen, die den Energiezustand einer Maschine ändern, wird eine Priorität zugewiesen und dann werden sie über einen Drosselungsmechanismus an den Hypervisor gesendet, damit es nicht zur Überlastung kommt. Die folgenden Einstellungen wirken sich auf die Drosselung aus. Diese Werte werden festgelegt, indem Sie eine Verbindung (Seite "Erweitert") in Studio bearbeiten.

So konfigurieren Sie Drosselungswerte für eine Verbindung:

1. Wählen Sie im Studio-Navigationsbereich Konfiguration > Hosting.
2. Wählen Sie die Verbindung und dann im Aktionsbereich Verbindung bearbeiten.

3. Sie können die folgenden Werte ändern:

- **Gleichzeitige Aktionen (alle Typen):** Das zulässige Maximum für gleichzeitig ausgeführte Energieaktionen. Diese Einstellung wird als absoluter Wert und als Prozentsatz der Verbindung mit dem Hypervisor angegeben. Der niedrigere der beiden Werte wird verwendet.  
Standard = 100 absolut, 20%
- **Gleichzeitige Updates für Personal vDisk-Bestand:** Das zulässige Maximum für gleichzeitige Personal vDisk-Energieaktionen. Diese Einstellung wird als absoluter Wert und als Prozentsatz der Verbindung angegeben. Der niedrigere der beiden Werte wird verwendet.  
Standard = 50 absolut, 25%

So kalkulieren Sie den absoluten Wert: Bestimmen Sie den Gesamtwert für IOPS (TIOPS), den der Datenträger des Endbenutzers unterstützt (dies sollte vom Hersteller festgelegt sein oder berechnet werden). Veranschlagen Sie 350 IOPS pro VM (IOPS/VM), um die Anzahl der VMs zu bestimmen, die jeweils auf dem Datenträger aktiv sein können. Sie berechnen diesen Wert, indem Sie den Gesamtwert für IOPS durch IOPS/VM teilen.

Beispiel: Wenn der Wert für den Datenträger des Endbenutzers 14.000 IOPS ist, ist die Anzahl der aktiven VMs  $14.000 \text{ IOPS} : 350 \text{ IOPS/VM} = 40$ .

- **Höchstanzahl neue Aktionen pro Minute:** Die maximale Anzahl der neuen Energieaktionen, die pro Minute an den Hypervisor gesendet werden können. Sie wird als absoluter Wert angegeben.  
Standard = 10

So identifizieren Sie die optimalen Werte für diese Einstellungen in der Bereitstellung:

1. Messen Sie mit den Standardwerten die Reaktionszeit für das Imageupdate eines Testkatalogs. Dies ist die Differenz zwischen der Startzeit eines Imageupdates (T1) und dem Zeitpunkt, wenn der VDA auf der letzten Maschine des Katalogs beim Controller registriert wird (T2). Reaktionszeit =  $T2 - T1$ .
2. Messen Sie die Ein- und Ausgabevorgänge pro Sekunde (IOPS) auf dem Hypervisorspeicher während des Imageupdates. Diese Daten können als Benchmark für die Optimierung dienen. (Die Standardwerte sind möglicherweise die beste Einstellung. Unter Umständen erreicht das System den maximalen IOPS-Wert, sodass die Einstellungswerte herabgesetzt werden müssen.)
3. Ändern Sie den Wert für "Gleichzeitige Updates für Personal vDisk-Bestand" wie unten beschrieben und lassen Sie alle anderen Einstellungen unverändert.
  1. Erhöhen Sie den Wert um 10 und messen Sie die Reaktionszeit nach jeder Änderung. Fahren Sie fort, den Wert um 10 zu erhöhen und messen Sie das Ergebnis, bis die Reaktionszeit abnimmt oder keine Änderung mehr auftritt.
  2. Wenn durch das Erhöhen des Werts im vorherigen Schritt keine Verbesserung erzielt wurde, verringern Sie den Wert schrittweise um 10 und messen Sie die Reaktionszeit nach jeder Verringerung. Wiederholen Sie diesen Vorgang, bis sich die Reaktionszeit nicht mehr ändert oder verbessert. Dieser Wert ist wahrscheinlich der optimale Wert für die PvD-Energieaktion.
4. Wenn Sie den Einstellungswert für die PvD-Energieaktion festgelegt haben, optimieren Sie nacheinander die Werte für die gleichzeitigen Aktionen (alle Typen) und die Höchstanzahl der neuen Aktionen pro Minute. Folgen Sie den oben erläuterten Schritten (schrittweises Erhöhen und Verringern der Werte), um verschiedene Werte zu testen.

So verwenden Sie Microsoft System Center Configuration Manager 2007 mit PvD

System Center Configuration Manager (Configuration Manager) 2012 erfordert keine besondere Konfiguration und kann auf die gleiche Weise wie alle anderen Anwendungen auf dem Masterimage installiert werden. Die folgenden Informationen gelten nur für System Center Configuration Manager 2007. Configuration Manager-Versionen vor Configuration Manager 2007 werden nicht unterstützt.

Führen Sie die folgenden Schritte aus, um die Configuration Manager 2007 Agent-Software in einer PvD-Umgebung zu verwenden.

1. Installieren Sie den Client-Agent auf dem Masterimage.
  1. Installieren Sie den Configuration Manager-Client auf dem Masterimage.
  2. Beenden Sie den ccmexec-Dienst (SMS-Agent) und deaktivieren Sie ihn.
  3. Löschen Sie SMS- oder Clientzertifikate wie folgt aus dem Zertifikatspeicher des lokalen Computers:
    - Gemischter Modus: Certificates (Lokaler Computer)\SMS\Certificates
    - Einheitlicher Modus
      - Certificates (Lokaler Computer)\Personal\Certificates
      - Löschen Sie das Clientzertifikat, das von Ihrer Zertifizierungsstelle ausgestellt wurde (normalerweise eine interne PKI).
  4. Löschen Sie C:\Windows\smscfg.ini oder benennen Sie die Datei um.
2. Entfernen Sie Informationen, die den Client eindeutig identifizieren.
  1. (Optional) Löschen Sie die Protokolldateien unter C:\Windows\System32\CCM\Logs oder verschieben Sie sie.
  2. Installieren Sie ggf. den Virtual Delivery Agent und nehmen Sie den PvD-Bestand auf.
  3. Fahren Sie das Masterimage herunter, erstellen Sie einen Snapshot und erstellen Sie dann mit diesem Snapshot einen Maschinenkatalog.
3. Überprüfen Sie Personal vDisk und starten Sie die Dienste. Führen Sie diese Schritte einmal auf jedem PvD-Desktop aus, nachdem er zum ersten Mal gestartet wurde. Dazu können Sie beispielsweise ein Domänen-GPO verwenden.
  - Bestätigen Sie, dass PvD aktiv ist, indem Sie prüfen, ob der Registrierungsschlüssel HKLM\Software\Citrix\personal vDisk\config\virtual vorhanden ist.
  - Stellen Sie den ccmexec-Dienst (SMS-Agent) auf "Automatic" ein und starten Sie den Dienst. Der Configuration Manager-Client kontaktiert den Configuration Manager-Server und ruft neue, eindeutige Zertifikate und GUIDs ab.



# Tools

May 10, 2016

Mit den folgenden Tools und Hilfsprogrammen können Sie PVD-Vorgänge anpassen, vereinfachen und überwachen.

## Benutzerdefinierte Regeldateien

Mit den von PVD bereitgestellten benutzerdefinierten Regeldateien können Sie das folgende Standardverhalten von PVD-Imageupdates ändern:

- Die Sichtbarkeit von Dateien auf der PVD
- Die Art der Zusammenführung von vorgenommenen Änderungen
- Einstellungen zur Beschreibbarkeit der Dateien

Detaillierte Anweisungen zu den benutzerdefinierten Regeldateien und dem CoW-Feature finden Sie in den Kommentaren zu den Dateien, die sich unter C:\ProgramData\Citrix\personal vDisk\Config auf der Maschine befinden, auf der PVD installiert ist. Die Dateien mit dem Namen custom\_\* erläutern die Regeln und wie sie aktiviert werden.

## Ändern der Größe und poolstats-Skripte

Es gibt zwei Skripts zum Überwachen und Verwalten der Größe der PVDs. Sie befinden sich im Ordner "Support\Tools\Scripts" auf dem XenDesktop-Installationsmedium. Sie können auch das Überwachungstool für PVD-Imageupdates verwenden, das sich im Ordner "Support\Tools\Scripts\PvdTool" befindet. Weitere Informationen finden Sie unter <http://blogs.citrix.com/2014/06/02/introducing-the-pvd-image-update-monitoring-tool/>.

Verwenden Sie "resize-personalvdisk-pool.ps1" zum Vergrößern der PVDs in allen Desktops eines Katalogs. Die folgenden Snap-Ins oder Module für den Hypervisor müssen auf der Maschine installiert werden, auf der Studio ausgeführt wird:

- XenServer erfordert XenServerPSSnapin
- vCenter erfordert vSphere PowerCLI
- System Center Virtual Machine Manager erfordert die VMM-Konsole

Mit "personal-vdisk-poolstats.ps1" können Sie den Status von Imageupdates überprüfen sowie den Speicherplatz für Anwendungen und Benutzerprofile in einer PVD-Gruppe. Führen Sie das Skript vor dem Update eines Images aus, um zu prüfen, ob Desktops genug Speicherplatz haben. Dadurch werden Fehler während des Updates verhindert. Für das Skript muss die Windows Management Instrumentation (WMI-In)-Firewall auf den PVD-Desktops aktiviert sein. Sie können die Firewall auf dem Masterimage aktivieren oder über GPO.

Wenn ein Imageupdate fehlschlägt, zeigt der Eintrag in der Spalte "Update" die Ursache an.

## Zurücksetzen des Anwendungsbereichs

Wenn ein Desktop durch die Installation einer fehlerhaften Anwendung oder aus einem anderen Grund beschädigt wird, können Sie den Anwendungsbereich der PVD auf den (leeren) Herstellerstandard zurücksetzen. Beim Zurücksetzen bleiben die Benutzerprofilaten erhalten.

So setzen Sie den Anwendungsbereich der PVD zurück:

- Melden Sie sich am Desktop des Benutzers als Administrator an. Starten Sie eine Eingabeaufforderung und führen Sie den Befehl C:\Programme\Citrix\Personal vDisk\bin\CtxPvD.exe -s Reset aus.
- Navigieren Sie in Citrix Director zum Desktop des Benutzers. Klicken Sie auf Reset Personal vDisk und anschließend auf OK.

## Exportieren und Importieren eines PVD-Bestands

Der Imageupdateprozess ist ein zentraler Teil der Bereitstellung neuer Images auf PVD-Desktops und umfasst Anpassungen, damit vorhandene persönliche vDisks mit dem neuen Basisimage funktionieren. Bei Bereitstellungen, die Maschinenerstellungsdienste (MCS) verwenden, können Sie einen Bestand von einer aktiven VM auf eine Netzwerkfreigabe exportieren und dann in ein Masterimage importieren. Mit diesem Bestand auf dem Masterimage wird eine Differenz berechnet. Obwohl das Feature zum Exportieren bzw. Importieren des Bestands nicht verwendet werden muss, kann es die Leistung des Imageupdateprozesses verbessern.

Sie müssen ein Administrator sein, um das Feature zum Exportieren/Importieren des Bestands zu verwenden. Authentifizieren Sie sich für den Export/Import ggf. bei der Dateifreigabe mit "net user". Der Benutzerkontext muss auf alle für den Export/Import verwendeten Dateifreigaben zugreifen können.

- Führen Sie zum Exportieren eines Bestands den Exportbefehl als Administrator auf einer Maschine aus, auf der sich ein VDA mit aktivierter PVD befindet (Mindestversion 7.6):  
Ctxpvdsvc.exe exportinventory "<Pfad zum Exportspeicherort>"

Die Software erkennt den Speicherort des aktuellen Bestands und exportiert den Bestand an den angegebenen Speicherort in einen Ordner mit dem Namen "ExportedPvdInventory". Hier ein Auszug aus der Befehlsausgabe:

- C:\Program Files\Citrix\personal vDisk\bin> .\CtxPvDsvc.exe exportinventory \\share location\ExportedInventory Current inventory source location C:\CitrixPvD\Settings\Inventory\VER-LAS ... Exporting
- Zum Importieren eines zuvor exportierten Bestands führen Sie den Importbefehl als Administrator auf dem Masterimage aus:

## Importieren

Führen Sie den Importbefehl als Administrator auf dem Masterimage aus.

Ctxpvdsvc.exe importinventory "<Pfad zum exportierten Bestand>"

Der <Pfad zum exportierten Bestand> muss der vollständige Pfad für die Bestandsdateien sein; in der Regel ist das <Netzwerkspeicherort\ExportedPvdInventory>.

Der Bestand wird aus dem Importspeicherort abgerufen (zuvor wurde er mit dem Befehl zum Exportieren des Bestands hierher exportiert) und in den Bestandspeicher auf dem Masterimage importiert. Hier ein Auszug aus der Befehlsausgabe:

C:\Program Files\Citrix\personal vDisk\bin> .\CtxPvDsvc.exe importinventory \\share location\ExportedInventory\ExportedPvdInventory Importing inventory \\share location\ExportedInventory\ExportedPvdInventory  
Nach dem Exportvorgang sollte die Netzwerkfreigabe die folgenden Dateinamen enthalten. Nach dem Importvorgang sollte der Bestandsspeicher auf dem Masterimage die gleichen Dateinamen enthalten.

- Components.DAT
- files\_rules
- folders\_rules
- regkey\_rules
- RINGTHREE.DAT
- S-1-5-18.DAT
- SAM.DAT
- SECURITY.DAT
- SNAPSHOT.DAT



- SOFTWARE.DAT
- SYSTEM.CurrentControlSet.DAT
- VDCATALOG.DAT
- vDiskJournalData

## Sichern und Wiederherstellen

Wichtig: Die Skripts verschieben die PVDs nicht an einen neuen Speicherort. Diesen Vorgang müssen Sie auf eine andere Weise ausführen.

Zwei PowerShell-Skripts im Ordner "Support\Tools\Scripts" auf dem Installationsmedium des Produkts ermöglichen das Sichern und Wiederherstellen persönlicher vDisks. Mit den Skripten zum Sichern und Wiederherstellen können Sie vorhandene PVDs und Benutzerzuweisungen von einem Katalog zu einem anderen migrieren. Dies kann nützlich sein, wenn Sie den PVD-Speicher ändern müssen. Das Skript zum Sichern erstellt eine XML-Datei mit Metadaten aus einem bestehenden Katalog. Die Metadaten enthalten den aktuellen Pfad der PVDs im Speicher und die Benutzerzuweisungen zu den PVDs. Das Skript zum Wiederherstellen verwendet die XML-Datei, um die PVDs einem neuen Katalog zuzuweisen und ihnen die richtigen Benutzer zuzuordnen.

- migration-backup.ps1 zeichnet die Zuordnung zwischen jedem Benutzer und dessen persönlicher vDisk in einem Maschinenkatalog auf und speichert diese Informationen in einer XML-Datei.
- migration-restore.ps1 erstellt mit der XML-Datei den Desktop des Benutzers in einem Maschinenkatalog neu.

Vor dem Sichern und Wiederherstellen beachten Sie Folgendes:

- Die Skripts verwenden die Hypervisor-API. Daher muss das PowerShell-Snap-In des Hypervisors auf dem Controller, auf dem die Skripts ausgeführt werden, installiert sein.
- Führen Sie die Skripts von einem Speicherort aus, der Zugriff auf den Controller hat, auf dem der Maschinenkatalog erstellt wurde.
- Die Skripts werden auf folgenden Hypervisor-Plattformen unterstützt: Citrix XenServer, Microsoft Hyper-V und VMware ESX.

## Sichern eines Maschinenkatalogs

Führen Sie ein Backup durch, wenn ein Maschinenkatalog geändert wird. Sie können ein Backup durchführen, während die Maschinen im Katalog aktiv sind.

Verwenden Sie migration-backup.ps1 zum Sichern beliebiger Maschinenkataloge, die persönliche vDisks enthalten. Das Skript fordert die Angabe des Namens des Maschinenkatalogs und der Verbindungsinformationen für den Hypervisor an. Anschließend durchläuft es sämtliche Benutzern zugewiesene Maschinen im Maschinenkatalog und speichert für jede Maschine die Zuordnung zwischen persönlichem vDisk-Speicher und dem jeweiligen Benutzer. Diese Informationen werden in einer XML-Datei gespeichert, die folgende Struktur hat:

- PvdMigration.hypervisor.Type unterstützt VMware ESX, Citrix XenServer und Microsoft Hyper-V.
- PvdMigration.PVD speichert Informationen dazu, wo die persönliche vDisk gespeichert und welchem Benutzer sie zugeordnet ist.
- PvdMigration.PVD.DiskId ist der eindeutige Bezeichner der vDisk auf dem Hypervisor, auf dem das Backup erstellt wurde.
- PvdMigration.PVD.DiskName ist der Name der VHD- bzw. VMDK-Datei.
- PvdMigration.PVD.SRName ist der Name des Speicheranbieters, wenn das Backup erstellt wurde.
- PvdMigration.PVD.SRID ist der eindeutige Bezeichner des Speicheranbieters auf dem Hypervisor, auf dem das Backup erstellt wurde.
- PvdMigration.PVD.UserName ist der Name des Benutzers, der der vDisk zugeordnet ist.
- PvdMigration.PVD.UserId ist die SID des Benutzers, der der vDisk zugeordnet ist.
- PvdMigration.PVD.State ist der Status der vDisk. Der Status kann "backed up" oder "processed" lauten. Er lautet "backed up" nach dem ersten Backup und ändert sich in "processed", wenn die XML-Datei für eine Wiederherstellung vom Backup verwendet wurde.

## Wiederherstellen eines Maschinenkatalogs

Beachten Sie vor dem Wiederherstellen Folgendes:

- Sie können nur einen Maschinenkatalog wiederherstellen, der dasselbe Masterimage hat wie der gesicherte Maschinenkatalog.
- Sie müssen ein neues Masterimage erstellen, indem Sie den Bestand des Masterimages aktualisieren, auf dessen Basis der gesicherte Maschinenkatalog erstellt wurde.

Stellen Sie mit migration-restore.ps1 beliebige Maschinenkataloge mit persönlichen vDisks wieder her. Das Skript erfordert folgende Eingaben:

- Während des Backups erstellte XML-Datei.
- Name des Maschinenkatalogs, der wiederhergestellt wird.
- Name des Speicherorts an dem die nicht zugewiesenen persönlichen vDisks gespeichert sind. Dies ist in der XML-Datei aufgelistet.
- Hypervisor-Verbindungsinformationen.

Das Skript migration-restore.ps1 findet alle nicht zugewiesenen Maschinen im Maschinenkatalog und weist ihnen Benutzer zu. Außerdem fügt es die persönlichen vDisks der Benutzer den Maschinen an.

## Beispielszenario 1: Wiederherstellung eines Maschinenkatalogs und seiner persönlichen vDisks mit neuen Maschinenennamen

In diesem Szenario werden ein ganzer Maschinenkatalog und die persönlichen vDisks, die den enthaltenen Maschinen angefügt sind, wiederhergestellt. Die Maschinen erhalten neue Namen. Dies kann auftreten, wenn der Hypervisor oder ein Speicherhost ausgefallen ist oder wenn Benutzer in eine neue Infrastruktur migriert werden.

1. Führen Sie migration-backup.ps1 aus, um die Zuweisung zwischen Benutzern und persönlichen vDisks in der XML-Datei aufzuzeichnen.
2. Verschieben oder sammeln Sie mit einer Backuplösung die persönlichen vDisks aus dem ursprünglichen Maschinenkatalog auf einen Datenträger.
  - VMware ESX oder Microsoft Hyper-V: Persönliche vDisks sind in dem vom Controller festgelegten Speicher in einem Ordner, der den Namen der Maschine enthält, dem die vDisk angefügt ist.
  - Citrix XenServer: Persönliche vDisks sind im Stamm des vom Controller festgelegten Speichers. Der Name jeder vDisk ist eine GUID.
3. Stellen Sie die persönlichen vDisks aus dem ursprünglichen Maschinenkatalog mit einer Speicherbackuplösung wieder her.
  - ESX oder Hyper-V: Platzieren Sie die vDisks in einen neuen Ordner auf der neuen Speicherressource. Alternativ können Sie die vDisks im ursprünglichen Pfad auf der neuen Speicherressource lassen.
  - XenServer: Platzieren Sie die vDisks im Stamm der neuen Speicherressource.
4. Erstellen Sie einen Provisioning Services-vDisk- oder Maschinenerstellungsdienste-Snapshot des Masterimages, mit dem Sie den ausgefallenen Maschinenkatalog erstellt haben.
5. Führen Sie Bestand aktualisieren über das Startmenü auf der vDisk oder dem Snapshot aus.
6. Erstellen Sie den Maschinenkatalog in Studio neu und verwenden Sie dabei eine andere Benennungskonvention als für den ausgefallenen ursprünglichen Maschinenkatalog. Dadurch wird ein Katalog neuer Maschinen mit jeweils einer neuen persönlichen vDisk erstellt, die die Sitedatenbank erkennt.
7. Stellen Sie sicher, dass der neu erstellte Maschinenkatalog der richtigen Bereitstellungsgruppe zugewiesen ist.
8. Stellen Sie sicher, dass die Bereitstellungsgruppe im Wartungsmodus ist und die enthaltenen Maschinen heruntergefahren sind.
9. Bearbeiten Sie die vom Backupskript generierte XML-Datei.
  - ESX oder Hyper-V: Wenn Sie vDisks in Schritt 3 in einem neuen Ordner auf der neuen Speicherressource wiederhergestellt haben, ersetzen Sie für jeden PVD-Abschnitt in der Datei den Ordernamen in DiskName durch den Speicherort der wiederhergestellten vDisks. Wenn Sie die vDisks im ursprünglichen Pfad im neuen Speicher wiederhergestellt haben, überspringen Sie diesen Schritt.
  - XenServer: Überspringen Sie diesen Schritt.
10. Führen Sie auf dem Controller migration-restore.ps1 aus und geben Sie den Namen der XML-Datei und den Speicherort der gesicherten vDisks an.

## Beispielszenario 2: Wiederherstellung eines Maschinenkatalogs und seiner persönlichen vDisks unter Wiederverwenden der bestehenden Maschinenennamen

In diesem Szenario werden ein ganzer Maschinenkatalog und die persönlichen vDisks, die den enthaltenen Maschinen angefügt sind, wiederhergestellt. Die bestehenden Namen der (ausgefallenen) Maschinen werden wiederverwendet. Dieses Szenario ist möglich, wenn der Hypervisor oder ein Speicherhost ausgefallen ist.

1. Führen Sie migration-backup.ps1 aus, um die Zuordnung zwischen Benutzern und persönlichen vDisks aufzuzeichnen.
2. Verschieben oder sammeln Sie mit einer Backuplösung die persönlichen vDisks aus dem ursprünglichen Maschinenkatalog auf einen Datenträger:
  - ESX oder Hyper-V: Persönliche vDisks sind in dem vom Controller festgelegten Speicher in einem Ordner, der den Namen der Maschine enthält, dem die vDisk angefügt ist.
  - XenServer: Persönliche vDisks sind im Stamm des vom Controller festgelegten Speichers. Der Name jeder vDisk ist eine GUID.
3. Stellen Sie die persönlichen vDisks aus dem ursprünglichen Maschinenkatalog mit einer Speicherbackuplösung wieder her:
  - ESX oder Hyper-V: Platzieren Sie die vDisks in einen neuen Ordner auf der neuen Speicherressource.
  - XenServer: Platzieren Sie die vDisks im Stamm der neuen Speicherressource.
4. Erstellen Sie einen Provisioning Services-vDisk- oder Maschinenerstellungsdienste-Snapshot des Masterimages, mit dem Sie den ausgefallenen Maschinenkatalog erstellt haben.
5. Führen Sie Bestand aktualisieren über das Startmenü auf der vDisk oder dem Snapshot aus.
6. Erstellen Sie den Maschinenkatalog in Studio neu und verwenden Sie dabei die gleiche Benennungskonvention wie für den ausgefallenen Maschinenkatalog. Dadurch wird ein Katalog neuer Maschinen mit jeweils einer neuen persönlichen vDisk erstellt, die die Sitedatenbank erkennt.
7. Stellen Sie sicher, dass der neu erstellte Maschinenkatalog der richtigen Bereitstellungsgruppe zugewiesen ist.
8. Stellen Sie sicher, dass die Desktopgruppe im Wartungsmodus ist und die enthaltenen Maschinen heruntergefahren sind.
9. Bearbeiten Sie die vom Backupskript generierte XML-Datei:
  - ESX oder Hyper-V: Ersetzen Sie für jeden PVD-Abschnitt in der Datei den Ordernamen in DiskName durch den Speicherort der wiederhergestellten vDisks.
  - XenServer: Überspringen Sie diesen Schritt.
10. Führen Sie das Skript migration-restore.ps1 auf dem Controller aus und verwenden Sie die geänderte XML-Datei als Eingabe. Das Skript fügt die vDisks an, ohne sie zu verschieben.
11. Überprüfen Sie, ob die Benutzerdaten erfolgreich wiederhergestellt wurden.

### **Beispielszenario 3: Wiederherstellung einer Teilmenge persönlicher vDisks in einem Maschinenkatalog**

In diesem Szenario sind nur einige der persönlichen vDisks in einem Maschinenkatalog ausgefallen und werden wiederhergestellt. Die virtuellen Maschinen im Katalog sind nicht ausgefallen.

1. Führen Sie migration-backup.ps1 aus, um die Zuweisung zwischen Benutzern und persönlichen vDisks in der XML-Datei aufzuzeichnen.
2. Die XML-Datei enthält einen PVD-Abschnitt für jeden Benutzer im Maschinenkatalog. Entfernen Sie alle Benutzer, deren persönliche vDisks nicht wiederhergestellt werden müssen, und die zugehörigen Abschnitte aus der Datei.
3. Stellen Sie die persönlichen vDisks mit einer Backuplösung aus dem ursprünglichen Maschinenkatalog wie in einem der anderen Szenarios beschrieben wieder her:
  - Zum Verwenden neuer Maschinennamen folgen Sie Beispielszenario 1.
  - Zur Erhalten bestehenden Maschinennamen folgen Sie Beispielszenario 2.
4. Stellen Sie sicher, dass der Katalog genügend nicht zugewiesene Maschinen hat. Fügen Sie, falls erforderlich, Maschinen hinzu. Sie benötigen eine neue Maschine für jeden Benutzer, dessen vDisk Sie wiederherstellen möchten.
5. Stellen Sie sicher, dass die Desktopgruppe im Wartungsmodus ist und die enthaltenen Maschinen heruntergefahren sind.
6. Führen Sie auf dem Controller migration-restore.ps1 mit der geänderten XML-Datei als Eingabe aus.
7. Überprüfen Sie, ob die Benutzerdaten erfolgreich wiederhergestellt wurden.

# Anzeigen, Meldungen und Problembehandlung

May 10, 2016

Wenn Sie in Studio eine PvD-aktivierte Maschine in einem Maschinenkatalog auswählen, können Sie auf der Registerkarte "PvD" den Überwachungsstatus während Imageupdates sowie die geschätzte Abschlusszeit und den Fortschritt verfolgen. Folgende Zustandsanzeigen sind während eines Imageupdates möglich: Ready, Preparing, Waiting, Failed und Requested.

Ein Imageupdate kann aus verschiedenen Gründen fehlschlagen, einschließlich zu wenig Speicherplatz oder weil ein Desktop die PvD nicht rechtzeitig findet. Wenn Studio angibt, dass ein Imageupdate fehlgeschlagen ist, wird ein Fehlercode mit beschreibendem Text angezeigt, um Sie bei der Problembehandlung zu unterstützen. Verwenden Sie das Überwachungstool für PvD-Imageupdates oder das Skript "personal-vdisk-poolstats.ps1" zum Überwachen des Imageupdatevorgangs und um Fehlercodes für das Problem zu erhalten.

Wenn ein Imageupdate fehlschlägt, finden Sie in den folgenden Protokolldateien weitere Informationen zur Problembehandlung:

- PvD-Dienstprotokoll: C:\ProgramData\Citrix\personal vDisk\Logs\PvDSvc.log.txt
- PvD-Aktivierungsprotokoll: P:\PVDLOGS\PvDActivation.log.txt

Der aktuelle Inhalt ist am Ende der Protokolldatei.

Fehlermeldungen: 7.6 und höher

Die folgenden Fehler gelten nur für PvD Version 7.6 und höher:

- **Ein interner Fehler ist aufgetreten. Weitere Informationen finden Sie in den Personal vDisk-Protokollen.**  
**Fehlercode% Disk (% s)**

Dieser Code gilt für alle nicht kategorisierten Fehler und hat daher keinen numerischen Wert. Alle unerwarteten Fehler während der Bestandserstellung oder dem Update von Personal vDisk sind mit diesem Fehlercode gekennzeichnet.

- Sammeln Sie die Protokolle und wenden Sie sich an den Citrix Support.
- Wenn dieser Fehler während der Aktualisierung des Katalogs auftritt, führen Sie ein Rollback des Katalogs auf die vorherige Version des Masterimages aus.

- **Die Regeldateien enthalten Syntaxfehler. Weitere Informationen finden Sie in den Protokollen.**

Fehlercode 2. Die Regeldatei enthält Syntaxfehler. Die Personal vDisk-Protokolldatei enthält den Namen der Regeldatei und die Zeile, in der der Syntaxfehler gefunden wurde. Beheben Sie den Syntaxfehler in der Regeldatei und wiederholen Sie den Vorgang.

- **Der in Personal vDisk gespeicherte Bestand, der der vorherigen Version des Masterimages entspricht, ist beschädigt oder nicht lesbar.**

Fehlercode 3. Der letzte Bestand ist in "UserData.V2.vhd" unter "\ProgramData\CitrixPvD\Settings\Inventory\VER-LAST" gespeichert. Stellen Sie den Bestand entsprechend der letzten Version des Masterimages wieder her, indem Sie den Ordner "VER-LAST" von einer funktionierenden PvD-Maschine importieren, die der vorherigen Version des Masterimages zugeordnet ist.

- **Der in Personal vDisk gespeicherte Bestand, der der vorherigen Version des Masterimages entspricht, ist eine höhere Version.**

Fehlercode 4. Die Ursache ist eine PvD-Versionsinkompatibilität zwischen dem letzten Masterimage und dem aktuellen Masterimage. Installieren Sie die aktuelle Version von Personal vDisk auf dem Masterimage und aktualisieren Sie den Katalog erneut.

- **Änderungsjournalüberlauf erkannt.**

Fehlercode 5. Ein USN-Journalüberlauf wurde durch zahlreiche Änderungen am Masterimage während der Bestandserstellung verursacht. Wenn dieses Problem nach mehreren Versuchen weiterhin auftritt, ermitteln Sie mit Process Monitor, ob die Drittanbietersoftware während der Erstellung des Bestands zahlreiche Dateien erstellt oder löscht.

- **Personal vDisk konnte keinen an das System angeschlossenen Datenträger zum Speichern von Benutzerdaten finden.**

Fehlercode 6. Überprüfen Sie zunächst, ob die PvD über die Hypervisor-Konsole mit der VM verbunden ist. Dieser Fehler tritt häufig auf, weil Software zum Vermeiden von Datenverlust den Zugriff auf die PvD verhindert. Wenn die PvD mit der VM verbunden ist, fügen Sie eine Ausnahme für den verbundenen Datenträger in der Konfiguration der Software zum Vermeiden von Datenverlust hinzu.

- **Das System wurde nach der Installation noch nicht neu gestartet. Starten Sie es neu, damit die Änderungen übernommen werden.**

Fehlercode 7. Starten Sie den Desktop neu und wiederholen Sie den Vorgang.

- **Beschädigte Installation. Installieren Sie Personal vDisk neu.**

Fehlercode 8. Installieren Sie Personal vDisk neu und versuchen Sie es noch einmal.

- **Der Personal vDisk-Bestand ist nicht auf dem aktuellen Stand. Aktualisieren Sie den Bestand auf dem Masterimage und versuchen Sie es noch einmal.**

Fehlercode 9. Der Personal vDisk-Bestand wurde vor dem Herunterfahren des Desktops nicht auf dem Masterimages aktualisiert. Starten Sie das Masterimage neu und fahren Sie den Desktop mit der Option "Persönliche vDisk aktualisieren" herunter. Erstellen Sie dann einen neuen Snapshot und aktualisieren Sie damit den Katalog.

- **Beim Start von Personal vDisk ist ein interner Fehler aufgetreten Weitere Informationen finden Sie in den Personal vDisk-Protokollen.**

Fehlercode 10. Die Ursache ist möglicherweise, dass der PvD-Treiber aufgrund eines internen Fehlers oder Beschädigung der persönlichen vDisk keine Virtualisierungssitzung starten kann. Starten Sie den Desktop über den Controller neu. Wenn das Problem weiterhin auftritt, sammeln Sie die Protokolle und wenden Sie sich an den Citrix Support.

- **Personal vDisk-Timeout bei dem Versuch, ein Speichermedium für die Personalisierungseinstellungen der Benutzer zu finden.**

Fehlercode 11. Dieser Fehler tritt auf, wenn der PvD-Treiber den PvD-Datenträger nicht innerhalb von 30 Sekunden nach dem Neustart findet. Die Ursache ist normalerweise ein nicht unterstützter SCSI-Controllertyp oder Speicherlatenz. Wenn dieses Problem bei allen Desktops im Katalog auftritt, ändern Sie den der Vorlagen-VM bzw. Master-VM zugeordneten SCSI-Controllertyp in einen Typ, der von Personal vDisk unterstützt wird. Wenn dieses Problem nur bei einigen Desktops im Katalog auftritt, wird dies möglicherweise durch Speicherlatenz-Spitzen verursacht, weil zahlreiche Desktops gleichzeitig gestartet werden. Beschränken Sie die Einstellung für die maximale Anzahl aktiver Energieaktionen für die Hostverbindung.

- **Personal vDisk wurde deaktiviert, da das System nicht sicher heruntergefahren wurde. Starten Sie die Maschine neu.**

Fehlercode 12. Möglicherweise kann ein Desktop bei aktivierter PvD den Startvorgang nicht abschließen. Starten Sie den Desktop neu. Wenn das Problem weiterhin auftritt, beobachten Sie den Startvorgang des Desktops über die Hypervisor-Konsole und prüfen Sie, ob der Desktop abstürzt. Wenn ein Desktop während des Starts abstürzt, stellen Sie die PvD aus einem Backup (wenn vorhanden) wieder her oder setzen Sie die PvD zurück.

- Der zum Bereitstellen von Personal vDisk angegebene Laufwerksbuchstabe ist nicht verfügbar.**  
 Fehlercode 13. Die Ursache ist möglicherweise, dass Personal vDisk den PvD-Datenträger nicht über das vom Administrator angegebene Laufwerk bereitstellen kann. Der PvD-Datenträger kann nicht bereitgestellt werden, wenn der Laufwerksbuchstabe bereits von anderer Hardware verwendet wird. Wählen Sie einen anderen Buchstaben als Bereitstellungspunkt für die persönliche vDisk aus.
- Fehler beim Installieren von Personal vDisk-Kernelmodustreibern.**  
 Fehlercode 14. Personal vDisk installiert Treiber während der ersten Bestandsaktualisierung nach der Installation. Einige Antivirenprodukte verhindern die Installation von Treibern außerhalb eines Installationsprogramms. Deaktivieren Sie vorübergehend den in Echtzeit ausgeführten Antivirenskan oder fügen Sie der Antivirensoftware während der ersten Bestandserstellung Ausnahmen für die PvD-Treiber hinzu.
- Es konnte kein Snapshot des Systemvolumens erstellt werden. Stellen Sie sicher, dass der Volumeschattenkopie-Dienst aktiviert ist.**  
 Fehlercode 15. Dieser Fehler kann auftreten, weil der Volumeschattenkopie-Dienst deaktiviert ist. Aktivieren Sie den Volumeschattenkopie-Dienst und versuchen Sie noch einmal, den Bestand aufzunehmen.
- Fehler beim Aktivieren des Änderungslogs. Warten Sie einige Minuten und versuchen Sie es noch einmal.**  
 Fehlercode 16. Personal vDisk verwendet das Änderungslog für das Verfolgen von Änderungen am Masterimage. Wenn PvD bei einer Bestandsaktualisierung erkennt, dass das Änderungslog deaktiviert ist, versucht PvD, es zu aktivieren. Der Fehler tritt auf, wenn die Aktivierung fehlschlägt. Warten Sie ein paar Minuten und versuchen Sie es noch einmal.
- Nicht genügend freier Speicherplatz auf dem Systemvolumen.**  
 Fehlercode 17. Für das Imageupdate ist nicht genug freier Speicherplatz auf dem Laufwerk C des Desktops vorhanden. Erweitern Sie das Systemvolumen oder entfernen Sie nicht verwendete Dateien vom Systemvolumen. Das Imageupdate sollte nach dem nächsten Neustart erneut beginnen.
- Es ist nicht genügend freier Speicherplatz im Personal vDisk-Speicher. Erweitern Sie den Personal vDisk-Speicher.**  
 Fehlercode 18. Es ist nicht genug freier Speicherplatz auf dem Laufwerk der persönlichen vDisk während des Imageupdatevorgangs vorhanden. Erweitern Sie den Personal vDisk-Speicher oder entfernen Sie nicht verwendete Dateien aus dem Personal vDisk-Speicher. Das Imageupdate sollte nach dem nächsten Neustart erneut beginnen.
- Der Personal vDisk-Speicher ist überbucht. Erweitern Sie den Personal vDisk-Speicher.**  
 Fehlercode 19. Es ist nicht genug freier Speicherplatz auf dem Laufwerk der persönlichen vDisk für die in vollem Umfang bereitgestellte "UserData.V2.vhd" vorhanden. Erweitern Sie den Personal vDisk-Speicher oder entfernen Sie nicht verwendete Dateien aus dem Personal vDisk-Speicher.
- Fehlerhafte Systemregistrierung.**  
 Fehlercode 20. Die Systemregistrierung ist beschädigt, nicht lesbar oder fehlt. Setzen Sie die persönliche vDisk zurück oder stellen Sie sie von einem früheren Backup wieder her.
- Beim Zurücksetzen von Personal vDisk ist ein interner Fehler aufgetreten. Weitere Informationen finden Sie in den Personal vDisk-Protokollen.**  
 Fehlercode 21. Dieser Code gilt für alle Fehler, die beim Zurücksetzen von Personal vDisk auftreten. Sammeln Sie die Protokolle und wenden Sie sich an den Citrix Support.
- Personal vDisk konnte nicht zurückgesetzt werden, da nicht genügend freier Speicherplatz im persönlichen vDisk-Speicher ist.**

Fehlercode 22. Es ist nicht genug freier Speicherplatz auf dem Laufwerk der persönlichen vDisk während eines Zurücksetzungsvorgangs vorhanden. Erweitern Sie den Personal vDisk-Speicher oder entfernen Sie nicht verwendete Dateien aus dem Personal vDisk-Speicher.

#### Fehlermeldungen: vor Version 7.6

Die folgenden Fehler gelten nur für PvD 7.x-Versionen vor Version 7.6:

- **Start fehlgeschlagen. Personal vDisk konnte kein Speichermedium für die Personalisierungseinstellungen der Benutzer finden.**

Die PvD-Software konnte die Personal vDisk (standardmäßig Laufwerk P:) nicht finden oder nicht als den vom Administrator bei der Erstellung des Katalogs ausgewählten Bereitstellungspunkt bereitstellen.

- Prüfen Sie, ob Sie im PvD-Dienstprotokoll den folgenden Eintrag finden: "PvD 1 status --> 18:183".
  - Wenn Sie eine ältere Version als PvD Version 5.6.12 verwenden, löst ein Upgrade auf die aktuelle Version das Problem.
  - Wenn Sie Version 5.6.12 oder höher verwenden, prüfen Sie mit dem Datenträgerverwaltungstool (diskmgmt.msc), ob das Laufwerk P: als nicht bereitgestelltes Volume vorhanden ist. Wenn es vorhanden ist, führen Sie chkdsk auf dem Volume aus, um festzustellen, ob es fehlerhaft ist. Versuchen Sie, das Volume mit chkdsk wiederherzustellen.
- **Start fehlgeschlagen. Fehler beim Start von Citrix Personal vDisk. Weitere Informationen ... Statuscode: 7, Fehlercode: 0x70**

Statuscode 7 bedeutet, dass ein Fehler beim Update der PvD aufgetreten ist. Es gibt die folgenden Fehler:

| Fehlercode | Beschreibung                                                                                                                                    |
|------------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| 0x20000001 | Fehler beim Speichern des Vergleichspakets, wahrscheinlich aufgrund von Speicherplatzmangel auf der virtuellen Festplatte.                      |
| 0x20000004 | Unzureichende Privilegien zum Aktualisieren der PvD.                                                                                            |
| 0x20000006 | Fehler beim Laden der Struktur aus dem PvD-Image oder dem PvD-Bestand, wahrscheinlich aufgrund eines fehlerhaften PvD-Images oder PvD-Bestands. |
| 0x20000007 | Fehler beim Laden des Dateisystembestands, wahrscheinlich aufgrund eines fehlerhaften PvD-Images oder PvD-Bestands.                             |
| 0x20000009 | Fehler beim Öffnen der Datei mit dem Dateisystembestand, wahrscheinlich aufgrund eines fehlerhaften PvD-Images oder PvD-Bestands.               |
| 0x2000000B | Fehler beim Speichern des Vergleichspakets, wahrscheinlich aufgrund von Speicherplatzmangel auf der virtuellen Festplatte.                      |
| 0x20000010 | Fehler beim Laden des Vergleichspakets.                                                                                                         |
| 0x20000011 | Regeldateien fehlen.                                                                                                                            |
| 0x20000021 | Fehlerhafter PvD-Bestand.                                                                                                                       |
| 0x20000027 | Der Katalog "MojoControl.dat" ist fehlerhaft.                                                                                                   |

| <b>0x2000002B<br/>Fehlercode</b> | <b>PvD-Bestand fehlerhaft oder fehlt.<br/>Beschreibung</b>                                                        |
|----------------------------------|-------------------------------------------------------------------------------------------------------------------|
| 0x2000002F                       | Fehler beim Registrieren des vom Benutzer installierten MOF beim Imageupdate. Lösung: Upgrade auf Version 5.6.12. |
| 0x20000032                       | Suchen Sie in PvDactivation.log.txt nach dem letzten Eintrag mit einem Win32-Fehlercode.                          |
| 0x20                             | Fehler beim Bereitstellen des Anwendungscontainers für Imageupdate. Lösung: Upgrade auf Version 5.6.12.           |
| 0x70                             | Nicht genügend Speicherplatz auf dem Datenträger.                                                                 |

- **Start fehlgeschlagen. Fehler beim Start von Citrix Personal vDisk [oder Personal vDisk hat einen internen Fehler festgestellt]. Weitere Informationen... Statuscode 20, Fehlercode 0x20000028**

Die persönliche vDisk wurde gefunden, eine PvD-Sitzung konnte jedoch nicht erstellt werden.

Sammeln Sie die Protokolle und prüfen Sie das Protokoll "SysVol-IvmSupervisor.log" auf Sitzungserstellungsfehler:

1. Suchen Sie nach dem Protokolleintrag "IvmpNativeSessionCreate: failed to create native session, status XXXXX".
2. Wenn der Status 0xc00002cf ist, können Sie das Problem lösen, indem Sie eine neue Version des Masterimages zum Katalog hinzufügen. Dieser Statuscode bedeutet, dass ein USN-Journalüberlauf aufgrund zahlreicher Änderungen nach einer Bestandsaktualisierung aufgetreten ist.
3. Starten Sie den betroffenen virtuellen Desktop neu. Wenn das Problem weiterhin besteht, wenden Sie sich an den technischen Support von Citrix.

- **Start fehlgeschlagen. Citrix Personal vDisk wurde deaktiviert, da das System nicht sicher heruntergefahren wurde. Wählen Sie "Noch einmal versuchen". Wenn das Problem weiterhin besteht, wenden Sie sich an den Systemadministrator.**

Die gepoolte VM kann den Start nicht ausführen, solange die PvD aktiviert ist. Stellen Sie zuerst fest, warum der Start nicht ausgeführt werden kann. Eine mögliche Ursache ist die Anzeige eines blauen Bildschirms aus einem der folgenden Gründe:

- Ein nicht kompatibles Antivirenprodukt ist auf dem Masterimage vorhanden, z. B. alte Versionen von Trend Micro.
- Der Benutzer hat Software installiert, die mit PvD nicht kompatibel ist. Dies ist zwar unwahrscheinlich, aber Sie können es überprüfen, indem Sie dem Katalog eine neue Maschine hinzufügen und testen, ob sie neu startet.
- Das PvD-Image ist fehlerhaft. Dieser Fehler wurde in Version 5.6.5 beobachtet.

So prüfen Sie, ob die gepoolte VM einen blauen Bildschirm anzeigt oder ob sie vorzeitig neu startet:

- Melden Sie sich bei der Maschine über die Hypervisor-Konsole an.
- Klicken Sie auf Noch einmal versuchen und warten Sie, bis die Maschine heruntergefahren ist.
- Starten Sie die Maschine über Studio.
- Beobachten Sie die Maschinenkonsole während des Starts mit der Hypervisor-Konsole.

Weitere Problembehandlungsschritte:

- Senden Sie das Speicherabbild der Maschine, die den blauen Bildschirm hat, zur Analyse an den technischen Support von Citrix.
- Prüfen Sie auch die der PvD zugeordneten Ereignisprotokolle auf Fehler:
  1. Stellen Sie "UserData.V2.vhd" (im Stamm von Laufwerk P:) mithilfe von DiskMgmt.msc bereit, indem Sie auf Aktion > Virtuelle Festplatte anfügen klicken.
  2. Starten Sie "Eventvwr.msc".
  3. Öffnen Sie das Systemereignisprotokoll (Windows\System32\winevt\logs\system.evtx) aus UserData.V2.vhd, indem

Sie auf Aktion > Gespeichertes Protokoll öffnen klicken.

- Öffnen Sie das Anwendungsereignisprotokoll (Windows\System32\winevt\logs\application.evtx) aus UserData.V2.vhd, indem Sie auf Aktion > Gespeichertes Protokoll öffnen klicken.

- Die persönliche vDisk kann nicht gestartet werden. Die persönliche vDisk konnte nicht gestartet werden, da der Bestand nicht aktualisiert worden ist. Aktualisieren Sie den Bestand auf dem Masterimage und versuchen Sie es noch einmal. Statuscode: 15, Fehlercode: 0x0**

Der Administrator hat beim Erstellen oder Aktualisieren des PvD-Katalogs den falschen Snapshot ausgewählt (d. h. das Masterimage wurde beim Erstellen des Snapshots nicht mit Persönliche vDisk aktualisieren heruntergefahren).

## Von Personal vDisk protokollierte Ereignisse

Wenn Personal vDisk nicht aktiviert ist, können Sie die folgenden Ereignisse in der Windows-Ereignisanzeige anzeigen. Wählen Sie im linken Bereich den Knoten Anwendungen, die Quelle im rechten Bereich ist Citrix Personal vDisk. Wenn Personal vDisk aktiviert ist, werden diese Ereignisse nicht angezeigt.

Die Ereignis-ID 1 bedeutet, dass es sich um eine Informationsmeldung handelt. ID 2 steht für einen Fehler. Möglicherweise werden nicht alle Ereignisse in jeder Personal vDisk-Version verwendet.

| Ereignis-ID | Beschreibung                                                                       |
|-------------|------------------------------------------------------------------------------------|
| 1           | Status von Personal vDisk: Bestandsaktualisierung gestartet.                       |
| 1           | Status von Personal vDisk: Bestandsaktualisierung abgeschlossen. GUID: %s.         |
| 1           | Status von Personal vDisk: Imageupdate gestartet.                                  |
| 1           | Status von Personal vDisk: Imageupdate abgeschlossen.                              |
| 1           | Zurücksetzen wird durchgeführt.                                                    |
| 1           | OK.                                                                                |
| 2           | Status von Personal vDisk: Bestandsaktualisierung fehlgeschlagen mit: %s.          |
| 2           | Status von Personal vDisk: Imageupdate fehlgeschlagen mit: %s.                     |
| 2           | Status von Personal vDisk: Imageupdate fehlgeschlagen, interner Fehler.            |
| 2           | Status von Personal vDisk: Bestandsaktualisierung fehlgeschlagen: interner Fehler. |
| 2           | Personal vDisk wurde unsachgemäß beendet und daher deaktiviert.                    |
| 2           | Imageupdate fehlgeschlagen. Fehlercode %d.                                         |



| Ereignis-ID | Beschreibung<br>Personal vDisk hat einen internen Fehler festgestellt. Statuscode[%d] Fehlercode[0x%X].    |
|-------------|------------------------------------------------------------------------------------------------------------|
| 2           | Zurücksetzung von Personal vDisk fehlgeschlagen.                                                           |
| 2           | Datenträger zum Speichern der angepassten Benutzereinstellungen kann nicht gefunden werden.                |
| 2           | Zum Erstellen eines Personal vDisk-Containers ist nicht genug Speicherplatz auf dem Datenträger vorhanden. |

# Benutzerprofile

May 10, 2016

Standardmäßig wird bei der Installation des Virtual Delivery Agents die Citrix Profilverwaltung ohne Benutzereingriff auf Masterimages installiert. Sie muss jedoch nicht als Profillösung verwendet werden.

Mit XenApp- und XenDesktop-Richtlinien können Sie auf die Maschinen jeder Bereitstellungsgruppe ein anderes Profilverhalten anwenden, um die Profile an unterschiedliche Benutzerbedürfnisse anzupassen. Beispiel: Eine Bereitstellungsgruppe erfordert möglicherweise verbindliche Citrix Profile, deren Vorlage an einem Netzwerkspeicherort gespeichert ist, aber eine andere Bereitstellungsgruppe erfordert möglicherweise Citrix Roamingprofile an einem anderen Speicherort mit mehreren umgeleiteten Ordnern.

- Wenn andere Administratoren in Ihrer Organisation für XenApp- und XenDesktop-Richtlinien zuständig sind, stimmen Sie gemeinsam ab, welche profilbezogenen Richtlinien für die Bereitstellungsgruppen gelten.
- Richtlinien zur Profilverwaltung können auch in der Gruppenrichtlinie sowie in der INI-Datei der Profilverwaltung und lokal auf einzelnen virtuellen Maschinen festgelegt werden. Diese verschiedenen Methoden zum Definieren des Profilverhaltens werden in der folgenden Reihenfolge gelesen:
  1. Gruppenrichtlinie (ADM- oder ADMX-Dateien)
  2. XenApp- und XenDesktop-Richtlinien im Knoten "Richtlinie"
  3. Lokale Richtlinien auf der virtuellen Maschine, zu der der Benutzer eine Verbindung herstellt
  4. INI-Datei der ProfilverwaltungBeispiel: Wenn Sie die gleiche Richtlinie sowohl in der Gruppenrichtlinie als auch im Knoten "Richtlinie" konfigurieren, wird die Richtlinieneinstellung in der Gruppenrichtlinie vom System gelesen und die XenApp- bzw. XenDesktop-Richtlinieneinstellung wird ignoriert.

Unabhängig davon, für welche Lösung Sie sich entscheiden, können Director-Administratoren auf Diagnoseinformationen zugreifen und Problembehandlung für Benutzerprofile durchführen. Weitere Informationen finden Sie in der Dokumentation für Director.

Wenn Sie das Personal vDisk-Feature verwenden, werden Citrix Benutzerprofile standardmäßig auf den persönlichen vDisks der virtuellen Desktops gespeichert. Löschen Sie die Kopie eines Profils im Benutzerspeicher nicht, wenn gleichzeitig eine Kopie auf der persönlichen vDisk verbleibt. Dies würde einen Profilverwaltungsfehler auslösen und dazu führen, dass für Anmeldungen an dem virtuellen Desktop ein temporäres Profil verwendet wird.

## Automatische Konfiguration

Der Desktoptyp wird automatisch basierend auf der VDA-Installation erkannt und entsprechende Standardwerte für die Profilverwaltung werden neben Ihrer Konfigurationsauswahl in Studio festgelegt.

Die Richtlinien, die von der Profilverwaltung angepasst werden, werden in der Tabelle unten angezeigt. Nicht-Standard-Richtlinieneinstellungen bleiben erhalten und werden nicht von diesem Feature überschrieben. Weitere Informationen zu jeder Richtlinie finden Sie in der Dokumentation zur Profilverwaltung. Die Maschinentypen, für die Profile erstellt werden, wirken sich auf die angepassten Richtlinien aus. Wichtig ist, ob Maschinen persistent oder bereitgestellt sind, und ob sie von mehreren Benutzern gemeinsam verwendet werden oder nur einem dedizierten Benutzer zugeordnet sind.

Persistente Systeme verfügen über einen lokalen Speicher, dessen Inhalt auch nach dem Abschalten des Systems bestehen bleibt. Persistente Systeme imitieren u. U. mit Speichertechnologien wie Speichernetzwerke (Storage Area Networks, SANs) einen lokalen Datenträger. Bereitgestellte Systeme werden dagegen bei Bedarf von einem Basisdatenträger und einem Identitätsdatenträger erstellt. Der lokale Speicher wird üblicherweise durch eine RAM-Disk oder Netzwerk-Disk imitiert.

Letztere wird oft über ein SAN mit einer Hochgeschwindigkeitsverbindung zur Verfügung gestellt. Für die Bereitstellung wird allgemein Provisioning Services oder die Maschinenerstellungsdienste (oder ein entsprechendes Produkt eines Drittanbieters) verwendet. Manchmal haben bereitgestellte Systeme persistenten lokalen Speicher, der möglicherweise durch persönliche vDisks zur Verfügung gestellt wird; diese werden als persistent klassifiziert.

Zusammen definieren diese beiden Faktoren die folgenden Maschinentypen:

- **Persistent und dediziert:** Beispiele sind Desktopbetriebssystemmaschinen mit festen Zuweisungen und einer persönlichen vDisk, die mit den Maschinenerstellungsdiensten erstellt wurden, Desktops mit persönlichen vDisks, die in VDI-in-a-Box erstellt wurden, physische Arbeitsstationen und Laptops.
- **Persistent und freigegeben:** Beispiele sind Serverbetriebssystemmaschinen, die mit den Maschinenerstellungsdiensten erstellt wurden.
- **Bereitgestellt und dediziert:** Beispiele sind Desktopbetriebssystemmaschinen mit statischen Zuweisungen, aber ohne persönliche vDisk, die mit Provisioning Services erstellt wurden.
- **Bereitgestellt und freigegeben:** Beispiele sind Desktopbetriebssystemmaschinen mit zufälliger Zuweisung, die mit Provisioning Services erstellt wurden, und Desktops ohne persönliche vDisks, die mit VDI-in-a-Box erstellt wurden.

Die folgenden Richtlinieneinstellungen der Profilverwaltung werden für die verschiedenen Maschinentypen empfohlen. Sie funktionieren in den meisten Fällen gut, aber Sie müssen sie ggf. an die Anforderungen Ihrer Bereitstellung anpassen.

Wichtig: Lokal zwischengespeicherte Profile nach Abmeldung löschen, Profilstreaming und Immer zwischenspeichern werden durch die automatische Konfiguration erzwungen. Passen Sie die anderen Richtlinien manuell an.

#### Persistente Maschinen

| Richtlinie                                                | Persistent und dediziert | Persistent und freigegeben |
|-----------------------------------------------------------|--------------------------|----------------------------|
| Lokal zwischengespeicherte Profile nach Abmeldung löschen | Deaktiviert              | Aktiviert                  |
| Profilstreaming                                           | Deaktiviert              | Aktiviert                  |
| Immer zwischenspeichern                                   | Aktiviert (Hinweis 1)    | Deaktiviert (Hinweis 2)    |
| Aktiv zurückschreiben                                     | Deaktiviert              | Deaktiviert (Hinweis 3)    |
| Anmeldungen lokaler Administratoren verarbeiten           | Aktiviert                | Deaktiviert (Hinweis 4)    |

#### Bereitgestellte Maschinen

| Richtlinie                                        | Bereitgestellt und dediziert | Bereitgestellt und freigegeben |
|---------------------------------------------------|------------------------------|--------------------------------|
| Lokal zwischengespeicherte Profile nach Abmeldung | Deaktiviert (Hinweis 5)      | Aktiviert                      |

| <b>löschen.<br/>Richtlinie</b>                  | <b>Bereitgestellt und<br/>dediziert</b> | <b>Bereitgestellt und<br/>freigegeben</b> |
|-------------------------------------------------|-----------------------------------------|-------------------------------------------|
| Profilstreaming                                 | Aktiviert                               | Aktiviert                                 |
| Immer zwischenspeichern                         | Deaktiviert (Hinweis 6)                 | Deaktiviert                               |
| Aktiv zurückschreiben                           | Aktiviert                               | Aktiviert                                 |
| Anmeldungen lokaler Administratoren verarbeiten | Aktiviert                               | Aktiviert (Hinweis 7)                     |

1. Da Profilstreaming für diesen Maschinentyp deaktiviert ist, wird die Einstellung Immer zwischenspeichern immer ignoriert.
2. Deaktivieren Sie Immer zwischenspeichern. Sie können sicherstellen, dass große Dateien möglichst bald nach der Anmeldung in Profile geladen werden, wenn Sie diese Richtlinie aktivieren und mit ihr ein Dateigrößenlimit definieren (in MB). Dateien, die diese Größe überschreiten, werden so schnell wie möglich lokal zwischengespeichert.
3. Deaktivieren Sie Aktiv zurückschreiben, außer wenn Sie Änderungen in Profilen für Benutzer speichern, die zwischen XenApp-Servern roamen. Aktivieren Sie in dieser Situation diese Richtlinie.
4. Deaktivieren Sie Anmeldungen lokaler Administratoren verarbeiten, außer für gehostete, freigegebene Desktops. Aktivieren Sie in dieser Situation diese Richtlinie.
5. Deaktivieren Sie Lokal zwischengespeicherte Profile nach Abmeldung löschen. Damit bleiben lokal zwischengespeicherte Profile erhalten. Da die Maschinen beim Abmelden zurückgesetzt werden, aber einzelnen Benutzern zugewiesen sind, ist die Anmeldung mit zwischengespeicherten Profile schneller.
6. Deaktivieren Sie Immer zwischenspeichern. Sie können sicherstellen, dass große Dateien möglichst bald nach der Anmeldung in Profile geladen werden, wenn Sie diese Richtlinie aktivieren und mit ihr ein Dateigrößenlimit definieren (in MB). Dateien, die diese Größe überschreiten, werden so schnell wie möglich lokal zwischengespeichert.
7. Aktivieren Sie Anmeldungen lokaler Administratoren verarbeiten, außer für Benutzer, die zwischen XenApp- und XenDesktop-Servern roamen. Deaktivieren Sie in dieser Situation diese Richtlinie.

## Ordnerumleitung

Die Ordnerumleitung ermöglicht das Speichern von Benutzerdaten auf Netzwerkfreigaben, die nicht zum Speichern von Profilen verwendet werden. Dies verringert die Profilgröße und die Ladezeit, hat aber möglicherweise Auswirkungen auf die Netzwerkbandbreite. Zur Ordnerumleitung müssen keine Citrix Benutzerprofile verwendet werden. Sie können die Benutzerprofile selbst verwalten und dennoch Ordner umleiten.

Konfigurieren Sie die Ordnerumleitung mit den Citrix Richtlinien in Studio.

- Stellen Sie sicher, dass die Netzwerkspeicherorte zum Speichern des Inhalts von umgeleiteten Ordnern verfügbar sind, und die erforderlichen Berechtigungen richtig sind. Die Speicherorteigenschaften werden überprüft.
- Umgeleitete Ordner werden im Netzwerk eingerichtet und mit Inhalten der virtuellen Desktops bei der Anmeldung aufgefüllt.

Hinweis: Konfigurieren Sie die Ordnerumleitung entweder mit den Citrix Richtlinien oder den Active Directory-Gruppenrichtlinienobjekten, jedoch nicht mit beiden. Das Konfigurieren der Ordnerumleitung mit beiden Richtlinienengines kann zu unvorhersehbarem Verhalten führen.

## Erweiterte Ordnerumleitung

In Bereitstellungen mit mehreren Betriebssystemen können Sie Teile eines Benutzerprofils für jedes Betriebssystem

freigeben. Der Rest des Profils ist nicht freigegeben und kann nur von einem Betriebssystem verwendet werden. Um sicherzustellen, dass die Benutzererfahrung für alle Betriebssysteme konsistent ist, benötigen Sie für jedes Betriebssystem eine andere Konfiguration. Dies ist die erweiterte Ordnerumleitung. Beispiel: Bei verschiedenen Versionen einer Anwendung auf zwei Betriebssystemen muss möglicherweise eine freigegebene Datei gelesen oder bearbeitet werden. Sie entscheiden daher, sie an einen einzigen Speicherort im Netzwerk umzuleiten, von dem beide Versionen auf sie zugreifen können. Oder Sie entscheiden, da die Inhalte des Startmenüordners der beiden Betriebssysteme unterschiedlich strukturiert sind, nur einen Ordner statt beide umzuleiten. Dadurch werden die Startmenüordner und die Inhalte auf jedem Betriebssystem getrennt, und die Benutzererfahrung ist konsistent.

Wenn Sie die erweiterte Ordnerumleitung in Ihrer Bereitstellung benötigen, müssen Sie die Struktur der Profildaten Ihrer Benutzer genau kennen und festlegen, welche Teile davon zwischen Betriebssystemen freigegeben werden können. Dies ist wichtig, weil eine falsch angewendete Ordnerumleitung zu unvorhersehbarem Verhalten führen kann.

So leiten Sie Ordner in erweiterten Bereitstellungen um

- Verwenden Sie eine separate Bereitstellungsgruppe für jedes Betriebssystem.
- Informieren Sie sich, wo die Benutzerdaten und -einstellungen von den virtuellen Anwendungen, einschließlich solcher auf virtuellen Desktops, gespeichert werden, und wie die Daten strukturiert sind.
- Leiten Sie die Ordner bei freigegebenen Profildaten, bei denen ein sicheres Datenroaming gewährleistet ist (da sie in jedem Betriebssystem identisch strukturiert sind), in jeder Bereitstellungsgruppe um.
- Bei nicht freigegebenen Profildaten, für die kein Roaming möglich ist, leiten Sie den Ordner nur in einer Desktopgruppe um. Dies ist in der Regel diejenige mit dem am häufigsten verwendeten Betriebssystem oder diejenige mit den relevantesten Daten. Alternativ können Sie bei nicht freigegebenen Daten, für die kein Roaming zwischen Betriebssystemen möglich ist, die Ordner beider Betriebssysteme an separate Netzwerkadressen umleiten.

**Beispiel einer erweiterten Bereitstellung:** Diese Bereitstellung umfasst Anwendungen einschließlich Versionen von Microsoft Outlook und Internet Explorer, die auf Windows 8-Desktops und Anwendungen ausgeführt werden, sowie andere Versionen von Outlook und Internet Explorer, die von Windows Server 2008 bereitgestellt werden. Sie haben hierfür bereits zwei Bereitstellungsgruppen für die beiden Betriebssysteme eingerichtet. Die Benutzer möchten auf dieselben Kontakte und Favoriten in beiden Versionen dieser beiden Anwendungen zugreifen.

Wichtig: Die folgenden Entscheidungen und Hinweise gelten für die hier beschriebenen Betriebssysteme und die beschriebene Bereitstellung. Die Ordner, die Sie in Ihrer Organisation umleiten oder freigeben, hängen von mehreren Faktoren ab, die nur für Ihre Bereitstellung relevant sind.

- Sie leiten mit Richtlinien, die auf Bereitstellungsgruppen angewendet werden, die folgenden Ordner um:

| Ordner          | Umleitung in Windows 8? | Umleitung in Windows Server 2008? |
|-----------------|-------------------------|-----------------------------------|
| Dokumente       | Ja                      | Ja                                |
| Anwendungsdaten | Nein                    | Nein                              |
| Kontakte        | Ja                      | Ja                                |
| Desktop         | Ja                      | Nein                              |
| Downloads       | Nein                    | Nein                              |
| Favoriten       | Ja                      | Ja                                |

| Ordner              | Umleitung in Windows 8? | Umleitung in Windows Server 2008? |
|---------------------|-------------------------|-----------------------------------|
| Verknüpfungen       | Ja                      | Nein                              |
| Musik               | Ja                      | Ja                                |
| Bilder              | Ja                      | Ja                                |
| Videos              | Ja                      | Ja                                |
| Suchen              | Ja                      | Nein                              |
| Gespeicherte Spiele | Nein                    | Nein                              |
| Startmenü           | Ja                      | Nein                              |

- Bei freigegebenen, umgeleiteten Ordnern:
  - Nach der Analyse der Datenstruktur der von anderen Versionen von Outlook und Internet Explorer gespeicherten Daten entscheiden Sie, dass es sicher ist, die Ordner für Kontakte und Favoriten freizugeben.
  - Sie wissen, dass die Struktur der Ordner Eigene Dokumente, Musik, Bilder und Videos auf allen Betriebssystemen gleich ist, und sie können daher für jede Bereitstellungsgruppe am gleichen Netzwerkspeicherort gespeichert werden.
- Bei nicht freigegebenen, umgeleiteten Ordnern:
  - Die Ordner Desktop, Verknüpfungen, Suchen oder Startmenü werden nicht in die Windows Server-Bereitstellungsgruppe umgeleitet, da die Daten dieser Ordner unterschiedlich in den beiden Betriebssystemen organisiert sind. Eine Freigabe ist daher nicht möglich.
  - Um ein vorhersagbares Verhalten für diese nicht freigegebenen Daten sicherzustellen, leiten Sie sie nur in der Windows 8-Bereitstellungsgruppe um. Sie entscheiden dies, da Windows 8 öfter von den Benutzern bei ihrer täglichen Arbeit verwendet wird, die vom Server bereitgestellten Anwendungen hingegen werden nur gelegentlich genutzt. Außerdem sind in diesem Fall die nicht freigegebenen Daten relevanter für eine Desktop- als für eine Anwendungsumgebung. Desktopverknüpfungen werden beispielsweise im Ordner "Desktop" gespeichert und sind nützlich, wenn sie von einer Windows 8-Maschine, aber nicht von einem Windows-Server stammen.
- Bei nicht umgeleiteten Ordnern:
  - Die Server sollen keine von Benutzern heruntergeladene Dateien ansammeln, und Sie leiten den Ordner "Downloads" daher nicht um.
  - Daten von einzelnen Anwendungen können zu Kompatibilitäts- und Leistungsproblemen führen. Daher leiten Sie den Ordner "Anwendungsdaten" nicht um.

Weitere Informationen zur Ordnerumleitung finden Sie unter <http://technet.microsoft.com/en-us/library/cc766489%28v=ws.10%29.aspx>.

## Ordnerumleitung und Ausschlüsse

In der Citrix Profilverwaltung (nicht aber in Studio) können Sie mit einer Leistungsverbesserung die Ordnerverarbeitung mit Ausschlüssen verhindern. Wenn Sie dieses Feature verwenden, schließen Sie keine umgeleiteten Ordner aus. Die Ordnerumleitung und die Ausschlussfunktion arbeiten zusammen, d. h. wenn Sie sicherstellen, dass keine umgeleiteten

Ordner ausgeschlossen sind, können sie von der Profilverwaltung wieder in die Profildnerstruktur verschoben werden, während gleichzeitig die Datenintegrität beibehalten wird, sollten sie später doch nicht umgeleitet werden. Weitere Informationen zu Ausschlüssen finden Sie unter [Aufnehmen und Ausschließen von Objekten](#).

# HDX

Nov 02, 2016

Citrix HDX enthält eine breite Palette an Technologien für eine High-Definition-Benutzererfahrung.

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Auf dem Gerät | <p>nutzt HDX die Computingfähigkeiten der Benutzergeräte und verbessert und optimiert die Benutzererfahrung. Die HDX MediaStream-Technologie liefert einen gleichmäßigen, nahtlosen Multimediainhalt in den virtuellen Desktops oder Anwendungen der Benutzer. Mit Workspace Control können Benutzer virtuelle Desktops und Anwendungen anhalten und auf einem anderen Gerät an derselben Stelle weiterarbeiten.</p>                                                                                                                                                                                                                                                                                                                                                                             |
| Im Netzwerk   | <p>enthält HDX erweiterte Optimierungs- und Beschleunigungsfunktionen und gewährleistet die beste Leistung in jedem Netzwerk, auch bei Verbindungen mit niedriger Bandbreite und bei WAN-Verbindungen mit hoher Latenz.</p> <p>HDX-Features passen sich den Änderungen in der Umgebung an und stimmen Lastausgleich und Bandbreite anhand optimaler Technologien für die jeweiligen Benutzerszenarien aufeinander ab, und zwar sowohl bei lokalem Zugriff auf die Desktops oder Anwendungen im Unternehmensnetzwerk als auch bei Remotezugriff von außerhalb des Unternehmens.</p>                                                                                                                                                                                                               |
| Im Datacenter | <p>nutzt HDX die Verarbeitungsleistung und die Skalierbarkeit von Servern für eine erweiterte Grafikleistung, unabhängig von den Funktionen des Clientgeräts. Komprimierte Multimediainformationen werden direkt im ursprünglichen Format an das Benutzergerät gesendet.</p> <p>Die in Citrix Director bereitgestellte HDX-Kanalüberwachung zeigt den Status der verbundenen HDX-Kanäle auf Benutzergeräten an.</p> <p>HDX Insight, die Integration von EdgeSight-Netzwerkinspektion und EdgeSight-Leistungsverwaltung mit Director, erfasst Daten zum ICA-Datenverkehr und bietet eine Dashboardansicht von Echtzeit- und historische Daten, wie die clientseitige und serverseitige ICA-Sitzungslatenz, die Bandbreitennutzung der ICA-Kanäle und die ICA-Roundtrip-Zeit für jede Sitzung.</p> |

So erleben Sie HDX-Funktionen mit Ihrem virtuellen Desktop:

- Sehen Sie selbst, wie HDX hochwertige Videoinhalte auf virtuellen Desktops bereitstellt: Zeigen Sie ein Video auf einer Website mit HD-Videos an, z. B. <http://www.microsoft.com/silverlight/iis-smooth-streaming/demo/>.
- Erleben Sie, wie die Flash-Umleitung die Bereitstellung von Flash-Multimediainhalten beschleunigt:
  1. Laden Sie Adobe Flash Player herunter (<http://get.adobe.com/flashplayer/>) und installieren Sie ihn auf dem virtuellen Desktop und dem Benutzergerät.
  2. Klicken Sie auf der Desktop Viewer-Symbolleiste auf Einstellungen. Klicken Sie im Dialogfeld Desktop Viewer-Einstellungen auf die Registerkarte Flash und wählen Sie Inhalt optimieren.
  3. Erleben Sie, wie die Flash-Umleitung die Bereitstellung von Flash-Multimediainhalten auf virtuellen Desktops beschleunigt und zeigen Sie auf dem Desktop ein Video von einer Website mit Flash-Videos an, z. B. YouTube. Die Flash-Umleitung wird nahtlos integriert, sodass Benutzer nicht wissen, wann die Software ausgeführt wird. Sie können überprüfen, ob die Flash-Umleitung ausgeführt wird, indem Sie nach einem Farbblock Ausschau halten, der vorübergehend vor dem Start von Flash Player angezeigt wird.



- So stellt HDX HD-Audio bereit:
  1. Konfigurieren Sie den Citrix Client für maximale Audioqualität; weitere Informationen hierzu finden Sie in der Receiver-Dokumentation.
  2. Geben Sie Musikdateien mit einem digitalen Audioplayer (z. B. iTunes) auf dem Desktop wieder.

HDX bietet standardmäßig qualitativ hochwertige Grafiken und Videos, sodass für die meisten Benutzer keine Konfiguration erforderlich ist. Die standardmäßig aktivierten Citrix Richtlinieneinstellungen liefern die beste Sofortlösung für die Mehrheit der Fälle.

- HDX wählt automatisch die beste Bereitstellungsmethode basierend auf Client, Plattform, Anwendung und Bandbreite und nimmt dann selbständig entsprechend der geänderten Bedingungen eine Einstellung vor.
- HDX optimiert die Leistung von 2D- und 3D-Grafiken und Video.
- HDX liefert den Windows-Desktopbenutzern eine Windows Aero-Oberfläche auf jedem Client.
- HDX ermöglicht das Streamen von Multimediadateien für die Benutzergeräte direkt vom Quellenanbieter im Internet oder Intranet, ohne dass der Hostserver beteiligt wird. Wenn die Anforderungen für diesen clientseitigen Inhaltsabruf nicht erfüllt sind, erfolgt die Medienbereitstellung automatisch über Windows Media-Umleitung zur Wiedergabe von Medieninhalten auf dem Benutzergerät statt auf dem Hostserver. In den meisten Fällen ist keine Änderung der Richtlinien für Windows Media erforderlich.

Gut zu wissen:

- Informationen zum Support und zu Systemanforderungen für HDX-Features finden Sie unter [Systemanforderungen für XenApp 7.6 und XenDesktop 7.6](#). Sofern nicht anders angegeben, stehen HDX-Features für unterstützte Windows-Serverbetriebssystemmaschinen, Windows-Desktopbetriebssystemmaschinen und Remote-PC-Zugriff-Desktops zur Verfügung.
- Nachfolgend wird beschrieben, wie Sie die Benutzererfahrung weiter optimieren, die Skalierbarkeit verbessern oder die Bandbreitenanforderungen reduzieren können. Weitere Informationen zum Arbeiten mit Citrix Richtlinien und Richtlinieneinstellungen finden Sie unter   
— *Citrix Richtlinien*  
zu diesem Release.
- Vorsicht beim Bearbeiten der Registrierung: Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

## Reduzieren der Bandbreite für Windows-Desktops

HDX bietet standardmäßig eine hochgradig dynamische Windows Aero- oder Windows 8-Desktopdarstellung auf virtuellen Desktops von unterstützten Windows-Benutzergeräten. Um dies zu gewährleisten, nutzt HDX den Grafikprozessor (GPU) oder den integrierten Grafikprozessor (IGP) auf den Benutzergeräten für lokale DirectX Grafikwiedergabe. Dieses Feature wird als Desktopgestaltungsumleitung bezeichnet und gewährleistet hohe Skalierbarkeit auf dem Server. Weitere Informationen finden Sie unter "What to do with all these choices" unter <http://blogs.citrix.com/2013/11/06/go-supersonic-with-xendesktop-7-x-bandwidth-supercoders/>.

Um die für Benutzersitzungen erforderliche Bandbreite zu reduzieren, sollten Sie die folgenden Richtlinieneinstellungen ändern. Beachten Sie, dass sich die Änderung der Einstellungen auf die Qualität der Benutzererfahrung auswirken kann.

- **Desktopgestaltungsumleitung:** gilt nur für Windows-Desktopbetriebssystemmaschinen, auf die über Windows-Benutzergeräte zugegriffen wird und nur für die Komposition des Windows-Desktops. Anwendungsfenster werden auf dem Server wiedergegeben, es sei denn, die Citrix Richtlinieneinstellung Zugriff auf lokale Anwendungen zulassen ist zugelassen.

- **Grafikqualität Desktopgestaltung:** verwendet hochwertige Grafiken für die Desktopgestaltung, es sei denn, Seamless-Anwendungen oder Zugriff auf lokale Anwendungen ist aktiviert. Reduzieren Sie die Grafikqualität, um die Bandbreite zu reduzieren.
- **Dynamische Fenstervorschau:** steuert die Anzeige von Seamlessfenstern in Flip, Flip 3D, Taskleistenvorschau und Peek-Fenstervorschau. Zum Reduzieren der Bandbreite deaktivieren Sie diese Richtlinieneinstellung.

## Verbessern der Bildqualität an Benutzergeräten

Die folgenden Richtlinieneinstellungen für "Visuelle Anzeige" steuern die Qualität der Bilder, die von virtuellen Desktops auf Benutzergeräte gesendet werden.

- **Bildqualität:** steuert die visuelle Qualität der Bilder auf dem Benutzergerät: Mittel, Hoch, Immer verlustfrei, Zu verlustfrei verbessern (Standardeinstellung = Mittel).
- **Frameratesollwert:** gibt die maximale Anzahl von Frames pro Sekunde an, die vom virtuellen Desktop zum Benutzergerät gesendet werden (Standardwert = 30). In vielen Fällen können Sie die Benutzererfahrung verbessern, indem Sie einen höheren Wert festlegen. Bei Geräten mit langsamen CPUs erzielen Sie durch Festlegen eines niedrigeren Werts eine bessere Benutzererfahrung.
- **Anzeigespeicherlimit:** gibt die maximale Größe des Videopuffers (in Kilobyte) für die Sitzung an (Standardwert = 65536 KB). Für Verbindungen, die eine größere Farbtiefe und eine höhere Auflösung erfordern, erhöhen Sie den Grenzwert. Sie können den maximal erforderlichen Speicher berechnen. Andere Farbtiefen als 32 Bit sind nur verfügbar, wenn die Richtlinieneinstellung Legacygrafikmodus aktiviert ist.

## Verbessern der Videokonferenzleistung

Die HDX-Webcam-Videokomprimierung verbessert die Bandbreiteneffizienz und Latenztoleranz für Webcams während Videokonferenzen in einer Sitzung. Mit dieser Technologie werden die Webcamdaten über einen dedizierten virtuellen Multimediakanal gestreamt. Dies beansprucht erheblich weniger Bandbreite im Vergleich zum isochronen HDX-Plug-n-Play und funktioniert gut über WAN-Verbindungen.

Receiver-Benutzer können das Standardverhalten überschreiben, wenn sie in Desktop Viewer unter Mikrofon & Webcam die Einstellung Mikrofon und Webcam nicht verwenden auswählen. Um zu verhindern, dass Benutzer die HDX-Webcam-Videokomprimierung ändern, deaktivieren Sie die Umleitung von USB-Geräten über die Richtlinieneinstellungen unter "ICA > USB-Geräte".

Die HDX-Videokomprimierung ist bei Receiver unter Windows standardmäßig aktiviert, muss jedoch für Receiver unter Linux konfiguriert werden. Weitere Informationen finden Sie in der Receiver-Dokumentation. HDX-Webcam-Videokomprimierung erfordert, dass die folgenden Richtlinieneinstellungen aktiviert sind (alle sind standardmäßig aktiviert).

- Clientaudioumleitung
- Clientmikrofonumleitung
- Multimediakonferenzen
- Windows Media-Umleitung (Windows Media Redirection)

Bei Webcams, die H.264-Hardwarecodierung unterstützen, verwendet HDX-Videokomprimierung die Hardwarecodierung standardmäßig. Die Hardware-Codierung beansprucht zusätzliche Bandbreite und ist nicht für Netzwerke mit geringer Bandbreite geeignet. Zum Erzwingen der Softwarekomprimierung bei Netzwerken mit geringer Bandbreite fügen Sie den folgenden DWORD-Schlüsselwert dem Registrierungsschlüssel hinzu: HKCU\Software\Citrix\HdxRealTime: DeepCompress\_ForceSWEncode=1.

## Bevorzugen der Skalierbarkeit des Servers vor der Benutzererfahrung

Bei Bereitstellungen, bei denen die Skalierbarkeit des Servers wichtiger ist als die Benutzererfahrung, können Sie das

Legacygrafiksystem verwenden, indem Sie die Richtlinieneinstellung Legacygrafikmodus hinzufügen und die einzelnen Richtlinieneinstellungen für Legacygrafiken konfigurieren. Ältere Grafiksysteme wirken sich auf die Benutzererfahrung über WAN- und mobile Verbindungen aus.

# Thinwire-Kompatibilitätsmodus

Nov 02, 2016

Dieser Abschnitt gilt für 7.6 FP3.

Durch neue Techniken bei der Bildschirmanalyse und -zwischenspeicherung gewährleistet der ThinWire-Kompatibilitätsmodus niedrige Bandbreitenauslastung und hohe Serverskalierbarkeit, ohne die Benutzererfahrung zu beeinträchtigen.

Der ThinWire-Kompatibilitätsmodus umfasst die folgenden Features:

- Intelligente Bitmapzuordnung für Anbieter von Bitmaps.
  - Bitmap-Übersetzungsanalyse ermöglicht effizientes Verschieben der Fenster und gleichmäßigen Bildlauf.
- Abwärtskompatibilität. Upgrades von Client oder Citrix Receiver oder Hardwarebeschleunigung sind nicht erforderlich.
  - Getestet mit zahlreichen älteren Thin Clients, bis zu 5 Jahren alt und älter.
- Optimierte für sehr niedrige Server-CPU-Auslastung und verbesserte Serverskalierbarkeit.
- Ein emulierter 16-Bit-Modus reduziert die Bandbreite um weitere 15-20 % bei typischen Arbeitslasten.
- Übergangserkennung für serverseitige Wiedergabe von Videoinhalt.
  - Verarbeitung mehrerer Übergangsbereiche für eine bessere Multimediaerfahrung. Beispiel: Wiedergabe mehrerer Videos oder Liveticker.
  - Selektives Scharfzeichnen für Bereiche, die einen Übergangszustand verlassen.
- Optimierte für CloudBridge-Beschleunigung. Tests ergaben eine Verringerung der Bandbreite im Verhältnis 6:1 bei Büroarbeitslasten.
- Adaptive Anzeige, die über Richtlinieneinstellungen angepasst werden kann. Weitere Informationen finden Sie unter **Bewegtbildkomprimierung** in [Einstellungen der Richtlinie "Bewegtbilder"](#).
- VDAs und Windows-Betriebssysteme bis zu Windows 10 VDA werden unterstützt.
- Der neue Modus "Zu verlustfrei verbessern" für 3D Pro erhöht die Reaktion, Interaktivität und unterbrechbares Scharfzeichnen für eine bessere Benutzererfahrung bei niedriger Bandbreite.
- Die Standardqualität statischer fotografischer Bilder ist höher als im Legacygrafikmodus.

Zur Steigerung der Clientleistung und Senkung der Bandbreitenauslastung werden für die Bildqualitätseinstellungen "Niedrig", "Mittel" (Standard) und "Hoch" Bildschirmaktualisierungen von der Übergangserkennung dynamisch ausgewertet, um zu entscheiden, ob hoch animierte Bereiche entsprechend der Richtlinie für die adaptive Anzeige mit niedrigerer Qualität gesendet werden.

Für die Bildqualität **Zu verlustfrei verbessern** verwendet der ThinWire-Kompatibilitätsmodus bei großen Bildschirmaktualisierungen das Prinzip "zunächst unscharf". Diese Einstellung ist für 3D Pro-Benutzer gedacht, die mit 3D-Modellen oder anderen grafikintensiven Anwendungen arbeiten. Wird die Aktivität fortgesetzt, wird ein Übergangsmodus angenommen und nach dessen Ende wird der betroffene Bereich scharfgezeichnet und zwischengespeichert. Bei der anfänglichen großen Änderung findet eine oberflächliche Bildanalyse am Änderungsbereich statt, um festzustellen, ob es sich um einen unscharfen Übergang oder um einen scharfen (verlustfreien) Übergang handelt, z. B. beim Drehen eines Drahtmodells. In puncto Framerate und Bandbreite ist die Codierung einfacher Bilder mit dem verlustfreien Citrix Codec effizienter, wobei kein Qualitätsverlust entsteht.

Der Schritt beim Scharfzeichnen bei der Einstellung "Zu verlustfrei verbessern" ist ebenfalls anders. Der Bereich wird nicht auf einmal scharfgezeichnet, sondern in vorgegebenen Blöcken für eine gleichmäßige und interaktive Benutzererfahrung. Das Scharfzeichnen eines großen Änderungsbereichs mitten im Übergang (z. B. beim Verschieben eines 3D-Modells, das

kurzzeitig angehalten und dann fortgesetzt wird) würde insbesondere bei Verbindungen mit geringer Bandbreite einen Stillstand verursachen. Die Größe der Blöcke beim Scharfzeichnen hängt davon ab, wie stark die Qualität reduziert wurde, um die Mindestframerate (eine Richtlinieneinstellung für die adaptive Anzeige) zu erzielen. Wurde die Qualität stark reduziert, sind die Blöcke beim Scharfzeichnen kleiner, die Mindestgröße beträgt 128 x 128 Pixel. Wurde die Qualität nicht reduziert, z. B. wenn der Client über ausreichend Rechenleistung und Bandbreite verfügt, können beim Scharfzeichnen Blöcke der Maximalgröße von 384 x 384 Pixel verwendet werden.

# HDX 3D Pro

Nov 02, 2016

Mit HDX 3D Pro können Sie Desktops und Anwendungen bereitstellen, die am besten mit einem Grafikprozessor (GPU) für die Hardware-Beschleunigung arbeiten, einschließlich professioneller auf OpenGL und DirectX basierender 3D-Grafikanwendungen. (Der Standard-VDA unterstützt die GPU-Beschleunigung nur für DirectX.)

Beispiele für professionelle 3D-Anwendungen:

- CAD-, CAM- und CAE-Anwendungen
- Geografische Informationssystemsoftware (GIS)
- Bildarchivierungskommunikationssystem (PACS) für bildgebende Diagnostik
- Anwendungen, die die aktuellen Versionen von OpenGL, DirectX, NVidia CUDA und OpenCL verwenden
- Rechenintensive Nichtgrafik-Anwendungen, die NVIDIA CUDA-GPUs (Compute Unified Device Architecture) für paralleles Computing verwenden

HDX 3D Pro bietet die beste bandbreitenunabhängige Benutzererfahrung:

- Für WAN-Verbindungen: Stellen Sie eine interaktive Benutzererfahrung über WAN-Verbindungen mit geringen Bandbreiten bis zu 1,5 MBit/s bereit.
- LAN-Verbindungen: Stellen Sie eine Benutzererfahrung wie bei einem lokalen Desktop bei LAN-Verbindungen mit Bandbreiten von 100 MBit/s bereit.

Sie können komplexe und teure Arbeitsstationen durch einfache Benutzergeräte ersetzen, da die Grafikverarbeitung in das Datacenter für eine zentralisierte Verwaltung verschoben wird.

HDX 3D Pro stellt Windows-Desktopbetriebssystemmaschinen und Windows-Serverbetriebssystemmaschinen die GPU-Beschleunigung bereit. HDX 3D Pro bietet vGPU-Beschleunigung für Windows-Desktopbetriebssystemmaschinen bei Verwendung mit Citrix XenServer und NVIDIA GRID-GPUs. Informationen zu den unterstützten XenServer-Versionen finden Sie unter [Citrix Virtual GPU Solution](#).

Mit dem HDX Monitor-Tool, das das Tool für die Systemdiagnose ersetzt, können Sie den Betrieb und die Konfiguration von HDX-Visualisierungstechnologien überprüfen und HDX-Probleme diagnostizieren und beheben. Das Tool und weitere Informationen stehen unter <https://taas.citrix.com/hdx/download> zur Verfügung.

# Flash-Umleitung

May 10, 2016

Bei der Flash-Umleitung wird die Verarbeitung der meisten Adobe Flash-Inhalte (einschließlich Animationen, Videos und Anwendungen) an über LAN- und WAN-angeschlossene Windows-Benutzergeräte übertragen, wodurch Server- und Netzwerklast verringert werden. Dies führt zu größerer Skalierbarkeit, während gleichzeitig eine High Definition-Benutzererfahrung sichergestellt wird. Das Konfigurieren der Flash-Umleitung erfordert sowohl server- als auch clientseitige Einstellungen.

Achtung: Bei der Flash-Umleitung findet eine erhebliche Anzahl von Interaktionen zwischen dem Benutzergerät und den Serverkomponenten statt. Verwenden Sie dieses Feature nur in Umgebungen, in denen eine sicherheitsbedingte Trennung zwischen dem Benutzergerät und dem Server nicht erforderlich ist. Zudem müssen Benutzergeräte so konfiguriert werden, dass sie dieses Feature nur mit vertrauenswürdigen Servern verwenden. Da für die Flash-Umleitung der Flash Player auf dem Benutzergerät installiert sein muss, aktivieren Sie diese Funktion nur, wenn der Flash Player gesichert ist.

Die zweite Generation und Legacyversionen der Flash-Umleitung sind unabhängige Lösungen und werden in separaten virtuellen Kanälen ausgeführt.

- Legacy-Flash-Umleitungsfeatures werden nur auf dem Client unterstützt. Wenn eine ältere Version oder gar keine Version des Flash Players auf dem Benutzergerät installiert ist, werden Flash-Inhalte auf dem Server wiedergegeben.
- Die zweite Generation der Flash-Umleitung wird sowohl auf Clients als auch auf Servern unterstützt. Wenn der Client die Flash-Umleitung der zweiten Generation unterstützt, werden Flash-Inhalte auf dem Client wiedergegeben. Flash-Umleitungsfeatures der zweiten Generation bieten Unterstützung für Benutzerverbindungen über das WAN, intelligentes Fallback und eine URL-Kompatibilitätsliste. Weitere Details finden Sie weiter unten.

Flash-Umleitung verwendet die Windows-Ereignisprotokollierung auf dem Server, um Flash-Ereignisse zu protokollieren. Das Ereignisprotokoll gibt an, ob Flash-Umleitung verwendet wird und ob Probleme vorliegen. Folgendes gilt für alle Ereignisse, die von der Flash-Umleitung protokolliert werden:

- Die Flash-Umleitung meldet Ereignisse an das Anwendungsprotokoll.
- Unter Windows 8 und Windows 7 wird ein spezifisches Protokoll für die Flash-Umleitung im Knoten "Anwendungs- und Dienstprotokolle" angezeigt.
- Der Quellwert ist Flash.
- Es wird keine Kategorie angegeben.

Aktuelle Updates für HDX Flash-Kompatibilität finden Sie unter [CTX136588](#).

Zum Konfigurieren der Flash-Umleitung auf dem Server verwenden Sie die folgenden Citrix Richtlinieneinstellungen: Weitere Informationen finden Sie unter [Einstellungen der Richtlinie "Flash-Umleitung"](#).

- Flash-Standardverhalten: Legt das Standardverhalten der Flash-Beschleunigung fest. Standardmäßig ist die Flash-Umleitung aktiviert. Zum Überschreiben dieses Standardverhaltens für einzelne Webseiten und Flash-Instanzen verwenden Sie die Einstellung Flash-URL-Kompatibilitätsliste.
- Flash - intelligentes Fallback: Erkennt kleinere Flash-Filme (z. B. Werbefilme) und gibt sie auf dem Server wieder statt sie zur Wiedergabe auf das Benutzergerät umzuleiten. Dies verursacht keine Unterbrechung oder Fehler beim Laden der Webseite oder der Flash-Anwendung. Standardmäßig ist "Flash - intelligentes Fallback" aktiviert. Zum Umleiten aller Flash-Inhalte für die Wiedergabe auf dem Benutzergerät deaktivieren Sie diese Richtlinieneinstellung.
- URL-Liste für serverseitigen Flash-Inhaltsabruf: Mit dieser Liste können Sie die Websites angeben, deren Flash-Inhalte zum Server heruntergeladen und dann zur Wiedergabe zum Benutzergerät gesendet werden. (Bei der standardmäßigen Flash-Umleitung werden Flash-Inhalte auf das Benutzergerät heruntergeladen und dort wiedergegeben.) Diese Einstellung ist zwingend mit der Einstellung Serverseitigen Inhaltsabruf aktivieren auf dem Benutzergerät verknüpft und ist hauptsächlich

für Intranetsites und interne Flash-Anwendungen vorgesehen. Weitere Informationen finden Sie weiter unten. Sie funktioniert auch mit den meisten Websites und kann verwendet werden, wenn das Benutzergerät keinen direkten Internetzugang hat (z. B. wenn der XenApp- oder XenDesktop-Server die Verbindung herstellt).

Hinweis: Beim serverseitigen Inhaltsabruf werden keine Flash-Anwendungen unterstützt, die Real Time Messaging Protocol (RTMP) verwenden; stattdessen wird serverseitiges Rendering verwendet, bei dem HTTP und HTTPS unterstützt werden.

- Flash-URL-Kompatibilitätsliste: Gibt an, wo Flash-Inhalte von aufgelisteten Websites wiedergegeben werden: auf dem Benutzergerät, auf dem Server oder ob die Wiedergabe blockiert wird.
- Flash-Hintergrundfarbenliste: Hiermit können Sie die Farben von Webseiten und Flash-Instanzen aufeinander abstimmen, wodurch die Darstellung der Webseite bei Flash-Umleitung verbessert wird.

Installieren Sie Citrix Receiver und Adobe Flash Player auf dem Benutzergerät. Es ist keine weitere Konfiguration auf dem Benutzergerät erforderlich.

Sie können die Standardeinstellungen mit Active Directory-Gruppenrichtlinienobjekten ändern. Importieren und fügen Sie die administrative Vorlage für HDX MediaStream Flash-Umleitung - Client (HdxFlashClient.adm) hinzu, die im folgenden Verzeichnis verfügbar ist:

- 32-Bit-Computer: %Programme%\Citrix\ICA Client\Configuration\Sprache
- 64-Bit-Computer: %Programme (x86)%\Citrix\ICA Client\Configuration\Sprache

Die Einstellungen werden unter Administrative Vorlagen > Klassische administrative Vorlage (ADM) > HDX MediaStream Flash-Umleitung - Client aufgeführt. Weitere Informationen über Gruppenrichtlinienobjekten und Vorlagen finden Sie in der Dokumentation zu Microsoft Active Directory.

## Ändern der Verwendung der Flash-Umleitung

Zusammen mit serverseitigen Einstellungen steuert die Richtlinie HDX MediaStream-Flash-Umleitung auf dem Benutzergerät, ob Adobe Flash-Inhalte für die lokale Wiedergabe auf dem Benutzergerät umgeleitet werden. Standardmäßig ist die Flash-Umleitung aktiviert und ermittelt mit der intelligenten Netzwerkerkennung, wann Flash-Inhalte auf dem Benutzergerät wiedergegeben werden.

Wenn keine Konfiguration festgelegt ist und Desktop Lock verwendet wird, wird die Flash-Umleitung auf dem Benutzergerät standardmäßig aktiviert.

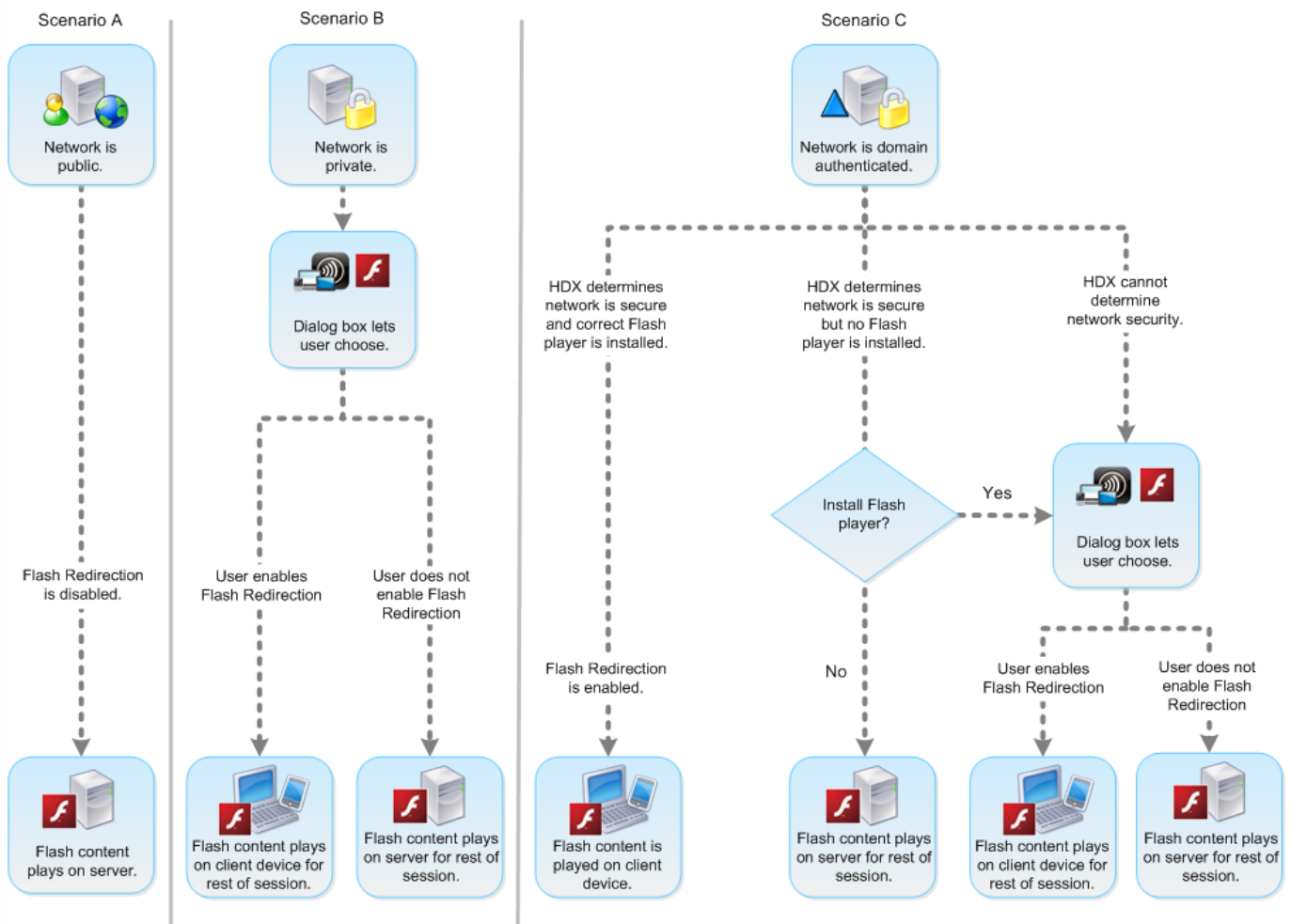
Gehen Sie zum Ändern der Verwendung von Flash-Umleitung bzw. zum Deaktivieren der Flash-Umleitung auf dem Benutzergerät wie folgt vor:

1. Wählen Sie aus der Einstellungsliste den Eintrag HDX MediaStream Flash-Umleitung auf dem Benutzergerät aktivieren und klicken Sie auf Richtlinieneinstellung.
2. Wählen Sie Nicht konfiguriert, Aktiviert (Standardeinstellung) oder Deaktiviert.
3. Wenn Sie Aktiviert eingestellt haben, wählen Sie eine Option unter HDX MediaStream Flash-Umleitung verwenden aus:
  - Die Einstellung Nur zweite Generation dient dazu, die aktuelle Flash-Umleitung zu verwenden, wenn die erforderliche Konfiguration vorhanden ist bzw. auf serverseitige Wiedergabe umzuschalten, wenn dies nicht der Fall ist.
  - Bei Wahl von Immer wird die Flash-Umleitung immer verwendet. Flash-Inhalte werden auf dem Benutzergerät wiedergegeben.
  - Bei Wahl von Nie wird die Flash-Umleitung nie verwendet. Flash-Inhalte werden auf dem Server wiedergegeben.
  - Mit Fragen (Standardeinstellung) wird festgelegt, dass die Entscheidung, ob Flash-Umleitung verwendet wird, durch intelligente Netzwerkerkennung zur Beurteilung der Sicherheitsstufe des clientseitigen Netzwerks erfolgt. Wenn die Sicherheit des Netzwerks nicht bestimmt werden kann, erhält der Benutzer die Entscheidung über die Flash-Umleitung.



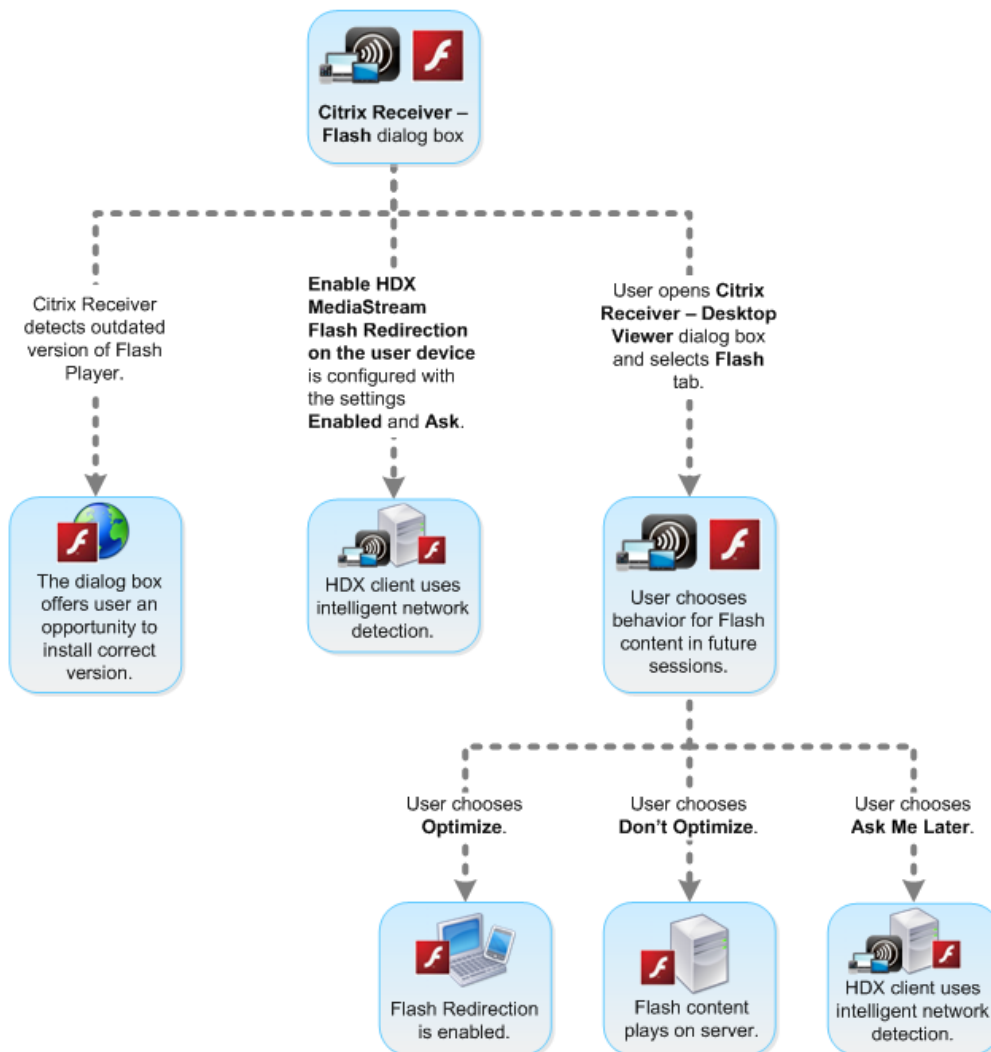
Es erscheint eine Aufforderung, die Flash-Umleitung zu aktivieren bzw. zu deaktivieren.  
Die folgende Abbildung zeigt, wie die Flash-Umleitung für verschiedene Netzwerktypen verarbeitet wird.

### Intelligent Network Detection for Flash Redirection



Benutzer können die intelligente Netzwerkerkennung über das Dialogfeld Citrix Receiver - Desktop Viewer-Einstellungen mit der Option Optimieren überschreiben oder über die Option Nicht optimieren auf der Registerkarte Flash. Welche Optionen zur Auswahl stehen, hängt davon ab, wie die Flash-Umleitung auf dem Benutzergerät konfiguriert ist (siehe folgende Abbildung).

## User control of Flash redirection



### Synchronisieren der clientseitigen HTTP-Cookies mit den serverseitigen HTTP-Cookies

Die Option zur serverseitigen HTTP-Cookie-Synchronisierung ist standardmäßig deaktiviert. Aktivieren Sie die Synchronisierung, um HTTP-Cookies vom Server herunterzuladen. Diese HTTP-Cookies werden dann für den clientseitigen Inhaltsabruf verwendet und können bei Bedarf von Sites mit Flash-Inhalten gelesen werden. Hinweis: Clientseitige Cookies werden bei der Synchronisierung nicht ersetzt; sie sind weiterhin verfügbar, wenn die Synchronisierungsrichtlinie zu einem späteren Zeitpunkt deaktiviert wird.

1. Wählen Sie aus der Einstellungsliste die Option Synchronisierung der clientseitigen HTTP-Cookies mit den serverseitigen HTTP-Cookies aktivieren und klicken Sie auf Richtlinieneinstellung.
2. Wählen Sie Nicht konfiguriert, Aktiviert oder Deaktiviert (Standardeinstellung).

#### Aktivieren Sie den serverseitigen Inhaltsabruf.

Bei der standardmäßigen Flash-Umleitung werden Flash-Inhalte auf das Benutzergerät heruntergeladen und dort wiedergegeben. Bei Aktivierung des serverseitigen Inhaltsabrufs werden die Flash-Inhalte zum Server heruntergeladen und dann an das Benutzergerät gesendet. Die Inhalte werden auf dem Benutzergerät wiedergegeben, sofern keine Richtlinie mit höherer Priorität dies verhindert, z. B. wenn eine Site durch die Richtlinieneinstellung Flash-URL-Kompatibilitätsliste blockiert wird.

Serverseitiger Inhaltsabruf wird häufig verwendet, wenn das Benutzergerät eine Verbindung zu internen Sites über

NetScaler Gateway herstellt und wenn das Benutzergerät keinen direkten Internetzugang hat.

Hinweis: Der serverseitige Inhaltsabruf unterstützt keine Flash-Anwendungen, die Real Time Messaging Protocol (RTMP) verwenden. Stattdessen wird die serverseitige Wiedergabe für solche Sites verwendet.

Die zweite Generation der Flash-Umleitung unterstützt drei Aktivierungsoptionen für den serverseitigen Inhaltsabruf. Zwei dieser Optionen umfassen das Zwischenspeichern serverseitiger Inhalte auf dem Benutzergerät, wodurch die Leistung verbessert wird, da wiederverwendeter Inhalt bereits wiedergabebereit auf dem Benutzergerät verfügbar ist. Der Inhalt dieses Caches wird separat von anderen HTTP-Inhalten auf dem Benutzergerät gespeichert.

Bei der Flash-Umleitung der zweiten Generation wird der Fallback auf serverseitigen Inhaltsabruf automatisch gestartet, wenn eine der Aktivierungsoptionen ausgewählt ist und der clientseitige Abruf von SWF-Dateien fehlschlägt.

Zum Aktivieren des serverseitigen Inhaltsabrufs müssen auf dem Clientgerät und dem Server Einstellungen festgelegt sein.

1. Wählen Sie in der Einstellungsliste den Eintrag Serverseitigen Inhaltsabruf aktivieren und klicken Sie auf Richtlinieneinstellung.
2. Wählen Sie Nicht konfiguriert, Aktiviert oder Deaktiviert (Standardeinstellung). Wenn Sie diese Einstellung aktivieren, wählen Sie eine der Optionen aus der Liste Serverseitiger Inhaltsabrufstatus:

| Option                                | Beschreibung                                                                                                                                                                                                                                                                                                                                            |
|---------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Deaktiviert                           | Deaktiviert serverseitigen Inhaltsabruf und überschreibt die Einstellung URL-Liste für serverseitigen Flash-Inhaltsabruf auf dem Server. Fallback auf serverseitigen Inhaltsabruf ist auch deaktiviert.                                                                                                                                                 |
| Aktiviert                             | Aktiviert serverseitigen Inhaltsabruf für Webseiten und Flash-Anwendungen, die in der URL-Liste für serverseitigen Flash-Inhaltsabruf angegeben sind. Fallback auf serverseitigen Inhaltsabruf ist verfügbar, aber der Inhalt wird nicht zwischengespeichert.                                                                                           |
| Aktiviert<br>(permanentes<br>Caching) | Aktiviert serverseitigen Inhaltsabruf für Webseiten und Flash-Anwendungen, die in der URL-Liste für serverseitigen Flash-Inhaltsabruf angegeben sind. Fallback auf serverseitigen Inhaltsabruf ist verfügbar. Über serverseitigen Inhaltsabruf erhaltener Inhalt wird auf dem Benutzergerät zwischengespeichert und von Sitzung zu Sitzung gespeichert. |
| Aktiviert<br>(temporäres<br>Caching)  | Aktiviert serverseitigen Inhaltsabruf für Webseiten und Flash-Anwendungen, die in der URL-Liste für serverseitigen Flash-Inhaltsabruf angegeben sind. Fallback auf serverseitigen Inhaltsabruf ist verfügbar. Über serverseitigen Inhaltsabruf erhaltener Inhalt wird auf dem Benutzergerät zwischengespeichert und am Ende der Sitzung gelöscht.       |

3. Aktivieren Sie auf dem Server die Richtlinieneinstellung URL-Liste für serverseitigen Flash-Inhaltsabruf und füllen Sie sie mit Ziel-URLs.

### Umleiten der Benutzergeräte für clientseitigen Inhaltsabruf an andere Server

Sie können einen Versuch, Flash-Inhalte abzurufen, mit der Einstellung URL-Neuschreiberegeln für clientseitigen Inhaltsabruf umleiten. Hierbei handelt es sich um ein Feature der Flash-Umleitung der zweiten Generation. Bei der Konfiguration dieses Features geben Sie zwei URL-Schemas an. Wenn das Benutzergerät versucht, Inhalte von einer Webseite abzurufen, die dem ersten Schema (dem URL-Übereinstimmungsschema) entspricht, werden sie an eine Webseite weitergeleitet, die mit dem zweiten Schema festgelegt wurde (dem Ersetzungsschema).

Mit dieser Einstellung können Sie die Auswirkungen von Inhaltsübermittlungsnetzwerken (Content Delivery Networks,

CDN) kompensieren. Manche Webseiten, die Flash-Inhalte bereitstellen, verwenden CDN-Umleitung, damit Benutzer den Inhalt vom nächstgelegenen Server in einer Gruppe von Servern, auf denen derselbe Inhalt zur Verfügung steht, abrufen können. Bei der Verwendung des clientseitigen Inhaltsabrufs der Flash-Umleitung wird der Flash-Inhalt vom Benutzergerät abgerufen, während der Rest der Webseite, auf der sich der Flash-Inhalt befindet, vom Server abgerufen wird. Wenn CDN verwendet wird, wird die Serveranfrage an den am nächsten gelegenen Server geleitet und die Benutzergeräatanfrage geht an denselben Standort. Dies möglicherweise nicht der Speicherort ist, der dem Benutzergerät am nächsten liegt; je nach Entfernung, kann eine sichtbare Verzögerung zwischen dem Laden der Webseite und die Wiedergabe von Flash-Inhalten entstehen.

1. Wählen Sie in der Einstellungsliste den Eintrag URL-Neuschreiberegeln für clientseitigen Inhaltsabruf und klicken Sie auf Richtlinieneinstellung.
2. Wählen Sie Nicht konfiguriert, Aktiviert oder Deaktiviert. Nicht konfiguriert ist der Standardwert; Deaktiviert bewirkt, dass alle URL-Neuschreiberegeln des nächsten Schrittes ignoriert werden.
3. Wenn Sie die Einstellung aktivieren, klicken Sie auf Anzeigen. Verwenden Sie die Perl-Syntax für reguläre Ausdrücke und geben Sie das URL-Schema im Feld Wert und das neu geschriebene URL-Format im Feld Wert ein.

In XenApp und XenDesktop 7.6 FP3 können Sie Registrierungseinstellungen hinzufügen, um die erforderliche Mindestversion für die Flash-Umleitung für Clientgeräte festzulegen, die auf VDAs mit Receiver für Windows oder Receiver für Linux zugreifen.

## Warnung

Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und ein erneutes Installieren des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

**ServerFlashPlayerVersionMinimum** ist ein Zeichenfolgenwert, der die erforderliche Mindestversion des Flash Players auf dem ICA-Server (VDA) angibt.

**ClientFlashPlayerVersionMinimum** ist ein Zeichenfolgenwert, der die erforderliche Mindestversion des Flash Players auf dem ICA-Client (Citrix Receiver) angibt.

Diese Versionszeichenfolgen können als "10", "10.2" oder "10.2.140" angegeben werden. Zurzeit werden nur die Hauptversions-, Nebenversions- und Buildnummern verglichen. Die Revisionsnummer wird ignoriert. Bei der Angabe von "10" als Versionszeichenfolge mit ausschließlicher Angabe der Hauptversionsnummer wird für die Nebenversions- und Buildnummer von 0 ausgegangen.

**FlashPlayerVersionComparisonMask** ist ein DWORD-Wert, der, wenn auf 0 festgelegt wird, den Vergleich der Flash Player-Version auf dem ICA-Client mit der auf dem ICA-Server deaktiviert. Die Vergleichsmaske hat auch andere Werte, die jedoch nicht verwendet werden sollten, da sich die Bedeutung jeder Maske mit einem Wert ungleich 0 ändern kann. Es wird empfohlen, die Vergleichsmaske für die gewünschten Clients nur auf 0 festzulegen. Es ist nicht empfehlenswert, die Vergleichsmaske unter den clientagnostischen Einstellungen festzulegen. Wenn keine Vergleichsmaske festgelegt ist, erfordert die Flash-Umleitung auf dem ICA-Client dieselbe Flash Player-Version wie auf dem ICA-Server oder eine höhere Version. Hierfür wird nur die Hauptversionsnummer verglichen.

Für die Umleitung müssen neben der Prüfung anhand der Vergleichsmaske die Prüfungen auf Mindestversion auf Client und

Server bestanden werden.

Der Unterschlüssel "ClientID0x51" gibt den Linux-ICA-Client an. Der Unterschlüssel "ClientID0x1" gibt den Windows-ICA-Client an. Der Name des Unterschlüssels entsteht durch Anfügen der hexadezimalen Clientprodukt-ID (ohne führende Nullen) an die Zeichenfolge "ClientID". Eine vollständige Liste der Client-IDs enthält die Dokumentation zum Mobilen SDK für Windows Apps unter <http://www.citrix.com/mobilitysdk/docs/clientdetection.html>.

### Beispiel für eine 32-Bit-VDA-Registrierungskonfiguration

[HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\HdxMediaStreamForFlash\Server\PseudoServer] clientagnostische Einstellungen

"ClientFlashPlayerVersionMinimum"="13.0" Mindestversion für ICA-Client "ServerFlashPlayerVersionMinimum"="13.0" Mindestversion für ICA-Server

[HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\HdxMediaStreamForFlash\Server\PseudoServer\ClientID0x1] Windows-ICA-Client-Einstellungen

"ClientFlashPlayerVersionMinimum"="16.0.0" Gibt die Flash-Player-Mindestversion für den Windows-Client an

[HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\HdxMediaStreamForFlash\Server\PseudoServer\ClientID0x51] Linux-ICA-Clienteeinstellungen

"FlashPlayerVersionComparisonMask"=dword:00000000 Deaktiviert den Versionsvergleich für den Linux-Client (Prüfung, um festzustellen, ob die Flash Player-Version auf dem Client neuer ist als auf dem Server)

"ClientFlashPlayerVersionMinimum"="11.2.0" Flash Player-Mindestversion für Linux-Client.

### Beispiel für eine 64-Bit-VDA-Registrierungskonfiguration

[HKEY\_LOCAL\_MACHINE\SOFTWARE\Wow6432Node\Citrix\HdxMediaStreamForFlash\Server\PseudoServer]

"ClientFlashPlayerVersionMinimum"="13.0" "ServerFlashPlayerVersionMinimum"="13.0"

[HKEY\_LOCAL\_MACHINE\SOFTWARE\Wow6432Node\Citrix\HdxMediaStreamForFlash\Server\PseudoServer\ClientID0x1]

"ClientFlashPlayerVersionMinimum"="16.0.0"

[HKEY\_LOCAL\_MACHINE\SOFTWARE\Wow6432Node\Citrix\HdxMediaStreamForFlash\Server\PseudoServer\ClientID0x51]

"FlashPlayerVersionComparisonMask"=dword:00000000 "ClientFlashPlayerVersionMinimum"="11.2.0"

# Host-zu-Client-Umleitung

Nov 02, 2016

Bei der Inhaltsumleitung können Sie steuern, wie die Benutzer auf die Informationen zugreifen: über die auf den Servern veröffentlichten Anwendungen oder über lokal auf den Benutzergeräten ausgeführte Anwendungen.

**Host-zu-Client-Umleitung** ist eine Art der Inhaltsumleitung. Sie wird nur auf Serverbetriebssystem-VDA's (nicht auf Desktopbetriebssystem-VDA's) unterstützt.

- Wenn die Host-zu-Client-Umleitung aktiviert ist, werden URLs auf dem Server-VDA abgefangen und an das Benutzergerät gesendet. Die URLs werden im Webbrowser oder Multimedia-Player auf dem Benutzergerät geöffnet.
- Kann bei aktivierter Host-zu-Client-Umleitung ein Benutzergerät keine Verbindung mit einer URL herstellen, wird die URL an den Server zurückgeleitet.
- Ist die Host-zu-Client-Umleitung deaktiviert, öffnen die Benutzer die URLs mit Webbrowsern oder Multimedia-Playern, die sich auf dem Server-VDA befinden.
- Wenn die Host-zu-Client-Umleitung aktiviert ist, können die Benutzer sie nicht deaktivieren.

Die Host-zu-Client-Umleitung wurde bisher als **Server-zu-Client-Umleitung** bezeichnet.

Der Einsatz der Host-zu-Client-Umleitung ist in bestimmten – allerdings seltenen – Fällen aus Leistungs-, Kompatibilitäts- oder Konformitätsgründen zu empfehlen. Normalerweise sind andere Formen der Inhaltsumleitung besser.

## Leistung

Sie können die Host-zu-Client-Umleitung zur Leistungssteigerung einsetzen, sodass auf den Benutzergeräten installierte Anwendungen den Vorzug vor denen auf dem VDA erhalten.

Die Host-zu-Client-Umleitung verbessert die Leistung nur unter bestimmten Bedingungen, da der VDA bereits Adobe Flash- und andere Arten von Multimediainhalten optimiert. Vor Einsatz der Host-zu-Client-Umleitung sollten Sie zunächst andere Methoden (Richtlinieneinstellungen, siehe Tabelle unten) in Betracht ziehen, da diese mehr Flexibilität und normalerweise, insbesondere für leistungsschwächere Benutzergeräte, eine bessere Benutzererfahrung bieten.

## Kompatibilität

Sie können die Host-zu-Client-Umleitung für Kompatibilitätszwecke in folgenden Fällen verwenden:

- Für Inhalte anderen Typs als HTML oder Multimedia (z. B. einen benutzerdefinierten URL-Typ).
- Für ältere Medienformate (z. B. Real Media), die vom Multimedia-Player des VDA's mit Multimediaumleitung nicht unterstützt werden.
- Die für den Inhaltstyp benötigte Anwendung wird nur von wenigen Benutzern verwendet und ist auf deren Geräten bereits installiert.
- Der VDA hat keinen Zugriff auf bestimmte Websites (beispielsweise interne Websites einer anderen Organisation).

## Konformität

Sie können die Host-zu-Client-Umleitung für Konformitätszwecke in folgenden Fällen verwenden:

- Der Lizenzvertrag für die Anwendung oder den Inhalt lässt keine Veröffentlichung über den VDA zu.

- Eine Organisationsrichtlinie verbietet das Hochladen eines Dokuments auf den VDA.

Einige Situationen kommen in komplexen Umgebungen, oder wenn Benutzergerät und VDA verschiedenen Organisationen angehören, eher vor.

Eine Umgebung kann viele verschiedene Benutzergerätetypen enthalten.

| Benutzergerät | Situation oder Umgebung                                                                  | Methode der Inhaltsumleitung                 |
|---------------|------------------------------------------------------------------------------------------|----------------------------------------------|
| Tablet        | -                                                                                        | Beliebig (siehe nächsten Tabelle)            |
| Laptop        | -                                                                                        | Beliebig (siehe nächsten Tabelle)            |
| Desktop-PC    | Benutzer verwenden viele verschiedene, auf den Benutzergeräten installierte Anwendungen. | Beliebig (siehe nächsten Tabelle)            |
| Desktop-PC    | Benutzer verwenden nur einige auf den Benutzergeräten installierte Anwendungen.          | Lokaler App-Zugriff                          |
| Desktop-PC    | Benutzer verwenden keine auf den Benutzergeräten installierten Anwendungen.              | Multimediaumleitung und/oder Flash-Umleitung |
| Desktopgerät  | Hersteller unterstützt Multimedia- und/oder Flash-Umleitung                              | Multimediaumleitung und/oder Flash-Umleitung |
| Thin Client   | Hersteller unterstützt Multimedia-, Flash- und Host-zu-Client-Umleitung.                 | Beliebig (siehe nächsten Tabelle)            |
| Zero Client   | Hersteller unterstützt Multimedia- und/oder Flash-Umleitung                              | Multimediaumleitung und/oder Flash-Umleitung |

Wählen Sie anhand der folgenden Beispiele eine geeignete Inhaltsumleitungsmethode.

| Link der URL           | Situation oder Umgebung                   | Methode der Inhaltsumleitung |
|------------------------|-------------------------------------------|------------------------------|
| Webseite oder Dokument | Der VDA hat keinen Zugriff auf die URL.   | Host-zu-Client-Umleitung     |
| Webseite               | Die Webseite enthält Adobe Flash-Inhalte. | Flash-Umleitung              |
|                        |                                           |                              |

|                             |                                                                    |                          |
|-----------------------------|--------------------------------------------------------------------|--------------------------|
| Multimediadatei oder Stream | Der VDA hat einen kompatiblen Multimedia-Player.                   | Multimediaumleitung      |
| Multimediadatei oder Stream | Der VDA hat keinen kompatiblen Multimedia-Player.                  | Host-zu-Client-Umleitung |
| Dokument                    | Der VDA hat keine Anwendung für den Dokumenttyp.                   | Host-zu-Client-Umleitung |
| Dokument                    | Das Dokument darf nicht auf Benutzergeräte heruntergeladen werden. | Keine Umleitung          |
| Dokument                    | Das Dokument darf nicht auf den VDA hochgeladen werden.            | Host-zu-Client-Umleitung |
| Benutzerdefinierter URL Typ | Der VDA hat keine Anwendung für den benutzerdefinierten URL-Typ.   | Host-zu-Client-Umleitung |

Die Host-zu-Client-Umleitung wird von Citrix Receiver für Windows, Receiver für Mac, Receiver für Linux, Receiver für HTML5 und Receiver für Chrome unterstützt.

Zur Verwendung der Host-zu-Client-Umleitung muss auf dem Benutzergerät ein Webbrowser, ein Multimedia-Player oder eine andere, für den Inhalt geeignete Anwendung vorliegen. Handelt es sich bei einem Benutzergerät um ein Desktopgerät, einen Thin Client oder einen Zero Client, vergewissern Sie sich, dass es über geeignete Anwendungen verfügt und ausreichend Leistung bietet.

Für den lokalen App-Zugriff aktivierte Benutzergeräte verwenden eine andere Inhaltsumleitungsmethode und erfordern keine Host-zu-Client-Umleitung.

Sie können Citrix Richtlinien verwenden, um eine Host-zu-Client-Inhaltsumleitung auf ungeeignete Geräte zu verhindern.

Die Host-zu-Client-Umleitung wird für folgende URLs verwendet:

- URLs, die als Hyperlink in einer Anwendung eingebettet sind (z. B. in einer E-Mail oder einem Dokument)
- URLs, die über ein Menü oder Dialogfeld einer VDA-Anwendung ausgewählt werden, wenn die Anwendung die Windows-API ShellExecuteEx verwendet
- URLs, die in das Windows-Fenster "Ausführen" eingegeben werden

Die Host-zu-Client-Umleitung wird nicht für URLs in einem Webbrowser verwendet (URLs in Webseiten oder in der Adressleiste eingegebene URLs).

## Hinweis

Wenn Benutzer ihren Standardwebbrowser auf dem VDA ändern (z. B. über "Standardprogramme festlegen"), kann dies die Host-



Wenn die Host-zu-Client-Umleitung aktiviert ist, hängt es von der Konfiguration des Benutzergeräts für den URL- und Inhaltstyp ab, welche Anwendung zum Öffnen einer URL verwendet wird. Beispiel:

- Eine HTTP-URL mit HTML-Inhalt wird im Standardbrowser geöffnet.
- Eine HTTP-URL mit PDF-Inhalt wird entweder im Standardbrowser oder einer anderen Anwendung geöffnet.

Diese Benutzergerätekonfiguration wird nicht über die Host-zu-Client-Inhaltsumleitung gesteuert. Wenn Sie die Konfiguration von Benutzergeräten nicht steuern, erwägen Sie die Verwendung der Flash-Umleitung oder der Multimedia-Umleitung anstelle der Host-zu-Client-Umleitung.

URLs des folgenden Typs werden lokal auf Benutzergeräten geöffnet, wenn die Host-zu-Client-Umleitung aktiviert ist:

- HTTP (Hypertext Transfer Protocol)
- HTTPS (Secure Hypertext Transfer Protocol)
- RTSP (Real Player und QuickTime)
- RTSPU (Real Player und QuickTime)
- PNM (Legacy Real Player)
- MMS (Microsoft Media Format)

Sie können der Liste der URL-Typen für die Host-zu-Client-Umleitung URL-Typen (einschließlich benutzerdefinierter Typen) hinzufügen und sie aus dieser löschen.

Zum Aktivieren der Host-zu-Client-Umleitung müssen Sie zunächst eine Citrix Richtlinieneinstellung aktivieren.

Die Richtlinieneinstellung für die Host-zu-Client-Umleitung ist Teil der [Einstellungen der Richtlinie "Dateiumleitung"](#). Diese Einstellung ist standardmäßig deaktiviert.

Außerdem müssen Sie möglicherweise, je nach VDA-Betriebssystem, Registrierungsschlüssel und die Gruppenrichtlinie für die Server-VDAs einrichten.

- Für Windows Server 2008 R2 SP1-VDAs müssen Sie weder Registrierungsschlüssel noch die Gruppenrichtlinie einrichten.
- Für VDAs unter Windows Server 2012, Windows Server 2012 R2 und Windows Server 2016 müssen Sie Registrierungsschlüssel und die Gruppenrichtlinie einrichten.

## Warnung

Die unsachgemäße Verwendung des Registrierungs-Editors kann zu schwerwiegenden Problemen führen, die nur durch eine Neuinstallation des Betriebssystems gelöst werden können. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

1. Kopieren Sie den Text zwischen **Reg file start** und **Reg file end** unten und fügen Sie ihn in den Editor ein.

2. Speichern Sie die Editor-Datei mit "Speichern unter" unter Auswahl des Typs "Alle Dateien" und Eingabe des Namens **ServerFTA.reg**.
3. Verteilen Sie die Datei **ServerFTA.reg** mit der Active Directory-Gruppenrichtlinie auf die Server.

- Reg file start --

Windows Registry Editor Version 5.00

[HKEY\_CLASSES\_ROOT\ServerFTAHTML\shell\open\command]

@="\"C:\\Program Files (x86)\\Citrix\\system32\\iexplore.exe\" %1"

[HKEY\_LOCAL\_MACHINE\\SOFTWARE\\Citrix\\ServerFTA]

@="ServerFTA"

[HKEY\_LOCAL\_MACHINE\\SOFTWARE\\Citrix\\ServerFTA\\Capabilities]

"ApplicationDescription"="Server FTA URL."

"ApplicationIcon"="C:\\Program Files (x86)\\Citrix\\system32\\iexplore.exe,0"

"ApplicationName"="ServerFTA"

[HKEY\_LOCAL\_MACHINE\\SOFTWARE\\Citrix\\ServerFTA\\Capabilities\\URLAssociations]

"http"="ServerFTAHTML"

"https"="ServerFTAHTML"

[HKEY\_LOCAL\_MACHINE\\SOFTWARE\\RegisteredApplications]

"Citrix.ServerFTA"="SOFTWARE\\Citrix\\ServerFTA\\Capabilities"

-- Reg file end --

Erstellen Sie eine XML-Datei. Kopieren Sie den Text zwischen "xml file start" und "xml file end" unten, fügen Sie ihn in die XML-Datei ein und speichern Sie die Datei unter dem Namen **ServerFTAdefaultPolicy.xml**.

```
-- xml file start --
```

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<DefaultAssociations>
```

```
<Association Identifier="http" ProgId="ServerFTAHTML" ApplicationName="ServerFTA" />
```

```
<Association Identifier="https" ProgId="ServerFTAHTML" ApplicationName="ServerFTA" />
```

```
</DefaultAssociations>
```

```
-- xml file end --
```

Navigieren Sie in der aktuellen Gruppenrichtlinien-Verwaltungskonsolle zu **Computerkonfiguration > Administrative Vorlagen > Windows-Komponenten > Datei-Explorer > Konfigurationsdatei für Standardzuordnungen festlegen** und geben Sie die zuvor erstellte Datei **ServerFTAdefaultPolicy.xml** an.

Zum Ändern der URL-Typenliste für die Host-zu-Client-Umleitung legen Sie auf dem Server-VDA den folgenden Registrierungsschlüssel fest:

Schlüssel: HKLM\Software\Wow6432Node\Citrix\SFTA

Zum Entfernen von URL-Typen aus der Liste legen Sie **DisableServerFTA** und **NoRedirectClasses** fest:

Name: DisableServerFTA

Typ: REG\_DWORD

Wert: 1

Name: NoRedirectClasses

Typ: REG\_MULTI\_SZ

Daten: Geben Sie eine beliebige Kombination der Werte "http", "https", "rtsp", "rtspu", "pnm" und "mms" ein. Geben Sie mehrere Werte auf separaten Zeilen an. Beispiel:

http

https

rtsp

Zum Hinzufügen von URL-Typen zu der Liste legen Sie ExtraURLProtocols fest:

Name: ExtraURLProtocols

Typ: REG\_MULTI\_SZ

Daten: Geben Sie eine beliebige Kombination von URL-Typen an. Die URL-Typen müssen das Suffix :// enthalten. Trennen Sie mehrere Werte durch Semikola. Beispiel:

eigenertyp1://;eigenertyp2://

Zum Aktivieren der Host-zu-Client-Umleitung für spezifische Websites legen Sie folgenden Registrierungsschlüssel auf dem Server-VDA fest:

Schlüssel: HKLM\Software\Wow6432Node\Citrix\SFTA

Name: ValidSites

Typ: REG\_MULTI\_SZ

Daten: Geben Sie eine beliebige Kombination vollständig qualifizierter Domännennamen (FQDN) an. Geben Sie mehrere FQDNs auf separaten Zeilen an. Ein FQDN darf einen Platzhalter (nur an der Stelle ganz links) enthalten. Dies entspricht einer Domänenebene und somit den Vorgaben von RFC 6125. Beispiel:

www.beispiel.com

\*.beispiel.com

# GPU-Beschleunigung für Windows Desktopbetriebssysteme

May 10, 2016

Mit HDX 3D Pro können Sie grafikintensive Anwendungen als Teil gehosteter Desktops oder Anwendungen auf Desktopbetriebssystemmaschinen bereitstellen. HDX 3D Pro unterstützt physische Hostcomputer (einschließlich Desktop-, Blade- und Rack-Arbeitsstationen), XenServer-VMs mit GPU-Passthrough und XenServer-VMs mit virtuellem GPU (vGPU).

Mit XenServer GPU-Passthrough können Sie VMs mit exklusivem Zugriff auf dedizierte Hardware für die Grafikverarbeitung erstellen. Sie können mehrere GPUs auf dem Hypervisor installieren und VMs jeder dieser GPUs einzeln zuweisen.

Mit XenServer vGPU können mehrere virtuelle Maschinen die Grafikverarbeitungsleistung eines einzelnen physischen GPU direkt nutzen. Die echte gemeinsame Hardware-GPU-Nutzung bietet vollständige Windows 7- oder Windows 2008 R2 SP1-Desktops, die für Benutzer mit komplexen und anspruchsvollen Designanforderungen geeignet sind. Die gemeinsame GPU-Nutzung wird für NVIDIA GRID K1- und K2-Karten unterstützt und verwendet die gleichen NVIDIA-Grafiktreiber, die auf nicht-virtualisierten Betriebssystemen bereitgestellt werden.

HDX 3D Pro bietet die folgenden Features:

- Adaptive, auf dem H.264-Standard basierende Tiefenkomprimierung für optimale Leistung bei WAN-Verbindungen und drahtlosen Verbindungen. HDX 3D Pro verwendet die auf CPU basierende Tiefenkomprimierung als Standardkomprimierungsverfahren zur Verschlüsselung. Dadurch wird eine optimale Komprimierung gewährleistet, die sich dynamisch an die Netzwerkbedingungen anpasst.  
Der auf dem H.264-Standard basierende Tiefenkomprimierungscodec konkurriert nicht mehr mit der Grafikwiedergabe um CUDA-Kerne auf der NVIDIA-GPU. Der Tiefenkomprimierungscodec wird auf der CPU ausgeführt und bietet Bandbreiteneffizienz.

- Verlustfreie Komprimierung für besondere Anwendungsfälle. HDX 3D Pro bietet einen verlustfreien CPU-basierten Codec zur Unterstützung von Anwendungen, in denen pixelgenaue Grafiken unerlässlich sind, z. B. für die medizinische Bilderstellung. Verlustfreie Komprimierung wird nur für besondere Anwendungsfälle empfohlen, da sie wesentlich mehr Netzwerk- und Verarbeitungsressourcen benötigt.

Bei Verwendung von verlustfreier Komprimierung:

- Die Anzeige für Verlustfreiheit, ein Symbol in der Taskleiste, zeigt an, ob es sich bei der Bildschirmanzeige um einen verlustreichen oder verlustfreien Frame handelt. Dies ist hilfreich, wenn die Richtlinieneinstellung Bildqualität auf Zu verlustfrei verbessern festgelegt ist. Die Anzeige für Verlustfreiheit wird grün, wenn die gesendeten Frames verlustfrei sind.
- Über die Umschaltung für Verlustfreiheit können die Benutzer jederzeit innerhalb der Sitzung in den immer verlustfreien Modus wechseln. Zum Aktivieren oder Deaktivieren des verlustfreien Modus in einer Sitzung können Sie jederzeit mit der rechten Maustaste auf das Symbol klicken oder verwenden Sie die Tastenkombination ALT + UMSCHALT + 1.

Für verlustfreie Komprimierung: HDX 3D Pro verwendet den verlustfreien Codec für die Komprimierung unabhängig von dem durch die Richtlinie ausgewählten Codec.

Für die verlustreiche Komprimierung: HDX 3D Pro verwendet den ursprünglichen Codec, entweder den Standard oder den über die Richtlinie ausgewählten Codec.

Einstellungen für die Umschaltung für Verlustfreiheit werden nicht für zukünftige Sitzungen gespeichert. Wenn Sie für alle Verbindungen den verlustfreien Codec verwenden möchten, legen Sie für die Richtlinie Bildqualität die Einstellung Immer verlustfrei fest.

- In **7.6 FP3** können Sie die Standardtastenkombination ALT + UMSCHALT + 1 zum Aktivieren oder Deaktivieren der Option "Verlustfrei" in einer Sitzung außer Kraft setzen. Konfigurieren Sie eine neue Registrierungseinstellung unter HKLM\SOFTWARE\Citrix\HDX3D\LLIndicator.
  - Name: HKLM\_HotKey, Typ: String
  - Das Format beim Konfigurieren einer Tastenkombination ist C=0|1, A=0|1, S=0|1, W=0|1, K=val. Schlüssel müssen durch Kommas getrennt werden. Die Reihenfolge der Tasten ist egal.
  - A, C, S, W und K sind Tasten, wobei Folgendes gilt: C=STRG, A=ALT, S=UMSCHALT, W=Win und K=eine gültige Taste. Zulässige Werte für K sind a-z, 0-9 und jeder virtuelle Tastencode. Weitere Informationen zu virtuellen Tastencodes finden Sie auf MSDN unter [Virtual-Key Codes](#).
  - Beispiel:
    - Taste F10 entspricht K=0x79
    - Taste STRG + F10 entspricht C=1, K=0x79
    - ALT + A entspricht A=1, K=a oder A=1, K=A oder K=A, A=1
    - STRG + ALT + 5 entspricht C=1, A=1, K=5 oder A=1, K=5, C=1
    - STRG + UMSCHALT + F5 entspricht A=1, S=1, K=0x74

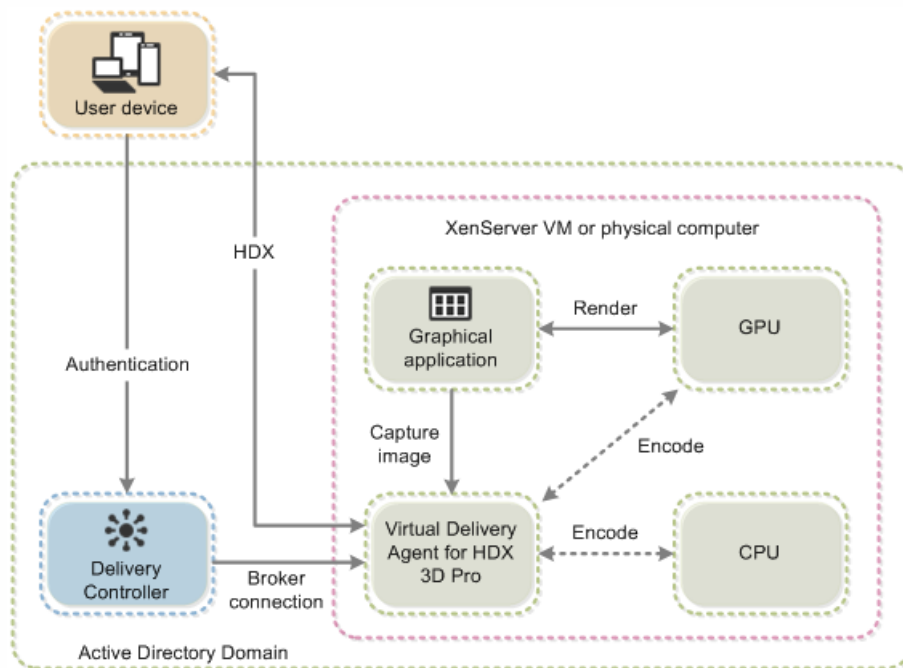
**Achtung:** Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors

zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

- Unterstützung für mehrere Monitore und hochauflösende Monitore: Für Windows 7- und Windows 8-Desktops unterstützt HDX 3D Pro Benutzergeräte mit bis zu vier Monitoren. Benutzer können ihre Monitore beliebig konfigurieren sowie Monitore mit unterschiedlichen Auflösungen und Ausrichtungen kombinieren. Die Anzahl der Monitore wird nur durch die Leistungsfähigkeit des GPU auf dem Hostcomputer, des Benutzergeräts und der verfügbaren Bandbreite begrenzt. HDX 3D Pro unterstützt alle Monitораuflösungen. Einschränkungen bestehen nur hinsichtlich der Leistungsfähigkeit der GPU auf dem Hostcomputer. Auf Windows XP-Desktops bietet HDX 3D Pro eingeschränkte Unterstützung für Dual-Monitor-Zugriff. Weitere Informationen hierzu finden Sie unter [VDAs auf Maschinen mit Windows XP oder Windows Vista](#).
- Dynamische Auflösung: Sie können das Fenster des virtuellen Desktops oder der Anwendung auf eine beliebige Auflösung einstellen. **Hinweis:** Die einzige unterstützte Methode zum Ändern der Auflösung ist das Anpassen des VDA-Sitzungsfensters. Das Ändern der Auflösung in der VDA-Sitzung (über "Systemsteuerung" > "Darstellung und Anpassung" > "Anzeige" > "Bildschirmauflösung") wird nicht unterstützt.
- Unterstützung für die NVIDIA-Kepler-Architektur. HDX 3D Pro unterstützt die NVIDIA GRID K1- und K2-Karten für GPU-Passthrough und die gemeinsame GPU-Verwendung. Der NVIDIA GRID-vGPU ermöglicht mehreren VMs den gleichzeitigen direkten Zugriff auf einen physischen GPU und die Verwendung derselben NVIDIA-Grafiktreiber, die auf nicht-virtualisierten Betriebssystemen bereitgestellt werden.
- Unterstützung für VMware vSphere und VMware ESX mit Virtual Direct Graphics Acceleration (vDGA): Sie können HDX 3D Pro mit vDGA sowohl für Remotedesktopdienste- als auch für VDI-Arbeitslasten verwenden. Wenn Sie HDX 3D Pro mit Virtual Shared Graphics Acceleration (vSGA) verwenden, wird nur ein Monitor unterstützt. Aufgrund der API-Abfangtechnologie von vSGA können bei der Verwendung mit großen 3D-Modellen Leistungsprobleme auftreten. Weitere Informationen finden Sie unter [VMware vSphere 5.1 - Citrix Known Issues](#).

Wie in der folgenden Abbildung dargestellt:

- Der Hostcomputer muss sich in derselben Active Directory-Domäne wie der Delivery Controller befinden.
- Wenn sich ein Benutzer bei Citrix Receiver anmeldet und auf die virtuelle Anwendung oder den Desktop zugreift, authentifiziert der Controller den Benutzer und kontaktiert den VDA für HDX 3D Pro, um eine Verbindung zum Computer, der die grafische Anwendung hostet, zu vermitteln. Der VDA für HDX 3D Pro komprimiert mit der entsprechenden Hardware auf dem Host die Ansicht des gesamten Desktops oder nur der grafischen Anwendung.
- Die Desktop- oder Anwendungsansichten und die dazugehörigen Interaktionen der Benutzer werden zwischen dem Hostcomputer und dem Benutzergerät über eine direkte HDX-Verbindung zwischen Citrix Receiver und VDA für HDX 3D Pro übertragen.



Wenn Sie mit der grafischen Benutzeroberfläche des Installationsprogramms einen VDA für Windows-Desktopbetriebssysteme installieren, wählen Sie Ja auf der Seite "HDX 3D Pro". Wenn Sie die Befehlszeilenschnittstelle verwenden, schließen Sie die Option `/enable_hdx_3d_pro` in den Befehl `XenDesktop VdaSetup.exe` ein.

Für das Upgrade von HDX 3D Pro müssen Sie die separate Komponente HDX 3D für professionelle Grafiken und den VDA

deinstallieren, bevor Sie den VDA für HDX 3D Pro installieren. Ebenso müssen Sie zum Wechsel vom standardmäßigen VDA zum VDA für HDX 3D Pro vor der Installation von VDA für HDX 3D Pro den standardmäßigen VDA deinstallieren.

Die NVIDIA GRID-API bietet direkten Zugriff auf den Framepuffer des Grafikprozessors und bietet die schnellste Framerate für eine gleichmäßige und interaktive Benutzererfahrung. Wenn Sie NVIDIA-Treiber vor einem VDA mit HDX 3D Pro installieren, ist NVIDIA GRID standardmäßig aktiviert.

Zum Aktivieren von NVIDIA GRID auf einer virtuellen Maschine müssen Sie den Microsoft Basic Display Adapter im Geräte-Manager deaktivieren. Führen Sie den folgenden Befehl aus und starten Sie dann den VDA neu: `Montereyenable.exe – enable – noreset`

Wenn Sie NVIDIA-Treiber nach einem VDA mit HDX 3D Pro installieren, ist NVIDIA GRID deaktiviert. Aktivieren Sie NVIDIA GRID mit dem von NVIDIA bereitgestellten Tool `Montereyenable`.

Führen Sie den folgenden Befehl aus, um NVIDIA GRID zu deaktivieren, und starten Sie dann den VDA neu: `Montereyenable.exe – disable – noreset`

Stellen Sie bei der Verwendung von HDX 3D Pro mit mehreren Monitoren sicher, dass der Hostcomputer mit mindestens so vielen Monitoren konfiguriert ist, wie an den Geräten der Benutzer angeschlossen sind. Die an den Hostcomputer angeschlossenen Monitore können physikalische oder virtuelle Monitore sein.

Schließen Sie Monitore (physikalische oder virtuelle) nicht an Hostcomputer an, während Benutzer mit dem virtuellen Desktop oder der virtuellen Anwendung, die die grafische Anwendung bereitstellen, verbunden sind. Dies kann für die Dauer der Benutzersitzung zu Instabilität führen.

Teilen Sie den Benutzern mit, dass das Ausführen von Änderungen (von ihnen oder einer Anwendung) an der Desktopauflösung, während eine grafische Anwendungssitzung ausgeführt wird, nicht unterstützt wird. Nach dem Beenden der Anwendungssitzung können Benutzer die Auflösung des Desktop Viewer-Fensters in "Citrix Receiver - Desktop Viewer-Einstellungen" ändern.

Wenn mehrere Benutzer eine Verbindung mit beschränkter Bandbreite gemeinsam verwenden, z. B. in einer Zweigstelle, empfiehlt Citrix die Verwendung der Richtlinieneinstellung `Bandbreitenlimit` für Sitzung insgesamt, um die für die einzelnen Benutzer verfügbare Bandbreite zu beschränken. Damit wird sichergestellt, dass die verfügbare Bandbreite beim Anmelden und Abmelden der Benutzer keinen großen Schwankungen unterworfen ist. Da HDX 3D Pro automatische Anpassungen durchführt, um die gesamte Bandbreite auszuschöpfen, kann sich die stark variierende verfügbare Bandbreite während der Benutzersitzungen negativ auf die Leistung auswirken.

Wenn beispielsweise 20 Benutzer eine Verbindung mit 60 MBit/s gemeinsam verwenden, kann die Bandbreite, die den einzelnen Benutzern zur Verfügung steht, abhängig von der Anzahl der gleichzeitigen Benutzer zwischen 3 MBit/s und 60 MBit/s variieren. Um die Benutzererfahrung in diesem Szenario zu optimieren, legen Sie die Bandbreite fest, die zu Spitzenzeiten pro Benutzer erforderlich ist, und stellen Sie sicher, dass die Benutzer diesen Wert nicht überschreiten können.

Citrix empfiehlt Benutzern einer 3D-Maus, die Priorität des virtuellen Kanals für die generische USB-Umleitung auf 0 zu erhöhen. Weitere Informationen dazu, wie Sie die Priorität virtueller Kanäle ändern können, finden Sie unter [CTX128190](#).

# GPU-Beschleunigung für Windows-Serverbetriebssystem

May 10, 2016

Mit HDX 3D Pro können grafikintensive Anwendungen, die in Windows-Serverbetriebssystemsitzen ausgeführt werden, auf der GPU des Servers wiedergegeben werden. Beim Verlagern der Wiedergabe von OpenGL, DirectX, Direct3D und Windows Presentation Foundation auf den GPU des Servers wird die CPU des Servers nicht durch die Grafikwiedergabe verlangsamt. Zusätzlich kann der Server so mehr Grafiken verarbeiten, weil die Arbeitslast zwischen Prozessor und Grafikprozessor aufgeteilt wird.

Bei Verwendung von HDX 3D Pro können mehrere Benutzer eine Grafikkarte gemeinsam verwenden. Wenn HDX 3D Pro mit XenServer GPU-Passthrough verwendet wird, hostet ein Server mehrere Grafikkarten, d. h. eine pro virtuelle Maschine.

Vorsicht beim Bearbeiten der Registrierung: Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

GPU Sharing ermöglicht die GPU-Hardwarewiedergabe von OpenGL- und DirectX-Anwendungen in Remotedesktopsitzen und hat die folgenden Merkmale:

- Verwenden auf Bare-Metal- oder virtuellen Maschinen, um die Anwendungsskalierbarkeit und -leistung zu steigern.
- Mehrere gleichzeitige Sitzungen können GPU-Ressourcen gemeinsam verwenden. (Die meisten Benutzer benötigen nicht die Wiedergabeleistung eines dedizierten GPU).
- Erfordert keine besonderen Einstellungen.

Sie können mehrere GPUs auf einem Hypervisor installieren und jedem GPU eine VM zuordnen: entweder durch Installation einer Grafikkarte mit mehreren GPUs oder durch Installation mehrerer Grafikkarten mit einem oder mehreren GPUs. Eine Mischung heterogener Grafikkarten auf einem Server wird nicht empfohlen.

Virtuelle Maschinen benötigen einen direkten Passthrough-Zugriff auf einen GPU; dies ist mit Citrix XenServer oder VMware vSphere verfügbar. Wenn HDX 3D mit GPU-Passthrough verwendet wird, unterstützt jeder GPU des Servers eine virtuelle Maschine mit mehreren Benutzern.

GPU Sharing hängt nicht von einer bestimmten Grafikkarte ab.

- Wählen Sie bei Ausführung auf einem Hypervisor eine Hardwareplattform und Grafikkarten aus, die mit der Implementierung von GPU-Passthrough des Hypervisors kompatibel sind. Die Liste der Hardware, die Zertifizierungstests mit XenServer GPU-Passthrough bestanden hat, finden Sie unter [GPU-Passthroughgeräte](#).
- Bei Ausführung auf Bare-Metal sollte eine Grafikkarte vom Betriebssystem aktiviert sein. Wenn mehrere GPUs auf der Hardware installiert sind, deaktivieren Sie mit dem Device Manager alle außer einem.

Die Skalierbarkeit mit GPU Sharing hängt von folgenden Faktoren ab:

- Ausgeführte Anwendungen
- Verbrauchter Videospeicher
- Verarbeitungsleistung der Grafikkarte



Beispiel: Skalierbarkeitswerte im Bereich von 8 bis 10 Benutzern wurden für NVIDIA Q6000- und M2070Q-Karten berichtet, die Anwendungen wie ESRI ArcGIS ausführen. Diese Karten haben 6 GB Videospeicher. Neuere NVIDIA GRID-Karten haben 8 GB Videospeicher und eine wesentlich höhere Verarbeitungsleistung (mehr CUDA-Kerne). Bei NVIDIA GRID K2-Karten wurde bei bis zu 20 Benutzern pro GRID K2-Karte eine gute Leistung beobachtet. Bei anderen Anwendungen kann die Skalierbarkeit wesentlich höher liegen; sie können 32 gleichzeitige Benutzer bei einem Profi-GPU erreichen.

Einige Anwendungen handhaben fehlenden Videospeicher besser als andere. Wenn die Hardware stark überlastet wird, kann der Grafikkartentreiber instabil werden oder abstürzen. Schränken Sie die Anzahl der gleichzeitigen Benutzer ein, um diese Probleme zu vermeiden.

Sie können die GPU-Beschleunigung mit einem Tool von Drittanbietern bestätigen, z. B. GPU-Z. GPU-Z finden Sie unter <http://www.techpowerup.com/gpuz/>.

Die Wiedergabe von DirectX, Direct3D und WPF steht nur auf Servern zur Verfügung, die einen Grafikprozessor haben, der eine Anzeigetreiberschnittstelle der Version 9x, 10 oder 11 unterstützt.

- Unter Windows Server 2008 R2 sind für DirectX und Direct3D keine Sondereinstellungen erforderlich, um einen einzelnen GPU zu verwenden.
- Unter Windows Server 2012 verwenden Remotedesktopdienste-Sitzungen auf dem RD-Sitzungshostserver als Standardadapter den Microsoft Basic Render-Treiber. Um den GPU in RDS-Sitzungen unter Windows Server 2012 zu verwenden, aktivieren Sie die Einstellung Use the hardware default graphics adapter for all Remote Desktop Services sessions in der Gruppenrichtlinie "Lokale Computerrichtlinie > Computerkonfiguration > Administrative Vorlagen > Windows-Komponenten > Remotedesktopdienste > Remotedesktop-Sitzungshost > Remotesitzungsumgebung".
- Um die Wiedergabe von WPF-Anwendungen mit der GPU des Servers zu aktivieren, müssen Sie in der Registrierung des Servers, der die Windows-Serverbetriebsystemsitzungen ausführt, die folgenden Einstellungen erstellen:
  - [HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\CtxHook\ApplInit\_Dlls\Multiple Monitor Hook]  
"EnableWPFHook"=dword:00000001
  - [HKEY\_LOCAL\_MACHINE\SOFTWARE\Wow6432Node\Citrix\CtxHook\ApplInit\_Dlls\Multiple Monitor Hook]  
"EnableWPFHook"=dword:00000001

Experimentelle Unterstützung wird bereitgestellt für GPU-Beschleunigung von CUDA- und OpenCL-Anwendungen, die in einer Benutzersitzung ausgeführt werden. Diese Unterstützung ist in der Standardeinstellung deaktiviert, Sie können sie jedoch für Test- und Evaluierungszwecke aktivieren.

Aktivieren Sie die folgenden Registrierungseinstellungen, um die experimentellen CUDA-Beschleunigungsfeatures zu verwenden:

- [HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\CtxHook\ApplInit\_Dlls\Graphics Helper] "CUDA"=dword:00000001
- [HKEY\_LOCAL\_MACHINE\SOFTWARE\Wow6432Node\Citrix\CtxHook\ApplInit\_Dlls\Graphics Helper]  
"CUDA"=dword:00000001

Aktivieren Sie die folgenden Registrierungseinstellungen, um die experimentellen OpenCL-Beschleunigungsfeatures zu verwenden:

- [HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\CtxHook\ApplInit\_Dlls\Graphics Helper] "OpenCL"=dword:00000001
- [HKEY\_LOCAL\_MACHINE\SOFTWARE\Wow6432Node\Citrix\CtxHook\ApplInit\_Dlls\Graphics Helper]  
"OpenCL"=dword:00000001

# OpenGL Software Accelerator

May 10, 2016

OpenGL Software Accelerator ist ein Softwarerasterizer für OpenGL-Anwendungen wie ArcGIS, Google Earth, Nehe, Maya, Blender, Voxler, CAD und CAM. In einigen Fällen bietet OpenGL Software Accelerator auch ohne die Verwendung einer Grafikkarte eine gute Benutzererfahrung mit OpenGL-Anwendungen.

Wichtig: OpenGL Software Accelerator wird ohne Modifikation bereitgestellt und muss für alle Anwendungen getestet werden. Er funktioniert möglicherweise mit einigen Anwendungen nicht, kann aber als Lösung dienen, wenn der Windows OpenGL-Rasterizer keine angemessene Leistung bietet. Wenn OpenGL Software Accelerator mit Ihren Anwendungen funktioniert, kann es Ihnen Kosten für GPU-Hardware sparen.

OpenGL Software Accelerator ist im Ordner "Support" auf dem Installationsmedium und wird auf allen gültigen VDA-Plattformen unterstützt.

## Einsatzmöglichkeiten für OpenGL Software Accelerator

- Wenn die Leistung von OpenGL-Anwendungen in virtuellen Maschinen auf XenServer oder anderen Hypervisors unzureichend ist, verwenden Sie OpenGL Accelerator. Bei einigen Anwendungen werden mit OpenGL Accelerator durch den Einsatz von SSE4.1 und AVX bessere Ergebnisse erzielt als mit dem in Windows enthaltenen Microsoft OpenGL Rasterizer. OpenGL Accelerator unterstützt zudem Anwendungen, die OpenGL-Versionen bis 2.1 verwenden.
- Bei Anwendungen, die auf einer Arbeitsstation ausgeführt werden, sollte zunächst die Standardversion von OpenGL verwendet werden, die von der Arbeitsstation des Grafikadapters bereitgestellt wird. Wenn es sich um eine aktuelle Grafikkarte handelt, wird damit in den meisten Fällen die beste Leistung erzielt. Bei Grafikkarten älterer Versionen oder weniger leistungsstarken Grafikkarten sollte OpenGL Software Accelerator eingesetzt werden.
- 3D OpenGL-Anwendungen, die nicht angemessen bereitgestellt wurden und CPU-basierte Softwarerasterisierung verwenden, profitieren möglicherweise von der OpenGL GPU-Hardwarebeschleunigung. Dieses Feature kann auf Bare-Metal- oder virtuellen Maschinen verwendet werden.

# Audiofeatures

May 10, 2016

Sie können die folgenden Citrix Richtlinieneinstellungen konfigurieren und einer Richtlinie hinzufügen, mit der HDX-Audiofeatures optimiert werden. Nutzungsinformationen sowie Beziehungen mit und Abhängigkeiten von anderen Richtlinieneinstellungen finden Sie unter [Einstellungen der Richtlinie "Audio"](#) und [Einstellungen der Richtlinie "Bandbreite"](#) und [Einstellungen der Richtlinie "Multistreamverbindungen"](#).

Wichtig: Die meisten Audiofeatures werden über den ICA-Stream transportiert und werden auf dieselbe Weise gesichert wie anderer ICA-Datenverkehr. UDP-Audio (User Datagram Protocol) verwendet einen separaten, ungesicherten Übertragungsmechanismus, wenn NetScaler Access Gateway nicht zum Pfad gehört. Wenn NetScaler Access Gateway für den Zugriff auf XenApp- und XenDesktop-Ressourcen konfiguriert ist, werden Audioübertragungen zwischen dem Endpunktgerät und dem NetScaler Access Gateway mit dem DTLS-Protokoll gesichert.

Im Allgemeinen erfordert eine höhere Audioqualität mehr Bandbreite und führt zu einer höheren CPU-Auslastung, da mehr Audiodaten an die Benutzergeräte gesendet werden. Mit der Audiokomprimierung können Sie die Audioqualität und die Sitzungsleistung aufeinander abstimmen; verwenden Sie Citrix Richtlinieneinstellungen, um den Komprimierungsgrad für Audiodateien zu konfigurieren.

Standardmäßig ist die Richtlinieneinstellung Audioqualität auf Hoch - High Definition-Audio eingestellt. Diese Einstellung bietet HiFi-Stereo-Audio, verbraucht aber mehr Bandbreite als die anderen Einstellungen. Verwenden Sie diese Audioqualität nicht für nicht optimierte Chat- und Videochat-Anwendungen (z. B. Softphones), da es hierbei zu Verzögerungen im Audiopfad kommen kann, was bei Echtzeitkommunikation ungeeignet ist.

Sie können auch die Erstellung separater Richtlinien für Benutzergruppen mit Einwählverbindungen und für Benutzer, die über ein LAN oder WAN eine Verbindung herstellen, in Erwägung ziehen. Benutzern ist bei DFÜ-Verbindungen, bei denen die Bandbreite im Allgemeinen begrenzt ist, die Download-Geschwindigkeit gewöhnlich wichtiger als die Audioqualität. Erstellen Sie daher für DFÜ-Verbindungen eine Richtlinie, die für Audio einen hohen Komprimierungsgrad anwendet, sowie eine weitere Richtlinie für LAN- oder WAN-Verbindungen, die einen geringen Komprimierungsgrad anwendet.

Informationen zu Einstellungen finden Sie unter [Einstellungen der Richtlinie "Audio"](#). Denken Sie daran, die Clientaudioeinstellungen auf dem Benutzergerät zu aktivieren (siehe [Audioeinstellungsrichtlinien für Benutzergeräte](#)).

Damit der Audioempfang von einer Anwendung auf dem Server über Lautsprecher oder andere Soundgeräte (z. B. Kopfhörer) auf dem Benutzergerät zugelassen wird, fügen Sie die Einstellung Clientaudioumleitung hinzu, die standardmäßig zugelassen ist.

Die Clientaudiozuordnung kann dazu führen, dass eine hohe Last auf dem Server und dem Netzwerk entsteht. Durch Nichtzulassen der Clientaudioumleitung werden jedoch alle HDX-Audiofunktionen deaktiviert.

Informationen zu Einstellungen finden Sie unter [Einstellungen der Richtlinie "Audio"](#). Denken Sie daran, die Clientaudioeinstellungen auf dem Benutzergerät zu aktivieren (siehe [Audioeinstellungsrichtlinien für Benutzergeräte](#)).

Damit die Audioaufzeichnung mit Eingabegeräten wie Mikrofonen auf dem Benutzergerät zugelassen wird, fügen Sie die Einstellung Clientmikrofonumleitung hinzu, die standardmäßig zugelassen ist.

Aus Sicherheitsgründen werden Benutzer darauf hingewiesen, wenn Server, die keine vertrauenswürdige Beziehung zu den Benutzergeräten haben, auf Mikrofone zugreifen und können dann den Zugriff vor Verwenden des Mikrofons akzeptieren oder ablehnen. Benutzer können die diesbezügliche Warnung in Citrix Receiver deaktivieren.

Informationen zu Einstellungen finden Sie unter [Einstellungen der Richtlinie "Audio"](#). Denken Sie daran, die Clientaudioeinstellungen auf dem Benutzergerät zu aktivieren (siehe [Audioeinstellungsrichtlinien für Benutzergeräte](#)).

Die Richtlinie Audio Plug & Play steuert, ob mehrere Audiogeräte zum Aufzeichnen und Wiedergeben zulässig sind. Diese Einstellung ist standardmäßig aktiviert.

Diese Einstellung gilt nur für Windows-Serverbetriebssystemmaschinen.

Informationen zu Einstellungen finden Sie unter [Einstellungen der Richtlinie "Audio"](#).

Die Richtlinieneinstellung Bandbreitenlimit für die Audioumleitung gibt die maximale Bandbreite (in Kilobits pro Sekunde) für die Wiedergabe und Aufzeichnung von Audio in einer Sitzung an. Die Einstellung Bandbreitenlimit für die Audioumleitung (Prozent) gibt die maximale Bandbreite für die Umleitung als Prozentsatz der insgesamt verfügbaren Bandbreite an. Standardmäßig ist Null (Maximum) für beide Einstellungen angegeben. Wenn beide Einstellungen konfiguriert sind, wird die Einstellung mit dem niedrigsten Bandbreitenlimit verwendet.

Informationen zu Einstellungen finden Sie unter [Einstellungen der Richtlinie "Bandbreite"](#). Denken Sie daran, die Clientaudioeinstellungen auf dem Benutzergerät zu aktivieren (siehe [Audioeinstellungsrichtlinien für Benutzergeräte](#)).

Standardmäßig ist **Audio über UDP - Real-time Transport** auf "Zugelassen" eingestellt (wenn die Auswahl während der Installation erfolgte), sodass ein UDP-Port auf dem Server für alle Verbindungen geöffnet wird, die für die Echtzeitübertragung von Audio über UDP konfiguriert wurden. Citrix empfiehlt, dass Sie UDP/RTP für Audio konfigurieren, um auch bei Netzwerküberlastung oder Paketverlust die beste Benutzererfahrung sicherzustellen.

**Wichtig:** Mit UDP übertragene Audiodaten werden nicht verschlüsselt, wenn NetScaler Access Gateway nicht zum Pfad gehört. Wenn NetScaler Access Gateway für den Zugriff auf XenApp- und XenDesktop-Ressourcen konfiguriert ist, werden Audioübertragungen zwischen dem Endpunktgerät und dem NetScaler Access Gateway mit dem DTLS-Protokoll gesichert.

Mit der Einstellung **Audio-UDP-Portbereich** geben Sie den Bereich der Portnummern an, die der Virtual Delivery Agent (VDA) zum Austausch von Audiopakdaten mit dem Benutzergerät verwendet.

Standardmäßig ist der Bereich 16500 bis 16509.

Informationen zu den Einstellungen von "Audio über UDP - Real-time Transport" finden Sie unter [Einstellungen der Richtlinie "Audio"](#). Informationen zu "Audio-UDP-Portbereich" finden Sie unter [Einstellungen der Richtlinie "Multistreamverbindungen"](#). Denken Sie daran, die **Clientaudioeinstellungen** auf dem Benutzergerät zu aktivieren (siehe [Audioeinstellungsrichtlinien für Benutzergeräte](#)).

1. Laden Sie die Gruppenrichtlinienvorlagen entsprechend den Anleitungen unter [Konfigurieren von Receiver mit der Gruppenrichtlinienobjektvorlage](#) herunter.

2. Navigieren Sie im Gruppenrichtlinien-Editor zu "Administrative Vorlagen > Citrix Components > Citrix Receiver > User Experience".
3. Wählen Sie für Clientaudioeinstellungen die Option Nicht konfiguriert, Aktiviert oder Deaktiviert.
  - **Nicht konfiguriert:** Standardmäßig ist die Audioumleitung mit hoher Audioqualität oder vorab konfigurierten benutzerdefinierten Audioeinstellungen aktiviert.
  - **Aktiviert:** Die Audioumleitung ist mit den ausgewählten Optionen aktiviert.
  - **Deaktiviert:** Die Audioumleitung ist deaktiviert.
4. Wenn Sie **Aktiviert** eingestellt haben, wählen Sie eine Tonqualität. Verwenden Sie für UDP-Audio die Standardeinstellung Mittel.
5. Aktivieren Sie nur für UDP-Audio die Einstellung **Real-Time Transport** und legen Sie den Bereich der eingehenden Ports so fest, dass der Durchgang durch die lokale Windows Firewall gewährleistet ist.
6. Um UDP-Audio mit NetScaler Access Gateway zu verwenden, wählen Sie die Option **Allow Real-Time Transport Through gateway**. NetScaler Access Gateway sollte mit DTLS konfiguriert sein. Weitere Informationen finden Sie unter [UDP Audio Through a NetScaler Gateway](#).

Wenn Sie als Administrator diese Änderungen nicht auf Endpunktgeräten vornehmen können, wie es beispielsweise bei BYOD-Geräten oder Heimcomputern der Fall sein kann, verwenden Sie die Attribute aus der Datei default.ica von StoreFront, um UDP-Audio zu aktivieren.

1. Öffnen Sie mit einem Texteditor, z. B. Editor, auf der StoreFront-Maschine die Datei C:\inetpub\wwwroot\Citrix\App\_Data\default.ica.
2. Nehmen Sie im Abschnitt [Application] die Einträge unten vor.



```
; This is to enable Real-Time Transport
```

```
EnableRtpAudio=true
```

```
; This is to Allow Real-Time Transport Through gateway
```

```
EnableUDPThroughGateway=true
```

```
; This is to set audio quality to Medium
```

```
AudioBandwidthLimit=1-
```

```
; UDP Port range
```

```
RtpAudioLowestPort=16500
```

```
RtpAudioHighestPort=16509
```

Wenn UDP-Audio durch Bearbeiten der Datei default.ica aktiviert wird, wird UDP-Audio für alle Benutzer aktiviert, die diesen Store verwenden.

Teilnehmer von Audio- oder Videokonferenzen hören eventuell ein Echo. Echos treten normalerweise auf, wenn der Abstand zwischen Lautsprechern und Mikrofonen nicht groß genug ist. Aus diesem Grund empfiehlt Citrix, dass Sie für Audio- und Videokonferenzen Kopfhörer verwenden.

HDX verfügt über eine Option zur Echounterdrückung (standardmäßig aktiviert), die das Auftreten von Echo minimiert. Die Qualität der Echounterdrückung hängt stark vom Abstand zwischen den Lautsprechern und dem Mikrofon ab. Der Abstand zwischen den Geräten darf nicht zu groß aber auch nicht zu klein sein.

Sie können eine Registrierungseinstellung ändern, um die Echounterdrückung zu deaktivieren. Vorsicht beim Bearbeiten der Registrierung: Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

1. Navigieren Sie mit dem Registrierungs-Editor auf dem Benutzergerät zu einer der folgenden Optionen:

- 32-Bit-Computer: HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\ICA Client\Engine\Configuration\Advanced\Modules\ClientAudio\EchoCancellation
- 64-Bit-Computer: HKEY\_LOCAL\_MACHINE\SOFTWARE\Wow6432Node\Citrix\ICA Client\Engine\Configuration\Advanced\Modules\ClientAudio\EchoCancellation

2. Ändern Sie den Wert im Feld Wert zu FALSE.



# Prioritäten für den Netzwerkdatenverkehr

May 10, 2016

Prioritäten für den Netzwerkdatenverkehr über mehrere Verbindungen für eine Sitzung werden zugewiesen, indem Diensteigenschaft (QoS)-fähige Router verwendet werden. Vier TCP/IP-Streams (real-time, interactive, background und bulk) und ein UDP/RTP-Stream (für Voice) sind zum Ausführen von ICA-Datenverkehr zwischen dem Benutzergerät und dem Server verfügbar. Jeder virtuelle Kanal ist mit einer bestimmten Priorität verknüpft und wird von der entsprechenden TCP-Verbindung transportiert. Sie können die Kanäle basierend auf der Portnummer, die für die Verbindung verwendet wird, unabhängig voneinander festlegen.

Gestreamte Mehrkanalverbindungen werden für Virtual Delivery Agents (VDAs) unterstützt, die auf Windows 8- und Windows 7-Maschinen installiert sind. Arbeiten Sie mit dem Netzwerkadministrator Ihres Unternehmens zusammen, um sicherzustellen, dass die in der Einstellung "Multiport-Richtlinie" konfigurierten Common Gateway Protocol (CGP)-Ports auf den Netzwerkroutern richtig zugewiesen sind.

Quality of Service wird nur unterstützt, wenn mehrere Sitzungszuverlässigkeitsports, oder CGP-Ports, konfiguriert sind.

Achtung: Verwenden Sie Transportsicherheit, wenn Sie dieses Feature einsetzen. Citrix empfiehlt die Verwendung von Internetprotokollsicherheit (IPsec) oder Secure Sockets Layer (SSL). SSL-Verbindungen werden nur unterstützt, wenn die Verbindungen durch ein NetScaler Gateway passieren, das Multistream unterstützt. Bei internen Unternehmensnetzwerken werden Multistreamverbindungen mit SSL nicht unterstützt.

Fügen Sie folgende Citrix Richtlinieneinstellungen einer Richtlinie hinzu, um die Servicequalität für mehrere Streamingverbindungen festzulegen (weitere Details finden Sie unter [Einstellungen der Richtlinie](#)

"**Multistreamverbindungen**"):

- **Multiportrichtlinie:** Diese Einstellung legt Ports für den ICA-Verkehr über mehrere Verbindungen fest und definiert die Netzwerkpriorität.
  - Wählen Sie eine Priorität aus der Liste CGP-Standardportpriorität aus. Standardmäßig hat der primäre Port (2598) eine hohe Priorität.
  - Geben Sie in den Feldern CGP-Port1, CGP-Port2 und CGP-Port3 je nach Bedarf zusätzliche CGP-Ports sowie entsprechende Prioritäten ein. Jeder Port muss eine eindeutige Priorität haben.Konfigurieren Sie die Firewalls auf VDAs explizit so, dass zusätzlicher TCP-Datenverkehr zulässig ist.
- **Multistreamcomputereinstellung:** Diese Einstellung ist standardmäßig deaktiviert. Wenn Sie Citrix Cloudbridge mit Multistream-Unterstützung in Ihrer Umgebung verwenden, müssen Sie diese Einstellung nicht konfigurieren. Konfigurieren Sie diese Richtlinieneinstellung, wenn Sie Router von Drittanbietern oder Legacy-Branch Repeater verwenden, um die gewünschte Quality of Service (QoS) zu erzielen.
- **Multistreambenutzereinstellung:** Diese Einstellung ist standardmäßig deaktiviert.

Damit die Richtlinien mit diesen Einstellungen wirksam werden, müssen sich Benutzer abmelden und dann am Netzwerk anmelden.



# Überlegungen zu USB und Clientlaufwerk

May 10, 2016

Mit HDX USB-Geräteumleitung können Benutzer eine Verbindung zum USB-Flashlaufwerk auf einem lokalen Computer herstellen und dann remote von einem virtuellen Desktop oder einer desktopgehosteten Anwendung darauf zugreifen. Während einer Sitzung können Benutzer Plug-n-Play-Geräte verwenden, einschließlich PTP-Geräte (Picture Transfer Protocol) wie digitale Kameras, MTP-Geräte (Media Transfer Protocol) wie digitale Audio Player oder tragbare Media Player und POS-Geräte (Point of Sale).

Double-Hop-USB wird bei Sitzungen mit über Desktop gehosteten Anwendungen nicht unterstützt.

USB-Umleitung ist verfügbar für Receiver für Windows und Receiver für Linux.

Standardmäßig ist die USB-Umleitung für bestimmte Klassen von USB-Geräten zulässig bzw. nicht zulässig. Weitere Informationen hierzu finden Sie in der Receiver-Dokumentation. Sie können festlegen, welche USB-Geräte einem virtuellen Desktop verfügbar gemacht werden, indem Sie die Liste der unterstützten USB-Geräte für die Umleitung beschränken.

## Important

In Umgebungen, in denen aus Sicherheitsgründen eine Trennung zwischen dem Benutzergerät und dem Server erforderlich ist, müssen Sie die Benutzer darüber informieren, welche Typen von USB-Geräten zu vermeiden sind.

Die beliebtesten USB-Geräte können über optimierte virtuelle Kanäle umgeleitet werden und liefern Leistung und Bandbreiteneffizienz über ein WAN. Der Grad der Unterstützung hängt vom auf dem Benutzergerät installierten Receiver ab. Optimierte virtuelle Kanäle sind normalerweise die beste Option, insbesondere in Umgebungen mit hoher Latenz.

Für die Umleitung von USB-Geräten gilt, dass ein SMART-Board ebenso wie eine Maus behandelt wird.

Das Produkt unterstützt die Verwendung von optimierten virtuellen Kanälen mit USB 3.0-Geräten und USB 3.0-Ports, wie ein virtueller CDM-Kanal, der zum Anzeigen von Dateien auf einer Kamera oder zur Audiowiedergabe über ein Headset verwendet wird. Das Produkt unterstützt zudem die generische USB-Umleitung von USB 3.0-Geräten, die mit einem USB 2.0-Port verbunden sind.

Spezialgeräte, für die es keine optimierten virtuellen Kanäle gibt, werden durch die Verwendung eines allgemeinen virtuellen Kanals für USB unterstützt, der eine unformatierte USB-Umleitung bietet. Informationen zu allen mit XenDesktop getesteten USB-Geräten finden Sie unter [CTX123569](#).

Einige erweiterte gerätespezifische Features, wie z. B. auf HID-Schaltflächen (Human Interface Device) auf einer Webcam, funktionieren möglicherweise mit dem optimierten virtuellen Kanal nicht wie erwartet. Wenn dies ein Problem darstellt, können Sie den allgemeinen virtuellen Kanal für USB verwenden.

Einige Geräte werden standardmäßig nicht umgeleitet und sind nur in der lokalen Sitzung verfügbar. Es wäre beispielsweise nicht angemessen, eine über internes USB an der Hauptplatine des Benutzergeräts angeschlossene Netzwerkkarte umzuleiten.

Die folgenden Citrix Richtlinieneinstellungen steuern die USB-Unterstützung:

- **Regeln für die Client-USB-Geräteoptimierung:** verfügbar in 7.6 FP3 mit GPMC. Der Optimierungsmodus wird für Eingabegeräte der Klasse 3 ("class=03") unterstützt, z. B. Signaturgeräte und Grafiktablets. Wenn keine Regel angegeben ist, wird das Gerät im interaktiven Modus behandelt (02). Der Erfassungsmodus (04) ist der empfohlene Modus für Signaturgeräte.
- **Client-USB-Geräteumleitung:** Die Standardeinstellung ist Nicht zugelassen.
- **Regeln für die Client-USB-Geräteumleitung:** Die Regeln gelten nur für Geräte, die generische USB-Umleitung verwenden, und nicht für Geräte, die spezielle oder optimierte Umleitung verwenden, wie CDM.
- **Client-USB-Geräteumleitung für Plug & Play-Geräte:** Die Standardeinstellung Zugelassen ermöglicht Plug & Play von PTP-, MTP- und POS-Geräten in einer Sitzung.
- **Bandbreitenlimit für Client-USB-Geräteumleitung:** Der Standardwert ist 0 (kein Maximum).
- **Bandbreitenlimit für Client-USB-Geräteumleitung (Prozent):** Der Standardwert ist 0 (kein Maximum).

Generische USB-Umleitung ist für spezielle USB-Geräte, für die es keinen optimierten virtuellen Kanal gibt. Diese Funktion leitet beliebige USB-Geräte von Clientmaschinen an virtuelle Desktops um. Durch dieses Feature können Endbenutzer mit vielen generischen USB-Geräten in der Desktopsitzung so interagieren, als wären die Geräte direkt angeschlossen.

Vorteile von generischer USB-Umleitung:

- Benutzer müssen keine Gerätetreiber auf den Benutzergeräten installieren.
- USB-Clienttreiber werden auf der VDA-Maschine installiert.

Dieses Feature wird für Desktopsitzungen auf VDAs für Desktop-OS 7.6 unterstützt.

Dieses Feature wird mit den folgenden Einschränkungen auch für Desktopsitzungen auf VDAs für Server-OS 7.6 unterstützt:

- Auf der VDA-Maschine muss Windows Server 2012 R2 ausgeführt werden
- Nur Single-Hop-Szenarios werden unterstützt
- Die USB-Clienttreiber müssen mit RDSH für Windows 2012 R2 kompatibel sein
- USB-Speichergeräte, Audiogeräte, Smartcardleser und Geräte, die nicht voll virtualisiert sind, werden nicht unterstützt

Weitere Informationen zur generischen USB-Umleitung finden Sie unter [CTX137939](#).

1. Fügen Sie die Einstellung Client-USB-Geräteumleitung einer Richtlinie hinzu und stellen Sie den Wert auf Zugelassen ein.
2. Optional: Zum Aktualisieren der für Remoting verfügbaren USB-Geräteleiste fügen Sie die Einstellung Regeln für die Client-USB-Geräteumleitung einer Richtlinie hinzu und stellen Sie die USB-Standardrichtlinienregeln ein.
3. Aktivieren Sie die USB-Unterstützung, wenn Sie Receiver auf Benutzergeräten installieren. Wenn Sie im vorigen Schritt die USB-Richtlinienregeln für Virtual Delivery Agent festgelegt haben, geben Sie nun die gleichen Richtlinienregeln für Receiver ein. Informationen zur USB-Unterstützung für Thin Clients und die erforderliche Konfiguration erhalten Sie vom Hersteller.

USB-Geräte werden automatisch umgeleitet, wenn die USB-Unterstützung aktiviert ist und die USB-Einstellungen für eine automatische Verbindung der USB-Geräte konfiguriert wurden. USB-Geräte werden auch automatisch umgeleitet, wenn das Gerät im Modus "Desktop Appliance" ist und der Verbindungsbalken nicht angezeigt wird. In manchen Fällen empfiehlt es sich jedoch, die automatische Umleitung nicht für alle USB-Geräte zu aktivieren. Weitere Informationen finden Sie unter [CTX123015](#).

Benutzer können Geräte, die nicht automatisch umgeleitet werden, explizit umleiten, indem sie sie aus der USB-Geräteleiste auswählen. Um zu verhindern, dass Geräte überhaupt aufgeführt oder umgeleitet werden, können Sie auf dem Client und für VDA die nachfolgenden Geräteregele festlegen.

Sie können die USB-Geräte, die für Remoting verfügbar sind, aktualisieren, indem Sie Regeln für die Client-USB-Geräteumleitung für Receiver und Virtual Desktop Agent definieren, die dann die USB-Standardrichtlinienregeln überschreiben.

- Bearbeiten Sie die Benutzergeräteregistrierung. Eine administrative Vorlage (ADM-Datei) ist auf dem Installationsmedium enthalten, sodass Sie das Gerät über die Active Directory-Gruppenrichtlinie ändern können: dvd root \os\lang\Support\Configuration\icacient\_usb.adm.
- Bearbeiten Sie die Administrator-Überschreibungsregeln der Serverbetriebssystemmaschinen mit den Gruppenrichtlinienregeln. Die Gruppenrichtlinien-Verwaltungskonsolle ist auf dem Installationsmedium enthalten:
  - Für x64: dvd root \os\lang\x64\Citrix Policy\ CitrixGroupPolicyManagement\_x64.msi
  - Für x86: dvd root \os\lang\x86\Citrix Policy\ CitrixGroupPolicyManagement\_x86.msi

### Warnung

Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

Die Produktstandardregeln sind in HKLM\SOFTWARE\Citrix\PortICA\GenericUSB\DeviceRules gespeichert. Ändern Sie diese Produktstandardregeln nicht. Verwenden Sie sie als Anleitung zum Erstellen von Administrator-Überschreibungsregeln, wie nachfolgend beschrieben. Die GPO-Überschreibungen werden ausgewertet, bevor die Produktstandardregeln angewendet werden.

Die Administrator-Override-Regeln sind in HKLM\SOFTWARE\Citrix\PortICA\GenericUSB\DeviceRules gespeichert. GPO-Richtlinienregeln haben das Format {Allow:| Deny;} gefolgt von Tag=Wert-Ausdrücken, die durch Leerzeichen getrennt sind. Die folgenden Tags werden unterstützt:

Tag	Beschreibung
VID	Vendor-ID vom Gerätedeskriptor
PID	Produkt-ID vom Gerätedeskriptor
REL	Release-ID vom Gerätedeskriptor
Klasse	Klasse vom Gerätedeskriptor oder einem Schnittstellendeskriptor; verfügbare USB-Klassencodes finden Sie auf der USB-Website unter <a href="http://www.usb.org/">http://www.usb.org/</a> .
SubClass	Unterklasse vom Gerätedeskriptor oder ein Schnittstellendeskriptor
Prot	Protokoll vom Gerätedeskriptor oder ein Schnittstellendeskriptor

- Wenn Sie neue Richtlinienregeln erstellen, beachten Sie Folgendes:
- Bei Regeln wird die Groß- und Kleinschreibung nicht berücksichtigt.
  - Regeln können optional von einem Kommentar gefolgt werden, der mit # eingeleitet wird. Ein Trennzeichen ist nicht erforderlich, der Kommentar wird beim Abgleichen ignoriert.
  - Leere Zeilen und Kommentare werden ignoriert.
  - Leerzeichen dienen als Trennzeichen, sie können nicht in einer Zahl oder Kennung verwendet werden. Beispielsweise ist Deny: Class = 08 SubClass=05 eine gültige Regel; Deny: Class=0 Sub Class=05 hingegen nicht.
  - Tags müssen den Übereinstimmungsoperator = verwenden. Beispielsweise VID=1230.
  - Jede Regel muss auf einer neuen Zeile beginnen oder Teil einer durch Semikolon getrennten Liste sein.

### Important

Wenn Sie die ADM-Vorlagendatei verwenden, müssen Sie die Regeln in einer einzigen Zeile mit Semikolons getrennt eingeben.

Bei optimierten Geräten, wie Massenspeichergeräten, wird das Gerät normalerweise über den speziellen CDM-Kanal umgeleitet und nicht mit Richtlinienregeln. Sie können dieses Verhalten auf die folgende Weise überschreiben:

- Leiten Sie das optimierte Gerät manuell mit der generischen USB-Umleitung um: Wählen Sie im Dialogfeld Einstellungen auf der Registerkarte Geräte die Option Zu allgemein wechseln.
- Leiten Sie das optimierte Gerät automatisch mit der generischen USB-Umleitung um, indem Sie die automatische Umleitung für das Speichergerät festlegen (z. B. AutoRedirectStorage = 1) und die USB-Einstellungen für eine automatische Verbindung der USB-Geräte konfigurieren. Weitere Informationen finden Sie unter [CTX123015](#).

- Beispiele:
- Das folgende Beispiel zeigt eine vom Administrator definierte USB-Richtlinienregel für Vendor- und Produkt-IDs:  
Allow: VID=046D PID=C626 # Allow Logitech SpaceNavigator 3D Mouse Deny: VID=046D # Deny all Logitech products

- Das folgende Beispiel zeigt eine vom Administrator definierte USB-Richtlinienregel für eine definierte Klasse, Unterklasse und ein Protokoll:  
Deny: Class=EF SubClass=01 Prot=01 # Deny MS Active Sync devices Allow: Class=EF SubClass=01 # Allow Sync devices Allow: Class=EF # Allow all USB-Miscellaneous devices

Standardmäßig werden USB-Geräte automatisch umgeleitet, wenn die USB-Unterstützung aktiviert ist und die USB-Einstellungen für eine automatische Verbindung der USB-Geräte konfiguriert wurden. USB-Geräte werden auch für Desktopgerätesites oder desktopgehostete Anwendungen automatisch umgeleitet. In manchen Fällen empfiehlt es sich jedoch, die automatische Umleitung nicht für alle USB-Geräte zu aktivieren. Weitere Informationen finden Sie unter [CTX123015](#).

Benutzer von Desktop Viewer können Geräte umleiten, die nicht automatisch umgeleitet werden, indem Sie sie aus der USB-Geräteleiste auswählen. Um zu verhindern, dass Geräte aufgeführt oder umgeleitet werden, legen Sie auf dem Benutzergerät und dem VDA die Geräteregeln fest.

Sie können die USB-Geräte, die für Remoting verfügbar sind, aktualisieren, indem Sie Regeln für die Client-USB-Geräteumleitung für Receiver und Virtual Desktop Agent definieren, die dann die USB-Standardrichtlinienregeln überschreiben. Geräteregeln werden sowohl für Receiver als auch Virtual Desktop Agent zwingend umgesetzt. Achten Sie darauf, die Regeln für beide zu ändern, damit die Geräteumleitung wie vorgesehen funktioniert.

- Bearbeiten Sie die Benutzergeräteregistrierung (oder die INI-Dateien bei Receiver für Linux). Eine administrative Vorlage (ADM-Datei) ist auf dem Installationsmedium enthalten, sodass Sie das Gerät über die Active Directory-Gruppenrichtlinie ändern können: dvd root \os\lang\Support\Configuration\icaclient\_usb.adm.
- Bearbeiten Sie die Administrator-Override-Regeln der VDA-Registrierung auf den Serverbetriebssystemmaschinen. Eine administrative Vorlage (ADM-Datei) ist auf dem Installationsmedium enthalten, sodass Sie VDA über die Active Directory-Gruppenrichtlinie ändern können: dvd root \os\lang\Support\Configuration\vda\_usb.adm.

## Warnung

Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

Die Produktstandardregeln sind in HKLM\SOFTWARE\Citrix\PortICA\GenericUSB\DeviceRules gespeichert. Ändern Sie diese Produktstandardregeln nicht. Verwenden Sie sie als Anleitung zum Erstellen von Administrator-Überschreibungsregeln, wie nachfolgend beschrieben. Die GPO-Überschreibungen werden ausgewertet, bevor die Produktstandardregeln angewendet werden.

Die Administrator-Override-Regeln sind in HKLM\SOFTWARE\Citrix\PortICA\GenericUSB\DeviceRules gespeichert. GPO-Richtlinienregeln haben das Format {Allow:|Deny:} gefolgt von Tag=Wert-Ausdrücken, die durch Leerzeichen getrennt sind. Die folgenden Tags werden unterstützt:

Tag	Beschreibung
VID	Vendor-ID vom Gerätedeskriptor
PID	Produkt-ID vom Gerätedeskriptor
REL	Release-ID vom Gerätedeskriptor
Klasse	Klasse vom Gerätedeskriptor oder einem Schnittstellendeskriptor; verfügbare USB-Klassencodes finden Sie auf der USB-Website unter <a href="http://www.usb.org/">http://www.usb.org/</a> .
SubClass	Unterklasse vom Gerätedeskriptor oder ein Schnittstellendeskriptor
Prot	Protokoll vom Gerätedeskriptor oder ein Schnittstellendeskriptor

Wenn Sie neue Richtlinienregeln erstellen, beachten Sie Folgendes:

- Bei Regeln wird die Groß- und Kleinschreibung nicht berücksichtigt.
- Regeln können optional von einem Kommentar gefolgt werden, der mit # eingeleitet wird. Ein Trennzeichen ist nicht erforderlich, der Kommentar wird beim Abgleichen ignoriert.
- Leere Zeilen und Kommentare werden ignoriert.
- Leerzeichen dienen als Trennzeichen, sie können nicht in einer Zahl oder Kennung verwendet werden. Beispielsweise ist Deny: Class = 08 SubClass=05 eine gültige Regel; Deny: Class=0 Sub Class=05 hingegen nicht.
- Tags müssen den Übereinstimmungsoperator = verwenden. Beispielsweise VID=1230.
- Jede Regel muss auf einer neuen Zeile beginnen oder Teil einer durch Semikolon getrennten Liste sein.

## Important

Wenn Sie die ADM-Vorlagendatei verwenden, müssen Sie die Regeln in einer einzigen Zeile mit Semikolons getrennt eingeben.

Bei optimierten Geräten, wie Massenspeichergeräten, wird das Gerät normalerweise über den speziellen CDM-Kanal umgeleitet und nicht mit Richtlinienregeln. Sie können dieses Verhalten auf die folgende Weise überschreiben:

- Leiten Sie das optimierte Gerät manuell mit der generischen USB-Umleitung um: Wählen Sie im Dialogfeld Einstellungen auf der Registerkarte Geräte die Option Zu allgemein wechseln.
- Leiten Sie das optimierte Gerät automatisch mit der generischen USB-Umleitung um, indem Sie die automatische Umleitung für das Speichergerät festlegen (z. B. AutoRedirectStorage = 1) und die USB-Einstellungen für eine automatische Verbindung der USB-Geräte konfigurieren. Weitere Informationen finden Sie unter [CTX123015](#).

Beispiele:

- Das folgende Beispiel zeigt eine vom Administrator definierte USB-Richtlinienregel für Vendor- und Produkt-IDs:  
Allow: VID=046D PID=C626 # Allow Logitech SpaceNavigator 3D Mouse Deny: VID=046D # Deny all Logitech products
- Das folgende Beispiel zeigt eine vom Administrator definierte USB-Richtlinienregel für eine definierte Klasse, Unterklasse und ein Protokoll:  
Deny: Class=EF SubClass=01 Prot=01 # Deny MS Active Sync devices Allow: Class=EF SubClass=01 # Allow Sync devices Allow: Class=EF # Allow all USB-Miscellaneous devices

Benutzer können ein USB-Gerät vor oder nach dem Starten einer virtuellen Sitzung anschließen.

Wenn Sie mit Receiver für Windows arbeiten, gilt Folgendes:

- Geräte, die nach dem Sitzungsbeginn angeschlossen werden, werden unmittelbar im USB-Menü von Desktop Viewer angezeigt.
- Wenn ein USB-Gerät nicht richtig umgeleitet wird, können Sie das Problem u. U. beheben, indem Sie das Gerät erst nach dem Beginn der virtuellen Sitzung anschließen.
- Um Datenverlust zu verhindern, verwenden Sie das Windows-Symbol "Hardware sicher entfernen", bevor Sie das USB-Gerät entfernen.

Nur bei Massenspeichergeräten ist Remotezugriff auch durch Clientlaufwerkszuordnung verfügbar. Hierbei werden Laufwerke auf dem Benutzergerät automatisch Laufwerksbuchstaben auf dem virtuellen Desktop zugeordnet, wenn Benutzer sich anmelden. Die Laufwerke werden als freigegebene Ordner mit zugeordneten Laufwerksbuchstaben angezeigt. Um Clientlaufwerkszuordnung zu konfigurieren, verwenden Sie unter den ICA-Richtlinieneinstellungen im Abschnitt "Dateiumleitung" die Einstellung Clientwechseldatenträger.

Die Hauptunterschiede zwischen den beiden Typen der Remotingrichtlinie sind:

Feature	Clientlaufwerkszuordnung	Generische USB-Umleitung
Diese Option ist in der Standardeinstellung aktiviert.	Ja	Nein
Konfigurierbare Leserechte	Ja	Nein
Sicheres Entfernen des Geräts in einer Sitzung	Nein	Ja, unter der Voraussetzung, dass Benutzer den Empfehlungen des Betriebssystems zum sicheren Entfernen von Geräten folgen.

Wenn die Richtlinien für die generische USB-Umleitung und die Clientlaufwerkszuordnung aktiviert sind und ein Massenspeichergerät vor oder nach dem Sitzungsstart angeschlossen wird, wird es mit der Clientlaufwerkszuordnung umgeleitet. Wenn die Richtlinien für die generische USB-Umleitung und die Clientlaufwerkszuordnung aktiviert sind und für ein Gerät die automatische Umleitung konfiguriert wurde (siehe <http://support.citrix.com/article/CTX123015>) und wenn ein Massenspeichergerät vor oder nach dem Sitzungsstart angeschlossen wird, wird es mit der generischen USB-Umleitung umgeleitet.

## Hinweis

Die USB-Umleitung wird für Verbindungen mit geringer Bandbreite (z. B. 50 KBit/s) unterstützt, das Kopieren großer Dateien funktioniert jedoch nicht.

Sie können steuern, ob Benutzer Dateien von ihren virtuellen Umgebungen auf ihre Benutzergeräte kopieren können. Standardmäßig ist in der Sitzung Lese-/Schreibzugriff auf Dateien und Ordner auf zugeordneten Clientlaufwerken möglich.

Um zu verhindern, dass Benutzer Dateien und Ordner auf zugeordneten Clientlaufwerken hinzufügen oder ändern, aktivieren Sie die Richtlinieneinstellung Schreibgeschützter Zugriff auf Clientlaufwerke. Wenn Sie diese Einstellung einer Richtlinie hinzufügen, müssen Sie die Einstellung Clientlaufwerkumleitung auf Zugelassen festlegen und zur Richtlinie hinzufügen.

# Überwachen

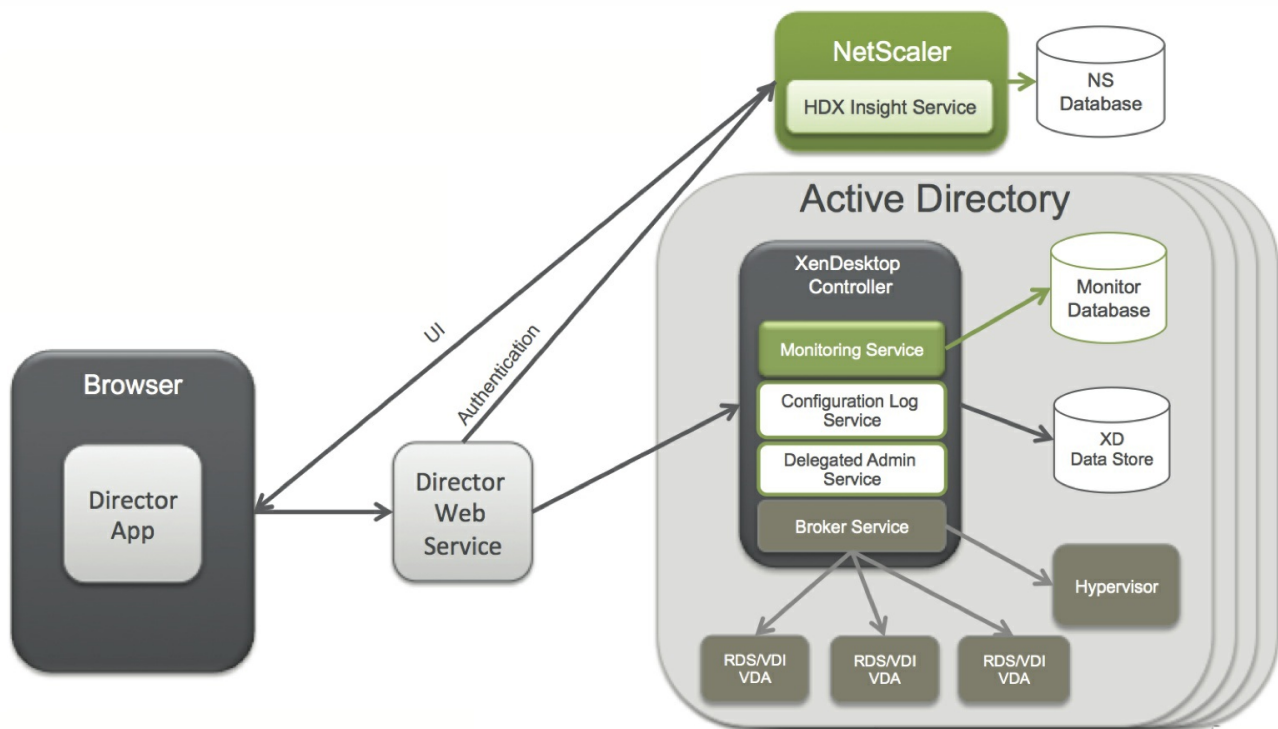
May 10, 2016

Administratoren und Helpdesk-Mitarbeiter können XenApp- und XenDesktop-Sites mit Director überwachen, indem Administratoren auf die Datenbank für die Konfigurationsprotokollierung zugreifen oder die Überwachungsdienst-OData-API der Site verwenden.

Administratoren können Folgendes überwachen:

- Sitzungsnutzung
- Anmeldungsleistung
- Verbindungs- und Maschinenfehler
- Lastauswertung
- Historische Trends
- Infrastruktur
- Benutzersitzungen
- Maschinen

Director ist ein Echtzeit-Webtool, mit dem Administratoren Endbenutzer überwachen, Fehler beheben und Support leisten können.



Director hat auf Folgendes Zugriff:

- Echtzeitdaten vom Broker Agent über eine einheitliche Konsole, die mit EdgeSight-Features, Leistungsverwaltung und Netzwerkinspektion integriert ist.
  - Die EdgeSight-Features umfassen Leistungsverwaltung zum Sicherstellen von Integrität und Kapazität, sowie

historische Trends und Netzwerkanalyse (von NetScaler HDX Insight) zum Identifizieren von Engpässen, die auf dem Netzwerk in der XenApp- oder XenDesktop-Umgebung beruhen.

- In der Überwachungsdatenbank gespeicherte historische Daten für den Zugriff auf die Datenbank für die Konfigurationsprotokollierung.
- ICA-Daten von NetScaler Gateway mit HDX Insight.
  - Übersicht über die Endbenutzererfahrung für virtuelle Anwendungen, Desktops und Benutzer in XenApp oder XenDesktop.
  - Korrelation von Netzwerkdaten mit Anwendungsdaten und Echtzeitmetrik für effektive Problembehandlung.
  - Integration mit dem Überwachungstool von XenDesktop 7 Director.
- Personal vDisk-Daten, die die Ausführungsüberwachung anhand von Basiszuordnung und Helpdesk-Mitarbeitern das Zurücksetzen von Personal vDisk ermöglichen (nur im Notfall zu verwenden).
  - Das Befehlszeilentool CtxPvdDiag.exe wird zum Sammeln von Benutzerinformationen in einer Datei zur Problembehandlung verwendet.

Director hat ein Dashboard zur Problembehandlung, das Echtzeitzustandsüberwachung der XenApp- oder XenDesktop-Site ermöglicht. Mit dieser Funktion können Administratoren Fehler in Echtzeit sehen und einen besseren Eindruck von der Endbenutzererfahrung erhalten.

Die Sitzungsaufzeichnung ermöglicht das Aufzeichnen von Bildschirmaktivitäten in der Sitzung eines Benutzers über eine beliebige Verbindung von einem XenApp-Server (im Einklang mit den Unternehmensrichtlinien und gesetzlichen Vorschriften). Mit der Sitzungsaufzeichnung können Sie Sitzungen aufzeichnen, katalogisieren und archivieren und sie erneut aufrufen und wiedergeben.

Die Sitzungsaufzeichnung verwendet flexible Richtlinien, mit denen automatisch Aufnahmen von Anwendungssitzungen ausgelöst werden können. Das IT-Team kann damit die Benutzeraktivität von Anwendungen überwachen und prüfen, z. B. Buchhaltungs- und Patienteninformationssysteme für das Gesundheitswesen, und interne Kontrollen zur Einhaltung von gesetzlichen Vorschriften und die Sicherheitsüberwachung unterstützen. Die Sitzungsaufzeichnung vereinfacht auch den technischen Support, da die Problemerkennung und Behebung der Probleme beschleunigt werden.

Die Konfigurationsprotokollierung ist ein Feature, mit dem Administratoren administrative Änderungen verfolgen, die an einer XenApp- oder XenDesktop-Site vorgenommen werden. Die Konfigurationsprotokollierung ermöglicht Administratoren die Diagnose und Problembehandlung nach der Durchführung von Konfigurationsänderungen, Hilfe beim Änderungsmanagement und der Nachverfolgung von Konfigurationen sowie Berichte über Administratoraktivitäten.

Über die Trendansicht der Konfigurationsprotokollierung in Director erhalten Administratoren, die keinen Zugriff auf XenDesktop Citrix Studio haben, Benachrichtigungen über Konfigurationsänderungen.

Die Trendansicht bietet historische Daten von Konfigurationsänderungen in einem bestimmten Zeitraum, sodass Administratoren beurteilen können, welche Änderungen wann und von wem an den Sites vorgenommen wurden, um die Ursache eines Problems zu finden. Konfigurationsinformationen werden in dieser Ansicht in drei Kategorien unterteilt.

- Verbindungsfehler
- Fehlgeschlagene Desktopmaschinen
- Fehlgeschlagene Servermaschinen

Mit Überwachungsdienst-OData-API der Site können Administratoren nach historischen Daten suchen. Dies ermöglicht IT-Mitarbeitern das Analysieren historischer Trends für die Planung, das Durchführen gezielter Problembehandlung bei Verbindungs- und Maschinenfehlern und das Extrahieren von Informationen für die Eingabe in andere Tools und Prozesse.

Das Schema des Überwachungsdiensts bietet folgende Datentypen:

- Daten zu Verbindungsfehlern
  - Maschinen in fehlerhaftem Zustand
  - Sitzungsnutzung
  - Anmeldedauer
  - Daten zum Lastausgleich
- 
- [Director](#)
  - [Sitzungsaufzeichnung – für XenApp 7.6 FP1 und FP2](#)
  - [Überwachen von Personal vDisks](#)
  - [Konfigurationsprotokollierung](#)
  - [Überwachungsdienst-OData-API](#)

# Director

May 10, 2016

Director bietet verschiedene Ansichten der Schnittstelle, die auf bestimmte Administratoren abgestimmt sind. Produktberechtigungen bestimmen, was angezeigt wird und welche Befehle verfügbar sind.

Beispiel: Helpdeskadministratoren sehen eine auf Helpdeskaufgaben abgestimmte Schnittstelle. Director ermöglicht Helpdeskadministratoren, nach dem Benutzer zu suchen, der das Problem gemeldet hat, und die diesem Benutzer zugeordneten Aktivitäten anzuzeigen, z. B. den Status der Anwendungen und Prozesse des Benutzers. So können Probleme schnell gelöst werden, indem Aktionen wie z. B. das Beenden einer nicht reagierenden Anwendung oder eines Prozesses, das Spiegeln von Vorgängen auf der Maschine des Benutzers, der Neustart der Maschine oder das Zurücksetzen des Benutzerprofils durchgeführt werden.

Im Gegensatz dazu sehen und verwalten Volladministratoren die gesamte Website und können Befehle für mehrere Benutzer und Maschinen ausführen. Das Dashboard bietet einen Überblick über die wichtigsten Aspekte einer Bereitstellung, z. B. den Status von Sitzungen und Benutzeranmeldungen und die Infrastruktur der Site. Die Informationen werden jede Minute aktualisiert. Wenn Probleme auftreten, werden automatisch Details zu Anzahl und Art der Fehler angezeigt.

Director ist standardmäßig als Website auf dem Delivery Controller installiert. Informationen zu Voraussetzungen und anderen Details finden Sie in den Systemanforderungen zu diesem Release.

Dieses Release von Director ist nicht kompatibel mit XenApp-Bereitstellungen vor Version 6.5 und XenDesktop-Bereitstellungen vor Version 7.

Wenn Director in einer Umgebung mit mehreren Sites verwendet wird, synchronisieren Sie die Systemuhren auf allen Servern, auf denen Controller, Director und andere wichtige Kernkomponenten installiert sind. Ansonsten werden die Sites in Director möglicherweise nicht richtig angezeigt.

Tipp: Wenn Sie beabsichtigen, neben XenApp 7.5- und XenDesktop 7.x-Sites auch XenApp 6.5-Sites zu überwachen, empfiehlt Citrix, Director und die Director-Konsole, die zur Überwachung von XenApp 6.5-Sites verwendet wird, auf separaten Servern zu installieren.

Wichtig: Zum Schutz von als Nur-Text über das Netzwerk gesendeten Benutzernamen und Kennwörtern empfiehlt Citrix dringend, nur Director-Verbindungen mit HTTPS und nicht mit HTTP zuzulassen. Bestimmte Tools können Nur-Text-Benutzernamen und -Kennwörter in (unverschlüsselten) HTTP-Netzwerkpaketen lesen, wodurch ein Sicherheitsrisiko für Benutzer entsteht.

## Konfigurieren von Berechtigungen

Um eine Anmeldung bei Director vornehmen zu können, müssen Administratoren mit den Berechtigungen für Director Active Directory-Domänenbenutzer sein und die folgenden Berechtigungen haben:

- Leseberechtigungen in allen zu durchsuchenden Active Directory-Gesamtstrukturen (siehe [Erweiterte Konfiguration](#))
- Konfigurierte delegierte Administratorrollen (siehe [Delegierte Administration und Director](#))
- Zum Spiegeln von Benutzern muss für Administratoren eine Microsoft-Gruppenrichtlinie für Windows-Remoteunterstützung konfiguriert werden. Darüber hinaus gilt Folgendes:
  - Bei der Installation von VDAs stellen Sie sicher, dass die Windows-Remoteunterstützung auf allen Benutzergeräten aktiviert ist (standardmäßig aktiviert).



- Wenn Sie Director auf einem Server installieren, stellen Sie sicher, dass die Windows-Remoteunterstützung installiert ist (standardmäßig ausgewählt). Allerdings ist sie auf dem Server standardmäßig deaktiviert. Das Feature muss für Director nicht aktiviert werden, um Benutzern zu helfen. Citrix empfiehlt, das Feature deaktiviert zu lassen, um die Sicherheit auf dem Server zu erhöhen.
- Damit Administratoren die Windows-Remoteunterstützung initiieren können, müssen Sie ihnen mit den entsprechenden Einstellungen der Microsoft-Gruppenrichtlinie die Berechtigungen für die Remoteunterstützung erteilen. Informationen finden Sie unter [CTX127388: How to Enable Remote Assistance for Desktop Director](#).
- Bei Benutzergeräten mit VDA-Versionen vor 7 ist eine zusätzliche Konfiguration erforderlich. Siehe [Konfigurieren von Berechtigungen für VDAs vor XenDesktop 7](#).

## Installieren von Director

**Hinweis:** Damit Director alle XenApp-Worker in der Farm findet, müssen Sie eine Reverse-DNS-Zone für die Subnetze hinzufügen, in denen sich die XenApp-Server auf den von der Farm verwendeten DNS-Servern befinden.

Installieren Sie Director mit dem Installer. Dieser prüft, ob die Voraussetzungen erfüllt sind, installiert fehlende Komponenten, richtet die Director Website ein und führt die Grundkonfiguration durch. Die Standardkonfiguration, die der Installer bietet, eignet sich für typische Bereitstellungen. Fügen Sie Director mit dem Installationsprogramm hinzu, falls dies während der Installation nicht geschehen ist. Zum Hinzufügen zusätzlicher Komponenten führen Sie den Installer erneut aus und wählen Sie die zu installierenden Komponenten aus. Informationen zur Verwendung des Installationsprogramms finden Sie in der Dokumentation zur Installation. Citrix empfiehlt, dass Sie die Installation ausschließlich mit dem Installer des Produkts und nicht über die MSI-Datei durchführen.

Wenn Director auf dem Controller installiert ist, erfolgt automatisch eine Konfiguration mit localhost als Serveradresse und Director kommuniziert standardmäßig mit dem lokalen Controller.

Zur Installation von Director auf einem dedizierten, Controller-remoten Server werden Sie zur Eingabe des FQDN oder der IP-Adresse eines Controllers aufgefordert. Director kommuniziert standardmäßig mit diesem angegebenen Controller. Geben Sie nur eine Controlleradresse für jede zu überwachende Site ein. Director ermittelt automatisch alle anderen Controller in derselben Site und wechselt zu diesen anderen Controllern, wenn der von Ihnen angegebene Controller ausfällt.

**Hinweis:** Director führt keinen Lastausgleich zwischen Controllern aus.

Citrix empfiehlt die Implementierung von SSL auf der IIS-Website, die Director hostet, um die Kommunikation zwischen dem Browser und dem Webserver zu schützen. In der Dokumentation von Microsoft zu IIS finden Sie entsprechende Anweisungen. Zum Aktivieren von SSL ist keine Director-Konfiguration erforderlich.

## So installieren Sie Director 7.6.300

Director 7.6.300 bietet Unterstützung für den virtuellen Framehawk-Kanal und enthält die aktuellen Produktkorrekturen. Es steht mit XenApp und XenDesktop 7.6 FP2 und mit XenApp 6.5 FP3 zur Verfügung.

**Hinweis:** Vergewissern Sie sich, dass Sie alle erforderlichen Features in IIS ausgewählt haben. Eine vollständige Liste finden Sie unter [CTX142260](#).

1. Laden Sie Director herunter und führen Sie die MSI-Datei, DesktopDirector.MSI oder DesktopDirector\_x64.MSI aus.
2. Installieren Sie CitrixGroupPolicyManagement.MSI, die Sie im Citrix Gruppenrichtlinienordner auf dem Installationsmedium von XenApp bzw. XenDesktop finden.
3. Konfigurieren Sie Director mit dem Delivery Controller unter Verwendung des Tools DirectorConfig.exe aus C:\inetpub\wwwroot\Director\tools. Weitere Informationen finden Sie unter [CTX137990](#).

4. Registrieren Sie ASP.net bei IIS. Führen Sie hierfür den Befehl `C:\inetpub\wwwroot\Director\tools>DirectorConfig.exe /registerdotnet` aus.
5. Installieren oder aktualisieren Sie WMIProxy auf dem VDA und starten Sie den Computer neu. WMIProxy\_x64.MSI und WMIProxy\_x86.MSI sind in den Director 7.6.300-Downloaddateien enthalten. **Hinweis:** Der virtuelle Framehawk-Kanal wird als nicht kompatibel mit einem VDA mit aktiviertem Framehawk angezeigt, wenn WMIProxy nicht aktualisiert wird.
6. Führen Sie ein Upgrade von XDPoshSnapin\_Hotfix auf dem Delivery Controller aus und starten Sie Studio neu. XDPoshSnapin.MSI ist in den in Schritt 1 heruntergeladenen Director-Downloaddateien enthalten. Dies ist erforderlich, damit delegierte Administratoren und benutzerdefinierte Administratoren die Informationen zum virtuellen Framehawk-Kanal anzeigen können.

## Anmelden an Director

Die Director Website befindet sich unter `https` oder `http://Director`.

Wenn eine der Sites einer Bereitstellung mit mehreren Sites ausfällt, dauert die Anmeldung für Director etwas länger, während Verbindungsversuche mit dieser Site laufen.

## Installieren von Director für XenApp 6.5

Wenn Director bereits für XenDesktop installiert ist, schließen Sie die Konfiguration für XenApp wie folgt ab:

- Verwenden Sie die IIS-Verwaltungskonsolle auf jedem Director-Server, um die Liste der XenApp-Serveradressen in den Anwendungseinstellungen zu aktualisieren, wie unter "So fügen Sie Director Sites hinzu" in [Erweiterte Konfiguration](#) beschrieben. Geben Sie pro XenApp-Farm die Serveradresse eines Controllers an: Die anderen Controller in einer XenApp-Farm werden dann automatisch zum Failover verwendet. Director führt keinen Lastausgleich zwischen Controllern aus.
- Konfigurieren Sie jeden XenApp-Workerserver für die Annahme von WinRM-Abfragen, wie unter [So konfigurieren Sie Berechtigungen](#) beschrieben.
- Konfigurieren Sie eine Firewallausnahme für Port 2513, der für die Kommunikation zwischen Director und XenApp verwendet wird.

## So installieren Sie Director für XenApp 6.5 zum ersten Mal

Führen Sie die folgenden Schritte aus, um Director für XenApp 6.5 zum ersten Mal zu installieren. In der Regel wird Director auf einem anderen Computer als die XenApp-Controller installiert.

1. Installieren Sie Director von dem XenApp 7.6-Installationsmedium.
2. Verwenden Sie die IIS-Verwaltungskonsolle auf jedem Director-Server, um die Liste der XenApp-Serveradressen in den Anwendungseinstellungen zu aktualisieren, wie unter "So fügen Sie Director Sites hinzu" in [Erweiterte Konfiguration](#) beschrieben.  
Geben Sie pro XenApp-Site die Serveradresse eines Controllers an: Die anderen Controller in einer XenApp-Site werden dann automatisch zum Failover verwendet. Director führt keinen Lastausgleich zwischen Controllern aus.  
  
Wichtig: Stellen Sie bei XenApp-Adressen sicher, dass Sie die Einstellung "Service.AutoDiscoveryAddressesXA" und nicht die Standardeinstellung "Service.AutoDiscoveryAddresses" verwenden.
3. Der Installer für den WMI Provider von Director ist auf der DVD im Ordner Support\DirectorWMIProvider. Installieren Sie ihn auf den jeweiligen XenApp-Servern (Controller und Worker, wenn Sitzungen ausgeführt werden).  
Wenn winrm nicht konfiguriert ist, führen Sie den Befehl `winrm qc` aus.
4. Konfigurieren Sie jeden XenApp-Workerserver für die Annahme von WinRM-Abfragen, wie unter [So konfigurieren Sie Berechtigungen](#) beschrieben.

5. Konfigurieren Sie eine Firewallausnahme für Port 2513, der für die Kommunikation zwischen Director und XenApp verwendet wird.
6. Um die Kommunikation zwischen dem Browser und dem Webserver zu schützen, empfiehlt Citrix die Implementierung von SSL auf der IIS-Website, die Director hostet.  
In der Dokumentation von Microsoft zu IIS finden Sie entsprechende Anweisungen. Zum Aktivieren von SSL ist keine Director-Konfiguration erforderlich.

# Delegierte Administration und Director

May 10, 2016

Bei der delegierten Administration werden drei Konzepte eingesetzt: Administratoren, Rollen und Geltungsbereiche. Berechtigungen richten sich nach der Administratorrolle und dem Geltungsbereich dieser Rolle. Beispiel: Einem Administrator wird die Helpdeskadministratorrolle zugewiesen, bei der der Geltungsbereich die Verantwortung für Endbenutzer an nur einer Site umfasst.

Weitere Informationen über das Erstellen von delegierten Administratoren finden Sie unter [Delegierte Administration](#).

Durch die administrativen Berechtigungen wird festgelegt, wie die Director-Benutzeroberfläche für Administratoren dargestellt wird und welche Aufgaben sie ausführen können. Mit Berechtigungen wird Folgendes festgelegt:

- Die Seiten, auf die der Administrator zugreifen kann, kollektiv als "Ansicht" bezeichnet
- Die Desktops, Maschinen und Sitzungen, die der Administrator anzeigen und verwenden kann
- Die Befehle, die der Administrator ausführen kann, z. B. das Spiegeln einer Benutzersitzung oder das Aktivieren des Wartungsmodus

Über die integrierten Rollen und Berechtigungen wird außerdem gesteuert, wie Administratoren Director verwenden:

Administratorrolle	Berechtigungen in Director
Volladministrator	Hat vollständigen Zugriff auf alle Ansichten und kann alle Befehle ausführen, einschließlich Spiegeln einer Benutzersitzung, Aktivieren des Wartungsmodus und Exportieren von Trenddaten.
Bereitstellungsgruppenadministrator	Hat vollständigen Zugriff auf alle Ansichten und kann alle Befehle ausführen, einschließlich Spiegeln einer Benutzersitzung, Aktivieren des Wartungsmodus und Exportieren von Trenddaten.
Lesezugriffadministrator	Kann auf alle Ansichten zugreifen und alle Objekte in angegebenen Geltungsbereichen sowie globale Informationen anzeigen. Kann Berichte aus HDX-Kanälen herunterladen und Trenddaten mit der Exportoption in der Ansicht "Trends" exportieren. Kann keine anderen Befehle ausführen oder Daten in den Ansichten ändern.
Helpdeskadministrator	Kann nur auf die Ansichten "Helpdesk" und "Benutzerdetails" zugreifen und nur Objekte anzeigen, die dem Administrator zur Verwaltung übertragen wurden. Kann eine Benutzersitzung spiegeln und Befehle für diesen Benutzer ausführen. Kann Vorgänge im Wartungsmodus ausführen. Kann Energieoptionen auf Desktopbetriebssystemmaschinen verwenden. Kann nicht auf das Dashboard, Trends oder Filteransichten zugreifen. Kann keine Energieoptionen auf Serverbetriebssystemmaschinen verwenden.
Maschinenkatalogadministrator	Kein Zugriff. Dieser Administrator wird für Director nicht unterstützt und er kann keine Daten anzeigen. Dieser Benutzer kann auf die Seite "Maschinendetails" zugreifen (maschinenbasierte Suche).

Hostadministratorrolle	Kein Zugriff. Dieser Administrator wird für Director nicht unterstützt und er kann keine Daten anzeigen.
------------------------	----------------------------------------------------------------------------------------------------------

## So konfigurieren Sie benutzerdefinierte Rollen für Director-Administratoren

In Studio können Sie auch Director-spezifische benutzerdefinierte Rollen konfigurieren, die den Anforderungen Ihrer Organisation besser gerecht werden und eine flexiblere Delegation von Berechtigungen ermöglichen. Sie können beispielsweise die integrierte Helpdeskadministratorrolle einschränken, sodass dieser Administrator keine Sitzungen abmelden kann.

Wenn Sie eine benutzerdefinierte Rolle mit Director-Berechtigungen erstellen, müssen Sie dieser auch andere allgemeine Berechtigungen erteilen:

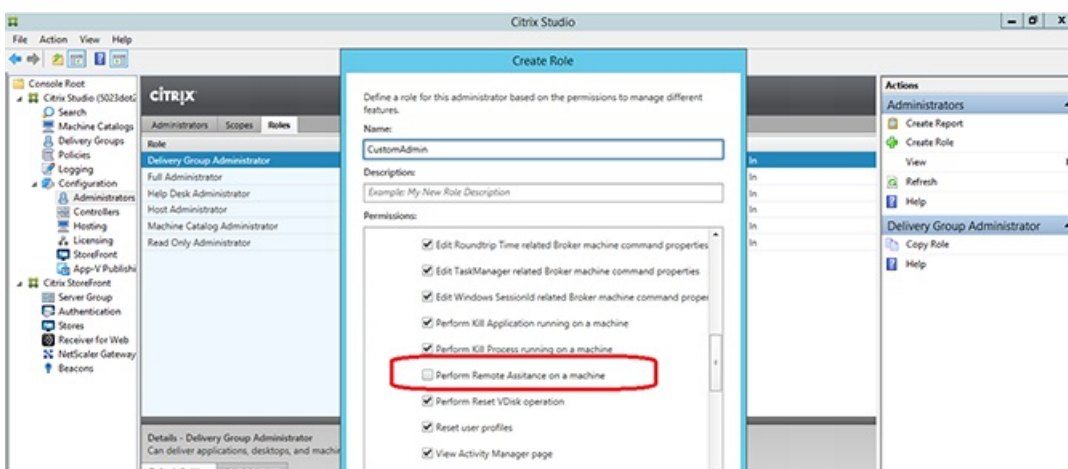
- Delivery Controller-Berechtigung zur Anmeldung bei Director.
- Berechtigungen für Bereitstellungsgruppen zum Anzeigen der zu diesen gehörigen Daten in Director.

Alternativ können Sie eine benutzerdefinierte Rolle erstellen, indem Sie eine vorhandene Rolle kopieren und dieser zusätzliche Berechtigungen für die verschiedenen Ansichten erteilen. Sie können beispielsweise die Rolle "Helpdesk" kopieren und Berechtigungen zum Anzeigen des Dashboards oder der Seiten "Filter" hinzufügen.

Wählen Sie die Director-Berechtigungen für die benutzerdefinierte Rolle, die Folgendes enthält:

- Abbrechen von auf Maschine ausgeführter Anwendung erzwingen
- Abbrechen von auf Maschine ausgeführtem Prozess erzwingen
- Remoteunterstützung für Maschine ausführen
- Vorgang zum Zurücksetzen von vDisk ausführen
- Benutzerprofile zurücksetzen
- Anzeigen der Seite "Clientdetails"
- Anzeigen der Seite "Dashboard"
- Anzeigen der Seite "Filter"
- Anzeigen der Seite "Maschinendetails"
- Anzeigen der Seite "Trends"
- Anzeigen der Seite "Benutzerdetails"

In diesem Beispiel ist das Spiegeln (Remoteunterstützung für Maschine ausführen) deaktiviert.



Aus der Liste der Berechtigungen für andere Komponenten sollten Sie zusätzlich folgende Berechtigungen berücksichtigen:

- Bereitstellungsgruppen:
  - Aktivieren/Deaktivieren des Wartungsmodus einer Maschine mit der Bereitstellungsgruppenmitgliedschaft
  - Ausführen von Energievorgängen auf Windows-Desktopmaschinen mit der Bereitstellungsgruppenmitgliedschaft
  - Ausführen der Sitzungsverwaltung auf Maschinen unter mit der Bereitstellungsgruppenmitgliedschaft

# Konfigurieren von Berechtigungen für VDAs vor XenDesktop 7

May 10, 2016

Wenn Benutzer auf ihren Geräten VDAs einer Version vor XenDesktop 7 installiert haben, ergänzt Director Informationen aus der Bereitstellung durch Echtzeitstatus und -metrik über Windows-Remoteverwaltung (WinRM).

Darüber hinaus können Sie mit dem hier beschriebenen Verfahren WinRM für die Verwendung mit Remote PC in XenDesktop 5.6 Feature Pack1 konfigurieren.

Standardmäßig verfügen nur lokale Administratoren des Desktopcomputers (in der Regel Domänenadministratoren und andere privilegierte Benutzer) über die Berechtigungen, die zum Anzeigen der Echtzeitdaten erforderlich sind.

Informationen zum Installieren und Konfigurieren von WinRM finden Sie unter [CTX125243](#).

Um anderen Benutzern das Anzeigen von Echtzeitdaten zu ermöglichen, müssen Sie diesen Berechtigungen erteilen. Beispiel: Angenommen, es gibt mehrere Director-Benutzer (HelpDeskUserA, HelpDeskUserB usw.), die zu einer Active Directory-Sicherheitsgruppe namens "HelpDeskUsers" gehören. Der Gruppe wurde in Studio die Helpdeskadministratorrolle mit den erforderlichen Delivery Controller-Berechtigungen zugewiesen. Die Gruppe benötigt jedoch auch Zugriff auf Informationen vom Desktopcomputer.

Sie haben zwei Möglichkeiten, die Berechtigungen für den erforderlichen Zugriff zu konfigurieren:

- Erteilen von Berechtigungen für die Director-Benutzer (Identitätswechselmodell)
- Erteilen von Berechtigungen für den Director-Dienst (Modell mit vertrauenswürdigen Subsystem)

## **So erteilen Sie Berechtigungen für die Director-Benutzer (Identitätswechselmodell)**

Director verwendet standardmäßig ein Identitätswechselmodell: Die WinRM-Verbindung mit der Desktopmaschine wird mit der Identität des Director-Benutzers hergestellt. Aus diesem Grund muss der Benutzer über die erforderlichen Berechtigungen auf dem Desktop verfügen.

Sie können diese Berechtigungen mit einer der folgenden beiden Methoden konfigurieren (nachfolgend beschrieben):

1. Hinzufügen von Benutzern zur lokalen Administratorgruppe auf der Desktopmaschine.
2. Erteilen der von Director benötigten spezifischen Berechtigungen für Benutzer. Bei dieser Option wird vermieden, dass Director-Benutzer (beispielsweise die HelpDeskUsers-Gruppe) Volladministratorrechte auf der Maschine erhalten.

## **So erteilen Sie Berechtigungen für den Director-Dienst (Modell mit vertrauenswürdigen Subsystem)**

Anstatt den Director-Benutzern Berechtigungen für die Desktopmaschinen zu erteilen, können Sie Director so konfigurieren, dass WinRM-Verbindungen mit einer Dienstidentität hergestellt werden, und ausschließlich dieser Dienstidentität die entsprechenden Berechtigungen erteilen.

Bei diesem Modell sind die Benutzer von Director nicht berechtigt, selbst WinRM-Aufrufe zu tätigen. Sie können nur mit Director auf die Daten zugreifen.

Der Director-Anwendungspool in IIS ist für die Ausführung als Dienstidentität konfiguriert. Dies ist standardmäßig das virtuelle Konto "APPPOOL\Director". Beim Herstellen von Remoteverbindungen wird dieses Konto als das Active Directory-Computerkonto des Servers angezeigt, z. B. MyDomain\DirectorServer\$. Sie müssen dieses Konto mit den entsprechenden

Berechtigungen konfigurieren.

Wenn mehrere Director-Websites bereitgestellt werden, müssen Sie das Computerkonto jedes Webserver in eine Active Directory-Sicherheitsgruppe setzen, die entsprechende Berechtigungen hat.

Um Director so einzustellen, dass anstelle der Benutzeridentität die Dienstidentität für WinRM verwendet wird, konfigurieren Sie die folgende Einstellung, wie unter [Erweiterte Konfiguration](#) beschrieben:

Service.Connector.WinRM.Identity = Service

Sie haben zwei Möglichkeiten, diese Berechtigungen zu konfigurieren:

1. Hinzufügen des Dienstkontos zur lokalen Administratorgruppe auf der Desktopmaschine.
2. Erteilen Sie dem Dienstkonto die für Director erforderlichen Berechtigungen (siehe weiter unten). So kann eine Erteilung von Volladministratorrechten für das Dienstkonto auf der Maschine vermieden werden.

### **So weisen Sie Berechtigungen einem bestimmten Benutzer oder einer bestimmten Gruppe zu**

Die folgenden Berechtigungen sind erforderlich, damit Director auf die von der Desktopmaschine benötigten Informationen über WinRM zugreifen kann:

- Lese- und Ausführberechtigungen in WinRM RootSDDL
- WMI-Namespace-Berechtigungen:
  - root/cimv2 – Remotezugriff
  - root/citrix – Remotezugriff
  - root/RSOP – Remotezugriff und Ausführen
- Zugehörigkeit zu diesen lokalen Gruppen:
  - Systemmonitorbenutzer
  - Ereignisprotokollleser

Das ConfigRemoteMgmt.exe-Tool, das zum automatischen Erteilen dieser Berechtigungen verwendet wird, befindet sich auf dem Installationsmedium in den Ordnern x86\Virtual Desktop Agent und x64\Virtual Desktop Agent und auf dem Installationsmedium im Ordner "tools". Sie müssen allen Director-Benutzern Berechtigungen erteilen.

Um einer Active Directory-Sicherheitsgruppe, einem Benutzer oder einem Computerkonto Berechtigungen zu erteilen, oder um Aktionen auszuführen wie "Anwendung beenden" und "Prozess beenden", führen Sie das Tool mit Administratorrechten an einer Eingabeaufforderung mit den folgenden Argumenten aus:

ConfigRemoteMgmt.exe /configwinrmuser domain\name

wobei name eine Sicherheitsgruppe, ein Benutzer oder ein Computerkonto ist.

So erteilen Sie einer Benutzersicherheitsgruppe die erforderlichen Berechtigungen:

ConfigRemoteMgmt.exe /configwinrmuser domain\HelpDeskUsers

So erteilen Sie einem bestimmten Computerkonto Berechtigungen:

ConfigRemoteMgmt.exe /configwinrmuser domain\DirectorServer\$

Für die Aktionen "Prozess beenden", "Anwendung beenden" und "Spiegeln":

ConfigRemoteMgmt.exe /configwinrmuser domain\name /all

So erteilen Sie einer Benutzergruppe Berechtigungen:

ConfigRemoteMgmt.exe /configwinrmuser domain\HelpDeskUsers /all



So zeigen Sie die Hilfe für das Tool an:

ConfigRemoteMgmt.exe

# Konfigurieren von HDX Insight

May 10, 2016

Hinweis: Die Verfügbarkeit dieser Funktion richtet sich nach der Lizenzierung der Organisation und den Administratorberechtigungen.

HDX Insight ist die Integration der EdgeSight-Netzwerkanalyse und der EdgeSight-Leistungsverwaltung in Director:

- Die EdgeSight-Netzwerkanalyse nutzt HDX Insight und stellt eine kontextbezogene Ansicht der Anwendungen und Desktops im Netzwerk bereit. Mit diesem Feature bietet Director eine erweiterte Analyse des ICA-Datenverkehrs in der Bereitstellung.
- Die EdgeSight-Leistungsverwaltung bietet eine Verlaufsspeicherung und Trendberichte. Anhand der Beibehaltung historischer Daten können Sie im Gegensatz zur Echtzeitbewertung Trendberichte über Kapazität und Integrität usw. erstellen.

Nachdem Sie dieses Feature in Director aktivieren, liefert HDX Insight zusätzliche Informationen an Director:

- Auf der Seite Trends werden bereitstellungsübergreifend Auswirkungen auf Latenz und Bandbreite für Anwendungen, Desktops und Benutzer angezeigt.
- Auf der Seite Benutzerdetails werden Latenz- und Bandbreiteninformationen zu spezifischen Benutzersitzungen angezeigt.

## Einschränkungen

- Die Roundtripzeitdaten für die ICA-Sitzung werden in Receiver für Windows 3.4 oder höher und Receiver für Mac 11.8 oder höher richtig angezeigt. Bei früheren Versionen von Receiver werden die Daten nicht richtig angezeigt.
- In der Trendansicht werden Anmeldedaten für HDX-Verbindungen für VDAs vor Version 7 nicht gesammelt. Für frühere VDAs werden die Diagrammdaten als 0 angezeigt.

## So konfigurieren Sie die EdgeSight-Netzwerkanalyse in Director

EdgeSight bietet eine Netzwerkanalyse unter Nutzung von NetScaler HDX Insight, anhand derer Administratoren der Citrix Anwendungen und Desktops Probleme behandeln und korrelieren können, die auf eine schlechte Netzwerkleistung zurückzuführen sind.

NetScaler Insight Center muss in Director installiert und konfiguriert sein, damit die EdgeSight-Netzwerkanalyse aktiviert werden kann. Insight Center ist eine virtuelle Maschine (Gerät), die von Citrix.com heruntergeladen wird. Mit der EdgeSight-Netzwerkanalyse sammelt Director Daten zur Bereitstellung. Diese Informationen werden aus HDX Insight verwendet, das eine stabile Analyse des Citrix ICA-Protokolls zwischen dem Client und der Citrix Back-End-Infrastruktur bietet.

1. Suchen Sie auf dem Server, auf dem Director installiert ist, das Befehlszeilentool DirectorConfig in C:\inetpub\wwwroot\Director\tools und führen Sie es mit dem Parameter /confignetscaler an der Eingabeaufforderung aus.
2. Wenn Sie dazu aufgefordert werden, konfigurieren Sie den Maschinennamen (FQDN oder IP-Adresse) für NetScaler Insight Center sowie den Benutzernamen, das Kennwort und den HTTP- oder HTTPS-Verbindungstyp.
3. Melden Sie sich zum Prüfen der Änderungen ab und wieder an.

# Erweiterte Konfiguration

May 10, 2016

Dieser Abschnitt enthält folgende Themen:

[Unterstützen von Benutzern in mehreren Active Directory-Domänen und -Gesamtstrukturen](#)

[Hinzufügen von Sites zu Director](#)

[Deaktivieren der Sichtbarkeit von ausgeführten Anwendungen im Aktivitätsmanager](#)

Einige Elemente der erweiterten Director-Konfigurationen, z. B. die Unterstützung mehrerer Sites oder mehrerer Active Directory-Gesamtstrukturen, wird durch die Einstellungen im Internetinformationsdienste-Manager (IIS) gesteuert. Wichtig: Wenn Sie eine Einstellung in IIS ändern, wird der Director-Dienst automatisch neu gestartet und die Benutzer werden abgemeldet.

So konfigurieren Sie erweiterte Einstellungen mit IIS

1. Öffnen Sie die IIS-Verwaltungskonsole.
2. Wechseln Sie zur Director-Website unter der Standardwebsite.
3. Doppelklicken Sie auf **Anwendungseinstellungen**.
4. Doppelklicken Sie auf eine Einstellung, um diese zu bearbeiten.

Platinum-Lizenzen behalten Daten standardmäßig 90 Tage lang bei. Weitere Informationen zu Konfigurationen finden Sie unter [Datengranularität und -beibehaltung](#).

Director sucht in Active Directory nach Benutzern und nach zusätzlichen Benutzer- und Maschineninformationen. Standardmäßig durchsucht Director die folgende Domäne oder Gesamtstruktur:

- In der das Konto des Administrators Mitglied ist
- In der der Director-Webserver Mitglied ist (falls unterschiedlich)

Director versucht, Suchen auf Gesamtstrukturebene mit dem globalen Active Directory-Katalog durchzuführen. Wenn der Administrator keine Berechtigungen zum Suchen auf der Gesamtstrukturebene hat, wird nur die Domäne durchsucht.

Für die Suche nach Daten aus einer anderen Active Directory-Domäne oder Gesamtstrukturebene müssen Sie explizit die zu durchsuchenden Domänen oder Gesamtstrukturen festlegen. Konfigurieren Sie die folgenden Einstellungen:

Connector.ActiveDirectory.Domains = (Benutzer),(Server)

Die Werte der Attribute Benutzer und Server stellen die Domänen des Director-Benutzers (Administrator) bzw. des Director-Servers dar.

Um Suchen von einer zusätzlichen Domäne oder Gesamtstruktur zu ermöglichen, fügen Sie, wie in diesem Beispiel gezeigt, den Namen der Domäne der Liste hinzu:

Connector.ActiveDirectory.Domains = (user),(server),,

Director versucht, Suchen für jede Domäne in der Liste auf der Gesamtstrukturebene durchzuführen. Wenn der Administrator keine Berechtigungen zum Suchen auf der Gesamtstrukturebene hat, wird nur die Domäne durchsucht.

Wenn Director bereits installiert ist, richten Sie das Programm für die Arbeit mit mehreren Sites ein. Verwenden Sie hierzu die IIS-Manager-Konsole auf jedem Director-Server zum Aktualisieren der Liste der Serveradressen in den Anwendungseinstellungen.

Fügen Sie folgender Einstellung eine Controller-Adresse von jedem Standort hinzu:

`Service.AutoDiscoveryAddresses = SiteAController,SiteBController`

wobei SiteAController und SiteBController die Adressen von XenDesktop-Controllern zweier verschiedener Sites sind.

Fügen Sie bei XenApp 6.5-Sites der folgenden Einstellung eine Controller-Adresse von den jeweiligen XenApp-Farmen hinzu:

`Service.AutoDiscoveryAddressesXA = FarmAController,FarmBController`

wobei FarmAController und FarmBController die Adressen von XenApp-Controllern von zwei verschiedenen Farmen sind.

Dies ist eine weitere Methode für XenApp 6.5-Sites, einen Controller von einer XenApp-Farm hinzuzufügen:

`DirectorConfig.exe /xenapp FarmControllerName`

Standardmäßig wird im Aktivitätsmanager von Director eine Liste aller ausgeführten Anwendungen für die Benutzersitzung angezeigt. Diese Informationen können von allen Administratoren angezeigt werden, die Zugriff auf den Aktivitätsmanager in Director haben. Bei delegierten Administratorrollen sind dies Volladministratoren, Bereitstellungsgruppenadministratoren und Helpdeskadministratoren.

Zum Schutz der Privatsphäre im Hinblick auf Benutzer und die von ihnen ausgeführten Anwendungen können Sie die Auflistung der ausgeführten Anwendungen auf der Registerkarte "Anwendungen" deaktivieren.

**Achtung:** Eine unsachgemäße Bearbeitung der Registrierung kann schwerwiegende Probleme verursachen und eine Neuinstallation des Betriebssystems erforderlich machen. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine unsachgemäße Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr. Sichern Sie die Registrierung auf jeden Fall vor dem Bearbeiten ab.

1. Ändern Sie für den VDA den Registrierungsschlüssel in `HKLM\Software\Citrix\Director\TaskManagerDataDisplayed`.

Standardmäßig ist dieser Schlüssel auf 1 eingestellt. Ändern Sie den Wert auf 0, was bedeutet, dass die Informationen nicht im Aktivitätsmanager angezeigt werden.

2. Bearbeiten Sie auf dem Server, auf dem Director installiert ist, die Einstellung zur Steuerung der Sichtbarkeit ausgeführter Anwendungen. In der Standardeinstellung ist der Wert "Wahr", wodurch die Sichtbarkeit der ausgeführten Anwendungen auf der Registerkarte Anwendungen zugelassen wird. Ändern Sie den Wert in "Falsch", wodurch die Sichtbarkeit deaktiviert wird. Diese Option gilt nur für den Aktivitätsmanager in Director, nicht für den VDA.

Ändern Sie den Wert der folgenden Einstellung:

`UI.TaskManager.EnableApplications = false`

**Wichtig:** Zum Deaktivieren der Ansicht ausgeführter Anwendungen empfiehlt Citrix, dass beide Änderungen durchgeführt werden, damit die Daten im Aktivitätsmanager nicht angezeigt werden.

# Überwachen von Bereitstellungen

Jan 05, 2017

Wenn Sie Director mit Volladministratorrechten öffnen, erscheint das Dashboard zur Überwachung der Integrität und Nutzung einer Site.

Wenn es zurzeit keine Fehler gibt und keine Fehler in den letzten 60 Minuten aufgetreten sind, bleiben Bereiche ausgeblendet. Wenn Fehler auftreten, wird der zugehörige Fehlerbereich automatisch angezeigt.

Hinweis: Je nachdem, über welche Lizenzierung Ihre Organisation verfügt und welche Administratorrechte vorliegen, stehen einige Optionen oder Features möglicherweise nicht zur Verfügung.

Bereich	Beschreibung
Benutzerverbindungsfehler	Verbindungsfehler während der letzten 60 Minuten. Klicken Sie auf die Kategorien neben der Gesamtzahl zum Anzeigen von Metriken für diesen Fehlertyp. In der nebenstehenden Tabelle wird angezeigt, wie sich dieser Wert auf die Bereitstellungsgruppen verteilt.
Fehlgeschlagene Desktopbetriebssystemmaschinen bzw. Fehlgeschlagene Serverbetriebssystemmaschinen	Gesamtanzahl der Fehler in den letzten 60 Minuten unterteilt nach Bereitstellungsgruppen. Fehler unterteilt nach Typ, einschließlich "konnte nicht gestartet werden", "beim Starten hängen geblieben" und "nicht registriert". Bei Serverbetriebssystemmaschinen wird auch das Erreichen der maximalen Last angegeben.
Lizenzierungsstatus	• Lizenzserverwarnungen werden vom Lizenzserver gesendet und enthalten für eine Auflösung der jeweiligen Warnung erforderlichen Aktionen.    • Delivery Controller-Warnungen enthalten vom Controller erfasste Zustandsangaben zur Lizenzierung und werden vom Delivery Controller gesendet.      Sie können den Schwellenwert für Warnungen in Studio festlegen.   Lizenzserver- bzw. Delivery Controller-Warnungen werden nicht angezeigt, wenn die Lizenzserverversion älter ist 11.12.1 bzw. wenn die Delivery Controller-Version älter ist als XenApp 7.6 oder XenDesktop 7.6.
Verbundene Sitzungen	Verbunden Sitzungen in allen Bereitstellungsgruppen in den letzten 60 Minuten.
Durchschnittliche Anmeldedauer	Anmeldedaten für die letzten 60 Minuten. Die große Zahl links ist die durchschnittliche Anmeldedauer während einer Stunde. Anmeldedaten für VDAs vor XenDesktop 7.0 sind nicht in diesem Durchschnitt enthalten.
Infrastruktur	Integritätsstatus der Hosts, Controller und Infrastruktur. Anzeige von Leistungswarnungen. Für Hosts werden der Verbindungsstatus sowie der Zustand der CPU,

Bereich	Beschreibung
	Arbeitsspeicher, Bandbreite (Netzwerkverwendung) und Speicher (Datenträgernutzung) anhand von Informationen von XenServer oder VMware überwacht.   Sie können beispielsweise XenCenter so konfigurieren, dass Warnungen zur Leistung generiert werden, wenn die CPU-, Netzwerk-E/A- oder Datenträger-E/A-Nutzung einen angegebenen Schwellenwert auf einem verwalteten Server oder einer virtuellen Maschine übersteigt. Standardmäßig ist das Warnungswiederholungsintervall 60 Minuten, Sie können jedoch auch eine andere Einstellung wählen. Weitere Informationen finden Sie in der XenServer-Dokumentation unter <a href="#">Konfigurieren von Leistungswarnungen</a>.

Hinweis: Wird für eine bestimmte Metrik kein Symbol angezeigt, bedeutet dies, dass diese Metrik von dem verwendeten Hosttyp nicht unterstützt wird. Beispiel: Für System Center Virtual Machine Manager-, AWS- und CloudStack-Hosts sind keine Integritätsdaten verfügbar.

Fahren Sie mit dem Beheben von Problemen mit den folgenden Optionen (Erläuterung siehe weiter unten) fort:

- Steuern der Energieverwaltung der Benutzermaschine
- Verhindern von Verbindungen mit Maschinen

Wenn eine Sitzung getrennt wird, bleibt sie aktiv und die Anwendungen werden weiter ausgeführt, das Benutzergerät kommuniziert jedoch nicht mehr mit dem Server.

Aktion	Beschreibung
Anzeigen einer zurzeit verbundenen Maschine oder Sitzung des Benutzers	Mit den Ansichten Aktivitäts-Manager und Benutzerdetails zeigen Sie die aktuell verbundene Maschine oder Sitzung des Benutzers an und eine Liste aller Maschinen und Sitzungen, auf die dieser Benutzer zugreifen kann. Klicken Sie auf das Symbol zum Sitzungswechsel in der Titelleiste des Benutzers, um auf diese Liste zuzugreifen. Siehe <a href="#">Wiederherstellen von Sitzungen</a> .
Anzeigen der Gesamtanzahl der verbundenen Sitzungen in allen Bereitstellungsgruppen	Im Dashboard im Bereich Verbundene Sitzungen zeigen Sie die Gesamtzahl der verbundenen Sitzungen in allen Bereitstellungsgruppen in den letzten 60 Minuten an. Wenn Sie anschließend auf die Gesamtzahl klicken, wird die Ansicht Filter angezeigt, in der Sie die grafischen Sitzungsdaten basierend auf ausgewählten Bereitstellungsgruppen und Bereichen und Nutzung von Bereitstellungsgruppen anzeigen.
Anzeigen der Daten über einen längeren Zeitraum	Klicken Sie in der Ansicht Trends auf die Registerkarte Sitzungen, um detailliertere Nutzungsdaten für verbundene und getrennte Sitzungen über einen längeren Zeitraum anzuzeigen (d. h. Sitzungsgesamtzahl, die länger als 60 Minuten zurückliegt). Klicken Sie zum Anzeigen dieser Informationen auf Verlaufstrends anzeigen.

Hinweis: Wenn auf dem Benutzergerät eine ältere Virtual Delivery Agent-Version ausgeführt wird, z. B. eine VDA-Version vor 7, kann Director keine vollständigen Sitzungsinformationen anzeigen. Stattdessen wird eine Meldung angezeigt, dass die Informationen in der Ansicht Benutzerdetails und Aktivitäts-Manager nicht verfügbar sind.

Wenn Sie auf Zahlen im Dashboard klicken oder im Menü Filter einen vordefinierten Filter auswählen, wird die Ansicht Filter geöffnet und zeigt Daten für die ausgewählte Maschine oder den Fehlertyp an.

Vordefinierten Filter können nicht bearbeitet werden. Sie können einen vordefinierten Filter jedoch als benutzerdefinierten Filter speichern und dann bearbeiten. Sie können auch benutzerdefinierten Ansichten mit Filter für Maschinen, Verbindungen und Sitzungen für alle Bereitstellungsgruppen erstellen.

1. Wählen Sie eine Ansicht aus:

- **Maschinen:** Wählen Sie Desktopbetriebssystemmaschinen oder Serverbetriebssystemmaschinen. Diese Ansicht zeigt die Anzahl der konfigurierten Computer. Die Registerkarte "Serverbetriebssystemmaschinen" enthält auch den Lastauswertungsprogrammindex, der die Verteilung der Leistungsindikatoren und Quickinfo der Sitzungsanzahl angibt, wenn Sie mit dem Mauszeiger auf den Link zeigen.
- **Sitzungen:** Sie können die Sitzungsanzahl auch in der Ansicht "Maschinen" anzeigen.
- **Verbindungen:** Filtern Sie Verbindungen nach verschiedenen Zeiträumen, u. a. die letzten 60 Minuten, die letzten 24 Stunden, oder die letzten 7 Tage.

2. Wählen Sie für "Fehler" die Kriterien aus.

3. Verwenden Sie die zusätzlichen Registerkarten für jede Ansicht ggf. zum Abschließen des Filters.

4. Wählen Sie zusätzliche Spalten bei Bedarf aus, um weitere Fehler zu beheben.

5. Speichern und benennen Sie den Filter.

Wenn sie den Filter später öffnen möchten, wählen Sie im Menü Filter den Filtertyp (Maschinen, Sitzungen oder Verbindungen) und dann den gespeicherten Filter.

6. Verwenden Sie u. U. für die Ansichten "Maschinen" oder "Verbindungen" die Energiesteuerelemente für alle in der gefilterten Liste ausgewählten Maschinen. Fehlerursachen und empfohlene Aktionen zum Beheben von Maschinen- und Verbindungsfehlern sind im folgenden Dokument aufgeführt: [Citrix Director 7.6 Failure Reasons Troubleshooting Guide](#).

7. Verwenden Sie in der Ansicht "Sitzungen" die Sitzungssteuerelemente oder die Option zum Senden von Nachrichten.

Fahren Sie mit dem Beheben von Problemen mit den folgenden Optionen (Erläuterung siehe weiter unten) fort:

- Steuern der Energieverwaltung der Benutzermaschine
- Verhindern von Verbindungen mit Maschinen

Die Ansicht "Trends" zeigt Verlaufstrenddaten für Sitzungen, Verbindungsfehler, Maschinenfehler, Anmeldeleistung und Lastauswertung für jede Site an. Klicken Sie zur Anzeige dieser Informationen im Dashboard oder in der Anzeige "Filter" auf Trends.

Das Vergrößerungsfeature beim Drilldown ermöglicht Ihnen das Navigieren durch Trenddiagramme, indem Sie bestimmte Zeiträume vergrößern (durch Klicken auf einen Datenpunkt im Diagramm) und die Detailinformationen zum Trend anzeigen. Durch dieses Feature können Sie die genauen Auswirkungen der angezeigten Trends besser verstehen.

Wenden Sie einen anderen Filter auf die Daten an, um den Standardgeltungsbereich der einzelnen Diagramme zu ändern.

Aktion	Beschreibung
Graphdaten exportieren	Wählen Sie die Registerkarte aus, die die Daten enthält, die Sie exportieren möchten. Klicken Sie auf "Exportieren" und wählen Sie das Format: PDF oder CVS.

Trends für Sitzungen anzeigen	Wählen Sie auf der Registerkarte "Sitzungen" die Bereitstellungsgruppe und den Zeitraum aus, um weitere Informationen zur Anzahl gleichzeitiger Sitzungen anzuzeigen.
Trends für Verbindungsfehler anzeigen	Wählen Sie auf der Registerkarte "Verbindungsfehler" den Maschinentyp, den Fehlertyp, die Bereitstellungsgruppe und den Zeitraum, um weitere Informationen über die Verbindungsfehler der Site anzuzeigen.
Trends für Maschinenfehler anzeigen	Wählen Sie auf der Registerkarte "Desktopbetriebssystemmaschinenfehler" bzw. "Serverbetriebssystemmaschinen" den Fehlertyp, die Bereitstellungsgruppe und den Zeitraum, um weitere Informationen über die Maschinenfehler der Site anzuzeigen.
Trends für die Anmeldeleistung anzeigen	Wählen Sie auf der Registerkarte "Anmeldeleistung" die Bereitstellungsgruppe und den Zeitraum, um ein Diagramm mit ausführlichen Informationen über die Dauer der Benutzeranmeldungen bei der Site und wie sich die Anzahl der Anmeldungen auf die Leistung auswirkt, anzuzeigen. In dieser Ansicht wird auch die durchschnittliche Dauer der Anmeldephasen angezeigt, u. a. Brokeringdauer, VM-Startzeit usw. Diese Daten beziehen sich speziell auf Benutzeranmeldungen und nicht auf Benutzer, die sich mit getrennten Sitzungen wiederverbinden.
Trends für die Lastauswertung anzeigen	Die Registerkarte "Lastauswertungsindex" bietet ein Diagramm, das ausführliche Informationen zur Last enthält, die auf die Serverbetriebssystemmaschinen verteilt ist. Als Filteroptionen für dieses Diagramm stehen Bereitstellungsgruppe oder Serverbetriebssystemmaschine in einer Bereitstellungsgruppe, Serverbetriebssystemmaschine (nur bei Auswahl von Serverbetriebssystemmaschine in einer Bereitstellungsgruppe) und Bereich zur Verfügung.
Anzeigen der Nutzung gehosteter Anwendungen	Die Verfügbarkeit dieser Funktion hängt von den Lizenzen des Unternehmens ab. Wählen Sie auf der Registerkarte "Verwendung gehosteter Anwendungen" die Bereitstellungsgruppe und den Zeitraum, um eine Kurve der höchsten gleichzeitigen Nutzung sowie eine Tabelle mit der anwendungsbasierten Verwendung anzuzeigen. In der Tabelle "Anwendungsbasierte Verwendung" können Sie eine bestimmte Anwendung auswählen, um Details und eine Liste der Benutzer anzuzeigen, die die Anwendung verwenden oder verwendet haben.
Nutzung virtueller Maschinen anzeigen	Wählen Sie auf der Registerkarte "Nutzung" die Option "Desktopbetriebssystemmaschinen" oder "Serverbetriebssystemmaschinen", um einen Überblick über die Nutzung der VMs in Echtzeit zu erhalten, sodass Sie den Kapazitätsbedarf der Site schnell einschätzen können. Verfügbarkeit der Desktopbetriebssysteme: Zeigt den aktuellen Zustand der Desktopbetriebssystemmaschinen (VDIs) nach Verfügbarkeit für die gesamte Site oder für bestimmte Bereitstellungsgruppen an.  Verfügbarkeit der Serverbetriebssysteme: Zeigt den aktuellen Zustand der Serverbetriebssystemmaschinen nach Verfügbarkeit für die gesamte Site oder für bestimmte Bereitstellungsgruppen an.
Netzwerkanalysedaten mit HDX Insight	Die Verfügbarkeit dieser Funktion richtet sich nach der Lizenzierung der Organisation und den Administratorberechtigungen.



anzeigen	Überwachen Sie auf der Registerkarte "Netzwerk" die Netzwerkanalyse, die eine kontextbezogene Ansicht der Benutzer, Anwendungen und Desktops im Netzwerk bereitstellt. Director bietet mit diesem Feature eine erweiterte Analyse des ICA-Datenverkehrs in der Bereitstellung.
----------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Die Flag-Symbole auf dem Diagramm weisen auf wichtige Ereignisse oder Aktionen für diesen Zeitraum hin. Bewegen Sie den Mauszeiger über das Flag und klicken Sie, um Ereignisse und Aktionen aufzulisten.

Hinweis: Anmeldedaten für HDX-Verbindungen werden für VDAs vor Version 7 nicht gesammelt. Für frühere VDAs werden die Diagrammdaten als 0 angezeigt.

Fahren Sie mit dem Beheben von Problemen mit den folgenden Optionen (Erläuterung siehe weiter unten) fort:

- Steuern der Energieverwaltung der Benutzermaschine
- Verhindern von Verbindungen mit Maschinen

Die Ansicht Trends zeigt Verlaufstrenddaten für Sitzungen, Verbindungsfehler, Maschinenfehler, Anmeldeleistung und Lastauswertung für jede Site an. Klicken Sie zur Anzeige dieser Informationen im Dashboard oder in der Anzeige Filter auf Trends.

Wenden Sie einen anderen Filter auf die Daten an, um den Standardgeltungsbereich der einzelnen Diagramme zu ändern.

Aktion	Beschreibung
Graphdaten exportieren	Wählen Sie die Registerkarte aus, die die Daten enthält, die Sie exportieren möchten. Klicken Sie auf Exportieren und wählen Sie das Format: PDF oder CVS.
Trends für Sitzungen anzeigen	Wählen Sie auf der Registerkarte Sitzungen die Bereitstellungsgruppe und den Zeitraum aus, um weitere Informationen zur Anzahl gleichzeitiger Sitzungen anzuzeigen.
Trends für Verbindungsfehler anzeigen	Wählen Sie auf der Registerkarte Verbindungsfehler den Maschinentyp, den Fehlertyp, die Bereitstellungsgruppe und den Zeitraum, um weitere Informationen über die Verbindungsfehler der Site anzuzeigen.
Trends für Maschinenfehler anzeigen	Wählen Sie auf der Registerkarte Desktopbetriebssystemmaschinenfehler bzw. Serverbetriebssystemmaschinen den Fehlertyp, die Bereitstellungsgruppe und den Zeitraum, um weitere Informationen über die Maschinenfehler der Site anzuzeigen.
Trends für die Anmeldeleistung anzeigen	Wählen Sie auf der Registerkarte Anmeldeleistung die Bereitstellungsgruppe und den Zeitraum, um ein Diagramm mit ausführlichen Informationen über die Dauer der Benutzeranmeldungen bei der Site und wie sich die Anzahl der Anmeldungen auf die Leistung auswirkt, anzuzeigen. In dieser Ansicht wird auch die durchschnittliche Dauer der Anmeldephasen angezeigt, u. a. Brokeringdauer, VM-Startzeit usw. Diese Daten beziehen sich speziell auf Benutzeranmeldungen und nicht auf Benutzer, die sich mit getrennten Sitzungen wiederverbinden.
Trends für die	Die Registerkarte Lastauswertungsindex bietet ein Diagramm, das ausführliche

Aktion	Beschreibung
Lastauswertung anzeigen	Informationen zur Last enthält, die auf die Serverbetriebssystemmaschinen verteilt ist. Als Filteroptionen für dieses Diagramm stehen Bereitstellungsgruppe oder Serverbetriebssystemmaschine in einer Bereitstellungsgruppe, Serverbetriebssystemmaschine (nur bei Auswahl von Serverbetriebssystemmaschine in einer Bereitstellungsgruppe) und Bereich zur Verfügung.
Anzeigen der Nutzung gehosteter Anwendungen	Die Verfügbarkeit dieser Funktion hängt von den Lizenzen des Unternehmens ab. Wählen Sie auf der Registerkarte Verwendung gehosteter Anwendungen die Bereitstellungsgruppe und den Zeitraum, um eine Kurve der höchsten gleichzeitigen Nutzung sowie eine Tabelle mit der anwendungsbasierten Verwendung anzuzeigen. In der Tabelle "Anwendungsbasierte Verwendung" können Sie eine bestimmte Anwendung auswählen, um Details und eine Liste der Benutzer anzuzeigen, die die Anwendung verwenden oder verwendet haben.
Netzwerkanalysedaten mit HDX Insight anzeigen	Die Verfügbarkeit dieser Funktion richtet sich nach der Lizenzierung der Organisation und den Administratorberechtigungen. Überwachen Sie auf der Registerkarte Netzwerk die Netzwerkanalyse, die eine kontextbezogene Ansicht der Benutzer, Anwendungen und Desktops im Netzwerk bereitstellt. Director bietet mit diesem Feature eine erweiterte Analyse des ICA-Datenverkehrs in der Bereitstellung.

Die Flag-Symbole auf dem Diagramm weisen auf wichtige Ereignisse oder Aktionen für diesen Zeitraum hin. Bewegen Sie den Mauszeiger über das Flag und klicken Sie, um Ereignisse und Aktionen aufzulisten.

Hinweis: Anmeldedaten für HDX-Verbindungen werden für VDAs vor Version 7 nicht gesammelt. Für frühere VDAs werden die Diagrammdaten als 0 angezeigt.

Fahren Sie mit dem Beheben von Problemen mit den folgenden Optionen (Erläuterung siehe weiter unten) fort:

- Steuern der Energieverwaltung der Benutzermaschine
- Verhindern von Verbindungen mit Maschinen

Zum Anzeigen der auf einem bestimmten Maschinen-VDA (physisch oder VM) installierten Hotfixes wählen Sie die Ansicht Maschinendetails.

Steuern Sie den Zustand der in Director ausgewählten Maschinen mit den Optionen für die Energieverwaltung. Diese Optionen stehen für Desktopbetriebssystemmaschinen, aber möglicherweise nicht für Serverbetriebssystemmaschinen zur Verfügung.

Hinweis: Diese Funktionen stehen nicht für physische Maschinen zur Verfügung, die Remote-PC-Zugriff verwenden.

Befehl	Funktion
Neu starten	Die VM wird ordnungsgemäß heruntergefahren und alle ausgeführten Prozesse werden einzeln angehalten, bevor die VM neu gestartet wird. Wählen Sie diese Option beispielsweise für den Neustart von Maschinen, die in Director mit "Konnten nicht gestartet werden" ausgewiesen

Befehl	werden. Funktion
<b>Neustart erzwingen</b>	Die VM wird neu gestartet, ohne dass sie heruntergefahren wird. Dieser Befehl funktioniert genauso wie das Trennen des Netzsteckers eines physischen Servers und erneutes Anschließen und Einschalten des Servers.
<b>Herunterfahren</b>	Die VM wird ordnungsgemäß heruntergefahren und alle ausgeführten Prozesse werden einzeln angehalten.
<b>Herunterfahren erzwingen</b>	Die VM wird zwingend heruntergefahren, ohne dass das Verfahren zum Herunterfahren durchgeführt wird. Dieser Befehl funktioniert genauso wie das Trennen des Netzsteckers eines physischen Servers. Es werden möglicherweise nicht immer alle ausgeführten Prozesse heruntergefahren, sodass bei diesem Verfahren die Gefahr von Datenverlust besteht.
<b>Anhalten</b>	Die laufende VM wird im aktuellen Zustand angehalten und dieser Zustand wird in einer Datei im Standardspeicherrepository gespeichert. Diese Option ermöglicht das Herunterfahren der VM auf dem Hostserver und später, nach einem Neustart, die Wiederaufnahme der VM mit dem ursprünglichen Ausführungsstatus.
<b>Fortfahren</b>	Nimmt eine angehaltene VM wieder auf und stellt den ursprünglichen Ausführungsstatus wieder her.
<b>Starten</b>	Startet eine ausgeschaltete VM.

Sollten die Energieverwaltungsaktionen fehlschlagen, zeigen Sie mit der Maus auf die Warnung und es wird eine Meldung mit Details zum Fehler angezeigt.

Verwenden Sie den Wartungsmodus, um vorübergehend neue Verbindungen zu verhindern, während der entsprechende Administrator Wartungsaufgaben am Image durchführt.

Wenn Sie den Wartungsmodus auf Maschinen aktivieren, werden keine neuen Verbindungen zugelassen, bis Sie ihn wieder deaktivieren. Wenn Benutzer momentan angemeldet sind, wird der Wartungsmodus erst wirksam, sobald alle Benutzer abgemeldet sind. Benutzern, die sich nicht abmelden, müssen Sie eine Nachricht senden, die sie darüber informiert, dass die Maschine zu einem bestimmten Zeitpunkt heruntergefahren wird. Verwenden Sie die Energieverwaltung, um die Maschinen zwingend herunterzufahren.

1. Wählen Sie die Maschine aus, z. B. auf der Ansicht Benutzerdetails, oder eine Gruppe von Maschinen in der Ansicht Filter.
2. Klicken Sie auf Wartungsmodus und aktivieren Sie die Option.

Wenn ein Benutzer versucht, eine Verbindung zu einem zugewiesenen Desktop herzustellen, während er im Wartungsmodus ist, wird eine Meldung angezeigt, dass der Desktop zurzeit nicht verfügbar ist. Es können keine neuen Verbindungen hergestellt werden, bis der Wartungsmodus deaktiviert wird.

# Behandeln von Benutzerproblemen

May 10, 2016

In der Ansicht **Helpdesk** (Seite **Aktivitäts-Manager**) in Director zeigen Sie Informationen über den Benutzer an:

- Überprüfen Sie die Details zur Anmeldung des Benutzers, zur Verbindung und zu den Anwendungen.
- Spiegeln Sie die Maschine des Benutzers.
- Behandeln Sie das Problem mit den in der folgenden Tabelle empfohlenen Aktionen und eskalieren Sie das Problem ggf. an den entsprechenden Administrator.

## Tipps zur Problembehandlung

Problem des Benutzers	Lösungsvorschläge:
Anmeldung dauert lange oder schlägt periodisch oder wiederholt fehl	<a href="#">Diagnose von Benutzeranmeldeproblemen</a>
Anwendung ist langsam oder reagiert nicht mehr	<a href="#">Beheben von Anwendungsfehlern</a>
Verbindung ist fehlgeschlagen	<a href="#">Wiederherstellen von Desktopverbindungen</a>
Sitzung ist langsam oder reagiert nicht	<a href="#">Wiederherstellen von Sitzungen</a>
Video ist langsam oder von schlechter Qualität	<a href="#">Ausführen von Systemberichten für HDX-Kanäle</a>

Hinweis: Um sicherzustellen, dass der Computer nicht im Wartungsmodus ist, überprüfen Sie in der Ansicht Benutzerdetails den Bereich Maschinendetails.

### Suchtipps

Wenn Sie den Namen des Benutzers im Suchfeld eingeben, sucht Director in Active Directory nach Benutzern in allen Sites, die für Director konfiguriert wurden.

Wenn Sie den Namen einer Maschine, die von mehreren Benutzern verwendet wird, in ein Suchfeld eingeben, zeigt Director die Maschinendetails für die angegebene Maschine an.

Wenn Sie einen Endpunktnamen in ein Suchfeld eingeben, verwendet Director die nicht authentifizierten (anonymen) und die authentifizierten Sitzungen, die mit einem bestimmten Endpunkt verbunden sind, sodass Probleme in nicht authentifizierten Sitzungen behandelt werden können. Stellen Sie sicher, dass Endpunktnamen eindeutig sind, damit die Problembehandlung von nicht authentifizierten Sitzungen durchgeführt werden kann.

Die Suchergebnisse schließen auch Benutzer ein, die derzeit keine Maschine verwenden bzw. keiner Maschine zugewiesen sind.

- Bei der Suche wird die Groß- und Kleinschreibung nicht beachtet.
- Teileinträge ergeben eine Liste möglicher Übereinstimmungen.
- Nachdem Sie einige Buchstaben eines zweiteiligen Namens (Benutzername, Nachname und Vorname oder Anzeigename)

durch Leerzeichen getrennt eingegeben haben, enthalten die Ergebnisse Übereinstimmungen für beide Zeichenfolgen. Wenn Sie beispielsweise `john robertson` eingeben, werden Zeichenfolgen wie "John Robertson" oder "Robert, Jones" als Ergebnisse angezeigt.

Klicken Sie auf das Director-Logo, um zur Startseite zurückzukehren.

### **Hochladen von Problembehandlungsinformationen zum technischen Support von Citrix**

Führen Sie Citrix Scout auf einem Delivery Controller oder VDA aus, um wichtige Datenpunkte und CDF-Traces (Citrix Diagnosis Facility) für die Fehlerbehebung auf ausgewählten Computern zu erfassen. Nach der Erfassung dieser Informationen lädt Scout die Datenpunkte sicher zum technischen Support von Citrix hoch. Die TaaS- (Tools-as-a-Service-)Plattform reduziert anhand dieser Informationen die Zeit zur Lösung der von Kunden gemeldeten Probleme.

Scout wird mit XenApp- bzw. XenDesktop-Komponenten installiert. Scout wird im Windows-Startmenü oder im Startbildschirm von Windows 8 oder 8.1 angezeigt, wenn Sie XenDesktop 7.1, XenDesktop 7.5 oder XenApp 7.5 installieren oder ein Upgrade darauf durchführen.

Zum Starten von Scout über das Startmenü oder den Startbildschirm wählen Sie Citrix > Citrix Scout.

Weitere Informationen zum Verwenden und Konfigurieren von Scout und häufig gestellte Fragen finden Sie unter <http://support.citrix.com/article/CTX130147>.

Im folgenden Video wird die Verwendung von Scout kurz erläutert.

# Spiegeln von Benutzern

May 10, 2016

Mit dem Feature Benutzer spiegeln in Director können Sie die virtuelle Maschine oder Sitzung eines Benutzers direkt anzeigen und damit arbeiten. Der Benutzer muss mit der zu spiegelnden Maschine verbunden sein. Wenn der Benutzer verbunden ist, wird der Name der verbundenen Maschine in der Titelleiste des Benutzers angezeigt.

1. Wählen Sie in der Ansicht Details die Benutzersitzung aus.
2. Aktivieren Sie die Spiegelung für die ausgewählte Benutzersitzung:
  - Klicken Sie zur Maschinenüberwachung in der Ansicht Aktivitäts-Manager auf Spiegeln.
  - Klicken Sie zur Sitzungsüberwachung in der Ansicht Benutzerdetails im Bereich Sitzungsdetails auf Spiegeln.
3. Nach Initialisierung der Verbindung werden Sie in einem Dialogfeld aufgefordert, die Datei .msrcincident zu öffnen oder zu speichern.
4. Öffnen Sie die Vorfalldatei mit dem Remoteunterstützung-Viewer, wenn er nicht standardmäßig ausgewählt ist. Auf dem Benutzergerät wird eine Bestätigungsaufforderung angezeigt.
5. Weisen Sie die Benutzer an, auf Ja zu klicken, um die Maschinen- oder die Sitzungs freigabe zu starten.

Fordern Sie den Benutzer auf, die Tastatur- und Maussteuerung freizugeben, damit Sie die Steuerung übernehmen können.

Richten Sie den Microsoft Internet Explorer-Browser so ein, dass die heruntergeladene Datei zur Microsoft-Remoteunterstützung (.msra) automatisch mit dem Remoteunterstützungsclient geöffnet wird.

Hierzu müssen Sie die Einstellung Automatische Eingabeaufforderung für Dateidownloads im Gruppenrichtlinien-Editor aktivieren:

Computerkonfiguration > Administrative Vorlagen > Windows-Komponenten > Internet Explorer > Internetsystemsteuerung > Sicherheitsseite > Internetzone > Automatische Eingabeaufforderung für Dateidownloads.

Diese Option ist standardmäßig für Sites in der lokalen Intranetzone aktiviert. Wenn die Director-Site nicht zur lokalen Intranetzone gehört, sollten Sie die Site manuell dieser Zone hinzufügen.

# Senden von Nachrichten an Benutzer

May 10, 2016

Sie können über Director eine Nachricht an einen Benutzer senden, der mit einer oder mehreren Maschinen verbunden ist. Sie können beispielsweise mit dieser Funktion sofortige Benachrichtigungen über administrative Aktionen senden, wie bevorstehende Desktopwartung, Abmeldungen bzw. Neustarts von Maschinen und die Rücksetzung von Profilen.

1. Wählen Sie in der Ansicht Aktivitäts-Manager den Benutzer aus und klicken Sie auf Details.
2. Klicken in der Ansicht Benutzerdetails im Bereich Sitzungsdetails auf Nachricht senden.
3. Füllen Sie die Felder Betreff und Nachricht aus und klicken Sie auf Senden.

Wenn die Nachricht gesendet wird, wird in Director eine Bestätigungsmeldung angezeigt. Wenn die Maschine des Benutzers verbunden ist, wird dort eine entsprechende Nachricht angezeigt.

Wenn die Nachricht nicht gesendet wird, wird in Director eine Fehlermeldung angezeigt. Gehen Sie bei der Problembehandlung gemäß der Anweisungen in der Fehlermeldung vor. Geben Sie abschließend den Betreff und Text der Nachricht erneut ein und klicken Sie auf Noch einmal versuchen.

# Diagnose von Benutzeranmeldeproblemen

May 10, 2016

Führen Sie diese allgemeinen Schritte aus:

1. Überprüfen Sie in der Ansicht Benutzerdetails im Bereich Anmeldedauer, welcher Anmeldezustand vorliegt.
  - Wenn Benutzer sich anmelden, wird der Anmeldeprozess in der Ansicht widergespiegelt.
  - Wenn der Benutzer bereits angemeldet ist, wird im Bereich Anmeldedauer angezeigt, wie viel Zeit für die Anmeldung des Benutzers an der aktuellen Sitzung benötigt wurde.
2. Fordern Sie die Benutzer auf, sich abmelden und neu anzumelden, damit Sie die Daten für Anmeldedauer beobachten können. Der Bereich wird ungefähr alle drei Minuten aktualisiert, es kann jedoch länger dauern, abhängig von der Dauer, die für den Abschluss der Anmeldung benötigt wird.
3. Untersuchen Sie die Phasen des Anmeldeprozess:
  - **Brokering**: Die zur Zuweisung des Desktops zum Benutzer benötigte Zeit.
  - **VM-Start**: Die zum Starten des Desktops benötigte Zeit.
  - **HDX-Verbindung**: Die je nach Netzwerk zum Herstellen der HDX-Verbindung benötigte Zeit.
  - **Gruppenrichtlinienobjekte**: Die zum Anwenden der Gruppenrichtlinienobjekte benötigte Zeit.
  - **Anmeldeskripts**: Die für die Skripts benötigte Zeit.
  - **Profilladezeit**: Die zum Laden des Benutzerprofils benötigte Zeit.
  - **Interaktive Sitzung**: Die zum Einrichten einer interaktiven Benutzersitzung benötigte Zeit.

Die Gesamtanmeldedauer ist keine genaue Summe der einzelnen Phasen. Beispiel: Einige Phasen treten parallel auf und in anderen Phasen wird eine zusätzliche Verarbeitung durchgeführt, die zu einer längeren Anmeldedauer als die Summe der einzelnen Phasen führen kann.

Tipp: Um ungewöhnliche oder unerwartete Werte im Diagramm zu finden, vergleichen Sie die in jeder Phase der aktuellen Sitzung benötigte Zeit mit der durchschnittlichen Dauer für diesen Benutzer in den letzten sieben Tagen sowie mit der durchschnittlichen Dauer in den letzten sieben Tagen für alle Benutzer der Bereitstellungsgruppe.

Eskalieren Sie wie erforderlich. Beispiel: Wenn der VM-Start langsam ist, liegt das Problem möglicherweise am Hypervisor, Sie können das Problem also an den Hypervisoradministrator eskalieren. Wenn die Brokeringdauer zu lang ist, können Sie das Problem dem Siteadministrator melden, damit der Lastausgleich auf dem Delivery Controller überprüft wird.

**Tipps zur Problembehandlung:** Überprüfen Sie ungewöhnliche Unterschiede, u. a.:

- Fehlende (aktuelle) Anmeldeleiten
- Große Abweichung zwischen der aktuellen und der durchschnittlichen Dauer für diesen Benutzer. Mögliche Ursachen:
  - Es wurde eine neue Anwendung installiert.
  - Das Betriebssystem wurde aktualisiert.
  - Es wurden Konfigurationsänderungen vorgenommen.
- Große Abweichung zwischen den Anmeldewerten des Benutzers (aktuelle und durchschnittliche Dauer) und der durchschnittlichen Dauer der Bereitstellungsgruppe.

Klicken Sie u. U. auf Neu starten, um den Anmeldeprozess des Benutzers zu beobachten und Probleme zu beheben, z. B. VM-Start oder Brokering.



# Beheben von Anwendungsfehlern

May 10, 2016

Klicken Sie in der Ansicht Aktivitäts-Manager auf die Registerkarte Anwendungen. Sie können alle Anwendungen auf allen Maschinen anzeigen, auf die dieser Benutzer zugreifen kann, einschließlich der lokalen und der gehosteten Anwendungen für die derzeit verbundene Maschine und den aktuellen Status der einzelnen Maschine.

Hinweis: Wenn die Registerkarte Anwendungen abgeblendet ist, wenden Sie sich an einen Administrator, der die Berechtigung hat, die Registerkarte zu aktivieren.

Die Liste enthält nur die Anwendungen, die in der Sitzung gestartet wurden.

Für Serverbetriebssystemmaschinen und Desktopbetriebssystemmaschinen werden Anwendungen für jede getrennte Sitzung angezeigt. Wenn der Benutzer nicht verbunden ist, werden keine Anwendungen angezeigt.

Aktion	Beschreibung
Beenden Sie die Anwendung, die nicht reagiert.	Wählen Sie die Anwendung aus, die nicht reagiert, und klicken Sie auf Anwendung beenden. Wenn die Anwendung beendet ist, fordern Sie den Benutzer auf, sie erneut zu starten.
Beenden Sie Prozesse, die nicht reagieren.	Wenn Sie die erforderlichen Berechtigungen haben, klicken Sie auf die Registerkarte Prozesse. Wählen Sie einen Prozess aus, der mit dieser Anwendung zusammenhängt oder der viele CPU-Ressourcen oder viel Speicher verbraucht, und klicken Sie auf Prozess beenden. Wenn Sie nicht die erforderlichen Berechtigungen zum Beenden des Prozesses haben, schlägt das Beenden fehl.
Starten Sie die Maschine des Benutzers neu.	Nur Desktopbetriebssystemmaschinen: Klicken Sie für die ausgewählte Sitzung auf Neu starten. Sie können auch in der Ansicht Maschinendetails die Maschine mit den Energiesteuerelementen neu starten oder herunterfahren. Fordern Sie den Benutzer auf, sich neu anzumelden, sodass Sie die Anwendung überprüfen können.  Für Serverbetriebssystemmaschinen steht die Option "Neu starten" nicht zur Verfügung. Melden Sie stattdessen den Benutzer ab und fordern Sie ihn auf, sich neu anzumelden.
Setzen Sie die Maschine in den Wartungsmodus.	Wenn das Image einer Maschine gewartet werden muss, z. B. mit Patches oder anderen Updates, setzen Sie die Maschine in den Wartungsmodus und eskalieren Sie das Problem an den entsprechenden Administrator. Klicken Sie in der Ansicht Maschinendetails auf Details und aktivieren Sie die Option "Wartungsmodus". Eskalieren Sie an den entsprechenden Administrator.

# Wiederherstellen von Desktopverbindungen

May 10, 2016

Überprüfen Sie von Director den Verbindungsstatus des Benutzers für die aktuelle Maschine in der Titelleiste des Benutzers.

Wenn die Desktopverbindung fehlgeschlagen ist, wird die Fehlerursache angezeigt, um Sie bei der Problembehandlung zu unterstützen.

Aktion	Beschreibung
Stellen Sie sicher, dass die Maschine nicht im Wartungsmodus ist.	Achten Sie auf der Seite Benutzerdetails darauf, dass der Wartungsmodus deaktiviert ist.
Starten Sie die Maschine des Benutzers neu.	Wählen Sie die Maschine aus und klicken Sie auf Neu starten. Verwenden Sie diese Option, wenn die Maschine des Benutzers nicht mehr reagiert oder keine Verbindung herstellen kann, z. B. wenn die Maschine sehr viele CPU-Ressourcen verbraucht und dies die CPU unbrauchbar macht.

# Wiederherstellen von Sitzungen

May 10, 2016

Wenn eine Sitzung getrennt wird, bleibt sie aktiv und die Anwendungen werden weiter ausgeführt, das Benutzergerät kommuniziert jedoch nicht mehr mit dem Server.

Die Problembehandlung von Sitzungsfehlern machen Sie in der Ansicht Benutzerdetails im Bereich Sitzungsdetails. Sie können die Details der aktuellen Sitzung (durch die Sitzungs-ID gekennzeichnet) anzeigen.

Aktion	Beschreibung
Beenden Sie Anwendungen und Prozesse, die nicht reagieren.	Klicken Sie auf die Registerkarte Anwendungen. Wählen Sie eine nicht reagierende Anwendung aus und klicken Sie auf Anwendung beenden. Sie können auch einen entsprechenden Prozess auswählen, der nicht reagiert, und auf Prozess beenden. Beenden Sie auch Prozesse, die ungewöhnlich viel Speicher oder CPU-Ressourcen verbrauchen, da sie die CPU unbrauchbar machen können.
Trennen Sie die Windows-Sitzung.	Klicken Sie auf Sitzungssteuerung und wählen Sie dann Trennen. Diese Option steht nur für vermittelte Serverbetriebssystemmaschinen zur Verfügung. Für nicht vermittelte Sitzungen ist die Option deaktiviert.
Melden Sie den Benutzer von der Sitzung ab.	Klicken Sie auf Sitzungssteuerung und wählen Sie dann Abmelden.

Zum Testen der Sitzung kann der Benutzer versuchen, sich neu anzumelden. Sie können den Benutzer auch spiegeln, um diese Sitzung genauer zu beobachten.

Hinweis: Wenn VDAs vor XenDesktop 7 auf Benutzergeräten ausgeführt werden, kann Director keine vollständigen Informationen über die Sitzung anzeigen. Stattdessen wird in einer Meldung angezeigt, dass die Informationen nicht verfügbar sind. Diese Meldungen werden möglicherweise auf der Seite Benutzerdetails und im Aktivitäts-Manager angezeigt.

# Ausführen von Systemberichten für HDX-Kanäle

May 10, 2016

Prüfen Sie in der Ansicht Benutzerdetails im Bereich HDX den Status der HDX-Kanäle auf der Maschine des Benutzers. Dieser Bereich ist nur verfügbar, wenn die Maschine des Benutzers mit HDX verbunden ist.

Wenn eine Meldung angibt, dass die Informationen zurzeit nicht verfügbar sind, warten Sie eine Minute, bis die Seite aktualisiert ist, oder klicken Sie auf die Schaltfläche Aktualisieren. Die Aktualisierung von HDX-Daten kann etwas länger dauern als bei anderen Daten.

Klicken Sie zur Anzeige weiterer Informationen auf das Fehler- oder Warnsymbol.

Tipp: Sie können Informationen über andere Kanäle in demselben Dialogfeld einblenden, indem Sie in der linken Ecke der Titelleiste auf den Pfeil nach links oder rechts klicken.

Systemberichte über die HDX-Kanäle werden hauptsächlich vom Citrix Support für die weitere Problembehandlung verwendet.

1. Klicken Sie im Bereich HDX auf Systembericht herunterladen.
2. Sie können die XML-Berichtdatei anzeigen oder speichern.
  - Klicken Sie zur Ansicht der XML-Datei auf Öffnen. Die XML-Datei wird in demselben Fenster wie die Anwendung Director angezeigt.
  - Klicken Sie zum Speichern der XML-Datei auf Speichern. Das Dialogfeld Speichern unter wird angezeigt, in dem Sie angeben, an welchem Speicherort auf der Director-Maschine die Datei heruntergeladen wird.

# Zurücksetzen von Personal vDisk

May 10, 2016

Achtung: Beim Zurücksetzen der Disk werden die Einstellungen auf die werksseitigen Standardwerte zurückgesetzt und alle Daten, einschließlich der Anwendungen, werden gelöscht. Die Profildaten bleiben gespeichert, es sei denn, Sie haben den Personal vDisk-Standard geändert (d. h. das Umleiten der Profile vom Laufwerk C:) oder Sie verwenden keine Profillösung eines Drittanbieters.

Zum Zurücksetzen muss die Maschine mit Personal vDisk ausgeführt werden, der Benutzer muss aber nicht angemeldet sein.

Diese Option steht nur für Desktopbetriebssystemmaschinen zur Verfügung, sie ist für Serverbetriebssystemmaschinen deaktiviert.

1. Wählen Sie in der Ansicht Helpdesk die gewünschte Desktopbetriebssystemmaschine aus.
2. Klicken Sie in dieser Ansicht oder im Bereich Personalisierung der Ansicht Benutzerdetails auf Personal vDisk zurücksetzen.
3. Klicken Sie auf Zurücksetzen. Eine Warnmeldung gibt an, dass der Benutzer abgemeldet wird. Wenn der Benutzer abgemeldet ist (falls er angemeldet war), wird die Maschine neu gestartet.

Wenn das Zurücksetzen erfolgreich ist, wird im Feld Status im Bereich Personalisierung der Ansicht Benutzerdetails der Wert Wird ausgeführt angezeigt. Wenn das Zurücksetzen fehlgeschlagen ist, wird rechts neben Wird ausgeführt ein rotes X angezeigt. Wenn Sie auf dieses X zeigen, werden Informationen zu dem Fehler angezeigt.

# Zurücksetzen eines Benutzerprofils

May 10, 2016

Achtung: Wenn ein Profil zurückgesetzt wird, werden die Ordner und Dateien des Benutzers zwar gespeichert und in das neue Profil kopiert, aber die meisten Benutzerdaten werden gelöscht (z. B. wird die Registrierung zurückgesetzt und die Anwendungseinstellungen werden möglicherweise gelöscht).

1. Suchen Sie in Director den Benutzer, dessen Profil Sie zurücksetzen möchten, und wählen Sie seine Benutzersitzung aus.
2. Klicken Sie auf **Profil zurücksetzen**.
3. Fordern Sie den Benutzer auf, sich von allen Sitzungen abzumelden.
4. Fordern Sie den Benutzer auf, sich erneut anzumelden. Der Ordner und Dateien, die aus dem Profil des Benutzers gespeichert wurden, werden in das neue Profil kopiert.

Wichtig: Wenn der Benutzer Profile auf mehreren Plattformen (z. B. Windows 8 und Windows 7) hat, fordern Sie den Benutzer auf, sich zuerst bei dem gleichen Desktop oder bei der gleichen App anzumelden, bei dem bzw. der dieser Benutzer Probleme hatte. Dies stellt sicher, dass das richtige Profil zurückgesetzt wird.

Wenn das Profil ein Citrix Benutzerprofil ist, ist es zum Zeitpunkt der Benutzerdesktopanzeige bereits zurückgesetzt. Bei Microsoft-Roamingprofilen dauert die Ordnerwiederherstellung möglicherweise noch kurze Zeit an. Der Benutzer muss angemeldet bleiben, bis die Wiederherstellung abgeschlossen ist.

Hinweis: Bei den zuvor erläuterten Schritten wird davon ausgegangen, dass Sie XenDesktop (Desktop-VDA) verwenden. Wenn Sie XenApp (Server-VDA) verwenden, müssen Sie angemeldet sein, um das Profil zurückzusetzen. Der Benutzer muss sich dann abmelden und neu anmelden, um das Zurücksetzen des Profils abzuschließen.

Wenn das Profil nicht erfolgreich zurückgesetzt wird (z. B. der Benutzer kann sich nicht wieder anmelden oder einige der Dateien fehlen), müssen Sie das ursprüngliche Profil manuell wiederherstellen.

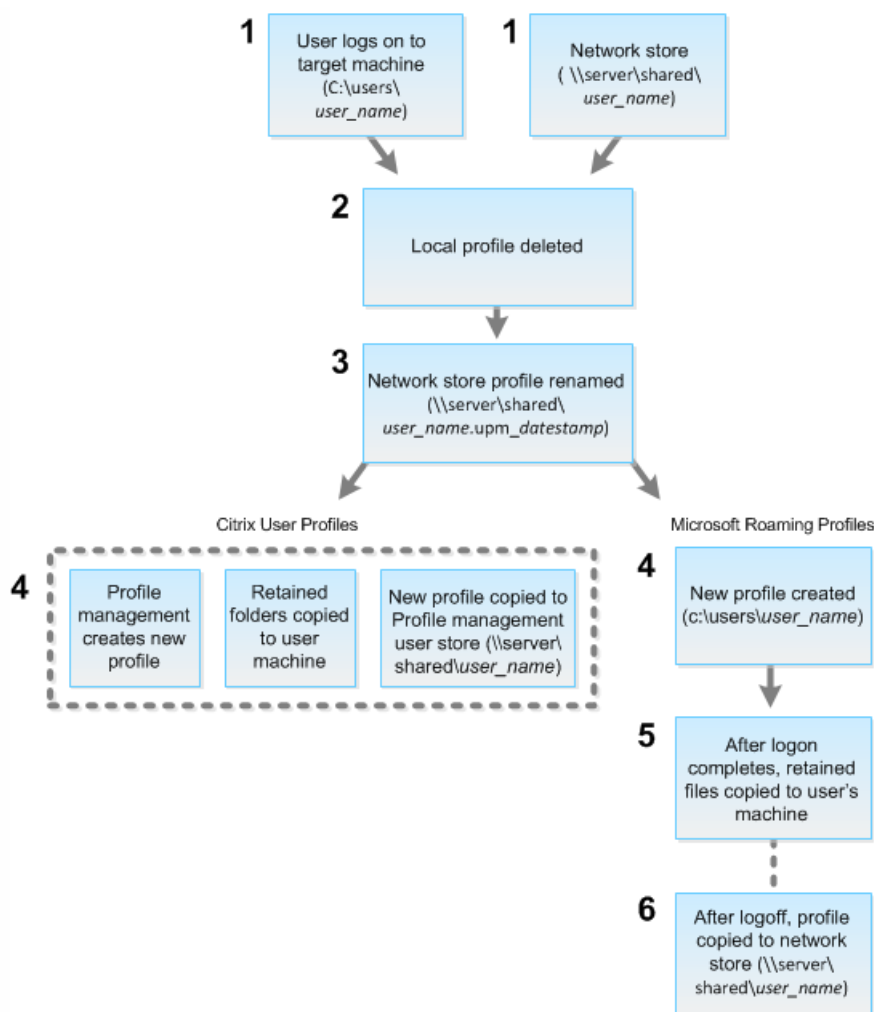
Die Ordner (und die Dateien) des Benutzerprofils werden gespeichert und in das neue Profil kopiert. Dabei gilt folgende Kopierreihenfolge:

- Desktop
- Cookies
- Favoriten
- Dokumente
- Bilder
- Musik
- Videos

Hinweis: In Windows 8 und höheren Versionen werden Cookies beim Zurücksetzen des Profils nicht kopiert.

Alle Citrix Benutzerprofile oder Microsoft Roamingprofile können zurückgesetzt werden. Wenn der Benutzer sich abmeldet und Sie den Befehl zum Zurücksetzen wählen (entweder in Director oder mit dem PowerShell SDK), identifiziert Director zunächst das verwendete Benutzerprofil und gibt dann den entsprechenden Befehl zum Zurücksetzen. Director erhält die Informationen über die Profilverwaltung, einschließlich Informationen zur Profilgröße, zum Typ und den Anmeldezeiten.

In diesem Diagramm wird die Verarbeitung dargestellt, die bei der nächsten Benutzeranmeldung ausgeführt wird.



1. Der Befehl zum Zurücksetzen von Director gibt den Profiltyp an. Der Profilverwaltungsdienst versucht dann, ein Profil dieses Typs zurückzusetzen und sucht die entsprechende Netzwerkfreigabe (Benutzerspeicher). Wenn der Benutzer von der Profilverwaltung verarbeitet wird, aber einen Roamingprofilbefehl erhält, wird er abgelehnt (oder umgekehrt).
2. Wenn ein lokales Profil vorhanden ist, wird es gelöscht.
3. Das Netzwerkprofil wird umbenannt.
4. Die nächste Aktion hängt davon ab, ob es sich bei dem Profil, das zurückgesetzt wird, um ein Citrix Benutzerprofil oder ein Microsoft Roamingprofil handelt.
  - Für Citrix Benutzerprofile wird das neue Profil mit den Importregeln der Profilverwaltung erstellt, die Ordner werden in das Netzwerkprofil zurückkopiert und der Benutzer kann die Anmeldung wie gewohnt fortsetzen. Wenn ein Roamingprofil für das Zurücksetzen verwendet wird, bleiben alle Registrierungseinstellungen im Roamingprofil im zurückgesetzten Profil gespeichert.  
Hinweis: Sie können in der Profilverwaltung konfigurieren, dass das Roamingprofil ggf. von einem Vorlagenprofil überschrieben wird.
  - Für Microsoft Roamingprofile wird ein neues Profil von Windows erstellt, und die Ordner werden bei Anmeldung des Benutzers auf das Benutzergerät zurückkopiert. Bei der nächsten Benutzerabmeldung wird das neue Profil in den Netzwerkspeicher kopiert.

1. Fordern Sie den Benutzer auf, sich von allen Sitzungen abzumelden.
2. Löschen Sie das lokale Profil, falls vorhanden.
3. Suchen Sie den archivierten Ordner auf der Netzwerkfreigabe, bei dem das Datum und die Uhrzeit dem Ordernamen

angehängt wurden, also den Ordner mit der Erweiterung .upm\_Datumsstempel.

4. Löschen Sie den aktuellen Profilnamen, d. h. die Datei ohne die Erweiterung upm\_Datumsstempel.
5. Benennen Sie den archivierten Ordner in den ursprünglichen Profilnamen um (entfernen Sie den Datumsstempel). Sie haben das Profil auf den ursprünglichen Zustand zurückgesetzt.



# Sitzungsaufzeichnung für XenApp 7.6 FP1, FP2 und LTSR

May 10, 2016

Die Sitzungsaufzeichnung ermöglicht das Aufzeichnen von Bildschirmaktivitäten in der Sitzung eines Benutzers von einer VDA-Serverbetriebssystemmaschine über eine beliebige Verbindung (im Einklang mit den Unternehmensrichtlinien und gesetzlichen Vorschriften). Mit der Sitzungsaufzeichnung können Sie Sitzungen aufzeichnen, katalogisieren und archivieren und sie erneut aufrufen und wiedergeben.

Die Sitzungsaufzeichnung verwendet flexible Richtlinien, mit denen automatisch Aufnahmen von Anwendungssitzungen ausgelöst werden können. Das IT-Team kann damit die Benutzeraktivität von Anwendungen überwachen und prüfen, z. B. Buchhaltungs- und Patienteninformationssysteme für das Gesundheitswesen, und interne Kontrollen zur Einhaltung von gesetzlichen Vorschriften und die Sicherheitsüberwachung unterstützen. Die Sitzungsaufzeichnung vereinfacht auch den technischen Support, da die Problemerkennung und Behebung der Probleme beschleunigt werden.

**Erhöhte Sicherheit durch Protokollierung und Überwachung:** Mit der Sitzungsaufzeichnung können Unternehmen die Aktivität der Benutzer auf dem Bildschirm für Anwendungen aufzeichnen, die vertrauliche Informationen verarbeiten. Dies ist besonders in stark regulierten Branchen wichtig, z. B. im Gesundheits- und Finanzwesen. Richtliniensteuerelemente ermöglichen eine selektive Aufzeichnung, wenn persönliche Daten ins Spiel kommen, die nicht aufgezeichnet werden dürfen.

**Leistungsfähige Aktivitätsüberwachung:** Die Sitzungsaufzeichnung erfasst und archiviert Bildschirmaktualisierungen, einschließlich Mausklicks und die visuelle Ausgabe von Tastaturanschlägen in gesicherten Videoaufzeichnungen, um die Aktivität für bestimmte Benutzer, Anwendungen und Server zu dokumentieren.

Die Sitzungsaufzeichnung ist nicht dafür ausgelegt oder vorgesehen, Beweismittel für Rechtsverfahren zu sammeln. Citrix empfiehlt, dass Unternehmen, die die Sitzungsaufzeichnung verwenden, andere Methoden der Beweismittelsammlung nutzen, z. B. konventionelle Videoaufzeichnungen in Kombination mit textbasierten eDiscovery-Tools.

**Schnellere Problembehebung:** Wenn sich Benutzer mit einem Problem an den Helpdesk wenden, das schwer zu reproduzieren ist, können die Supportmitarbeiter die Aufzeichnung von Benutzersitzungen aktivieren. Wenn das Problem wieder auftritt, stellt die Sitzungsaufzeichnung eine visuelle Aufzeichnung des Fehlers bereit, einschließlich Zeitstempel, mit der Benutzerprobleme schneller gelöst werden können.

# Erste Schritte mit der Sitzungsaufzeichnung

May 10, 2016

Nach dem Durchführen der folgenden Schritte können Sie XenApp-Sitzungen aufzeichnen und prüfen.

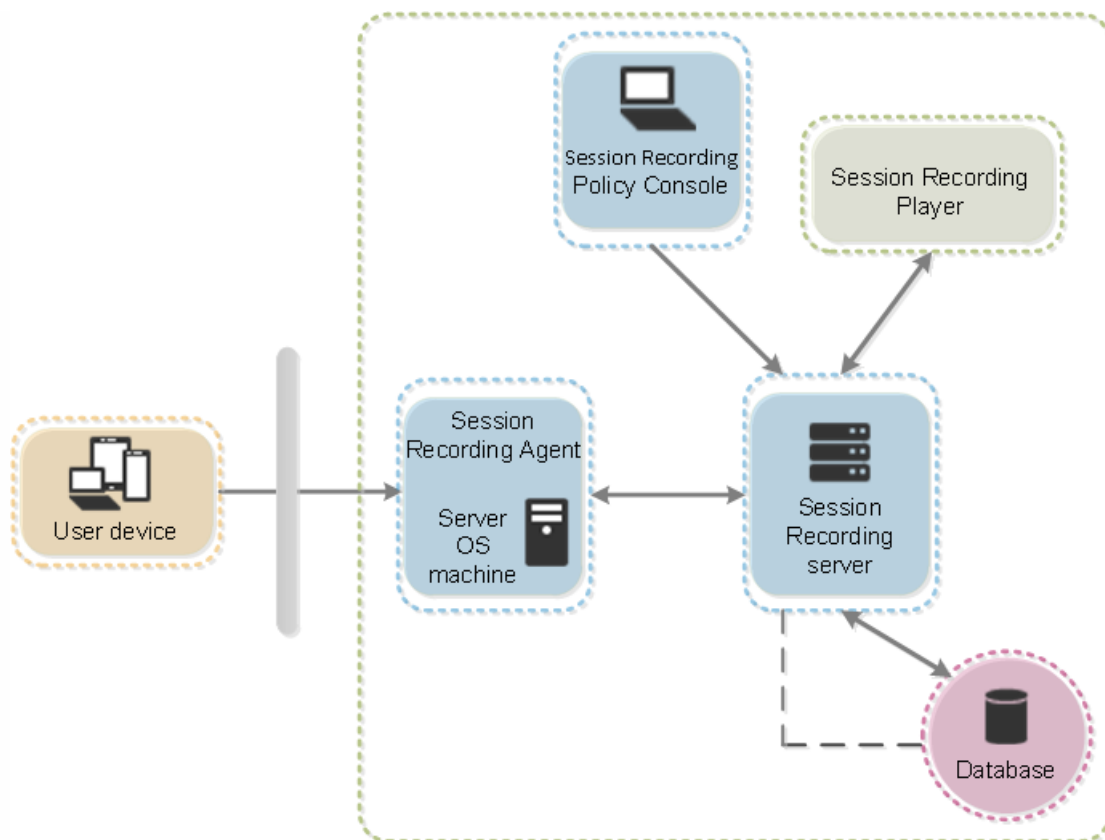
1. Machen Sie sich mit den Komponenten der Sitzungsaufzeichnung vertraut.
2. Wählen Sie das Bereitstellungsszenario für die Umgebung.
3. Prüfen Sie die Installationsanforderungen.
4. Installieren Sie die Sitzungsaufzeichnung.
5. Konfigurieren Sie die Sitzungsaufzeichnungskomponenten für das Aufzeichnen und Anzeigen von Sitzungen.

Die Sitzungsaufzeichnung besteht aus fünf Komponenten:

- **Sitzungsaufzeichnungsagent:** eine Komponente, die auf jeder Serverbetriebssystemmaschine zum Aktivieren der Aufzeichnungen installiert ist. Mit dieser Komponente werden Sitzungsdaten aufgezeichnet.
- **Sitzungsaufzeichnungsserver:** Ein Server, auf dem die folgenden Programme ausgeführt werden:
  - **Broker:** Eine von IIS 6.0+ gehostete Webanwendung, die Such- und Dateidownloadanfragen vom Player und Richtlinienverwaltungsanforderungen von der Richtlinienkonsole für die Sitzungsaufzeichnung handhabt und Aufzeichnungsrichtlinien für jede XenApp-Sitzung auswertet.
  - **Speichermanager:** Ein Windows-Dienst, der Sitzungsaufzeichnungsdateien verwaltet, die von jedem für die Sitzungsaufzeichnung aktivierten Computer mit XenApp empfangen werden.
- **Sitzungsaufzeichnungsplayer:** Eine Benutzeroberfläche, auf die Benutzer von der Arbeitsstation aus zugreifen und mit der aufgezeichnete XenApp-Sitzungsdateien wiedergegeben werden.
- **Datenbank für die Sitzungsaufzeichnung:** Eine SQL-Datenbank zum Speichern der Sitzungsaufzeichnungsdaten.
- **Richtlinienkonsole für die Sitzungsaufzeichnung:** Konsole zum Erstellen von Richtlinien, um anzugeben, welche Sitzungen aufgezeichnet werden sollen.

In dieser Abbildung werden die Komponenten der Sitzungsaufzeichnung und deren Beziehung zueinander dargestellt:

Im dargestellten Beispiel einer Bereitstellung befinden sich der Sitzungsaufzeichnungsagent, der Sitzungsaufzeichnungsserver, die Datenbank für die Sitzungsaufzeichnung, die Richtlinienkonsole für die Sitzungsaufzeichnung und der Sitzungsaufzeichnungsplayer hinter einer Sicherheitsfirewall. Der Sitzungsaufzeichnungsagent ist auf einer Serverbetriebssystemmaschine installiert. Auf einem zweiten Server wird die Richtlinienkonsole für die Sitzungsaufzeichnung ausgeführt, ein dritter Server ist der Sitzungsaufzeichnungsserver und auf einem vierten Server wird die Datenbank für die Sitzungsaufzeichnung ausgeführt. Der Sitzungsaufzeichnungsplayer ist auf einer Arbeitsstation installiert. Ein Clientgerät außerhalb der Firewall kommuniziert mit der Serverbetriebssystemmaschine, auf der der Sitzungsaufzeichnungsagent installiert ist. Innerhalb der Firewall kommunizieren der Sitzungsaufzeichnungsagent, die Richtlinienkonsole für die Sitzungsaufzeichnung, der Sitzungsaufzeichnungsplayer und die Datenbank für die Sitzungsaufzeichnung mit dem Sitzungsaufzeichnungsserver.



# Planen der Bereitstellung

May 10, 2016

Abhängig von der Umgebung können Sie die Sitzungsaufzeichnungskomponenten in verschiedenen Szenarios bereitstellen.

Eine Sitzungsaufzeichnungsbereitstellung muss nicht auf eine Site begrenzt sein. Außer dem Sitzungsaufzeichnungsagent sind alle Komponenten von der Serversite unabhängig. Sie können beispielsweise mehrere Sites für einen Sitzungsaufzeichnungsserver konfigurieren.

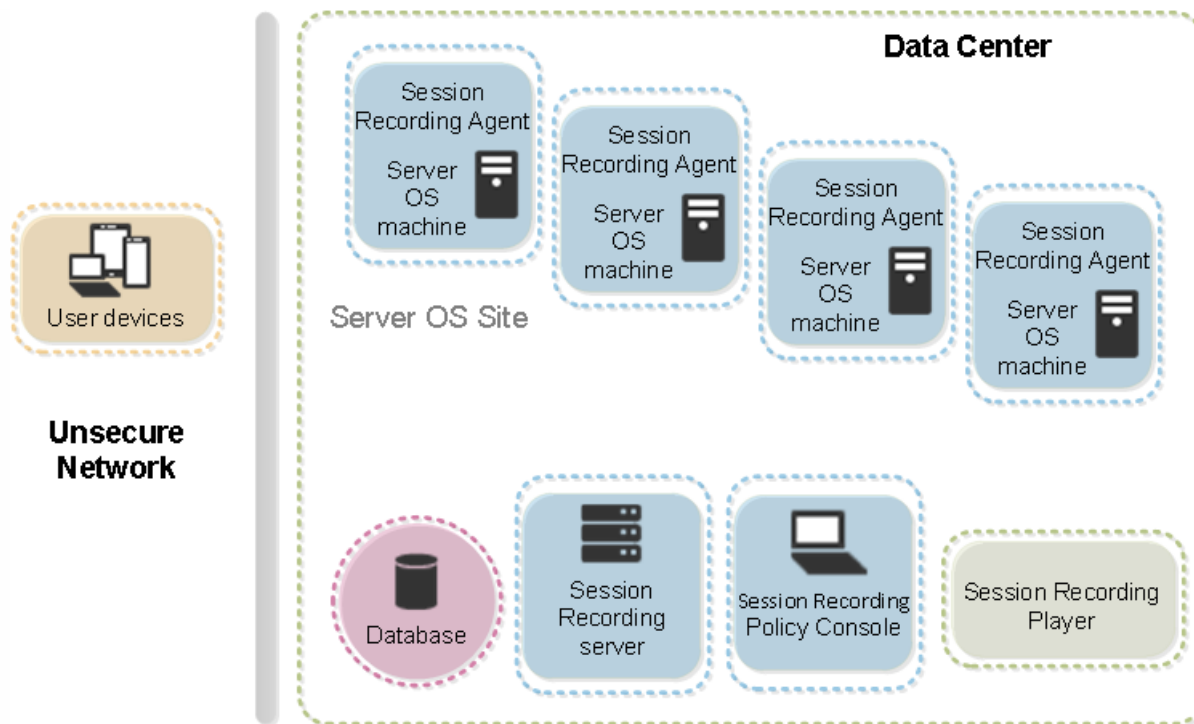
Wenn Sie jedoch eine große Site mit vielen Instanzen der Agentsoftware haben und viele grafikintensive Anwendungen (z. B. AutoCAD-Anwendungen) oder viele Sitzungen aufzeichnen möchten, kann dies zu einem hohen Leistungsbedarf auf einem Sitzungsaufzeichnungsserver führen. Sie können zur Vermeidung von Leistungsproblemen mehrere Sitzungsaufzeichnungsserver auf verschiedenen Computern installieren und die Instanzen des Sitzungsaufzeichnungsagents auf verschiedene Computer verweisen. Vergessen Sie jedoch nicht, dass eine Instanz der Agentsoftware nur jeweils auf einen Server verweisen kann.

Im Anschluss finden Sie die zwei empfohlenen Konfigurationen für eine Sitzungsaufzeichnungsbereitstellung:

- Bereitstellen des Sitzungsaufzeichnungsagents auf einer einzelnen Serverbetriebssystemmaschine
- Bereitstellen des Sitzungsaufzeichnungsagents auf mehreren Serverbetriebssystemmaschinen

## Serversitebereitstellung

Verwenden Sie diesen Typ der Bereitstellung für die Aufzeichnung von Sitzungen für eine oder mehrere Sites. Der Sitzungsaufzeichnungsagent wird auf jeder Serverbetriebssystemmaschine der Site installiert. Die Site befindet sich in einem Datacenter hinter einer Sicherheitsfirewall. Die Komponenten der Sitzungsaufzeichnungsverwaltung (Datenbank für die Sitzungsaufzeichnung, Sitzungsaufzeichnungsserver, Richtlinienkonsole für die Sitzungsaufzeichnung) sind außerhalb des Datacenters hinter der Firewall auf anderen Servern und der Sitzungsaufzeichnungsplayer ist auf einer Arbeitsstation installiert. Außerhalb der Firewall, in einer nicht gesicherten Netzwerkumgebung, befinden sich die XenApp-Clients, z. B. Arbeitsstationen, mobile Geräte, PDAs und Laptops.



# Sicherheitsempfehlungen

Nov 30, 2016

Die Sitzungsaufzeichnung ist für die Bereitstellung in einem sicheren Netzwerk und den Zugriff durch Administratoren konzipiert und ist daher sicher. Die Standardbereitstellung ist einfach, und Sicherheitsfunktionen, z. B. digitale Signatur und Verschlüsselung, können optional konfiguriert werden.

Die Komponenten der Sitzungsaufzeichnung kommunizieren über die Internetinformationsdienste (IIS) und Microsoft Message Queuing (MSMQ). Internetinformationsdienste stellen die Webdienste-Kommunikationsverbindung zwischen jeder Komponente der Sitzungsaufzeichnung bereit. MSMQ bietet eine zuverlässige Datentransportmethode zum Senden von Sitzungsaufzeichnungsdaten vom Sitzungsaufzeichnungsagent zum Sitzungsaufzeichnungsserver.

Berücksichtigen Sie diese Sicherheitsempfehlungen bei der Planung der Bereitstellung:

- Stellen Sie sicher, dass Server mit Sitzungsaufzeichnungskomponenten physisch gesichert sind. Schließen Sie diese Computer falls möglich in einem sicheren Raum ein, zu dem nur autorisierte Person direkten Zugang haben.
- Isolieren Sie Server mit Sitzungsaufzeichnungskomponenten in einem separaten Subnetz oder einer separaten Domäne.
- Installieren Sie eine Firewall zwischen dem Sitzungsaufzeichnungsserver und den anderen Servern, um die aufgezeichneten Sitzungsdaten vor Benutzern zu schützen, die auf andere Server zugreifen.
- Halten Sie den Verwaltungsserver und die SQL-Datenbank für die Sitzungsaufzeichnung durch Installation der aktuellen Sicherheitsupdates von Microsoft auf dem neuesten Stand.
- Verhindern Sie, dass Personen ohne Administratorberechtigung sich beim Verwaltungscomputer anmelden.
- Schränken Sie genau ein, welche Benutzer die Aufzeichnungsrichtlinien ändern und Sitzungsaufzeichnungen anzeigen können.
- Installieren Sie digitale Zertifikate, verwenden Sie die Funktion zur Dateisignatur der Sitzungsaufzeichnung und richten Sie die SSL-Kommunikation in IIS ein.
- Legen Sie im Dialogfeld Sitzungsaufzeichnungsagent - Eigenschaften für MSMQ die Verwendung von HTTPS als Transportprotokoll fest. Weitere Informationen finden Sie unter [Problembehandlung bei MSMQ](#).
- Verwenden Sie TLS 1.0 und deaktivieren Sie die Verschlüsselungsverfahren SSLv2, SSLv3 und RC4 auf dem Sitzungsaufzeichnungsserver und in der Datenbank für die Sitzungsaufzeichnung. Weitere Informationen finden Sie in der Microsoft-Artikeln <http://support.microsoft.com/default.aspx?scid=kb;en-us;187498> und <http://support.microsoft.com/kb/245030/en-us>.
- Verwenden Sie den Wiedergabeschutz. Der Wiedergabeschutz ist eine Funktion der Sitzungsaufzeichnung, mit der aufgezeichnete Dateien vor dem Download zum Sitzungsaufzeichnungsplayer verschlüsselt werden. Diese Option ist in der Standardeinstellung aktiviert und befindet sich in den Eigenschaften des Sitzungsaufzeichnungservers.
- Folgen Sie den Anleitungen von NSIT für die Länge der Kryptografieschlüssel und den Kryptografiealgorithmen.

Weitere Informationen über die Konfiguration der Sitzungsaufzeichnungsfeatures finden Sie unter <http://support.citrix.com/article/CTX200868>.

-

Auf dem Computer mit dem Sitzungsaufzeichnungsserver sendet der IIS-Webserver das Serverzertifikat an den Client, wenn eine SSL-Verbindung von Sitzungsaufzeichnungsagent, dem Sitzungsaufzeichnungsplayer oder der Richtlinienkonsole für die Sitzungsaufzeichnung hergestellt wird. Beim Empfang eines Serverzertifikats prüft der Sitzungsaufzeichnungsagent, der Sitzungsaufzeichnungsplayer oder die Richtlinienkonsole für die Sitzungsaufzeichnung, welche Zertifizierungsstelle das

Zertifikat ausgestellt hat, und ob die Zertifizierungsstelle vom Client als vertrauenswürdig angesehen wird. Wenn die Zertifizierungsstelle nicht als vertrauenswürdig angesehen wird, wird das Zertifikat abgelehnt, und ein Fehler wird im Anwendungsereignisprotokoll für den Sitzungsaufzeichnungsagent aufgezeichnet oder eine Fehlermeldung wird dem Benutzer im Sitzungsaufzeichnungsplayer bzw. der Richtlinienkonsole für die Sitzungsaufzeichnung angezeigt.

Zur Installation eines Serverzertifikats werden Informationen über den Server gesammelt und eine Zertifizierungsstelle wird zur Ausstellung eines Zertifikats für den Server aufgefordert. Sie müssen beim Anfordern eines Serverzertifikats die richtigen Informationen angeben und sicherstellen, dass der Servername richtig geschrieben ist. Wenn der vollqualifizierte Domänenname (FQDN) für die Clientverbindungen (Sitzungsaufzeichnungsagent, Sitzungsaufzeichnungsplayer und Richtlinienkonsole) verwendet wird, müssen die Zertifikatsinformationen, die an die Zertifizierungsstelle gesendet werden, den vollqualifizierten Domännennamen des Servers und nicht den NetBIOS-Namen verwenden. Wenn Sie einen NetBIOS-Namen angeben, geben Sie beim Anfordern eines Serverzertifikats nicht den FQDN an. Installieren Sie das Serverzertifikat im lokalen Zertifikatsspeicher des Servers. Installieren Sie das Zertifikat der ausstellenden Zertifizierungsstelle auf jedem Client, der eine Verbindung herstellt.

Im Unternehmen wird möglicherweise eine private Zertifizierungsstelle verwendet, die Serverzertifikate ausstellt und mit der Sitzungsaufzeichnung verwendet werden kann. Bei Verwendung einer privaten Zertifizierungsstelle müssen Sie sicherstellen, dass das Zertifikat der ausstellenden Zertifizierungsstelle auf jedem Clientgerät installiert ist. Weitere Informationen zu Zertifikaten und Zertifizierungsstellen finden Sie in der Microsoft Dokumentation. Einige Unternehmen und Organisationen fungieren derzeit als Zertifizierungsstellen, u. a. VeriSign, Baltimore, Entrust und ihre jeweiligen verbundenen Unternehmen.

Alle Zertifikate haben ein Ablaufdatum, das die Zertifizierungsstelle festlegt. Das Ablaufdatum wird in den Eigenschaften des Zertifikats angegeben. Stellen Sie sicher, dass Zertifikate vor dem Ablauf erneuert werden, um Fehler bei der Sitzungsaufzeichnung zu vermeiden.

Die Installation der Sitzungsaufzeichnung verwendet HTTPS und die Standardwebsite muss mit einem Serverzertifikat konfiguriert werden, das von einer Zertifizierungsstelle ausgestellt wurde. Weitere Informationen zur Installation von Serverzertifikaten unter IIS finden Sie in der IIS-Dokumentation.

# Überlegungen zur Skalierbarkeit

May 10, 2016

Für die Installation und Ausführung der Sitzungsaufzeichnung sind im Vergleich zur Ausführung von XenApp nur wenige zusätzliche Ressourcen erforderlich. Wenn Sie jedoch viele Sitzungen mit der Sitzungsaufzeichnung aufzeichnen möchten oder die Sitzungen, die Sie aufzeichnen möchten, große Sitzungsdateien erstellen (z. B. Anwendungen mit hohem Grafikanteil), sollten Sie die Leistung des Systems berücksichtigen, wenn Sie die Bereitstellung der Sitzungsaufzeichnung planen.

Weitere Informationen finden zum Konstruieren eines flexibel skalierbaren Sitzungsaufzeichnungssystems finden Sie unter <http://support.citrix.com/article/CTX200869>.

Dieser Abschnitt enthält folgende Themen:

[Empfehlungen zur Hardware](#)

[Datenträger- und Datenspeicherhardware](#)

[Netzwerkkapazität](#)

[Verarbeitungskapazität des Computers](#)

[Bereitstellen mehrerer Sitzungsaufzeichnungsserver](#)

[Datenbankskalierbarkeit](#)

Berücksichtigen Sie, wie viele Daten an jeden Sitzungsaufzeichnungsserver gesendet werden und wie schnell die Server diese Daten verarbeiten und speichern können. Die Rate, mit der das System die eingehenden Daten speichern kann, muss größer als die Dateneingaberate sein.

Sie können die Dateneingaberate schätzen, wenn Sie die Anzahl der aufgezeichneten Sitzungen mit der durchschnittlichen Größe jeder Sitzungsaufzeichnung multiplizieren und durch die Länge der Sitzungsaufzeichnungen dividieren. Beispiel: An einem 8-stündigen Arbeitstag zeichnen Sie 5000 Microsoft Outlook-Sitzungen auf, deren Größe 20 MB ist. Die Dateneingaberate beträgt dann ungefähr 3,5 MBit/s. (5000 Sitzungen multipliziert mit 20 MB, dividiert durch 8 Stunden und dividiert durch 3600 Sekunden.)

Sie können die Leistung steigern, wenn Sie die Leistung eines Sitzungsaufzeichnungsservers optimieren oder mehrere Sitzungsaufzeichnungsserver auf verschiedenen Computern installieren.

Die Datenträger- und Datenspeicherhardware ist der wichtigste Faktor bei der Planung der Bereitstellung der Sitzungsaufzeichnung. Die Schreibleistung der Datenspeicherlösung ist besonders wichtig. Je schneller Daten zum Datenträger geschrieben werden, je höher ist die Leistung des Gesamtsystems.

Zu den für die Sitzungsaufzeichnung geeigneten Datenspeicherlösungen gehören lokale Datenträger, die als RAID-Arrays von einem lokalen Datenträgercontroller oder von einem verbundenen SAN gesteuert werden.

Hinweis: Die Sitzungsaufzeichnung sollte nicht mit NAS (Network-Attached Storage) verwendet werden, da Leistungs- und



Sicherheitsprobleme auftreten, die mit dem Schreiben von Aufzeichnungsdaten in Netzlaufwerke verbunden sind. Bei der Installation auf einem lokalen Datenträger steigert ein Datenträgercontroller mit integriertem Cachespeicher die Leistung. Ein Datenträgercontroller mit Cachespeicher muss eine Notstromversorgung durch Batterie haben, um die Datenintegrität bei einem Stromausfall sicherzustellen.

Ein Netzwerk mit 100 MBit/s ist für die Verbindung eines Sitzungsaufzeichnungsservers geeignet. Eine Gigabit-Ethernet-Verbindung kann die Leistung verbessern, führt jedoch nicht zu einer zehn Mal besseren Leistung als eine Verbindung mit 100 MBit/s.

Stellen Sie sicher, dass Netzwerkschwitches, die von der Sitzungsaufzeichnung verwendet werden, nicht mit Anwendungen von Drittherstellern gemeinsam verwendet werden, die ggf. um die verfügbare Netzwerkbandbreite konkurrieren. Netzwerkschwitches sollten nur vom Sitzungsaufzeichnungsserver verwendet werden.

Der Computer, auf dem ein Sitzungsaufzeichnungsserver installiert ist, muss die folgenden Spezifikationen erfüllen:

- Eine Dual-CPU oder Dual-Core-CPU (empfohlen)
- 2 GB bis 4 GB RAM

Bei höheren Spezifikationen wird die Leistung nicht wesentlich gesteigert.

Wenn ein einzelner Sitzungsaufzeichnungsserver die Leistungsanforderungen nicht erfüllt, können Sie weitere Sitzungsaufzeichnungsserver auf anderen Computern installieren. Bei diesem Typ der Bereitstellung hat jeder Sitzungsaufzeichnungsserver eine dedizierte Datenspeicherung, Datenbank und dedizierte Netzwerkschwitches. Verweisen Sie die Instanzen des Sitzungsaufzeichnungsagents in der Bereitstellung auf verschiedene Sitzungsaufzeichnungsserver, um die Last aufzuteilen.

Die Datenbank für die Sitzungsaufzeichnung erfordert Microsoft SQL Server 2014, Microsoft SQL Server 2012 oder Microsoft SQL Server 2008 R2. Die an die Datenbank gesendete Datenmenge ist sehr gering, da nur die Metadaten über die aufgezeichneten Sitzungen in der Datenbank gespeichert werden. Die Sitzungsaufzeichnungsdateien werden auf einem separaten Datenträger gespeichert. Normalerweise benötigt jede aufgezeichnete Sitzung nur 1 KB in der Datenbank, es sei denn, Sie fügen durchsuchbare Ereignisse mit der Sitzungsaufzeichnungs-Ereignis-API in die Sitzung ein.

Bei den Express-Editionen von Microsoft SQL Server 2014, Microsoft SQL Server 2012, und Microsoft SQL Server 2008 R2 ist die Größe der Datenbank auf 10 GB beschränkt. Bei 1 KB pro aufgezeichneter Sitzung kann die Datenbank ungefähr 4 Millionen Sitzungen katalogisieren. Andere Editionen von Microsoft SQL Server haben keine Beschränkungen hinsichtlich Datenbankgröße und werden nur durch den verfügbaren Speicherplatz auf dem Datenträger beschränkt. Wenn die Zahl der Sitzungen in der Datenbank ansteigt, wird die Leistung der Datenbank und die Geschwindigkeit der Suchen nur geringfügig beeinträchtigt.

Wenn Sie keine Anpassungen über die Sitzungsaufzeichnungs-Ereignis-API machen, generiert jede aufgezeichnete Sitzung vier Datenbanktransaktionen: Zwei beim Start der Aufzeichnung, eine bei der Benutzeranmeldung bei der aufgezeichneten Sitzung und eine am Ende der Aufzeichnung. Beim Anpassen der Sitzungen mit der Sitzungsaufzeichnungs-Ereignis-API erstellt jedes durchsuchbare Ereignis, das aufgezeichnet wurde, eine Transaktion. Da selbst bei der einfachsten

Datenbankbereitstellung mehrere Hundert Transaktionen pro Sekunde gehandhabt werden können, wird die Verarbeitungslast für die Datenbank nie überstrapaziert. Die Auswirkung ist so gering, dass die Datenbank für die Sitzungsaufzeichnung normalerweise auf demselben SQL-Server wie andere Datenbanken ausgeführt werden kann, u. a. der XenApp- oder XenDesktop-Datenbank des Datenspeichers.

Wenn Sie in der Bereitstellung der Sitzungsaufzeichnung viele Millionen aufgezeichneter Sitzungen in der Datenbank katalogisieren müssen, halten Sie die Microsoft-Richtlinien zur Skalierbarkeit von SQL Server ein.

# Wichtige Bereitstellungshinweise

May 10, 2016

- Die Komponenten der Sitzungsaufzeichnung können nur miteinander kommunizieren, wenn sie in derselben Domäne oder in vertrauenswürdigen Domänen mit einer gegenseitigen Vertrauensbeziehung installiert sind. Das System kann nicht in einer Arbeitsgruppe oder in Domänen mit einer externen Vertrauensbeziehung installiert werden.
- Die Sitzungsaufzeichnung unterstützt kein Clustering von zwei oder mehr Sitzungsaufzeichnungsservern in einer Bereitstellung.
- Aufgrund des hohen Grafikanteils und der Speichernutzung bei der Wiedergabe von großen Aufzeichnungen sollte der Sitzungsaufzeichnungsplayer nicht als veröffentlichte Anwendung installiert werden.
- Die Installation der Sitzungsaufzeichnung ist für die SSL/HTTPS-Kommunikation konfiguriert. Stellen Sie sicher, dass Sie ein Zertifikat auf dem Sitzungsaufzeichnungsserver installieren, und dass die Komponenten der Sitzungsaufzeichnung der Stammzertifizierungsstelle vertrauen.
- Wenn Sie die Datenbank für die Sitzungsaufzeichnung auf einem eigenständigen Server mit SQL Server 2014 Express Edition, SQL Server 2012 Express Edition oder SQL Server 2008 R2 Express Edition ausführen, muss auf dem Server das TCP/IP-Protokoll aktiviert sein und der SQL Server-Browserdienst muss ausgeführt werden. Diese Einstellungen sind standardmäßig deaktiviert, müssen jedoch für die Kommunikation zwischen dem Sitzungsaufzeichnungsserver und der Datenbank aktiviert werden. Weitere Informationen zur Aktivierung dieser Einstellungen finden Sie in der Microsoft Dokumentation.
- Berücksichtigen Sie bei der Planung der Sitzungsaufzeichnungsbereitstellung die Auswirkungen der Sitzungs freigabe. Die Sitzungs freigabe für veröffentlichte Anwendungen kann Konflikte mit Richtlinienregeln für Sitzungsaufzeichnungen der Sitzungsaufzeichnung für veröffentlichte Anwendungen verursachen. Die Sitzungsaufzeichnung ordnet die aktive Richtlinie der ersten veröffentlichten Anwendung zu, die ein Benutzer öffnet. Wenn der Benutzer die erste Anwendung geöffnet hat, halten weitere Anwendungen, die in derselben Sitzung geöffnet werden, die Richtlinie ein, die für die erste Anwendung gilt. Beispiel: Wenn eine Richtlinie festlegt, dass nur Outlook aufgezeichnet wird, beginnt die Aufzeichnung, wenn der Benutzer Outlook öffnet. Wenn der Benutzer jedoch Microsoft Word als zweite veröffentlichte Anwendung öffnet (während Outlook ausgeführt wird), wird Word auch aufgezeichnet. Sollte die aktive Richtlinie jedoch nicht festlegen, dass Word aufgezeichnet wird, und der Benutzer startet Word vor Outlook (gemäß der Richtlinie wird Outlook aufgezeichnet), wird Outlook nicht aufgezeichnet.

# Installieren der Sitzungsaufzeichnung

May 10, 2016

Dieser Artikel enthält in jeweils eigenen Abschnitten Installationsanweisungen für XenApp 7.6 Feature Pack 2 und LTSR und Feature Pack 1.

## Checkliste vor der Installation

Stellen Sie vor der Installation sicher, dass Sie alle in dieser Liste aufgeführten Schritte abgeschlossen haben:

	Schritt
	Installieren Sie die Voraussetzungen, bevor Sie mit der Installation beginnen. Weitere Informationen finden Sie unter <a href="#">Systemanforderungen</a> .
	Wählen Sie die Computer aus, auf denen Sie die Komponenten der Sitzungsaufzeichnung installieren möchten, und stellen Sie sicher, dass jeder Computer die Hardware- und Softwareanforderungen für die zu installierenden Komponenten erfüllt.
	Laden Sie die ZIP-Datei der Sitzungsaufzeichnung von der LTSR-Downloadseite herunter.
	Installieren Sie die relevanten Zertifikate in der Umgebung, wenn die Kommunikation zwischen den Komponenten der Sitzungsaufzeichnung mit dem SSL-Protokoll erfolgt. Weitere Informationen finden Sie unter <a href="#">Installieren von Zertifikaten</a> .
	Installieren Sie die für die Komponenten der Sitzungsaufzeichnung benötigten Hotfixes. Die Hotfixes sind unter <a href="#">Citrix Support</a> verfügbar.
	Konfigurieren Sie Director zum Erstellen und Aktivieren von Sitzungsaufzeichnungsrichtlinien.

## Hinweise:

- Citrix empfiehlt, die veröffentlichten Anwendungen in separate Bereitstellungsgruppen basierend auf den Aufzeichnungsrichtlinien aufzuteilen, da die Sitzungsfreigabe veröffentlichter Anwendungen einen Konflikt mit aktiven Richtlinien verursachen kann, wenn sie in derselben Bereitstellungsgruppe sind. Die Sitzungsaufzeichnung ordnet die aktive Richtlinie der ersten veröffentlichten Anwendung zu, die ein Benutzer öffnet.
- Wenn Sie beabsichtigen, Maschinenerstellungsdienste (MCS) oder Provisioning Services mit XenApp zu verwenden, bereiten Sie den Server für eine eindeutige QMID vor. Weitere Informationen finden Sie in der Beschreibung unter "Bekannte Probleme". Wenn Sie diesen Schritt nicht ausführen, verlieren Sie u. U. Aufzeichnungsdaten.
- Für SQL Server muss TCP/IP aktiviert sein und der SQL Server-Browserdienst sowie die Windows-Authentifizierung müssen ausgeführt werden.
- Wenn Sie HTTPS verwenden, konfigurieren Sie Serverzertifikate für SSL/HTTPS.

## Installationsdateien für die Sitzungsaufzeichnung

Sie benötigen die folgenden Installationsdateien von der Citrix Downloadseite:

- Verwaltungsdateien der Sitzungsaufzeichnung
  - Broker\_PowerShellSnapIn\_x64.msi
  - SessionRecordingAdministrationx64.msi
- Agentdateien der Sitzungsaufzeichnung
  - SessionRecordingAgentx64.msi
- Playerdateien der Sitzungsaufzeichnung
  - SessionRecordingPlayer.msi

Die Verwaltungskomponenten der Sitzungsaufzeichnung sind die Datenbank für die Sitzungsaufzeichnung, der Sitzungsaufzeichnungsserver und die Richtlinienkonsole für die Sitzungsaufzeichnung. Sie können festlegen, welche dieser Komponenten auf einem Server installiert werden.

Stellen Sie sicher, dass alle Voraussetzungen installiert sind, bevor Sie die Verwaltungskomponenten der Sitzungsaufzeichnung installieren. Weitere Informationen finden Sie unter [Verwaltungskomponenten der Sitzungsaufzeichnung](#).

Zum Verbessern der Sicherheit können Sie diese Berechtigungen nach der Installation der Datenbank entfernen.

1. Führen Sie **Broker\_PowerShellSnapIn\_x64.msi** aus und folgen Sie den Anweisungen zum Durchführen der Installation.
2. Starten Sie die Windows-Eingabeaufforderung als Administrator und führen Sie den folgenden Befehl aus:

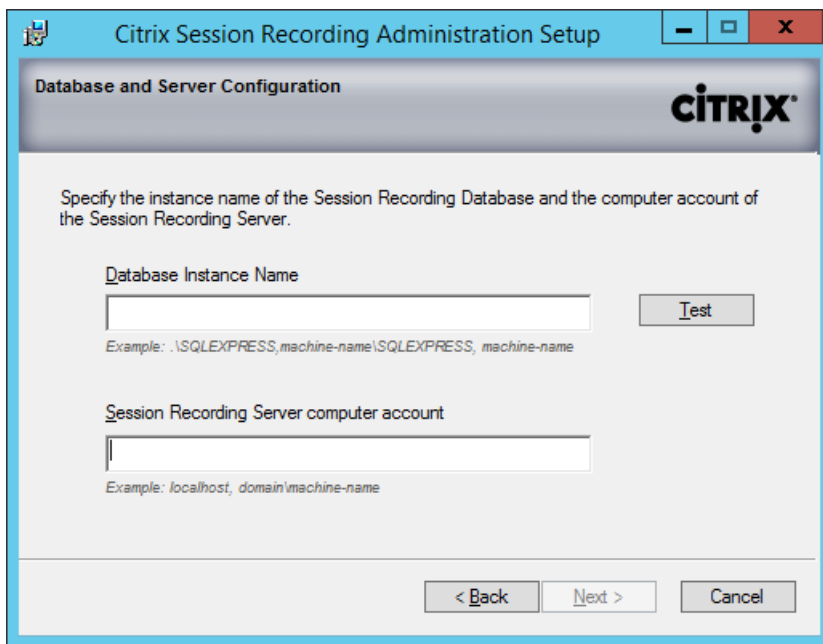
```
msiexec /i SessionRecordingAdministrationx64.msi
```

oder doppelklicken Sie auf die MSI-Datei.

3. Klicken Sie im Installationsprogramm auf "Weiter" und akzeptieren Sie die Lizenzvereinbarung.
4. Wählen Sie im Setupbildschirm der Sitzungsaufzeichnungsverwaltung die Verwaltungskomponenten aus, die Sie installieren möchten.

Stellen Sie sicher, dass alle Voraussetzungen installiert sind, bevor Sie die Datenbank für die Sitzungsaufzeichnung installieren. Weitere Informationen finden Sie unter [Verwaltungskomponenten der Sitzungsaufzeichnung](#).

1. Führen Sie auf der Seite "Datenbankkonfiguration" folgende Schritte aus:
  - Wenn Sie alle Verwaltungskomponenten auf demselben Server installieren, geben Sie **localhost** im Feld "Name des Sitzungsaufzeichnungsservers" ein.
  - Wenn Sie den Sitzungsaufzeichnungsserver und die Datenbank für die Sitzungsaufzeichnung auf verschiedenen Servern installieren, geben Sie den Namen des Computers, auf dem der Sitzungsaufzeichnungsserver gehostet wird, im folgenden Format ein: Domäne\Computername. Der Name des Sitzungsaufzeichnungsservers ist das Benutzerkonto für den Zugriff auf die Datenbank.



Wenn die Datenbankinstanz keine benannte Instanz ist, die Sie beim Setup der Instanz konfiguriert haben, können Sie nur den Maschinennamen des SQL Servers verwenden. Wenn Sie die Instanz benannt haben, verwenden Sie Maschinename\Instanzname als Datenbankinstanzname. Um den verwendeten Servernamen zu ermitteln, führen Sie **select @@servername** auf dem SQL Server aus. Der zurückgegebene Name ist der genaue Datenbankinstanzname. Klicken Sie auf **Test**, um die Verbindung mit dem SQL-Server zu testen. Stellen Sie sicher, dass der aktuelle Benutzer die öffentliche SQL Server-Rollenberechtigung hat, ansonsten schlägt der Test aufgrund einer fehlenden Berechtigung fehl. Klicken Sie auf **Weiter**, um die Installation fortzusetzen.

2. Befolgen Sie die Anweisungen zum Abschließen der Installation. Wenn der aktuelle Benutzer nicht der Datenbankadministrator ist, wird während der Installation ein Dialogfeld angezeigt, in dem die Anmeldeinformationen eines Datenbankadministrators mit der **sysadmin**-Serverrollenberechtigung eingegeben werden müssen. Geben Sie die richtigen Anmeldeinformationen ein und klicken Sie auf **OK**, um mit der Installation fortzufahren. Bei der Installation wird die neue Datenbank für die Sitzungsaufzeichnung erstellt und das Maschinenkonto des Sitzungsaufzeichnungsservers als **db-owner** hinzugefügt.

Stellen Sie sicher, dass alle Voraussetzungen installiert sind, bevor Sie den Sitzungsaufzeichnungsserver installieren. Weitere Informationen finden Sie unter [Verwaltungskomponenten der Sitzungsaufzeichnung](#).

1. Geben Sie den Namen des SQL Servers im Textfeld "Datenbankinstanzname" ein. Wenn Sie eine benannte Instanz verwenden, geben Sie Maschinename\Instanzname ein, andernfalls geben Sie nur einen Maschinennamen ein.
2. Klicken Sie auf "Test", um die Verbindung mit dem SQL-Server zu testen. Stellen Sie sicher, dass der aktuelle Benutzer die öffentliche SQL Server-Rollenberechtigung hat, ansonsten schlägt der Test aufgrund einer fehlenden Berechtigung fehl. Klicken Sie dann auf **Weiter** und folgen Sie den Anweisungen, um die Installation abzuschließen.
3. Am Ende des Installations-Assistenten können Sie entscheiden, ob Sie sich an dem Programm zur Verbesserung der Benutzerfreundlichkeit von Citrix beteiligen möchten. Wenn Sie diesem Programm beitreten, werden anonyme Statistik- und Nutzungsinformationen an Citrix gesendet. Weitere Informationen finden Sie unter [Citrix Programm zur Verbesserung der Benutzerfreundlichkeit \(CEIP\)](#).

Installieren Sie den Sitzungsaufzeichnungsagent auf der Serverbetriebssystem-VDA-Maschine, auf der Sie Sitzungen aufzeichnen möchten.

1. Installieren Sie mit dem Server-Manager .NET Framework 3.5 und Microsoft Message Queuing (MSMQ) mit HTTP-Unterstützung auf dem XenApp 7.6-Serverbetriebssystem-VDA.
2. Starten Sie die Windows-Eingabeaufforderung als Administrator und führen Sie den folgenden Befehl aus:

```
msiexec /i SessionRecordingAgentx64.msi
```

oder doppelklicken Sie auf die MSI-Datei.

3. Klicken Sie im Installationsprogramm auf **Weiter** und akzeptieren Sie die Lizenzvereinbarung.
4. Geben Sie auf der Seite "Sitzungsaufzeichnungsagent - Konfiguration" den Namen des Computers ein, auf dem Sie den Sitzungsaufzeichnungsserver installiert haben, sowie das Protokoll und die Portinformationen für die Verbindung mit dem Sitzungsaufzeichnungsserver.

The screenshot shows the 'Citrix Session Recording Agent Setup' window, specifically the 'Session Recording Agent Configuration' page. The window has a blue title bar and a Citrix logo in the top right corner. Below the title bar, there is a header bar with the text 'Session Recording Agent Configuration'. The main area contains the following elements:

- A note: 'Enter the Session Recording Server connection information below. Note that you can modify these values at a later date by running the Session Recording Agent properties.'
- A text input field labeled 'Session Recording Server Name' with a placeholder example: 'Example: machine-name, machine-name.domain-name.net'.
- A 'Protocol' dropdown menu with 'HTTP' and 'HTTPS' options. 'HTTPS' is currently selected and highlighted in blue.
- A 'Port' text input field with the value '443' entered.
- A 'Test' button below the port field.
- Navigation buttons at the bottom: '< Back', 'Next >' (which is highlighted with a dashed border), and 'Cancel'.

Standardmäßig verwendet die Sitzungsaufzeichnung HTTPS/SSL für die sichere Kommunikation. Wenn SSL nicht konfiguriert ist, verwenden Sie HTTP. Navigieren Sie dazu zur Sitzungsaufzeichnungsbroker-Site und deaktivieren Sie SSL in der IIS-Verwaltungskonsole. Öffnen Sie die SSL-Einstellungen und deaktivieren Sie das Kontrollkästchen "SSL erfordern".

5. Für Feature Pack 2: Wenn Sie das Rollover-Feature verwenden möchten, installieren Sie Update ICATS760WX64010 (<http://support.citrix.com/article/CTX142037>). Für LTSR: Das Rollover-Feature ist in der LTSR-Version des VDA (7.6.300) enthalten. Wenn Ihr VDA auf LTSR-Stufe ist, ist das Feature verfügbar, ohne dass Sie eine weitere Aktion ausführen müssen.

6. Befolgen Sie die Anweisungen zum Abschließen der Installation.

Installieren Sie den Sitzungsaufzeichnungsplayer auf dem Sitzungsaufzeichnungsserver oder auf mindestens einer Arbeitsstation in der Domäne für Benutzer, die Sitzungsaufzeichnungen anzeigen.

- Führen Sie **SessionRecordingPlayer.msi** aus und folgen Sie den Anweisungen zum Durchführen der Installation.

Für ein Upgrade der Sitzungsaufzeichnung von XenApp 7.6 Feature Pack 1 oder 2 führen Sie das Installationsprogramm auf der Maschine aus, auf der Sie die Sitzungsaufzeichnungsfeatures installiert haben.

Befolgen Sie die Anweisungen zum Abschließen des Upgrades. Sie müssen während des Upgrades keine Informationen angeben.

### Checkliste vor der Installation

Stellen Sie vor der Installation sicher, dass Sie alle in dieser Liste aufgeführten Schritte abgeschlossen haben:

	Schritt
	Installieren Sie die Voraussetzungen, bevor Sie mit der Installation beginnen. Weitere Informationen finden Sie unter <a href="#">Systemanforderungen</a> .
	Wählen Sie die Computer aus, auf denen Sie die Komponenten der Sitzungsaufzeichnung installieren möchten, und stellen Sie sicher, dass jeder Computer die Hardware- und Softwareanforderungen für die zu installierenden Komponenten erfüllt.
	Laden Sie die ZIP-Datei der Sitzungsaufzeichnung von der Citrix Downloadseite unter <b>XenApp &gt; Components</b> herunter.
	Installieren Sie die relevanten Zertifikate in der Umgebung, wenn die Kommunikation zwischen den Komponenten der Sitzungsaufzeichnung mit dem SSL-Protokoll erfolgt. Weitere Informationen finden Sie unter <a href="#">Installieren von Zertifikaten</a> .
	Installieren Sie die für die Komponenten der Sitzungsaufzeichnung benötigten Hotfixes. Die Hotfixes sind unter <a href="#">Citrix Support</a> verfügbar.
	Konfigurieren Sie Director zum Erstellen und Aktivieren von Sitzungsaufzeichnungsrichtlinien.

### Hinweise:

- Citrix empfiehlt, die veröffentlichten Anwendungen in separate Bereitstellungsgruppen basierend auf den Aufzeichnungsrichtlinien aufzuteilen, da die Sitzungs freigabe veröffentlichter Anwendungen einen Konflikt mit aktiven Richtlinien verursachen kann, wenn sie in derselben Bereitstellungsgruppe sind. Die Sitzungsaufzeichnung ordnet die aktive Richtlinie der ersten veröffentlichten Anwendung zu, die ein Benutzer öffnet.
- Wenn Sie beabsichtigen, Maschinenerstellungsdienste (MCS) oder Provisioning Services mit XenApp zu verwenden, bereiten Sie den Server für eine eindeutige QMID vor. Weitere Informationen finden Sie in der Beschreibung unter Bekannte Probleme. Wenn Sie diesen Schritt nicht ausführen, verlieren Sie u. U. Aufzeichnungsdaten.



- Für SQL Server muss TCP/IP aktiviert sein und der SQL Server-Browserdienst sowie die Windows-Authentifizierung müssen ausgeführt werden.
- Wenn Sie HTTPS verwenden, konfigurieren Sie Serverzertifikate für SSL/HTTPS.

## Installationsdateien für die Sitzungsaufzeichnung

Sie benötigen die folgenden Installationsdateien von der Citrix Downloadseite:

- Verwaltungsdateien der Sitzungsaufzeichnung
  - Broker\_PowerShellSnapIn\_x64.msi
  - SessionRecordingAdministrationx64.msi
- Agentdateien der Sitzungsaufzeichnung
  - SessionRecordingAgentx64.msi
- Playerdateien der Sitzungsaufzeichnung
  - SessionRecordingPlayer.msi

Die Verwaltungskomponenten der Sitzungsaufzeichnung sind die Datenbank für die Sitzungsaufzeichnung, der Sitzungsaufzeichnungsserver und die Richtlinienkonsole für die Sitzungsaufzeichnung. Sie können festlegen, welche dieser Komponenten auf einem Server installiert werden.

Stellen Sie sicher, dass alle Voraussetzungen installiert sind, bevor Sie die Verwaltungskomponenten der Sitzungsaufzeichnung installieren. Weitere Informationen finden Sie unter [Verwaltungskomponenten der Sitzungsaufzeichnung](#).

Die Installation der Datenbank für die Sitzungsaufzeichnung auf einem lokalen SQL-Server erfordert die Anmeldung NT AUTHORITY\SYSTEM mit den SQL Server-Rollenberechtigungen sysadmin.

Die Installation der Datenbank für die Sitzungsaufzeichnung auf einem Remote-SQL-Server erfordert die Anmeldung unter dem Maschinenkonto, unter dem das Installationsprogramm ausgeführt wird, mit der SQL Server-Rollenberechtigungen sysadmin.

Zum Verbessern der Sicherheit können Sie diese Berechtigungen nach der Installation der Datenbank entfernen.

1. Führen Sie Broker\_PowerShellSnapIn\_x64.msi aus und folgen Sie den Anweisungen zum Durchführen der Installation.
2. Starten Sie die Windows-Eingabeaufforderung als Administrator und führen Sie den folgenden Befehl aus:

```
msiexec /i SessionRecordingAdministrationx64.msi
```

oder doppelklicken Sie auf die MSI-Datei.

3. Klicken Sie im Installationsprogramm auf **Weiter** und akzeptieren Sie die Lizenzvereinbarung.
4. Wählen Sie im Setupbildschirm der Sitzungsaufzeichnungsverwaltung die Verwaltungskomponenten aus, die Sie installieren möchten.

Stellen Sie sicher, dass alle Voraussetzungen installiert sind, bevor Sie die Datenbank für die Sitzungsaufzeichnung installieren. Weitere Informationen finden Sie unter [Verwaltungskomponenten der Sitzungsaufzeichnung](#).

1. Führen Sie auf der Seite "Datenbankkonfiguration" folgende Schritte aus:

- Wenn Sie alle Verwaltungskomponenten auf demselben Server installieren, geben Sie **localhost** im Feld "Name des Sitzungsaufzeichnungsservers" ein.
- Wenn Sie den Sitzungsaufzeichnungsserver und die Datenbank für die Sitzungsaufzeichnung auf verschiedenen Servern installieren, geben Sie den Namen des Computers, auf dem der Sitzungsaufzeichnungsserver gehostet wird, im folgenden Format ein: *Domäne\Computername*. Der Name des Sitzungsaufzeichnungsservers ist das Benutzerkonto für den Zugriff auf die Datenbank.

Wenn die Datenbankinstanz keine benannte Instanz ist, die Sie beim Setup der Instanz konfiguriert haben, können Sie nur den Maschinennamen des SQL Servers verwenden. Wenn Sie die Instanz benannt haben, verwenden Sie *Maschinennamen\Instanzname* als Datenbankinstanzname. Um den verwendeten Servernamen zu ermitteln, führen Sie **select @@servername** auf dem SQL Server aus. Der zurückgegebene Name ist der genaue Datenbankinstanzname.

2. Befolgen Sie die Anweisungen zum Abschließen der Installation. Bei der Installation wird die neue Datenbank für die Sitzungsaufzeichnung erstellt und das Maschinenkonto des Sitzungsaufzeichnungsservers als **db-owner** hinzugefügt.

## Installieren des Sitzungsaufzeichnungsservers

Stellen Sie sicher, dass alle Voraussetzungen installiert sind, bevor Sie den Sitzungsaufzeichnungsserver installieren. Weitere Informationen finden Sie unter [Verwaltungskomponenten der Sitzungsaufzeichnung](#).

- Geben Sie den Namen des SQL Servers im Textfeld Datenbankinstanzname ein. Wenn Sie eine benannte Instanz verwenden, geben Sie *Maschinennamen\Instanzname* ein, andernfalls geben Sie nur einen Maschinennamen ein.

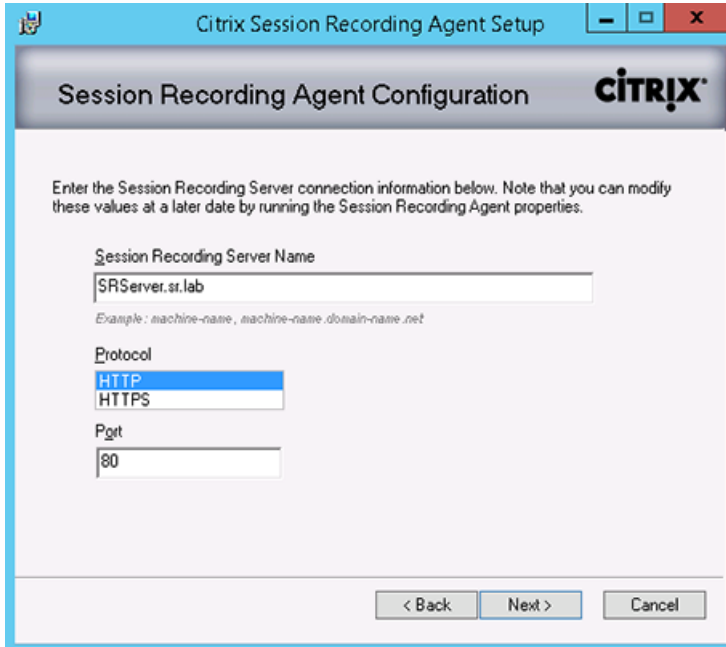
Installieren Sie den Sitzungsaufzeichnungsagent auf der Serverbetriebssystem-VDA-Maschine, auf der Sie Sitzungen aufzeichnen möchten.

1. Installieren Sie mit dem Server-Manager .NET Framework 3.5 und Microsoft Message Queuing (MSMQ) mit HTTP-Unterstützung auf dem XenApp 7.6-Serverbetriebssystem-VDA.
2. Starten Sie die Windows-Eingabeaufforderung als Administrator und führen Sie den folgenden Befehl aus:

```
msiexec /i SessionRecordingAgentx64.msi
```

oder doppelklicken Sie auf die MSI-Datei.

3. Klicken Sie im Installationsprogramm auf "Weiter" und akzeptieren Sie die Lizenzvereinbarung.
4. Geben Sie auf der Seite Sitzungsaufzeichnungsagent - Konfiguration den Namen des Computers ein, auf dem Sie den Sitzungsaufzeichnungsserver installiert haben, sowie das Protokoll und die Portinformationen für die Verbindung zum Sitzungsaufzeichnungsserver.



Standardmäßig verwendet die Sitzungsaufzeichnung HTTPS/SSL für die sichere Kommunikation. Wenn SSL nicht konfiguriert ist, verwenden Sie HTTP. Navigieren Sie dazu zur Sitzungsaufzeichnungsbroker-Site und deaktivieren Sie SSL in der IIS-Verwaltungskonsole. Öffnen Sie die SSL-Einstellungen und deaktivieren Sie das Kontrollkästchen SSL erfordern.

5. Wenn Sie das Rollover-Feature verwenden möchten, installieren Sie folgenden Hotfix:

<http://support.citrix.com/article/CTX142037>.

6. Befolgen Sie die Anweisungen zum Abschließen der Installation.

Installieren Sie den Sitzungsaufzeichnungsplayer auf dem Sitzungsaufzeichnungsserver oder auf mindestens einer Arbeitsstation in der Domäne für Benutzer, die Sitzungsaufzeichnungen anzeigen.

- Führen Sie SessionRecordingPlayer.msi aus und folgen Sie den Anweisungen zum Durchführen der Installation.

Sie entfernen Komponenten der Sitzungsaufzeichnung von einem Server oder einer Arbeitsstation über das Objekt "Programme und Funktionen" in der Windows-Systemsteuerung. Zum Entfernen der Datenbank für die Sitzungsaufzeichnung benötigen Sie die gleiche SQL Server-Rollenberechtigung sysadmin wie bei der Installation.

Sie können mit der Director-Konsole Sitzungsaufzeichnungsrichtlinien erstellen und aktivieren.

1. Zur Verwendung einer HTTPS-Verbindung installieren Sie das Zertifikat zum Vertrauen des Sitzungsaufzeichnungsservers

im Ordner mit den vertrauenswürdigen Stammzertifikaten des Director-Servers.

2. Zum Konfigurieren des Director-Servers für die Verwendung des Sitzungsaufzeichnungsservers führen Sie folgenden Befehl aus: `C:\inetpub\wwwroot\Director\tools\DirectorConfig.exe /configsessionrecording`
3. Geben Sie die IP-Adresse bzw. den FQDN des Sitzungsaufzeichnungsservers, die Portnummer und den Verbindungstyp (http/https) für die Verbindung zwischen Sitzungsaufzeichnungsagent und Sitzungsaufzeichnungsbroker auf dem Director-Server ein.

# Automatisieren der Installationen

May 10, 2016

Zur Installation des Sitzungsaufzeichnungsagents auf mehreren Servern können Sie ein Skript erstellen, das eine Installation ohne Benutzereingriffe ausführt.

Mit der folgenden Befehlszeile installieren Sie den Sitzungsaufzeichnungsagent und erstellen eine Protokolldatei, in der die Installationsinformationen aufgezeichnet werden.

`msiexec /i SessionRecordingAgentx64.msi sessionrecordingservername=yourservename sessionrecordingbrokerprotoco=yourbrokerprotocol sessionrecordingbrokerport=yourbrokerport /!v yourinstall;`  
Wobei Folgendes gilt:

`yourservename` ist der NetBIOS-Name oder FQDN des Computers, auf dem der Sitzungsaufzeichnungsserver ausgeführt wird. Wenn Sie keinen Namen eingeben, wird der Standardwert "localhost" verwendet.

`yourbrokerprotocol` ist entweder HTTP oder HTTPS und ist das Protokoll, mit dem der Sitzungsaufzeichnungsagent mit dem Sitzungsaufzeichnungsbroker kommuniziert. Wenn Sie keine Eingabe machen, wird als Standardwert HTTPS verwendet.

`yourbrokerport` ist eine Ganzzahl, die den Port angibt, über den der Sitzungsaufzeichnungsagent mit dem Sitzungsaufzeichnungsbroker kommuniziert. Wenn Sie keine Eingabe machen, wird als Standardwert 0 verwendet, d. h. der Sitzungsaufzeichnungsagent verwendet den Standardport für das ausgewählte Protokoll: 80 für HTTP oder 443 für HTTPS.

`/!v` legt die ausführliche Protokollierung fest.

Installationsprotokoll ist der Speicherort, an dem die Setupprotokolldatei erstellt wird.

`/q` legt den stillen Modus fest.

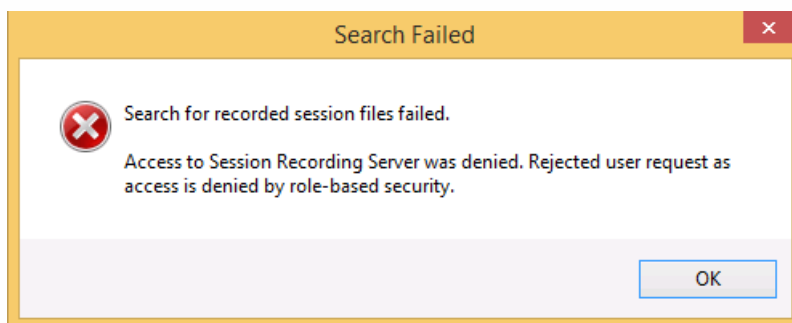
# Konfigurieren der Sitzungsaufzeichnung für die Wiedergabe und Aufzeichnung von Sitzungen

May 10, 2016

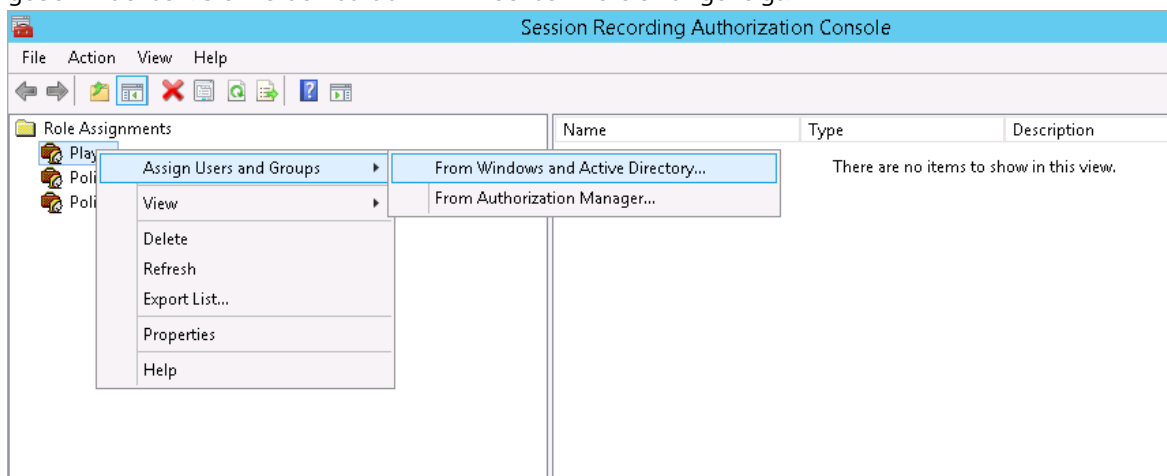
Nach der Installation der Sitzungsaufzeichnungskomponenten konfigurieren Sie die Sitzungsaufzeichnung mit den folgenden Schritten für die Aufzeichnung von XenApp-Sitzungen und das benutzerseitige Anzeigen der Sitzungen:

- Berechtigen von Benutzern zur Wiedergabe von Aufzeichnungen
- Berechtigen von Benutzern zur Verwaltung von Aufzeichnungsrichtlinien
- Ändern der aktiven Aufzeichnungsrichtlinie zu einer Richtlinie, die Sitzungsaufzeichnungen ermöglicht
- Konfigurieren des Sitzungsaufzeichnungsplayers für die Verbindung mit dem Sitzungsaufzeichnungsserver

Nach der Installation der Sitzungsaufzeichnung ist zunächst kein Benutzer berechtigt, Sitzungsaufzeichnungen wiederzugeben. Sie müssen jedem Benutzer, einschließlich des Administrators, Berechtigungen zuweisen. Ein Benutzer ohne Berechtigung zur Wiedergabe von Sitzungsaufzeichnungen erhält die folgende Fehlermeldung bei dem Versuch, eine Sitzungsaufzeichnung wiederzugeben:



1. Melden Sie sich bei dem Computer mit dem Sitzungsaufzeichnungsserver als Administrator an.
2. Starten Sie die Sitzungsaufzeichnungsautorisierungskonsole.
3. Klicken Sie in der Sitzungsaufzeichnungsautorisierungskonsole auf Player.
4. Wählen Sie die Benutzer und Gruppen aus, denen Sie Berechtigungen zur Wiedergabe von Sitzungsaufzeichnungen geben möchten. Sie werden daraufhin im rechten Bereich angezeigt.

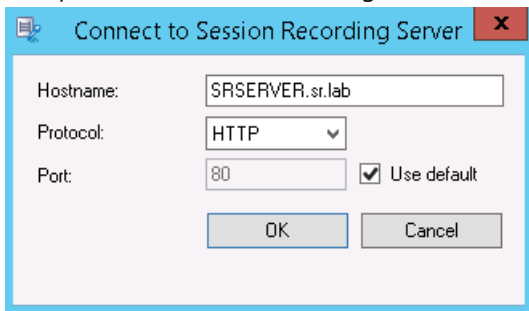


Bei der Installation der Sitzungsaufzeichnung gewähren standardmäßig Domänenadministratoren Berechtigung zum Steuern von Aufzeichnungsrichtlinien. Sie können die Autorisierungseinstellung ändern.

1. Melden Sie sich bei dem Computer mit dem Sitzungsaufzeichnungsserver als Administrator an.
2. Starten Sie die Sitzungsaufzeichnungsautorisierungskonsole und wählen Sie PolicyAdministrators aus.
3. Fügen Sie die Benutzer und Gruppen hinzu, die Aufzeichnungsrichtlinien verwalten sollen.

Die aktive Aufzeichnungsrichtlinie legt das Verhalten zur Sitzungsaufzeichnung auf allen Serverbetriebssystem-VDA's fest, auf denen der Sitzungsaufzeichnungsagent installiert ist und die eine Verbindung mit dem Sitzungsaufzeichnungsserver herstellen. Bei der Installation der Sitzungsaufzeichnung wird als aktive Richtlinie Nicht aufzeichnen verwendet. Sitzungen können erst aufgezeichnet werden, wenn Sie die aktive Aufzeichnungsrichtlinie ändern.

1. Melden Sie sich bei dem Server an, auf dem die Richtlinienkonsole für die Sitzungsaufzeichnung installiert ist, als autorisierter Richtlinienadministrator an.
2. Starten Sie die Richtlinienkonsole für die Sitzungsaufzeichnung.
3. Wenn das Dialogfeld Mit Sitzungsaufzeichnungsserver verbinden angezeigt wird, stellen Sie sicher, dass der Name des Computers, auf dem der Sitzungsaufzeichnungsserver gehostet wird, das Protokoll und der Port richtig sind.



4. Erweitern Sie in der Richtlinienkonsole für die Sitzungsaufzeichnung Aufzeichnungsrichtlinien. Die bei der Installation der Sitzungsaufzeichnung erstellten Aufzeichnungsrichtlinien werden angezeigt. Ein Häkchen gibt die aktive Richtlinie an:
  - Nicht aufzeichnen: Die Standardrichtlinie. Wenn Sie keine andere Richtlinie festlegen, wird keine Sitzung aufgezeichnet.
  - Alle mit Benachrichtigung aufzeichnen: Bei Auswahl dieser Richtlinie werden alle Sitzungen aufgezeichnet. Der Benutzer wird in einem Pop-up-Fenster über die Aufzeichnung informiert.
  - Alle ohne Benachrichtigung aufzeichnen: Bei Auswahl dieser Richtlinie werden alle Sitzungen aufgezeichnet. Der Benutzer wird nicht in einem Pop-up-Fenster über die Aufzeichnung informiert.
5. Wählen Sie die Richtlinie aus, die Sie als aktive Richtlinie verwenden möchten.
6. Klicken Sie im Menü auf Aktion > Richtlinie aktivieren.

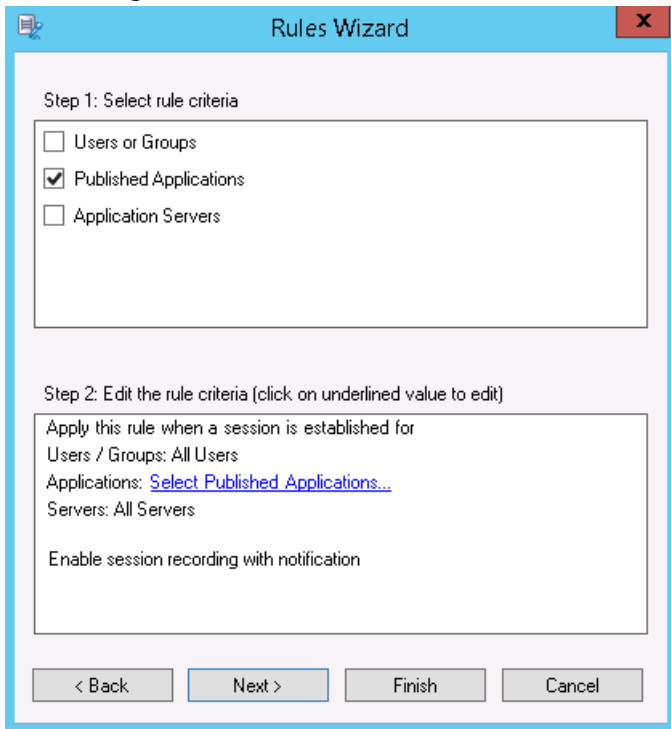
Hinweis: Sie können in der Sitzungsaufzeichnung eigene Aufzeichnungsrichtlinien erstellen. Die erstellten Aufzeichnungsrichtlinien werden im Ordner "Aufzeichnungsrichtlinien" in der Richtlinienkonsole für die Sitzungsaufzeichnung angezeigt.

Die generische Aufzeichnungsrichtlinie entspricht u. U. nicht Ihren Anforderungen. Sie können Richtlinien basierend auf Benutzern, VDA-Servern und Anwendungen konfigurieren.

Wichtig: Eine Richtlinie kann viele Regeln enthalten, aber es kann jeweils nur eine aktive Richtlinie ausgeführt werden.

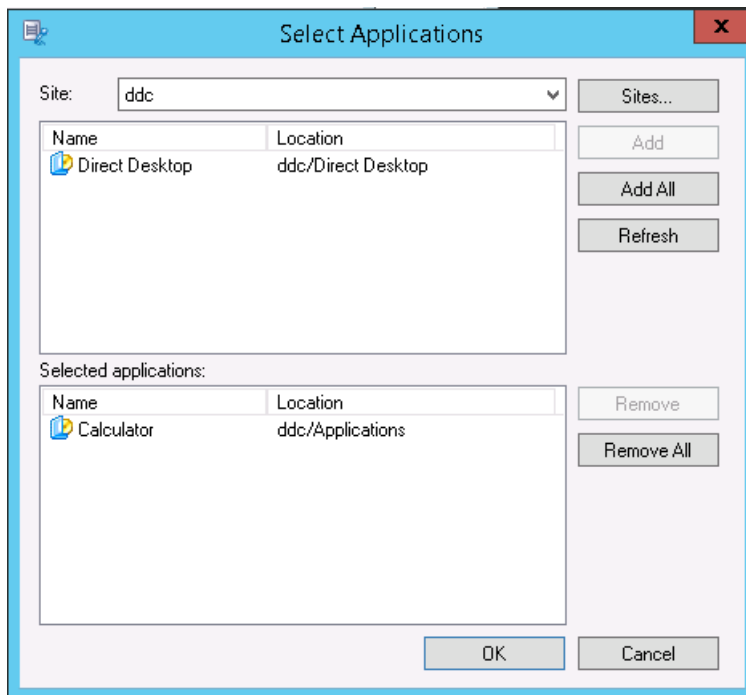
1. Melden Sie sich bei dem Server an, auf dem die Richtlinienkonsole für die Sitzungsaufzeichnung installiert ist, als autorisierter Richtlinienadministrator an.

2. Starten Sie die Richtlinienkonsole für die Sitzungsaufzeichnung und wählen Sie Aufzeichnungsrichtlinien > Neue Richtlinie hinzufügen.
3. Klicken Sie mit der rechten Maustaste auf Neue Richtlinie und wählen Sie Neue Regel hinzufügen.
4. Wählen Sie im Regelasistenten die Option Sitzungsaufzeichnung mit Benachrichtigung aktivieren und klicken Sie auf Weiter.
5. Aktivieren Sie das Kontrollkästchen Veröffentlichte Anwendungen und klicken Sie auf den Link für Veröffentlichte Anwendungen auswählen.



6. Klicken Sie im Bildschirm "Anwendungen auswählen" auf Sites und Hinzufügen.
7. Geben Sie den Servernamen eines XenApp 7.6 FP1 Delivery Controllers ein.
8. Klicken Sie auf Verbinden und die Site wird enumeriert.
9. Klicken Sie auf Schließen und eine Liste der veröffentlichten Anwendungen wird angezeigt. Fügen Sie einige der Anwendungen aus der Liste hinzu, klicken Sie auf OK und Fertig stellen.





10. Klicken Sie mit der rechten Maustaste auf die Richtlinie und wählen Sie Aktivieren. Bei Bedarf können Sie die Richtlinie umbenennen.

Damit der Sitzungsaufzeichnungsplayer Sitzungen wiedergeben kann, müssen Sie die Verbindung mit dem Sitzungsaufzeichnungsserver konfigurieren, auf dem die Sitzungsaufzeichnungen gespeichert sind. Jeder Sitzungsaufzeichnungsplayer kann eine Verbindung mit mehreren Sitzungsaufzeichnungsservern herstellen, jedoch kann nur jeweils eine Verbindung aktiv sein. Wenn ein Sitzungsaufzeichnungsplayer eine Verbindung mit mehreren Sitzungsaufzeichnungsservern herstellen kann, können Benutzer den gewünschten Sitzungsaufzeichnungsserver durch Auswahl eines Kontrollkästchen festlegen.

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Starten Sie den Sitzungsaufzeichnungsplayer.
3. Wählen Sie auf der Menüleiste des Sitzungsaufzeichnungsplayers Extras > Optionen aus.
4. Klicken Sie auf der Registerkarte Verbindungen auf Hinzufügen.
5. Geben Sie im Feld Hostname den Namen oder die IP-Adresse des Computers mit dem Sitzungsaufzeichnungsserver ein und wählen Sie das Protokoll aus. Standardmäßig verwendet die Sitzungsaufzeichnung HTTPS/SSL für die sichere Kommunikation. Wenn SSL nicht konfiguriert ist, wählen Sie "HTTP".
6. Wenn der Sitzungsaufzeichnungsplayer eine Verbindung mit mehreren Sitzungsaufzeichnungsservern herstellen kann, wiederholen Sie Schritt 4 und 5 für jeden Server.
7. Stellen Sie sicher, dass das Kontrollkästchen für den Sitzungsaufzeichnungsserver aktiviert ist, mit dem Sie eine Verbindung herstellen möchten.

# Gewähren von Zugriffsrechten für Benutzer

May 10, 2016

Hinweis: Aus Sicherheitsgründen sollten Sie den Benutzern nur die Rechte geben, die sie zum Ausführen bestimmter Funktionen benötigen, z. B. zum Anzeigen aufgezeichneter Sitzungen.

Sie weisen den Benutzern in der Sitzungsaufzeichnungsautorisierungskonsole auf dem Sitzungsaufzeichnungsserver Rollen zu und gewähren den Sitzungsaufzeichnungsbenutzern Rechte. Sitzungsaufzeichnungsbenutzer haben drei Rollen:

- **Player:** Mitglieder der Rolle können aufgezeichnete XenApp-Sitzungen anzeigen. Für diese Rolle besteht keine Standardeinstellung.
- **PolicyQuery:** Die Server mit dem Sitzungsaufzeichnungsagent können Auswertungen der Aufzeichnungsrichtlinie anfordern. In der Standardeinstellung sind authentifizierte Benutzer Mitglieder dieser Rolle.
- **PolicyAdministrator:** Mitglieder dieser Rolle können Aufzeichnungsrichtlinien anzeigen, erstellen, bearbeiten, löschen und aktivieren. In der Standardeinstellung sind Administratoren des Computers, auf dem der Sitzungsaufzeichnungsserver ausgeführt wird, Mitglieder dieser Rolle.

Die Sitzungsaufzeichnung unterstützt in Active Directory definierte Benutzer und Gruppen.

1. Melden Sie sich bei dem Computer, auf dem der Sitzungsaufzeichnungsserver ausgeführt wird, als Administrator oder Richtlinienadministrator an.
2. Starten Sie die Sitzungsaufzeichnungsautorisierungskonsole.
3. Wählen Sie die Rolle, der Sie Benutzer zuordnen möchten.
4. Klicken Sie auf Aktion > Windows-Benutzer und -Gruppen zuweisen.
5. Fügen Sie die Benutzer und Gruppen hinzu.

In der Konsole vorgenommene Änderungen werden beim Update (das jede Minute erfolgt) übernommen.

# Erstellen und Aktivieren von Aufzeichnungsrichtlinien

May 10, 2016

Mit der Richtlinienkonsole für die Sitzungsaufzeichnung erstellen und aktivieren Sie Richtlinien, die festlegen, wie Sitzungen aufgezeichnet werden.

Sie können verfügbare Systemrichtlinien nach der Installation der Sitzungsaufzeichnung aktivieren oder eigene Richtlinien erstellen und aktivieren. Die Systemrichtlinien für die Sitzungsaufzeichnung wenden eine Regel auf alle Benutzer, veröffentlichten Anwendungen und Server an. In benutzerdefinierten Richtlinien geben Sie an, welche Benutzer, veröffentlichten Anwendungen und Server aufgezeichnet werden.

Die aktive Richtlinie legt fest, welche Sitzungen aufgezeichnet werden. Nur jeweils eine Richtlinie ist aktiv.

Die Sitzungsaufzeichnung enthält die folgenden Systemrichtlinien:

- **Nicht aufzeichnen:** Bei Auswahl dieser Richtlinie werden keine Sitzungen aufgezeichnet. Dies ist die Standardrichtlinie. Wenn Sie keine andere Richtlinie angeben, werden keine Sitzungen aufgezeichnet.
- **Alle mit Benachrichtigung aufzeichnen:** Bei Auswahl dieser Richtlinie werden alle Sitzungen aufgezeichnet. Der Benutzer wird in einem Popupfenster über die Aufzeichnung informiert.
- **Alle ohne Benachrichtigung aufzeichnen:** Bei Auswahl dieser Richtlinie werden alle Sitzungen aufgezeichnet. Der Benutzer wird nicht in einem Popupfenster über die Aufzeichnung informiert.

Systemrichtlinien können nicht geändert oder gelöscht werden.

1. Melden Sie sich bei dem Server an, auf dem die Richtlinienkonsole für die Sitzungsaufzeichnung installiert ist.
2. Starten Sie die Richtlinienkonsole für die Sitzungsaufzeichnung.
3. Wenn das Dialogfeld Mit Sitzungsaufzeichnungsserver verbinden angezeigt wird, stellen Sie sicher, dass der Name des Sitzungsaufzeichnungsservers, das Protokoll und der Port richtig sind. Klicken Sie auf OK.
4. Erweitern Sie in der Richtlinienkonsole für die Sitzungsaufzeichnung Aufzeichnungsrichtlinien.
5. Wählen Sie die Richtlinie aus, die Sie als aktive Richtlinie verwenden möchten.
6. Klicken Sie im Menü auf Aktion > Richtlinie aktivieren.

Wenn Sie eine eigene Richtlinie erstellen, legen Sie Regeln fest und geben an, welche Sitzungen von Benutzern und Gruppen, veröffentlichten Anwendungen und Servern aufgezeichnet werden. Ein Assistent in der Richtlinienkonsole für die Sitzungsaufzeichnung unterstützt Sie beim Erstellen der Regeln. Zum Abrufen einer Liste der veröffentlichten Anwendungen und Server ist Siteadministrator-Leseberechtigung erforderlich. Konfigurieren Sie dies auf dem Delivery Controller der Site.

Für jede erstellte Regel geben Sie eine Aufzeichnungsaktion und ein Regelkriterium an. Die Aufzeichnungsaktion gilt für Sitzungen, die das Regelkriterium erfüllen.

Wählen Sie für jede Regel eine Aufzeichnungsaktion aus:

- Nicht aufzeichnen: (Wählen Sie Sitzungsaufzeichnung deaktivieren im Assistenten für Regeln.) Diese

Aufzeichnungsaktion gibt an, dass Sitzungen, die das Regelkriterium erfüllen, nicht aufgezeichnet werden.

- Mit Benachrichtigung aufzeichnen: (Wählen Sie Sitzungsaufzeichnung mit Benachrichtigung aktivieren im Assistenten für Regeln.) Diese Aufzeichnungsaktion gibt an, dass Sitzungen, die das Regelkriterium erfüllen, nicht aufgezeichnet werden. Der Benutzer wird in einem Pop-up-Fenster über die Aufzeichnung informiert.
- Ohne Benachrichtigung aufzeichnen: (Wählen Sie Sitzungsaufzeichnung ohne Benachrichtigung aktivieren im Assistenten für Regeln.) Diese Aufzeichnungsaktion gibt an, dass Sitzungen, die das Regelkriterium erfüllen, nicht aufgezeichnet werden. Benutzer werden nicht über die Aufzeichnung informiert.

Wählen Sie für jede Regel mindestens eines der folgenden Elemente, um ein Regelkriterium zu erstellen:

- Benutzer oder Gruppen: Sie erstellen eine Liste der Benutzer oder Gruppen, für die die Aufzeichnungsaktion der Regel gilt.
- Veröffentlichte Anwendungen: Sie erstellen eine Liste der veröffentlichten Anwendungen, auf die die Aufzeichnungsaktion der Regel angewendet wird. Wählen Sie im Assistenten für Regeln die XenApp-Sites aus, auf denen die Anwendungen verfügbar sind.
- Anwendungsserver: Sie erstellen eine Liste der Serverbetriebssystemmaschinen, auf die die Aufzeichnungsaktion der Regel angewendet wird. Wählen Sie im Assistenten für Regeln die XenApp-Sites, in denen sich die Server befinden.

Wenn Sie mehrere Regeln in einer Aufzeichnungsrichtlinie erstellen, können einige Sitzungen die Kriterien für mehrere Regeln erfüllen. In diesen Situationen wird die Regel mit der höchsten Priorität auf die Sitzung angewendet.

Die Aufzeichnungsaktion einer Regel legt die Priorität fest:

- Regeln mit der Aktion Sitzungsaufzeichnung deaktivieren haben die höchste Priorität.
- Regeln mit der Aktion Sitzungsaufzeichnung mit Benachrichtigung aktivieren haben die nächsthöhere Priorität.
- Regeln mit der Aktion Sitzungsaufzeichnung ohne Benachrichtigung aktivieren haben die niedrigste Priorität.

Einige Sitzungen erfüllen ggf. kein Regelkriterium in einer Aufzeichnungsrichtlinie. Für diese Sitzungen gilt die Aufzeichnungsaktion der Fallbackregel der Richtlinie. Die Aufzeichnungsaktion der Fallbackregel ist immer Nicht aufzeichnen. Die Fallbackregel kann nicht geändert oder gelöscht werden.

## Verwenden von Active Directory-Gruppen

Beim Erstellen von Richtlinien können Sie in der Sitzungsaufzeichnung Active Directory-Gruppen verwenden. Active Directory-Gruppen statt einzelner Benutzer vereinfachen die Erstellung und Verwaltung von Regeln und Richtlinien. Beispiel: Wenn Benutzer in der Buchhaltungsabteilung des Unternehmens zur Active Directory-Gruppe "Finanz" gehören, können Sie eine Regel erstellen, die für alle Mitglieder dieser Gruppe gilt, indem Sie die Gruppe "Finanz" beim Erstellen der Regel im Assistenten für Regeln auswählen.

## Weißer Liste der Benutzer

Sie können Richtlinien für die Sitzungsaufzeichnung erstellen, die sicherstellen, dass die Sitzungen bestimmter Benutzer im Unternehmen nie aufgezeichnet werden. Dies wird weiße Liste der Benutzer genannt. Eine weiße Liste der Benutzer ist beispielsweise für Benutzer geeignet, die mit vertraulichen Informationen arbeiten, oder in Situationen, in denen das Unternehmen die Sitzungen einer bestimmten Mitarbeitergruppe nicht aufzeichnen möchte.

Wenn beispielsweise alle Mitglieder der Geschäftsleitung im Unternehmen zu einer Active Directory-Gruppe "Geschäftsführung" gehören, können Sie sicherstellen, dass die Sitzungen dieser Benutzer nie aufgezeichnet werden, wenn Sie eine Regel erstellen, mit der die Sitzungsaufzeichnung für die Gruppe "Geschäftsführung" deaktiviert wird. Während die Richtlinie, die diese Regel enthält, aktiv ist, werden keine Sitzungen der Mitglieder der Gruppe "Geschäftsführung" aufgezeichnet. Die Sitzungen anderer Mitarbeiter im Unternehmen werden basierend auf den anderen Regeln in der aktiven

Richtlinie aufgezeichnet.

## Erstellen einer Richtlinie

Hinweis: Bei der Verwendung des Richtlinienassistenten werden Sie möglicherweise aufgefordert, zum Bearbeiten auf einen unterstrichenen Wert zu klicken, obwohl kein unterstrichener Wert angezeigt wird. Unterstrichene Werte werden nur in bestimmten Situationen angezeigt. Ignorieren Sie den Schritt, wenn kein unterstrichener Wert angezeigt wird.

1. Melden Sie sich bei dem Server an, auf dem die Richtlinienkonsole für die Sitzungsaufzeichnung installiert ist.
2. Starten Sie die Richtlinienkonsole für die Sitzungsaufzeichnung.
3. Wenn das Dialogfeld Mit Sitzungsaufzeichnungsserver verbinden angezeigt wird, stellen Sie sicher, dass der Name des Sitzungsaufzeichnungsservers, das Protokoll und der Port richtig sind. Klicken Sie auf OK.
4. Wählen Sie in der Richtlinienkonsole für die Sitzungsaufzeichnung Aufzeichnungsrichtlinien.
5. Klicken Sie auf der Menüleiste auf Aktion > Neue Richtlinie hinzufügen. Im linken Bereich wird eine Neue Richtlinie genannte Richtlinie angezeigt.
6. Markieren Sie die neue Richtlinie und klicken Sie auf der Menüleiste auf Aktion > Umbenennen.
7. Geben Sie einen Namen für die Richtlinie ein, die Sie erstellen, und drücken Sie die Eingabetaste oder klicken Sie außerhalb des neuen Namens.
8. Markieren Sie die Richtlinie und klicken Sie auf der Menüleiste auf Aktion > Neue Regel hinzufügen, um den Assistenten für Regeln zu starten.
9. Folgen Sie den Anweisungen, um die Regeln für diese Richtlinie zu erstellen.

## Ändern von Richtlinien

1. Melden Sie sich bei dem Server an, auf dem die Richtlinienkonsole für die Sitzungsaufzeichnung installiert ist.
2. Starten Sie die Richtlinienkonsole für die Sitzungsaufzeichnung.
3. Wenn das Dialogfeld Mit Sitzungsaufzeichnungsserver verbinden angezeigt wird, stellen Sie sicher, dass der Name des Sitzungsaufzeichnungsservers, das Protokoll und der Port richtig sind. Klicken Sie auf OK.
4. Erweitern Sie in der Richtlinienkonsole für die Sitzungsaufzeichnung Aufzeichnungsrichtlinien.
5. Wählen Sie die Richtlinie aus, die Sie ändern möchten. Die Regeln für die Richtlinie werden im rechten Bereich angezeigt.
6. Fügen Sie eine Regel hinzu, ändern oder löschen eine Regel:
  - Klicken Sie auf der Menüleiste auf Aktion > Neue Regel hinzufügen. Wenn die Richtlinie aktiv ist, werden Sie in einem Pop-upfenster zum Bestätigen der Aktion aufgefordert. Erstellen Sie mit dem Assistenten für Regeln eine neue Regel.
  - Markieren Sie die Regel, die Sie ändern möchten, klicken Sie mit der rechten Maustaste und wählen Sie Eigenschaften. Ändern Sie die Regel mit dem Assistenten für Regeln.
  - Markieren Sie die Regel, die Sie löschen möchten, klicken Sie mit der rechten Maustaste und wählen Sie Regel löschen.

## Löschen von Richtlinien

Hinweis: Eine Systemrichtlinie oder aktive Richtlinie kann nicht gelöscht werden.

1. Melden Sie sich bei dem Server an, auf dem die Richtlinienkonsole für die Sitzungsaufzeichnung installiert ist.
2. Starten Sie die Richtlinienkonsole für die Sitzungsaufzeichnung.
3. Wenn das Dialogfeld Mit Sitzungsaufzeichnungsserver verbinden angezeigt wird, stellen Sie sicher, dass der Name des Sitzungsaufzeichnungsservers, das Protokoll und der Port richtig sind. Klicken Sie auf OK.
4. Erweitern Sie in der Richtlinienkonsole für die Sitzungsaufzeichnung Aufzeichnungsrichtlinien.
5. Wählen Sie im linken Bereich die Richtlinie aus, die Sie löschen möchten. Wenn die Richtlinie aktiv ist, müssen Sie eine andere aktivieren.
6. Klicken Sie auf der Menüleiste auf Aktion > Richtlinie löschen.
7. Klicken Sie auf Ja, um die Aktion zu bestätigen.

Wenn Sie eine Richtlinie aktivieren, bleibt die vorher aktive Richtlinie bis zum Ende der Benutzersitzung in Kraft. Manchmal wird die neue Richtlinie jedoch gültig, wenn ein Dateirollover erfolgt. Ein Dateirollover tritt auf, wenn das Limit für die maximale Größe erreicht wird. Weitere Informationen zur maximalen Dateigröße für Aufzeichnungen finden Sie unter [Angeben der Dateigröße für Aufzeichnungen](#).

In der folgenden Tabelle werden die Vorgänge beschrieben, die beim Anwenden einer neuen Richtlinie auftreten, während eine Sitzung aufgezeichnet wird und ein Rollover erfolgt:

Vorherige Richtlinie:	Neue Richtlinie:	Richtlinie nach Rollover:
Nicht aufzeichnen:	Jede andere Richtlinie	Keine Änderung. Die neue Richtlinie wird nur gültig, wenn sich der Benutzer an einer neuen Sitzung anmeldet.
Ohne Benachrichtigung aufzeichnen	Nicht aufzeichnen:	Aufzeichnung wird gestoppt.
	Mit Benachrichtigung aufzeichnen	Aufzeichnung wird fortgesetzt, und eine Benachrichtigung wird angezeigt.
Mit Benachrichtigung aufzeichnen	Nicht aufzeichnen:	Aufzeichnung wird gestoppt.
	Ohne Benachrichtigung aufzeichnen	Aufzeichnung wird fortgesetzt. Bei der nächsten Anmeldung des Benutzers wird keine Meldung angezeigt.

# Deaktivieren oder Aktivieren der Aufzeichnung

May 10, 2016

Der Sitzungsaufzeichnungsagent wird auf jeder Serverbetriebssystemmaschine installiert, auf der Sie Sitzungen aufzeichnen möchten. Jeder Agent hat eine Einstellung, mit der die Aufzeichnung auf dem Server aktiviert wird, auf dem die Agentsoftware installiert ist. Nach dem Aktivieren der Aufzeichnung wertet die Sitzungsaufzeichnung die aktive Aufzeichnungsrichtlinie aus, mit der festgelegt wird, welche Sitzungen aufgezeichnet werden.

Bei Installation des Sitzungsaufzeichnungsagents ist die Aufzeichnung aktiviert. Citrix empfiehlt, dass Sie die Sitzungsaufzeichnung auf Servern deaktivieren, für die keine Aufzeichnung vorgesehen ist, da die Anwendung die Serverleistung geringfügig beeinträchtigt, selbst wenn keine Aufzeichnung erfolgt.

1. Melden Sie sich bei dem Server an, auf dem der Sitzungsaufzeichnungsagent installiert ist.
2. Klicken Sie im Startmenü auf Sitzungsaufzeichnungsagent - Eigenschaften.
3. Aktivieren oder deaktivieren Sie unter Sitzungsaufzeichnung das Kontrollkästchen Sitzungsaufzeichnung für diesen XenApp-Server aktivieren, um anzugeben, ob Sitzungen für den Server aufgezeichnet werden sollen.
4. Starten Sie den Sitzungsaufzeichnungsagent-Dienst neu, um die Änderung zu übernehmen.

Hinweis: Bei der Installation der Sitzungsaufzeichnung wird als aktive Richtlinie Nicht aufzeichnen (auf keinem Server werden Sitzungen aufgezeichnet) verwendet. Aktivieren Sie in der Richtlinienkonsole für die Sitzungsaufzeichnung eine andere Richtlinie, um die Aufzeichnung zu aktivieren.

# Konfigurieren der Verbindung mit dem Sitzungsaufzeichnungsserver

May 10, 2016

Die Verbindung zwischen Sitzungsaufzeichnungsagent und dem Sitzungsaufzeichnungsserver wird normalerweise bei der Installation des Agents konfiguriert. Die Verbindung kann nach der Installation des Sitzungsaufzeichnungsagents in dessen Eigenschaften konfiguriert werden.

1. Melden Sie sich bei dem Server an, auf dem der Sitzungsaufzeichnungsagent installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsagent - Eigenschaften.
3. Klicken Sie auf die Registerkarte Verbindungen.
4. Geben Sie im Feld Sitzungsaufzeichnungsserver den Namen oder die IP-Adresse des Servers ein.
5. Wählen Sie im Bereich Nachrichtenwarteschlange des Speichermanagers der Sitzungsaufzeichnung das Protokoll aus, das der Speichermanager der Sitzungsaufzeichnung für die Kommunikation verwendet. Sie können auch die Standardportnummer ändern.
6. Akzeptieren Sie im Feld Nachrichtenlebensdauer den Standard von 7200 Sekunden (zwei Stunden) oder geben Sie einen neuen Wert für die Anzahl der Sekunden ein, für die jede Nachricht in der Warteschlange gespeichert wird, wenn ein Kommunikationsfehler auftritt. Nach dem Ablauf dieses Zeitraums wird die Nachricht gelöscht und die Datei kann nur bis an die Stelle wiedergegeben werden, an der die Daten verloren wurden.
7. Wählen Sie im Bereich Sitzungsaufzeichnungsbroker das Protokoll aus, das der Sitzungsaufzeichnungsbroker für die Kommunikation verwendet. Sie können auch die Standardportnummer ändern.
8. Starten Sie den Sitzungsaufzeichnungsagent-Dienst neu, um die Änderung zu übernehmen.



# Erstellen von Benachrichtigungen

May 10, 2016

Wenn in der aktiven Aufzeichnungsrichtlinie festgelegt ist, dass Benutzer über das Aufzeichnen der Sitzung benachrichtigt werden, wird ein Pop-up-Fenster mit einer Benachrichtigung angezeigt, nachdem die Benutzer die Anmeldeinformationen eingegeben haben. Die folgende Meldung ist die Standardbenachrichtigung: "Ihre Aktivität mit den von Ihnen vor kurzem gestarteten Programmen wird aufgezeichnet. Schließen Sie alle Programme, wenn Sie dies ablehnen." Der Benutzer klickt auf OK und setzt die Sitzung fort.

Die Standardbenachrichtigung wird in der Sprache des Betriebssystems des Computers angezeigt, auf dem der Sitzungsaufzeichnungsserver ausgeführt wird.

Sie können benutzerdefinierte Benachrichtigungen in der gewünschten Sprache erstellen; jedoch ist nur eine Benachrichtigung pro Sprache möglich. Den Benutzern wird die Benachrichtigung in der Sprache angezeigt, die den lokalen Einstellungen des Benutzers entspricht.

1. Melden Sie sich bei dem Computer mit dem Sitzungsaufzeichnungsserver an.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsserver - Eigenschaften.
3. Klicken Sie unter Sitzungsaufzeichnungsserver - Eigenschaften auf die Registerkarte Benachrichtigungen.
4. Klicken Sie auf Hinzufügen.
5. Wählen Sie die Sprache für die Benachrichtigung und geben Sie den Text ein. Sie können nur eine Nachricht für jede Sprache erstellen.

Nach der Annahme und Aktivierung wird die neue Benachrichtigung im Feld Sprachspezifische Benachrichtigungen angezeigt.

# Aktivieren der Aufzeichnung benutzerdefinierter Ereignisse

May 10, 2016

In der Sitzungsaufzeichnung können Sie mit Anwendungen von Drittanbietern benutzerdefinierte Daten, so genannte Ereignisse, in die Sitzungsaufzeichnungen einfügen. Diese Ereignisse werden angezeigt, wenn die Sitzung mit dem Sitzungsaufzeichnungsplayer wiedergegeben wird. Die Ereignisse sind Teil der Sitzungsaufzeichnungsdatei und können nach dem Aufzeichnen der Sitzung nicht geändert werden.

Ein Ereignis kann beispielsweise den folgenden Text enthalten: Jedes Mal, wenn ein Benutzer einen Browser in einer Sitzung öffnet, die aufgezeichnet wird, wird der Text zu diesem Zeitpunkt in die Aufzeichnung eingefügt. Wenn die Sitzung mit dem Sitzungsaufzeichnungsplayer wiedergegeben wird, kann der Benutzer die Anzahl der Marker in der Liste Ereignisse und Textmarken im Player notieren, und damit schnell die Stellen suchen und zählen, an denen der Benutzer einen Browser geöffnet hat.

So fügen Sie benutzerdefinierte Ereignisse in Aufzeichnungen auf einem Server ein

- Aktivieren Sie über die Eigenschaften des Sitzungsaufzeichnungsagents eine Einstellung auf jedem Server, auf dem Sie benutzerdefinierte Ereignisse einfügen möchten. Sie müssen jeden Server einzeln aktivieren. Das globale Aktivieren aller Server in einer Site ist nicht möglich.
- Entwickeln Sie Anwendungen, die auf der Event-API basieren, die in der XenApp-Sitzung jedes Benutzers ausgeführt werden (zum Einfügen der Daten in die Aufzeichnung).

Bei der Installation der Sitzungsaufzeichnung wird auch eine Ereignisaufzeichnungs-COM-Anwendung (API) installiert, mit der Sie Text von Anwendungen von Drittherstellern in die Aufzeichnung einfügen können. Sie können die API von vielen Programmiersprachen verwenden, u. a. Visual Basic, C++ oder C#. Die DLL-Datei der Sitzungsaufzeichnungs-Event-API ist Teil der Installation der Sitzungsaufzeichnung. Die Datei ist unter `C:\Programme\Citrix\SessionRecording\Agent\Bin\Interop.UserApi.dll` gespeichert.

So aktivieren Sie die Aufzeichnung von benutzerdefinierten Ereignissen auf einem Server

1. Melden Sie sich bei dem Server an, auf dem der Sitzungsaufzeichnungsagent installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsagent - Eigenschaften.
3. Klicken Sie unter Sitzungsaufzeichnungsagent - Eigenschaften auf die Registerkarte Aufzeichnung.
4. Aktivieren Sie unter Benutzerdefinierte Ereignisaufzeichnung das Kontrollkästchen Anwendungen von Drittherstellern können benutzerdefinierte Daten auf diesem XenApp-Server aufzeichnen.

# Aktivieren oder Deaktivieren der Wiedergabe von Livesitzungen

May 10, 2016

Mit dem Sitzungsaufzeichnungsplayer können Sie eine Sitzung nach oder während der Aufzeichnung anzeigen. Wenn Sie eine Sitzung anzeigen, die momentan aufgezeichnet wird, ähnelt es der Liveanzeige von Aktionen. Es besteht jedoch eine Verzögerung von bis zu zwei Sekunden, bis die Daten vom XenApp-Server zur Arbeitsstation übermittelt werden.

Einige Funktionen stehen nicht zur Verfügung, wenn Sie Sitzungen anzeigen, deren Aufzeichnung noch nicht abgeschlossen ist:

- Eine digitale Signatur kann erst nach dem Abschluss der Aufzeichnung zugewiesen werden. Wenn die digitale Signatur aktiviert ist, können Sie Livesitzungen wiedergeben, die jedoch nicht signiert sind. Zertifikate können erst nach dem Abschluss der Aufzeichnung angezeigt werden.
- Der Wiedergabeschutz kann erst nach dem Abschluss der Aufzeichnung angewendet werden. Wenn der Wiedergabeschutz aktiviert ist, können Sie Livesitzungen wiedergeben, die jedoch erst nach dem Abschluss der Aufzeichnung verschlüsselt werden.
- Eine Datei kann erst nach dem Abschluss der Aufzeichnung zwischengespeichert werden.

In der Standardeinstellung ist die Wiedergabe von Livesitzungen aktiviert.

1. Melden Sie sich bei dem Computer mit dem Sitzungsaufzeichnungsserver an.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsserver - Eigenschaften.
3. Klicken Sie unter Sitzungsaufzeichnungsserver - Eigenschaften auf die Registerkarte Wiedergabe.
4. Aktivieren Sie das Kontrollkästchen Wiedergabe von Livesitzungen zulassen oder heben Sie die Markierung auf.

# Aktivieren oder Deaktivieren des Wiedergabeschutzes

May 10, 2016

Als Sicherheitsmaßnahme werden Aufzeichnungsdateien automatisch von der Sitzungsaufzeichnung verschlüsselt, bevor sie zum Anzeigen im Sitzungsaufzeichnungsplayer heruntergeladen werden. Dieser Wiedergabeschutz stellt sicher, dass die Dateien nur von dem Benutzer, der die Datei heruntergeladen hat, und nicht von anderen Benutzern kopiert oder angezeigt werden können. Die Wiedergabe der Dateien kann nicht auf einer anderen Arbeitsstation oder von einem anderen Benutzer durchgeführt werden. Verschlüsselte Dateien haben eine ICLE-Erweiterung, nicht verschlüsselte Dateien haben eine ICL-Erweiterung. Die Dateien bleiben während der Speicherung im Cache auf der Arbeitsstation mit dem Sitzungsaufzeichnungsplayer verschlüsselt, bis sie von einem autorisierten Benutzer geöffnet werden.

Citrix empfiehlt die Verwendung von HTTPS für den Schutz der übermittelten Daten.

In der Standardeinstellung ist der Wiedergabeschutz aktiviert.

1. Melden Sie sich bei dem Computer mit dem Sitzungsaufzeichnungsserver an.
2. Wählen Sie im Menü Start die Option Sitzungsaufzeichnungsserver - Eigenschaften.
3. Klicken Sie unter Sitzungsaufzeichnungsserver - Eigenschaften auf die Registerkarte Wiedergabe.
4. Aktivieren Sie das Kontrollkästchen Für die Wiedergabe heruntergeladene Sitzungsaufzeichnungsdateien verschlüsseln oder heben Sie die Markierung auf.

# Aktivieren oder Deaktivieren der digitalen Signatur

May 10, 2016

Wenn Sie Zertifikate auf dem Computer installiert haben, auf dem die Komponenten der Sitzungsaufzeichnung installiert sind, können Sie die Sicherheit der Sitzungsaufzeichnungsbereitstellung erhöhen, indem Sie den Sitzungsaufzeichnungen digitale Signaturen zuweisen.

In der Standardeinstellung sind digitale Signaturen deaktiviert.

## So aktivieren Sie digitale Signaturen

1. Melden Sie sich bei dem Computer mit dem Sitzungsaufzeichnungsserver an.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsserver - Eigenschaften.
3. Klicken Sie unter Sitzungsaufzeichnungsserver - Eigenschaften auf die Registerkarte Signatur.
4. Navigieren Sie zu dem Zertifikat, das die sichere Kommunikation zwischen den Computern ermöglicht, auf denen die Komponenten der Sitzungsaufzeichnung installiert sind.

## So deaktivieren Sie digitale Signaturen

1. Melden Sie sich bei dem Computer mit dem Sitzungsaufzeichnungsserver an.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsserver - Eigenschaften.
3. Klicken Sie unter Sitzungsaufzeichnungsserver - Eigenschaften auf die Registerkarte Signatur.
4. Klicken Sie auf Entfernen.

# Angeben des Speicherorts für Aufzeichnungen

May 10, 2016

In den Eigenschaften des Sitzungsaufzeichnungsservers geben Sie den Speicherort für Aufzeichnungen und archivierte Aufzeichnungen an.

Hinweis: Archivieren oder stellen Sie gelöschte Dateien mit dem Befehl icldb wieder her.

So geben Sie den Speicherort für aufgezeichnete Dateien an

In der Standardeinstellung werden Aufzeichnungen im Verzeichnis Laufwerk:\SessionRecordings des Computers mit dem Sitzungsaufzeichnungsserver gespeichert. Sie können das Verzeichnis ändern, in dem die Aufzeichnungen gespeichert werden, weitere Verzeichnisse hinzufügen, um die Last auf mehrere Volumes zu verteilen oder freien Speicherplatz zu nutzen. Mehrere Verzeichnisse in der Liste geben an, dass die Last der Aufzeichnungen auf mehrere Verzeichnisse verteilt wird. Sie können ein Verzeichnis mehrmals hinzufügen. Der Lastausgleich durchläuft alle Verzeichnisse.

1. Melden Sie sich bei dem Computer mit dem Sitzungsaufzeichnungsserver an.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsserver - Eigenschaften.
3. Klicken Sie unter Sitzungsaufzeichnungsserver - Eigenschaften auf die Registerkarte Speicherung.
4. Mit der Liste Verzeichnisse für Dateispeicherung verwalten Sie die Verzeichnisse, in denen Aufzeichnungen gespeichert werden.

Sie können die Verzeichnisse für die Dateispeicherung auf dem lokalen Datenträger, einem SAN-Volume oder an einem durch UNC-Pfad angegebenen Speicherort erstellen. Buchstaben zugeordneter Netzwerklaufräume werden nicht unterstützt. Die Sitzungsaufzeichnung darf nicht zusammen mit NAS (Network-Attached Storage) verwendet werden, da schwere Leistungs- und Sicherheitsprobleme auftreten, die mit dem Schreiben von Aufzeichnungsdaten in ein Netzlaufwerk verbunden sind.

So geben Sie ein Wiederherstellungsverzeichnis für archivierte Dateien an

In der Standardeinstellung werden Aufzeichnungen im Verzeichnis Laufwerk:\SessionRecordingsRestore des Computers mit dem Sitzungsaufzeichnungsserver wiederhergestellt. Sie können das Verzeichnis für die archivierten Aufzeichnungen ändern.

1. Melden Sie sich bei dem Computer mit dem Sitzungsaufzeichnungsserver an.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsserver - Eigenschaften.
3. Klicken Sie unter Sitzungsaufzeichnungsserver - Eigenschaften auf die Registerkarte Speicherung.
4. Geben Sie im Feld Wiederherstellungsverzeichnis für archivierte Dateien das Verzeichnis für wiederhergestellte Archivdateien an.

# Angeben der Dateigröße für Aufzeichnungen

May 10, 2016

Wenn die Größe der Aufzeichnungsdateien zunimmt, kann der Download länger dauern und die Reaktionszeit kann sich verlangsamen, wenn Sie mit dem Schieberegler durch die Wiedergabe navigieren. Sie können die Dateigröße durch Festlegen eines Schwellenwerts für eine Datei steuern. Wenn die Aufzeichnung dieses Limit erreicht, schließt die Sitzungsaufzeichnung die Datei und öffnet eine neue, um die Aufzeichnung fortzusetzen. Dies wird Rollover genannt.

Sie können zwei Schwellenwerte für ein Rollover angeben:

- **Dateigröße:** Wenn die Größe der Datei die angegebene Zahl Megabytes erreicht, schließt die Sitzungsaufzeichnung die Datei und öffnet eine neue. In der Standardeinstellung wird ein Dateirollover durchgeführt, wenn die Dateigröße 50 Megabytes erreicht. Sie können jedoch ein Limit von 10 Megabytes bis zu 1 Gigabyte festlegen.
- **Dauer:** Wenn die Sitzung für die angegebene Anzahl der Stunden aufgezeichnet wurde, wird die Datei geschlossen und eine neue Datei wird geöffnet. In der Standardeinstellung wird ein Dateirollover durchgeführt, wenn die Sitzung für 12 Stunden aufgezeichnet wurde. Sie können jedoch ein Limit von einer bis zu 24 Stunden festlegen.

Die Sitzungsaufzeichnung prüft beide Felder und ermittelt, welches Ereignis zuerst auftritt, und legt dann den Rollover fest. Beispiel: Wenn Sie 17 MB für die Dateigröße und sechs Stunden für die Dauer eingeben, und die Aufzeichnung erreicht 17 MB nach drei Stunden, reagiert die Sitzungsaufzeichnung auf die Dateigröße von 17 MB, schließt die Datei und öffnet eine neue.

Unabhängig vom eingegebenen Wert für die Dateigröße führt die Sitzungsaufzeichnung ein Rollover frühestens nach einer Stunde durch (dies ist der Mindestwert, den Sie eingeben können), um das Erstellen von zu vielen kleinen Dateien zu vermeiden. Diese Regel gilt nicht, wenn die Dateigröße über ein Gigabyte ansteigt.

So geben Sie ein Höchstlimit für eine Datei an

1. Melden Sie sich bei dem Computer mit dem Sitzungsaufzeichnungsserver an.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsserver - Eigenschaften.
3. Klicken Sie unter Sitzungsaufzeichnungsserver - Eigenschaften auf die Registerkarte Rollover.
4. Geben Sie eine Ganzzahl zwischen 10 und 1024 ein, mit der Sie die maximale Dateigröße in Megabyte angeben.
5. Geben Sie eine Ganzzahl zwischen 1 und 24 ein, mit der Sie die maximale Aufzeichnungslänge in Stunden angeben.

# Anzeigen von Aufzeichnungen

May 10, 2016

Mit dem Sitzungsaufzeichnungsplayer zeigen Sie aufgezeichnete XenApp- bzw. XenDesktop-Sitzungen an, suchen die Sitzungsaufzeichnungsdateien und fügen Textmarken hinzu.

Wenn Sitzungen mit aktiviertem Liveplayback aufgezeichnet werden, können Sie aktuell ausgeführte Sitzungen mit einer Verzögerung von ein paar Minuten und abgeschlossene Sitzungen anzeigen.

Sitzungen, die länger als die vom Sitzungsaufzeichnungsadministrator festgelegten Höchstwerte sind oder deren Dateigröße das Limit übersteigt, werden in mehreren Sitzungsdateien aufgezeichnet.

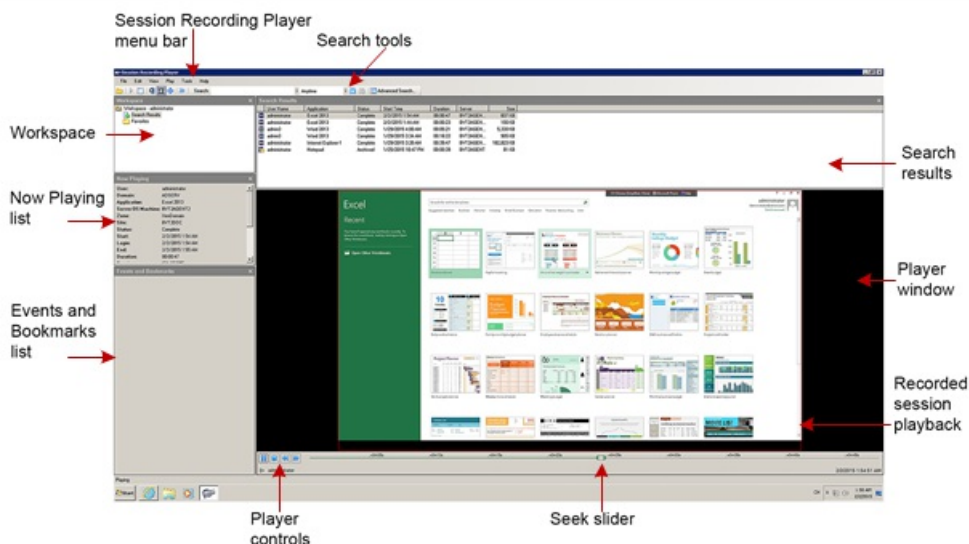
Hinweis: Der Sitzungsaufzeichnungsadministrator muss die Benutzer für den Zugriff auf aufgezeichnete Serverbetriebssystemmaschinen-Sitzungen berechtigen. Wenden Sie sich an den Sitzungsaufzeichnungsadministrator, wenn Sie nicht auf Sitzungsaufzeichnungen zugreifen können.

Bei der Installation des Sitzungsaufzeichnungsplayers richtet der Administrator normalerweise die Verbindung zwischen dem Player und einem Sitzungsaufzeichnungsserver ein. Wenn die Verbindung nicht eingerichtet ist, müssen Sie die Verbindung beim ersten Suchen nach Dateien angeben. Weitere Informationen zum Einrichten der Verbindung erhalten Sie vom Sitzungsaufzeichnungsadministrator.

So starten Sie den Sitzungsaufzeichnungsplayer

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.  
Der Sitzungsaufzeichnungsplayer wird angezeigt.

Die Abbildung zeigt den Sitzungsaufzeichnungsplayer mit Beschriftungen seiner wichtigsten Elemente. Die Funktion dieser Elemente wird in den folgenden Artikeln beschrieben.



So blenden Sie Fensterelemente ein oder aus

Der Sitzungsaufzeichnungsplayer hat Fensterelemente, die Sie ein- und ausblenden können.

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.



3. Klicken Sie auf der Menüleiste des Sitzungsaufzeichnungsplayers auf Ansicht.
4. Wählen Sie die Elemente aus, die Sie anzeigen möchten. Bei der Auswahl eines Elements wird es sofort angezeigt. Ein Häkchen gibt die Auswahl des Elements an.

### So wechseln Sie den Sitzungsaufzeichnungsserver

Wenn der Administrator eine Verbindung mit mehreren Sitzungsaufzeichnungsservern im Sitzungsaufzeichnungsplayer ermöglicht hat, können Sie den Server wählen, mit dem der Player eine Verbindung herstellen soll. Der Sitzungsaufzeichnungsplayer kann nur jeweils eine Verbindung mit einem Sitzungsaufzeichnungsserver herstellen.

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Klicken Sie auf der Menüleiste des Sitzungsaufzeichnungsplayers auf Extras > Optionen > Verbindungen.
4. Wählen Sie den Sitzungsaufzeichnungsserver aus, mit dem Sie eine Verbindung herstellen möchten.

# Öffnen und Wiedergeben von Aufzeichnungen

May 10, 2016

Sie öffnen Sitzungsaufzeichnungen im Sitzungsaufzeichnungsplayer auf dreierlei Weise:

- Führen Sie eine Suche mit dem Sitzungsaufzeichnungsplayer durch. Sitzungsaufzeichnungen, die die Suchkriterien erfüllen, werden im Bereich der Suchergebnisse angezeigt.
- Greifen Sie auf Sitzungsaufzeichnungsdateien direkt von der lokalen Festplatte oder einem freigegebenen Laufwerk zu.
- Greifen Sie auf Sitzungsaufzeichnungsdateien vom Ordner "Favoriten" zu

Beim Öffnen einer Datei, die ohne digitale Signatur aufgezeichnet wurde, werden Sie in einer Warnmeldung darauf hingewiesen, dass der Ursprung und die Integrität der Datei nicht geprüft werden konnte. Bestätigen Sie die Warnmeldung mit Ja und öffnen Sie die Datei, wenn Sie hinsichtlich der Integrität der Datei keine Bedenken haben.

So öffnen und geben Sie eine Aufzeichnung im Bereich der Suchergebnisse wieder

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Führen Sie eine Suche durch.
4. Wählen Sie im Arbeitsbereich Suchergebnisse aus, wenn der Bereich "Suchergebnisse" nicht angezeigt wird.
5. Wählen Sie im Bereich der Suchergebnisse die Sitzung aus, die Sie wiedergeben möchten.
6. Tun Sie Folgendes:
  - Doppelklicken Sie auf die Sitzung.
  - Klicken Sie mit der rechten Maustaste und wählen Sie Wiedergabe.
  - Klicken Sie auf der Sitzungsaufzeichnungsplayer-Menüleiste auf Wiedergabe > Wiedergabe.

So öffnen und geben eine Aufzeichnung durch Zugriff auf die Datei wieder

Namen der Sitzungsaufzeichnungsdateien beginnen mit i\_, dann folgt eine eindeutige alphanumerische Datei-ID, der die Dateierweiterung .icl bzw. .icle folgt, wobei: .icl für Aufzeichnungen ohne Wiedergabeschutz und .icle für Aufzeichnungen mit Wiedergabeschutz verwendet wird. Die Sitzungsaufzeichnung speichert die Sitzungsaufzeichnungsdateien in einer Dateistruktur, in die das Datum der Aufzeichnung integriert ist. Beispiel: Die Datei für eine Sitzung, die am Montag, 22. Dezember 2014 aufgezeichnet wurde, wird im Ordnerpfad 2014\12\22 gespeichert.

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Tun Sie Folgendes:
  - Wählen Sie auf der Sitzungsaufzeichnungsplayer-Menüleiste auf Datei > Öffnen aus und navigieren Sie zu der Datei.
  - Navigieren Sie mit dem Windows-Explorer auf die Datei und ziehen Sie sie in das Player-Fenster.
  - Navigieren Sie mit dem Windows-Explorer auf die Datei und doppelklicken Sie.
  - Wenn Sie Favoriten im Arbeitsbereich erstellt haben, wählen Sie Favoriten und öffnen Sie die Datei im Bereich "Favoriten" genauso wie Sie die Datei im Bereich der Suchergebnisse öffnen.

Verwenden von Favoriten

Das Erstellen von Favoriten-Ordern ermöglicht den schnellen Zugriff auf oft angezeigte Sitzungsaufzeichnungen. Diese Favoriten-Ordner verweisen auf Sitzungsaufzeichnungsdateien, die auf der Arbeitsstation oder auf einem Netzwerklaufwerk gespeichert sind. Sie können diese Dateien von anderen Arbeitsstationen importieren, zu anderen exportieren und die Ordner für andere Sitzungsaufzeichnungsplayer-Benutzer freigeben.

Hinweis: Nur Benutzer mit Zugriffsrechten für den Sitzungsaufzeichnungsplayer können die Sitzungsaufzeichnungsdateien herunterladen, die den Favoriten-Ordern zugeordnet sind. Wenden Sie sich für Zugriffsrechte an den Sitzungsaufzeichnungsadministrator.

So erstellen Sie einen Favoriten-Unterordner

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Wählen Sie im Sitzungsaufzeichnungsplayer den Ordner Favoriten im Bereich "Arbeitsbereich".
4. Klicken Sie auf der Menüleiste auf Datei > Ordner > Neuer Ordner. Ein neuer Ordner wird unter dem Ordner Favoriten angezeigt.
5. Geben Sie den Namen des Ordners ein und drücken Sie die Eingabetaste oder klicken Sie an beliebiger Stelle, um den neuen Namen einzugeben.

Mit den anderen Optionen im Menü Datei > Ordner können Sie die Ordner löschen, umbenennen, verschieben, kopieren, importieren und exportieren.

# Suchen nach Sitzungsaufzeichnungen

May 10, 2016

Im Sitzungsaufzeichnungsplayer können Sie Schnellsuchen und erweiterte Suchen durchführen und Optionen festlegen, die für alle Suchen gelten. Die Suchergebnisse werden im Bereich "Suchergebnisse" des Sitzungsaufzeichnungsplayers angezeigt.

Hinweis: Wenn Sie alle verfügbaren Sitzungsaufzeichnungen (bis zur Höchstanzahl der in einer Suche angezeigten Sitzungen) anzeigen möchten, führen Sie die Suche ohne Suchparameter durch.

So führen Sie eine Schnellsuche aus

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Definieren Sie die Suchkriterien:
  - Geben Sie im Feld Suchen ein Suchkriterium ein. Als Hilfestellung:
  - Zeigen Sie auf Suchen, um eine Liste der Parameter als Richtlinie zu anzeigen.
  - Klicken Sie auf den Pfeil rechts neben dem Feld Suchen, um den Text für die letzten 64 Suchen anzuzeigen.
  - Wählen Sie in der rechts neben dem Feld Suchen angezeigten Dropdownliste den Zeitraum oder die Dauer aus und geben Sie an, wann die Sitzung aufgezeichnet wurde.
4. Klicken Sie auf das Fernglas-Symbol rechts von der Dropdownliste, um die Suche zu starten.

So führen Sie eine erweiterte Suche durch

Tipp: Die erweiterte Suche kann bis zu 20 Sekunden dauern, wenn das Ergebnis über 150.000 Einheiten umfasst. Citrix empfiehlt die Verwendung gezielterer Suchbedingungen, z. B. einen Datumsbereich oder Benutzer, um den Umfang des Ergebnisses zu limitieren.

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Klicken Sie im Sitzungsaufzeichnungsplayer-Fenster auf der Symbolleiste auf Erweiterte Suche oder wählen Sie Extras > Erweiterte Suche.
4. Legen Sie die Suchkriterien auf den Registerkarten im Dialogfeld Erweiterte Suche fest:
  - Allgemein ermöglicht die Suche nach Domäne oder Kontoautorität, Site, Gruppe, Serverbetriebssystemmaschine, Anwendung oder Datei-ID.
  - Datum/Uhrzeit: Die Suche wird nach Datum, Wochentag und Tageszeit durchgeführt.
  - Ereignisse dient zur Suche nach benutzerdefinierten Ereignissen, die der Sitzungsaufzeichnungsadministrator in die Sitzungen eingefügt hat.
  - Sonstiges: Die Suche wird nach Sitzungsname, Clientname, Clientadresse und Aufzeichnungsdauer durchgeführt. Sie können für diese Suche auch die Höchstzahl der angezeigten Suchergebnisse und den Einschluss von archivierten Dateien in der Suche festlegen.

Bei der Angabe von Suchkriterien wird die Abfrage, die Sie erstellen, im unteren Bereich des Dialogfelds angezeigt.

5. Klicken Sie auf Suchen, um die Suche zu starten.

Tipp: Sie können erweiterte Suchen speichern und abrufen. Klicken Sie im Dialogfeld Erweiterte Suche auf Speichern, um die aktuelle Abfrage zu speichern. Klicken Sie im Dialogfeld Erweiterte Suche auf Öffnen, um eine gespeicherte Abfrage abzurufen. Abfragen werden als Dateien mit der ISQ-Erweiterung gespeichert.

So stellen Sie Suchoptionen ein

Mit den Suchoptionen im Sitzungsaufzeichnungsplayer beschränken Sie die Höchstzahl der Sitzungsaufzeichnungen, die in

den Suchergebnissen angezeigt werden, und legen den Ein- oder Ausschluss von archivierten Sitzungsaufzeichnungsdateien fest.

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Klicken Sie auf der Menüleiste des Sitzungsaufzeichnungsplayers auf Extras > Optionen > Suche.
4. Geben Sie im Feld Angezeigte Ergebnisse (max.) die Anzahl der Suchergebnisse ein, die angezeigt werden. Sie können maximal 500 Ergebnisse anzeigen.
5. Abhängig davon, ob Sie archivierte Dateien in Suchen einschließen möchten, aktivieren oder deaktivieren Sie Archivierte Dateien einschließen.

# Wiedergeben aufgezeichneter Sitzungen

May 10, 2016

Nach dem Öffnen einer aufgezeichneten Sitzung im Sitzungsaufzeichnungsplayer können Sie in der Aufzeichnung folgendermaßen navigieren:

- Mit den Player-Bedienelementen können Sie die Aufzeichnung wiedergeben, anhalten oder stoppen und die Wiedergabegeschwindigkeit erhöhen oder verringern.
- Mit dem Schieberegler für das Positionieren gehen Sie vorwärts oder rückwärts.

Wenn Sie Marker in die Aufzeichnung eingefügt haben, oder die aufgezeichnete Sitzung benutzerdefinierte Ereignisse enthält, können Sie mit diesen Markern oder Ereignissen durch die Sitzungsaufzeichnung navigieren.

Hinweis:

- Bei der Wiedergabe einer aufgezeichneten Sitzung wird möglicherweise ein zweiter Mauszeiger angezeigt. Der zweite Mauszeiger wird an der Stelle in der Aufzeichnung angezeigt, an der der Benutzer in Internet Explorer navigiert und auf ein Bild geklickt hat, das im Original größer als der Bildschirm war, das jedoch von Internet Explorer automatisch skaliert wurde. Ein Zeiger wird während der Sitzung angezeigt, jedoch können zwei bei der Wiedergabe angezeigt werden.
- Diese Version der Sitzungsaufzeichnung unterstützt weder die SpeedScreen-Multimediabeschleunigung für XenApp noch die Richtlinieneinstellung zum Optimieren der Flash-Inhalte für XenApp. Wenn diese Option aktiviert ist, wird bei der Wiedergabe ein schwarzes Rechteck angezeigt.
- Die Sitzungsaufzeichnung kann keine Lync-Webcamvideos aufzeichnen, wenn das HDX RealTime Optimization Pack für Microsoft Lync verwendet wird.

## Verwenden der Player-Bedienelemente

Sie können auf die Player-Bedienelemente unten im Player-Fenster klicken oder im Sitzungsaufzeichnungsplayer-Menü auf Wiedergabe klicken. Die Player-Bedienelemente ermöglichen Folgendes:

	Wiedergeben der ausgewählten Sitzungsdatei.
	Anhalten der Wiedergabe.
	Stoppen der Wiedergabe. Wenn Sie auf Stopp und dann auf Wiedergabe klicken, wird die Aufzeichnung vom Anfang der Datei an wiedergegeben.
	Halbieren der momentanen Wiedergabegeschwindigkeit um die Hälfte auf mindestens ein Viertel der Normalgeschwindigkeit.
	Verdoppeln der momentanen Wiedergabegeschwindigkeit auf maximal das Zweiunddreißigfache der Normalgeschwindigkeit.

## Verwenden des Schiebereglers für das Positionieren

Mit dem Schieberegler für das Positionieren unter dem Player-Fenster springen Sie auf eine andere Stelle in der aufgezeichneten Sitzung. Sie können den Schieberegler für das Positionieren auf eine Stelle in der Aufzeichnung ziehen, die Sie anzeigen möchten, oder auf eine Stelle auf dem Schieberegler klicken, um auf diese Stelle zu gehen.

Sie können den Schieberegler für das Positionieren auch mit den folgenden Tasten auf der Tastatur steuern:

<b>Schlüssel:</b>	<b>Positionieren:</b>
Pos1	Auf den Anfang positionieren
Ende	Auf das Ende positionieren
Nach-Rechts-Taste	5 Sekunden vorwärts positionieren
Nach-Links-Taste	5 Sekunden rückwärts positionieren
Verschieben des Mausekursors um einen Anschlag nach unten	15 Sekunden vorwärts positionieren
Verschieben des Mausekursors um einen Anschlag nach oben	15 Sekunden rückwärts positionieren
Strg + Nach-Rechts-Taste	30 Sekunden vorwärts positionieren
Strg + Nach-Links-Taste	30 Sekunden rückwärts positionieren
Bild-Ab	1 Minute vorwärts positionieren
Bild-Auf	1 Minute rückwärts positionieren
Strg + Mausekursor einen Klick nach unten bewegen	90 Sekunden vorwärts positionieren
Strg + Mausekursor einen Klick nach oben bewegen	90 Sekunden rückwärts positionieren
Strg + Bild-Ab	6 Minuten vorwärts positionieren
Strg + Bild-Auf	6 Minuten rückwärts positionieren

Hinweis: Klicken Sie im Menü des Sitzungsaufzeichnungsplayers auf Extras > Optionen > Player und verschieben Sie den Schieberegler, um die Reaktionszeit des Schiebereglers zu verlängern oder zu verkürzen. Für eine schnellere Reaktionszeit wird mehr Speicher benötigt. Die Reaktion kann, abhängig von der Größe der Aufzeichnungen und der Computerhardware, langsam sein.

So ändern Sie die Wiedergabegeschwindigkeit

Sie können die Wiedergabe aufgezeichneter Sitzungen im Sitzungsaufzeichnungsplayer in exponentiellen Erhöhungen von einem Viertel der normalen Wiedergabegeschwindigkeit bis zum 32-fachen der normalen Wiedergabegeschwindigkeit

einstellen.

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Klicken Sie auf der Sitzungsaufzeichnungsplayer-Menüleiste auf Wiedergabe > Wiedergabegeschwindigkeit.
4. Wählen Sie eine der Optionen für die Geschwindigkeit aus.

Die Geschwindigkeit wird sofort geändert. Eine Zahl gibt die erhöhte oder verringerte Wiedergabegeschwindigkeit unter den Bedienelementen im Player-Fenster an. Text, der die exponentielle Rate angibt, wird kurz in grün im Player-Fenster angezeigt.

#### So überspringen Sie Stellen ohne Aktionen

Im Schnellprüfmodus überspringt der Sitzungsaufzeichnungsplayer die Teile von aufgezeichneten Sitzungen, in denen keine Aktion stattfindet. Mit dieser Einstellung sparen Sie Zeit bei der Wiedergabe, animierte Folgen werden jedoch nicht übersprungen, z. B. ein animierter Mauszeiger, blinkende Cursor oder angezeigte Uhren, bei denen sich die zweite Hand bewegt.

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Klicken Sie im Menü des Sitzungsaufzeichnungsplayers auf Wiedergabe > Schnellprüfmodus.

Sie können die Option aktivieren oder deaktivieren. Bei jeder Auswahl wird der Status kurz in grün im Player -Fenster angezeigt.



# Verwenden von Ereignissen und Textmarken

May 10, 2016

Ereignisse und Textmarken erleichtern das Navigieren in aufgezeichneten Sitzungen.

Ereignisse werden bei der Aufzeichnung mit der Ereignis-API oder einer Anwendung eines Drittherstellers in die Sitzungen eingefügt. Ereignisse werden als Teil der Sitzungsdatei gespeichert. Sie können mit dem Sitzungsaufzeichnungsplayer nicht gelöscht oder geändert werden.

Textmarken sind Marker, die Sie mit dem Sitzungsaufzeichnungsplayer in Sitzungsaufzeichnungen einfügen. Textmarken werden der aufgezeichneten Sitzung zugeordnet, bis sie gelöscht werden, sie werden jedoch nicht mit der Sitzungsdatei gespeichert. In der Standardeinstellung ist jede Textmarke mit Textmarke beschriftet. Sie können diese Beschriftung beliebig ändern und maximal 128 Zeichen eingeben.

Ereignisse und Textmarken werden als Punkte unter dem Player-Fenster angezeigt. Ereignisse werden als gelbe Punkte und Textmarken als blaue Punkte angezeigt. Wenn Sie die Maus auf diese Punkte verschieben, wird der zugeordnete Text angezeigt. Sie können die Ereignisse und Textmarken auch in der Liste "Ereignisse und Textmarken" im Sitzungsaufzeichnungsplayer anzeigen. Sie werden in dieser Liste in chronologischer Reihenfolge mit den Textbeschriftungen und den Uhrzeiten angezeigt, zu denen sie in der aufgezeichneten Sitzung erscheinen.

Ereignisse und Textmarken erleichtern das Navigieren in aufgezeichneten Sitzungen. Wenn Sie auf ein Ereignis oder eine Textmarke gehen, springen Sie auf die Stelle in der aufgezeichneten Sitzung, an der das Ereignis oder die Textmarke eingefügt ist.

## So zeigen Sie Ereignisse und Textmarken in der Liste an

In der Liste "Ereignisse und Textmarken" werden die Ereignisse und Textmarken angezeigt, die in der momentan wiedergegebenen Sitzungsaufzeichnung eingefügt sind. Sie können in der Liste nur Ereignisse, nur Textmarken oder Ereignisse und Textmarken anzeigen.

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Verschieben Sie den Mauszeiger auf die Liste "Ereignisse und Textmarken" und klicken Sie mit der rechten Maustaste, um das Menü anzuzeigen.
4. Wählen Sie Nur Ereignisse anzeigen, Nur Textmarken anzeigen oder Alle anzeigen.

## So fügen Sie eine Textmarke ein

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Starten Sie die Wiedergabe der aufgezeichneten Sitzung, der Sie eine Textmarke hinzufügen möchten.
4. Schieben Sie den Schieberegler für das Positionieren auf die Stelle, an der Sie die Textmarke einfügen möchten.
5. Schieben Sie den Mauszeiger in den Player-Fensterbereich und klicken Sie mit der rechten Maustaste, um das Menü anzuzeigen.
6. Fügen Sie eine Textmarke mit der Standardbeschriftung Textmarke hinzu oder erstellen Sie eine Anmerkung:
  - Zum Hinzufügen einer Textmarke mit der Standardbeschriftung Textmarke wählen Sie Textmarke hinzufügen.
  - Zum Hinzufügen einer Textmarke mit einer eigenen Beschriftung wählen Sie Anmerkung hinzufügen. Geben Sie die Anmerkung (max. 128 Zeichen) ein, die Sie der Textmarke zuordnen möchten. Klicken Sie auf OK.

## So fügen Sie eine Anmerkung hinzu oder ändern sie

Nach dem Erstellen einer Textmarke können Sie eine Anmerkung hinzufügen oder eine bestehende Anmerkung ändern.

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Starten Sie die Wiedergabe der aufgezeichneten Sitzung, die die Textmarke enthält.
4. Stellen Sie sicher, dass Textmarken in der Liste "Ereignisse und Textmarken" angezeigt werden.
5. Wählen Sie die Textmarke aus der Liste "Ereignisse und Textmarken" aus und klicken Sie mit der rechten Maustaste, um das Menü anzuzeigen.
6. Wählen Sie Anmerkung bearbeiten.
7. Geben Sie im angezeigten Dialogfeld die neue Anmerkung ein und klicken Sie auf OK.

#### So löschen Sie eine Textmarke

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Starten Sie die Wiedergabe der aufgezeichneten Sitzung, die die Textmarke enthält.
4. Stellen Sie sicher, dass Textmarken in der Liste "Ereignisse und Textmarken" angezeigt werden.
5. Wählen Sie die Textmarke aus der Liste "Ereignisse und Textmarken" aus und klicken Sie mit der rechten Maustaste, um das Menü anzuzeigen.
6. Wählen Sie Löschen.

#### So gehen Sie auf ein Ereignis oder eine Textmarke

Wenn Sie auf ein Ereignis oder eine Textmarke gehen, springt der Sitzungsaufzeichnungsplayer an die Stelle in der Sitzungsaufzeichnung, an der das Ereignis oder die Textmarke eingefügt ist.

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Starten Sie eine Sitzungsaufzeichnung, die Ereignisse oder Textmarken enthält.
4. Gehen Sie auf ein Ereignis oder eine Textmarke.
  - Klicken Sie im Bereich unter dem Player-Fenster auf den Punkt, der das Ereignis oder die Textmarke darstellt, um auf das Ereignis oder die Textmarke zu gehen.
  - Doppelklicken Sie in der Liste "Ereignisse und Textmarken" auf ein Ereignis oder eine Textmarke, um an die entsprechende Stelle zu gehen. Wenn Sie auf das nächste Ereignis oder die nächste Textmarke gehen möchten, wählen Sie ein Ereignis oder eine Textmarke aus der Liste aus, klicken Sie mit der rechten Maustaste, um das Menü anzuzeigen, und wählen Sie Auf Textmarke positionieren.

# Ändern der Wiedergabeanzeige

May 10, 2016

Mit Optionen können Sie ändern, wie die aufgezeichneten Sitzungen im Player-Fenster angezeigt werden. Sie können das Bild mit Panning und Skalieren ändern, die Wiedergabe im Vollbildmodus anzeigen, das Player-Fenster in einem eigenen Fenster und einen Rahmen um die Sitzungsaufzeichnung anzeigen, um sie vom Hintergrund des Player-Fensters zu unterscheiden.

So zeigen Sie das Player-Fenster im Vollbildmodus an

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Klicken Sie im Menü des Sitzungsaufzeichnungsplayers auf Ansicht > Player-Vollbild.
4. Drücken Sie ESC oder F11, um die Originalgröße des Fensters wiederherzustellen.

So zeigen Sie das Player-Fenster in einem eigenen Fenster an

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Klicken Sie im Menü des Sitzungsaufzeichnungsplayers auf Ansicht > Player in neuem Fenster. Ein neues Fenster wird mit dem Player-Fenster angezeigt. Sie können das Fenster verschieben oder die Fenstergröße ändern.
4. Wenn Sie das Player-Fenster im Hauptfenster einbetten möchten, wählen Sie Ansicht > Player in neuem Fenster oder drücken Sie F10.

So skalieren Sie die Sitzungswiedergabe auf die Größe des Player-Fensters

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Klicken Sie im Menü des Sitzungsaufzeichnungsplayers auf Wiedergabe > Panning und Skalieren > Passend skalieren.
  - Bei Passend skalieren (Schnellrendering) wird das Bild verkleinert, die Bildqualität ist jedoch noch gut. Bilder werden schneller als mit der Option "Passend skalieren (hohe Qualität)" aufgebaut, die Bilder und der Text sind jedoch nicht so scharf. Verwenden Sie diese Option, wenn Sie Leistungsprobleme mit der Option "Passend skalieren (hohe Qualität)" feststellen.
  - Bei Passend skalieren (hohe Qualität) wird das Bild verkleinert, die Qualität des Bildes und des Textes ist sehr gut. Bei dieser Option werden die Bilder möglicherweise langsamer als bei der Option "Passend skalieren (Schnellrendering)" aufgebaut.

So führen Sie ein Panning des Bilds durch

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Klicken Sie im Menü des Sitzungsaufzeichnungsplayers auf Wiedergabe > Panning und Skalieren > Panning. Der Zeiger ändert sich zu einer Hand und eine kleine Darstellung des Bildschirms wird oben rechts im Player-Fenster angezeigt.
4. Ziehen Sie das Bild. Die kleine Darstellung gibt die Stelle im Bild an, an der Sie sich befinden.
5. Wählen Sie eine der Skalierungsoptionen, um das Panning anzuhalten.

So zeigen Sie einen roten Rahmen um die Sitzungsaufzeichnung an

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.

2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Klicken Sie im Menü des Sitzungsaufzeichnungsplayers auf Extras > Optionen > Player.
4. Aktivieren Sie Rahmen um Sitzungsaufzeichnung anzeigen.

Tipp: Wenn Rahmen um Sitzungsaufzeichnung anzeigen nicht aktiviert ist, können Sie den roten Rahmen vorübergehend anzeigen, wenn Sie die linke Maustaste gedrückt halten, wenn der Zeiger im Player-Fenster ist.

# Zwischenspeichern von Sitzungsaufzeichnungsdateien

May 10, 2016

Bei jedem Öffnen einer Sitzungsaufzeichnungsdatei lädt der Sitzungsaufzeichnungsplayer die Datei vom Speicherort herunter, auf dem die Aufzeichnungen gespeichert sind. Wenn Sie dieselben Dateien oft herunterladen, sparen Sie Zeit, wenn Sie die Dateien auf der Arbeitsstation zwischenspeichern. Zwischengespeicherte Dateien werden auf der Arbeitsstation in diesem Ordner gespeichert:

userprofile\AppData\Local\Citrix\SessionRecording\Player\Cache

Sie können die verwendete Cachegröße angeben. Wenn die Aufzeichnungen den angegebenen Speicherplatz auf der Festplatte vollständig belegen, löscht die Sitzungsaufzeichnung die ältesten und am wenigsten verwendeten Aufzeichnungen, um Platz für neue Aufzeichnungen zu machen. Sie können den Cache jederzeit leeren, um freien Speicherplatz auf der Festplatte zu erhalten.

## So aktivieren Sie das Zwischenspeichern

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Klicken Sie auf der Menüleiste des Sitzungsaufzeichnungsplayers auf Extras > Optionen > Cache.
4. Aktivieren Sie das Kontrollkästchen Heruntergeladene Dateien lokal zwischenspeichern.
5. Wenn Sie den für die Zwischenspeicherung verwendeten Speicherplatz auf dem Datenträger beschränken möchten, aktivieren Sie das Kontrollkästchen Verwendeten Speicherplatz auf Datenträger beschränken, und geben Sie den Speicherplatz in Megabyte an.
6. Klicken Sie auf OK.

## So löschen Sie den Cache

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
3. Klicken Sie auf der Menüleiste des Sitzungsaufzeichnungsplayers auf Extras > Optionen > Cache.
4. Aktivieren Sie das Kontrollkästchen Heruntergeladene Dateien lokal zwischenspeichern.
5. Klicken Sie im Sitzungsaufzeichnungsplayer auf Extras > Optionen > Cache.
6. Klicken Sie auf Cache löschen und dann zur Bestätigung auf OK.

# Problembehandlung bei der Sitzungsaufzeichnung

May 10, 2016

In diesen Informationen finden Sie Lösungen zu Problemen, auf die Sie möglicherweise während oder nach der Installation der Sitzungsaufzeichnungskomponenten stoßen:

- Probleme bei der Komponentenverbindung
- Probleme bei der Sitzungsaufzeichnung
- Probleme beim Sitzungsaufzeichnungsplayer oder der Richtlinienkonsole für die Sitzungsaufzeichnung
- Probleme beim Kommunikationsprotokoll

## Sitzungsaufzeichnungsagent kann keine Verbindung herstellen

Wenn der Sitzungsaufzeichnungsagent keine Verbindung herstellen kann, wird die Ereignismeldung `Exception caught while sending poll messages to Session Recording Broker` gefolgt vom Ausnahmetext protokolliert. Der Ausnahmetext gibt den Grund für den Verbindungsfehler an. Diese Gründe sind wie folgt:

- Die zugrunde liegende Verbindung wurde geschlossen. Eine vertrauenswürdige Beziehung konnte für den sicheren Kanal (SSL/TLS) nicht erstellt werden. Diese Ausnahme bedeutet, dass der Sitzungsaufzeichnungsserver ein Zertifikat verwendet, das von einer Zertifizierungsstelle signiert ist, die der Server mit dem Sitzungsaufzeichnungsagent nicht als vertrauenswürdig einstuft oder für die der Server kein Zertifikat der Zertifizierungsstelle hat. Das Zertifikat kann auch abgelaufen oder widerrufen sein.

Lösung: Stellen Sie sicher, dass das richtige Zertifikat der Zertifizierungsstelle auf dem Server mit dem Sitzungsaufzeichnungsagent installiert ist oder ein vertrauenswürdiges Zertifikat verwendet wird.

- Der Remoteserver gab einen Fehler zurück: (403) verboten. Dies ist ein HTTPS-Standardfehler, der angezeigt wird, wenn Sie eine Verbindung mit HTTP (nicht sicheres Protokoll) versuchen. Der Computer mit dem Sitzungsaufzeichnungsserver lehnt die Verbindung ab, da er nur sichere Verbindungen annimmt.

Lösung: Ändern Sie in den Eigenschaften des Sitzungsaufzeichnungsagents das Protokoll des Sitzungsaufzeichnungsbrokers in HTTPS.

Der Sitzungsaufzeichnungsbroker gab beim Auswerten einer Aufzeichnungsrichtlinienabfrage einen unbekannten Fehler zurück. Fehlercode 5 (Zugriff verweigert). Weitere Informationen finden Sie im Ereignisprotokoll auf dem Sitzungsaufzeichnungsserver. Dieser Fehler tritt auf, wenn Sitzungen gestartet werden und eine Anfrage für eine Auswertung der Aufzeichnungsrichtlinie gemacht wird. Der Fehler tritt auf, wenn die Gruppe der authentifizierten Benutzer (die Standardmitglieder) von der Rolle "PolicyQuery" in der Sitzungsaufzeichnungsautorisierungskonsole entfernt werden.

Lösung: Fügen Sie die Gruppe der authentifizierten Benutzer wieder der Rolle hinzu oder fügen Sie alle Server mit dem Sitzungsaufzeichnungsagent der Rolle "PolicyQuery" hinzu.

Die zugrunde liegende Verbindung wurde geschlossen. Eine Verbindung, die aktiv bleiben sollte, wurde vom Server geschlossen. Dieser Fehler bedeutet, dass der Sitzungsaufzeichnungsserver nicht ausgeführt wird oder keine Anfragen annehmen kann. IIS können offline geschaltet sein oder werden neu gestartet, oder der Server ist offline geschaltet.

Lösung: Stellen Sie sicher, dass der Sitzungsaufzeichnungsserver gestartet wurde, IIS auf dem Server ausgeführt wird und der Server eine Verbindung zum Netzwerk hat.

## Sitzungsaufzeichnungsplayer kann keine Verbindung mit der Datenbank für die Sitzungsaufzeichnung

## herstellen

Wenn der Sitzungsaufzeichnungsserver keine Verbindung mit der Datenbank für die Sitzungsaufzeichnung herstellen kann, wird möglicherweise eine Fehlermeldung mit ungefähr dem folgenden Wortlaut angezeigt:

Ereignisquelle:

Citrix Speichermanager der Sitzungsaufzeichnung - Beschreibung: Citrix: Ausnahme beim Herstellen der Datenbankverbindung. Dieser Fehler wird im Ereignisprotokoll der Anwendungen in der Ereignisanzeige des Computers mit dem Sitzungsaufzeichnungsserver aufgezeichnet.

Verbindung mit dem Sitzungsaufzeichnungsserver kann nicht hergestellt werden. Überprüfen Sie, ob der Sitzungsaufzeichnungsserver ausgeführt wird. Diese Fehlermeldung wird beim Start der Richtlinienkonsole für die Sitzungsaufzeichnung angezeigt.

Lösung:

- Die Express Edition von Microsoft SQL Server 2008 R2, Microsoft SQL Server 2012 oder Microsoft SQL Server 2014 ist auf einem eigenständigen Server installiert und Dienste oder Einstellungen sind nicht richtig für die Sitzungsaufzeichnung konfiguriert. Auf dem Server muss das TCP/IP-Protokoll aktiviert sein, und der SQL Server Browser-Dienst muss ausgeführt werden. Weitere Informationen zur Aktivierung dieser Einstellungen finden Sie in der Microsoft Dokumentation.
- Bei der Installation der Sitzungsaufzeichnung (Verwaltungskomponenten) wurden falsche Server- und Datenbankinformationen angegeben. Deinstallieren Sie die Datenbank für die Sitzungsaufzeichnung und installieren Sie sie mit den richtigen Informationen neu.
- Der Server mit der Datenbank für die Sitzungsaufzeichnung ist ausgefallen. Prüfen Sie die Serverkonnektivität.
- Der Computer mit dem Sitzungsaufzeichnungsserver oder der Computer mit dem Server für die Datenbank für die Sitzungsaufzeichnung kann den vollqualifizierten Domänennamen oder den NetBIOS-Namen des jeweils anderen nicht auflösen. Stellen Sie mit "Ping" sicher, dass die Namen aufgelöst werden können.

Fehler bei der Anmeldung für Benutzer "NT\_AUTHORITY\ANONYMOUS LOGON". Diese Fehlermeldung gibt an, dass die Dienste falsch als .\administrator angemeldet sind.

Lösung: Starten Sie die Dienste als lokaler Systembenutzer und die SQL-Dienste neu.

## Kein Aufzeichnen von Sitzungen

Wenn die Anwendungssitzungen nicht aufgezeichnet werden, prüfen Sie zuerst das Anwendungsereignisprotokoll in der Ereignisanzeige auf der Serverbetriebssystemmaschine, auf der der Sitzungsaufzeichnungssagent und der Sitzungsaufzeichnungsserver ausgeführt werden. Sie können nützliche Diagnoseinformationen erhalten.

Wenn Sitzungen nicht aufgezeichnet werden, kann dies folgende Gründe haben:

- **Komponentenverbindungen und Zertifikate:** Wenn die Sitzungsaufzeichnungskomponenten nicht miteinander kommunizieren können, können Sitzungsaufzeichnungen fehlschlagen. Zur Behebung von Aufzeichnungsproblemen sollten Sie sicherstellen, dass alle Komponenten richtig konfiguriert sind und auf die richtigen Computer verweisen, und dass alle Zertifikate gültig und richtig installiert sind.
- **Umgebungen ohne Active Directory-Domäne:** Die Sitzungsaufzeichnung ist für eine Ausführung in einer Microsoft Active Directory-Domänenumgebung konzipiert. Wenn Sie keine Active Directory-Umgebung ausführen, können Aufzeichnungsprobleme auftreten. Stellen Sie sicher, dass alle Komponenten der Sitzungsaufzeichnung auf Computern ausgeführt werden, die Mitglieder einer Active Directory-Domäne sind.

- **Die Sitzungsfreigabe verursacht einen Konflikt mit der aktiven Richtlinie:** Die Sitzungsaufzeichnung ordnet die aktive Richtlinie der ersten veröffentlichten Anwendung zu, die ein Benutzer öffnet. Anwendungen, die später in derselben Sitzung geöffnet werden, halten die Richtlinie ein, die für die erste Anwendung gilt. Veröffentlichen Sie die problematischen Anwendungen auf eigenständigen Serverbetriebssystemmaschinen oder deaktivieren Sie die Sitzungsfreigabe, um einen Konflikt zwischen der Sitzungsfreigabe und der aktiven Richtlinie zu vermeiden.
- **Die Aufzeichnung ist nicht aktiviert:** Wenn Sie den Sitzungsaufzeichnungsagent auf einer Serverbetriebssystemmaschine installieren, wird die Aufzeichnung auf dem Server standardmäßig aktiviert. Die Aufzeichnung erfolgt erst, wenn eine aktive Aufzeichnungsrichtlinie konfiguriert ist, die Aufzeichnungen zulässt.
- **Die aktive Aufzeichnungsrichtlinie lässt die Aufzeichnung nicht zu.** Zum Aufzeichnen einer Sitzung muss die aktive Aufzeichnungsrichtlinie zulassen, dass die Sitzungen für den Benutzer, Server oder die veröffentlichte Anwendung aufgezeichnet werden können.
- **Die Dienste der Sitzungsaufzeichnung werden nicht ausgeführt.** Zum Aufzeichnen von Sitzungen muss der Sitzungsaufzeichnungsagent-Dienst auf der Serverbetriebssystemmaschine und der Dienst des Speichermanagers der Sitzungsaufzeichnung auf dem Computer mit dem Sitzungsaufzeichnungsserver ausgeführt werden.
- **MSMQ ist nicht konfiguriert:** Wenn MSMQ auf dem Computer mit dem Sitzungsaufzeichnungsagent und dem Computer mit dem Sitzungsaufzeichnungsserver falsch konfiguriert ist, können Aufzeichnungsprobleme auftreten.

### Wiedergabe von Livesitzungen nicht möglich

Wenn Sie Probleme beim Wiedergeben von Aufzeichnungen im Sitzungsaufzeichnungsplayer haben, werden möglicherweise die folgenden Fehlermeldungen angezeigt:

Fehler beim Download der Sitzungsaufzeichnungsdatei. Wiedergabe einer Livesitzung ist nicht zulässig. Die Konfiguration des Servers lässt diese Funktion nicht zu. Dieser Fehler gibt an, dass der Server die Aktion nicht zulässt.

Lösung: Klicken Sie in den Sitzungsaufzeichnungsserver - Eigenschaften auf die Registerkarte Wiedergabe und aktivieren Sie das Kontrollkästchen Wiedergabe von Livesitzungen zulassen.

### Aufzeichnungen sind beschädigt oder unvollständig

Wenn Aufzeichnungen beim Anzeigen im Sitzungsaufzeichnungsplayer beschädigt oder unvollständig sind, werden u. U. auch Warnungen in den Ereignisprotokollen auf dem Sitzungsaufzeichnungsagent protokolliert.

**Event Source:** Citrix Session Recording Storage Manager

**Description:** Data lost while recording file

Dies passiert normalerweise, wenn mit Maschinenerstellungsdienste (MCS) oder Provisioning Services VDAs mit einem konfigurierten Masterimage und installiertem Microsoft Message Queuing (MSMQ) erstellt werden. In diesem Fall haben die VDAs die gleiche QMId für MSMQ.

Lösung: Erstellen Sie eine eindeutige QMId für jeden VDA. Einen Workaround finden Sie unter [Bekannte Probleme](#).

### Test der Verbindung mit Datenbankinstanz schlägt bei Installation der Datenbank für Sitzungsaufzeichnung oder Sitzungsaufzeichnungsserver fehl

Bei der Installation der Datenbank für die Sitzungsaufzeichnung oder des Sitzungsaufzeichnungservers schlägt der Test der Verbindung fehl und es wird eine **Fehlermeldung mit der Aufforderung** zur Korrektur des Namens der Datenbankinstanz **angezeigt, selbst wenn dieser richtig ist**.

Lösung: Stellen Sie sicher, dass der aktuelle Benutzer die öffentliche SQL Server-Rollenberechtigung hat, damit der Test



nicht aufgrund einer fehlenden Berechtigung fehlschlägt.

# Prüfen der Komponentenverbindungen

May 10, 2016

Beim Setup der Sitzungsaufzeichnung stellen die Komponenten möglicherweise keine Verbindung mit den anderen Komponenten her. Alle Komponenten kommunizieren mit dem Sitzungsaufzeichnungsserver (Broker). In der Standardeinstellung ist der Broker (eine IIS-Komponente) mit dem Standardwebsitezertifikat von IIS gesichert. Wenn eine Komponente keine Verbindung mit dem Sitzungsaufzeichnungsserver herstellen kann, kann der Verbindungsversuch der anderen Komponenten auch fehlschlagen.

Der Sitzungsaufzeichnungsagent und der Sitzungsaufzeichnungsserver (Speichermanager und Broker) protokollieren Verbindungsfehler im Ereignisprotokoll der Anwendungen in der Ereignisanzeige auf dem Computer mit dem Sitzungsaufzeichnungsserver. Die Richtlinienkonsole für die Sitzungsaufzeichnung und der Sitzungsaufzeichnungsplayer zeigen Fehlermeldungen auf dem Bildschirm an, wenn keine Verbindung hergestellt werden kann.

## Prüfen, ob der Sitzungsaufzeichnungsagent verbunden ist

1. Melden Sie sich bei dem Server an, auf dem der Sitzungsaufzeichnungsagent installiert ist.
2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsagent - Eigenschaften.
3. Klicken Sie in den Sitzungsaufzeichnungsserver - Eigenschaften auf Verbindung.
4. Stellen Sie sicher, dass der Wert für Sitzungsaufzeichnungsserver den Namen des Computers enthält, auf dem der Sitzungsaufzeichnungsserver ausgeführt wird.
5. Stellen Sie sicher, dass der unter Sitzungsaufzeichnungsserver angegebene Server von der Serverbetriebssystemmaschine kontaktiert werden kann.

Hinweis: Prüfen Sie das Anwendungsereignisprotokoll auf Fehler und Warnungen.

## Prüfen, ob der Sitzungsaufzeichnungsserver verbunden ist

Achtung: Die Verwendung des Registrierungs-Editors kann zu schwerwiegenden Problemen führen, die nur durch eine Neuinstallation des Betriebssystems gelöst werden können. Citrix übernimmt keine Garantie dafür, dass Probleme, die auf eine falsche Verwendung des Registrierungs-Editors zurückzuführen sind, behoben werden können. Die Verwendung des Registrierungs-Editors geschieht daher auf eigene Gefahr.

1. Melden Sie sich bei dem Computer mit dem Sitzungsaufzeichnungsserver an.
2. Öffnen Sie den Registrierungs-Editor.
3. Navigieren Sie auf HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\SmartAuditor\Server.
4. Stellen Sie sicher, dass der Wert für SmAudDatabaseInstance richtig auf die Datenbank für die Sitzungsaufzeichnung verweist, die Sie in der SQL Server-Instanz installiert haben.

## Prüfen, ob die Datenbank für die Sitzungsaufzeichnung verbunden ist

1. Öffnen Sie die SQL-Instanz, die die installierte Datenbank für die Sitzungsaufzeichnung enthält, in einem SQL-Verwaltungstool.
2. Öffnen Sie die Sicherheitsberechtigungen der Datenbank für die Sitzungsaufzeichnung.
3. Stellen Sie sicher, dass das Computerkonto der Sitzungsaufzeichnung Zugriffsrechte auf die Datenbank hat. Beispiel: Wenn der Server mit dem Sitzungsaufzeichnungsserver "SsRecSrv" in der MIS-Domäne genannt wird, muss das Computerkonto in der Datenbank als MIS\SsRecSrv\$ konfiguriert werden. Dieser Wert wird während der Installation der Datenbank für die Sitzungsaufzeichnung konfiguriert.

## Testen der IIS-Konnektivität

Mit dem Testen der Verbindungen zwischen Sitzungsaufzeichnungsserver und IIS-Site unter Zugriff auf die Webseite des Sitzungsaufzeichnungsbrokers mit einem Webbrowser können Sie feststellen, ob Kommunikationsprobleme zwischen den Sitzungsaufzeichnungskomponenten auf eine falsche Protokollkonfiguration, Zertifizierungsprobleme oder Probleme beim Start des Sitzungsaufzeichnungsbrokers zurückzuführen sind.

## So prüfen Sie die IIS-Konnektivität für den Sitzungsaufzeichnungsagent

1. Melden Sie sich bei dem Server an, auf dem der Sitzungsaufzeichnungsagent installiert ist.
2. Öffnen Sie einen Webbrowser und geben Sie die folgende Adresse ein:
  - HTTPS:https://servername/SessionRecordingBroker/RecordPolicy.rem?wsdlwobei servername der Name des Computers ist, auf dem der Sitzungsaufzeichnungsserver ausgeführt wird.
  - HTTP:http://servername/SessionRecordingBroker/RecordPolicy.rem?wsdlwobei servername der Name des Computers ist, auf dem der Sitzungsaufzeichnungsserver ausgeführt wird.
3. Wenn Sie zur NTLM-Authentifizierung aufgefordert werden, melden Sie sich mit dem Domänenadministratorkonto an.

Wenn ein XML-Dokument im Browser angezeigt wird, bestätigt dies, dass der Computer, auf dem der Sitzungsaufzeichnungsagent ausgeführt wird, mit dem Sitzungsaufzeichnungsserver-Computer verbunden ist und das konfigurierte Protokoll verwendet.

## So prüfen Sie die IIS-Konnektivität für den Sitzungsaufzeichnungsplayer

1. Melden Sie sich bei der Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
2. Öffnen Sie einen Webbrowser und geben Sie die folgende Adresse ein:
  - HTTPS:https://servername/SessionRecordingBroker/Player.rem?wsdlwobei servername der Name des Computers ist, auf dem der Sitzungsaufzeichnungsserver ausgeführt wird.
  - HTTP:http://servername/SessionRecordingBroker/Player.rem?wsdlwobei servername der Name des Computers ist, auf dem der Sitzungsaufzeichnungsserver ausgeführt wird.
3. Wenn Sie zur NTLM-Authentifizierung aufgefordert werden, melden Sie sich mit dem Domänenadministratorkonto an.

Wenn ein XML-Dokument im Browser angezeigt wird, bestätigt dies, dass der Computer, auf dem der Sitzungsaufzeichnungsplayer ausgeführt wird, mit dem Sitzungsaufzeichnungsserver-Computer verbunden ist und das konfigurierte Protokoll verwendet.

## So prüfen Sie die IIS-Konnektivität für die Richtlinienkonsole für die Sitzungsaufzeichnung

1. Melden Sie sich bei dem Server an, auf dem die Richtlinienkonsole für die Sitzungsaufzeichnung installiert ist.
2. Öffnen Sie einen Webbrowser und geben Sie die folgende Adresse ein:
  - HTTPS:https://servername/SessionRecordingBroker/PolicyAdministration.rem?wsdlwobei servername der Name des Computers ist, auf dem der Sitzungsaufzeichnungsserver ausgeführt wird.
  - HTTP:http://servername/SessionRecordingBroker/PolicyAdministration.rem?wsdlwobei servername der Name des Computers ist, auf dem der Sitzungsaufzeichnungsserver ausgeführt wird.
3. Wenn Sie zur NTLM-Authentifizierung aufgefordert werden, melden Sie sich mit dem Domänenadministratorkonto an.

Wenn ein XML-Dokument im Browser angezeigt wird, bestätigt dies, dass der Computer, auf dem die Richtlinienkonsole für die Sitzungsaufzeichnung ausgeführt wird, mit dem Sitzungsaufzeichnungsserver-Computer verbunden ist und das konfigurierte Protokoll verwendet.

### Problembehandlung bei Zertifikaten

Wenn Sie HTTPS als Kommunikationsprotokoll verwenden, muss der Computer mit dem Sitzungsaufzeichnungsserver mit

einem Serverzertifikat konfiguriert werden. Alle Komponentenverbindungen mit dem Sitzungsaufzeichnungsserver müssen ein Zertifikat der Stammzertifizierungsstelle haben. Sonst schlagen die Verbindungen zwischen den Komponenten fehl.

Sie können die Zertifikate genauso wie beim Testen der IIS-Konnektivität durch Zugriff auf die Webseite des Sitzungsaufzeichnungsbrowsers testen. Wenn Sie auf die XML-Seite für jede Komponente zugreifen können, sind die Zertifikate richtig konfiguriert.

Im Anschluss finden Sie Gründe, warum Zertifikate zu Verbindungsproblemen führen:

- **Ungültige oder fehlende Zertifikate:** Wenn der Server mit dem Sitzungsaufzeichnungsagent kein Stammzertifikat für die Vertrauenswürdigkeit des Serverzertifikats hat, den Sitzungsaufzeichnungsserver nicht als vertrauenswürdig ansieht und keine Verbindung über HTTPS herstellen kann, schlägt die Verbindung fehl. Stellen Sie sicher, dass alle Komponenten das Serverzertifikat auf dem Sitzungsaufzeichnungsserver als vertrauenswürdig einstufen.
- **Inkonsistente Benennung:** Wenn das Serverzertifikat, das dem Computer mit dem Sitzungsaufzeichnungsserver zugewiesen ist, mit einem vollqualifizierten Domännennamen (FQDN) erstellt wurde, müssen alle Komponenten, die eine Verbindung herstellen, für die Verbindung mit dem Sitzungsaufzeichnungsserver den vollqualifizierten Domännennamen verwenden. Wenn ein NetBIOS-Name verwendet wird, konfigurieren Sie die Komponenten mit einem NetBIOS-Namen für den Sitzungsaufzeichnungsserver.
- **Abgelaufene Zertifikate:** Wenn ein Serverzertifikat abgelaufen ist, schlägt eine Verbindung mit dem Sitzungsaufzeichnungsserver über HTTPS fehl. Stellen Sie sicher, dass das Zertifikat, das dem Computer mit dem Sitzungsaufzeichnungsserver zugewiesen ist, gültig und nicht abgelaufen ist. Wenn dasselbe Zertifikat für die digitale Signatur der Sitzungsaufzeichnungen verwendet wird, enthält das Ereignisprotokoll des Computers mit dem Sitzungsaufzeichnungsserver Fehlermeldungen, dass das Zertifikat abgelaufen ist, oder Warnmeldungen, wenn das Zertifikat bald abläuft.

# Probleme beim Suchen nach Aufzeichnungen im Sitzungsaufzeichnungsplayer

May 10, 2016

Wenn Sie Probleme beim Suchen nach Aufzeichnungen im Sitzungsaufzeichnungsplayer haben, werden möglicherweise die folgenden Fehlermeldungen angezeigt:

- Fehler bei der Suche nach Sitzungsaufzeichnungsdateien. Der Name des Remoteservers konnte nicht aufgelöst werden: Servername. Wobei Servername der Name des Servers ist, mit dem der Sitzungsaufzeichnungsplayer eine Verbindung herstellen möchte. Der Sitzungsaufzeichnungsplayer kann den Sitzungsaufzeichnungsserver nicht kontaktieren. Zu den zwei möglichen Gründen gehören ein falsch eingegebener Servername, oder DNS kann den Servernamen nicht auflösen. Lösung: Klicken Sie im Player-Menü auf Extras > Optionen > Verbindungen und prüfen Sie die Richtigkeit des Servernamens, der in der Liste Sitzungsaufzeichnungsserver aufgeführt ist. Wenn der Name richtig ist, stellen Sie mit dem Ping-Befehl sicher, dass der Name aufgelöst werden kann. Wenn der Sitzungsaufzeichnungsserver nicht betriebsbereit oder offline geschaltet ist, tritt ein Fehler beim Suchen nach Sitzungsaufzeichnungsdateien auf. Die Fehlermeldung lautet Fehler beim Verbinden mit dem Remoteserver.
- Fehler beim Verbinden mit dem Remoteserver. Dieser Fehler tritt auf, wenn der Sitzungsaufzeichnungsserver nicht betriebsbereit oder offline geschaltet ist. Lösung: Stellen Sie sicher, dass der Sitzungsaufzeichnungsserver verbunden ist.
- Fehler: Zugriff verweigert. Ein Fehler "Zugriff verweigert" kann auftreten, wenn der Benutzer nicht berechtigt ist, Sitzungsaufzeichnungsdateien zu suchen und herunterzuladen. Lösung: Weisen Sie den Benutzer in der Sitzungsaufzeichnungs-Autorisierungskonsole der Rolle "Player" zu.
- Fehler bei der Suche nach Sitzungsaufzeichnungsdateien. Die zugrunde liegende Verbindung wurde geschlossen. Eine vertrauenswürdige Beziehung konnte für den sicheren Kanal (SSL/TLS) nicht erstellt werden. Diese Ausnahme tritt auf, wenn der Sitzungsaufzeichnungsserver ein Zertifikat verwendet, das von einer Zertifizierungsstelle signiert ist, die das Clientgerät nicht vertrauenswürdig ansieht oder für die das Clientgerät kein Zertifikat der Zertifizierungsstelle hat. Lösung: Installieren Sie das richtige oder vertrauenswürdige Zertifikat der Zertifizierungsstelle auf der Arbeitsstation, auf der der Sitzungsaufzeichnungsplayer installiert ist.
- Der Remoteserver gab einen Fehler zurück: (403) verboten. Dies ist ein HTTPS-Standardfehler, der angezeigt wird, wenn Sie eine Verbindung mit HTTP (nicht sicheres Protokoll) versuchen. Der Server lehnt die Verbindung ab, da er standardmäßig nur sichere Verbindungen annimmt. Lösung: Klicken Sie auf der Menüleiste des Sitzungsaufzeichnungsplayers auf Extras > Optionen > Verbindungen. Wählen Sie den Server aus der Liste Sitzungsaufzeichnungsserver aus und klicken Sie auf Ändern. Ändern Sie das Protokoll von HTTP in HTTPS.

## Problembehandlung bei MSMQ

Wenn den Benutzern eine Benachrichtigung angezeigt wird, der Benutzer die Aufzeichnungen jedoch nicht nach dem Durchführen einer Suche im Sitzungsaufzeichnungsplayer finden kann, kann ein Problem mit MSMQ bestehen. Stellen Sie sicher, dass die Warteschlange mit dem Sitzungsaufzeichnungsserver (Speichermanager) verbunden ist. Testen Sie mit einem Webbrowser, ob Verbindungsfehler bestehen (wenn Sie HTTP oder HTTPS als MSMQ-Kommunikationsprotokoll verwenden).

So stellen Sie sicher, dass die Warteschlange verbunden ist

1. Melden Sie sich bei dem Server an, auf dem der Sitzungsaufzeichnungsagent gehostet wird, und zeigen Sie die ausgehenden Warteschlangen an.
2. Stellen Sie sicher, dass die Warteschlange des Computers mit dem Sitzungsaufzeichnungsserver verbunden ist.
  - Wenn der Zustand "Warten auf Verbindung" ist, sind mehrere Nachrichten in der Warteschlange und als Protokoll wird HTTP oder HTTPS verwendet (gemäß der Auswahl auf der Registerkarte Verbindungen im Dialogfeld Sitzungsaufzeichnungsagent - Eigenschaften); führen Sie Schritt 3 aus.
  - Wenn der Zustand "Verbunden" ist und keine Nachrichten in der Warteschlange sind, besteht möglicherweise ein Problem mit dem Server, auf dem der Sitzungsaufzeichnungsserver ausgeführt wird. Überspringen Sie Schritt 3 und führen Sie Schritt 4 aus.
3. Wenn mehrere Nachrichten in der Warteschlange sind, starten Sie einen Webbrowser und geben Sie folgende Adresse ein:
  - HTTPS:https://servername/msmq/private\$/CitrixSmAudData wobei servername der Name des Computers ist, auf dem der Sitzungsaufzeichnungsserver ausgeführt wird.
  - HTTP:http://servername/msmq/private\$/CitrixSmAudData wobei servername der Name des Computers ist, auf dem der Sitzungsaufzeichnungsserver ausgeführt wird.

Wenn die Seite einen Fehler zurückgibt, z. B. Der Server nimmt nur sichere Verbindungen an, ändern Sie das für MSMQ im Dialogfeld Sitzungsaufzeichnungsagent - Eigenschaften aufgeführte Protokoll in HTTPS. Wenn die Seite ein Problem mit dem Sicherheitszertifikat der Website meldet, besteht möglicherweise ein Problem mit der Vertrauensbeziehung für den sicheren Kanal (SSL/TLS). Installieren Sie dann das richtige Zertifikat der Zertifizierungsstelle oder verwenden Sie eine vertrauenswürdige Zertifizierungsstelle.

4. Wenn die Warteschlange keine Nachrichten enthält, melden Sie sich bei dem Computer mit dem Sitzungsaufzeichnungsserver an und zeigen Sie private Warteschlangen an. Wählen Sie citrixsmduddata. Wenn mehrere Nachrichten in der Warteschlange sind (Spalte "Nachrichtenzahl"), stellen Sie sicher, dass der Dienst des Speichermanagers der Sitzungsaufzeichnung gestartet ist. Starten Sie sonst den Dienst neu.

# Ändern des Kommunikationsprotokolls

May 10, 2016

Aus Sicherheitsgründen empfiehlt Citrix, HTTP nicht als Kommunikationsprotokoll zu verwenden. Die Sitzungsaufzeichnung ist für die Verwendung von HTTPS konfiguriert. Wenn Sie HTTP statt HTTPS verwenden möchten, müssen Sie mehrere Einstellungen ändern.

## So verwenden Sie HTTP als Kommunikationsprotokoll

1. Melden Sie sich bei dem Computer mit dem Sitzungsaufzeichnungsserver an und deaktivieren Sie sichere Verbindungen für den Sitzungsaufzeichnungsbroker in IIS.
2. Ändern Sie das eingestellte Protokoll von HTTPS in HTTP in jedem Dialogfeld Sitzungsaufzeichnungsagent - Eigenschaften:
  1. Melden Sie sich bei jedem Server an, auf dem der Sitzungsaufzeichnungsagent installiert ist.
  2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsagent - Eigenschaften.
  3. Klicken unter "Sitzungsaufzeichnungsagent - Eigenschaften" auf die Registerkarte Verbindungen.
  4. Klicken Sie im Bereich Sitzungsaufzeichnungsbroker in der Dropdownliste Protokoll auf HTTP und bestätigen Sie die Änderung mit OK. Bestätigen Sie den Neustart des Dienstes mit Ja.
3. Ändern Sie die Protokolleinstellung in den Sitzungsaufzeichnungsplayer-Einstellungen von HTTPS in HTTP:
  1. Melden Sie sich bei jeder Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
  2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.
  3. Klicken Sie im Menü von Sitzungsaufzeichnungsplayer auf Extras > Optionen > Verbindungen, wählen Sie den Server aus und klicken Sie auf Ändern.
  4. Wählen Sie HTTP aus der Dropdownliste Protokoll aus und klicken Sie zwei Mal auf OK, um die Änderung zu akzeptieren und das Dialogfeld zu schließen.
4. Ändern Sie die Protokolleinstellung in den Einstellungen der Richtlinienkonsole für die Sitzungsaufzeichnung von HTTPS in HTTP:
  1. Melden Sie sich bei dem Server an, auf dem die Richtlinienkonsole für die Sitzungsaufzeichnung installiert ist.
  2. Klicken Sie im Menü Start auf Richtlinienkonsole für die Sitzungsaufzeichnung.
  3. Wählen Sie HTTP aus der Dropdownliste Protokoll und klicken Sie auf OK, um die Verbindung herzustellen. Wenn die Verbindung erfolgreich hergestellt wird, wird diese Einstellung gespeichert und beim nächsten Öffnen der Richtlinienkonsole für die Sitzungsaufzeichnung verwendet.

## So setzen Sie das Kommunikationsprotokoll auf HTTPS zurück

1. Melden Sie sich bei dem Computer mit dem Sitzungsaufzeichnungsserver an und aktivieren Sie sichere Verbindungen für den Sitzungsaufzeichnungsbroker in IIS.
2. Ändern Sie das eingestellte Protokoll von HTTP in HTTPS in jedem Dialogfeld Sitzungsaufzeichnungsagent - Eigenschaften:
  1. Melden Sie sich bei jedem Server an, auf dem der Sitzungsaufzeichnungsagent installiert ist.
  2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsagent - Eigenschaften.
  3. Klicken unter "Sitzungsaufzeichnungsagent - Eigenschaften" auf die Registerkarte Verbindungen.
  4. Klicken Sie im Bereich Sitzungsaufzeichnungsbroker in der Dropdownliste Protokoll auf HTTPS und bestätigen Sie die Änderung mit OK. Bestätigen Sie den Neustart des Dienstes mit Ja.
3. Ändern Sie die Protokolleinstellung in den Sitzungsaufzeichnungsplayer-Einstellungen von HTTP in HTTPS:
  1. Melden Sie sich bei jeder Arbeitsstation an, auf der der Sitzungsaufzeichnungsplayer installiert ist.
  2. Klicken Sie im Menü Start auf Sitzungsaufzeichnungsplayer.

3. Klicken Sie im Menü von Sitzungsaufzeichnungsplayer auf Extras > Optionen > Verbindungen, wählen Sie den Server aus und klicken Sie auf Ändern.
4. Wählen Sie HTTPS aus der Dropdownliste Protokoll aus und klicken Sie zwei Mal auf OK, um die Änderung zu akzeptieren und das Dialogfeld zu schließen.
4. Ändern Sie die Protokolleinstellung in den Einstellungen der Richtlinienkonsole für die Sitzungsaufzeichnung von HTTP in HTTPS:
  1. Melden Sie sich bei dem Server an, auf dem die Richtlinienkonsole für die Sitzungsaufzeichnung installiert ist.
  2. Klicken Sie im Menü Start auf Richtlinienkonsole für die Sitzungsaufzeichnung.
  3. Wählen Sie HTTPS aus der Dropdownliste Protokoll und klicken Sie auf OK, um die Verbindung herzustellen. Wenn die Verbindung erfolgreich hergestellt wird, wird diese Einstellung gespeichert und beim nächsten Öffnen der Richtlinienkonsole für die Sitzungsaufzeichnung verwendet.



# Verweis: Verwalten der Datensätze in der Datenbank

May 10, 2016

Das ICLDB-Dienstprogramm (ICA Log database) ist ein Datenbankbefehlszeilenprogramm, mit dem Sie Sitzungsaufzeichnungsdatensätze in der Datenbank manipulieren. Dieses Dienstprogramm wird mit der Sitzungsaufzeichnung im Verzeichnis Laufwerk:\Programme\Citrix\SessionRecording\Server\Bin auf dem Server mit der Serversoftware der Sitzungsaufzeichnung installiert.

## Übersichtstabelle

In der folgenden Tabelle finden Sie die Befehle und Optionen für das ICLDB-Dienstprogramm. Geben Sie die Befehle im folgenden Format ein:

icldb [version | locate | dormant | import | archive | remove | removeall] Befehlsoptionen [/l] [/f] [/s] [/?]

Hinweis: Weitere Informationen finden Sie in der Hilfe des Hilfsprogramms. Zum Öffnen der Hilfe geben Sie an der Eingabeaufforderung Laufwerk:\%Programme%\Citrix\SessionRecording\Server\Bin den Befehl icldb /? ein. Hilfe für bestimmte Befehle rufen Sie mit dem Befehl icldb Befehl /? auf.

Befehl	Beschreibung
archive	Archiviert Sitzungsaufzeichnungsdateien, die älter als der angegebene Speicherzeitraum sind. Archivieren Sie Dateien mit diesem Befehl.
dormant	Zählt oder zeigt die Sitzungsaufzeichnungsdateien an, die als inaktiv angesehen werden. Inaktive Dateien sind Sitzungsaufzeichnungen, die aufgrund von Datenverlust nicht abgeschlossen wurden. Verwenden Sie diesen Befehl, wenn Sie den Verdacht haben, dass Sie Daten verlieren. Sie können prüfen, ob die Sitzungsaufzeichnungsdateien für die ganze Datenbank oder nur Aufzeichnungen inaktiv werden, die in der angegebenen Anzahl von Tagen, Stunden oder Minuten aufgezeichnet wurden.
import	Importiert Sitzungsaufzeichnungsdateien in die Datenbank für die Sitzungsaufzeichnung. Mit diesem Befehl erstellen Sie die Datenbank neu, wenn Sie Datensätze der Datenbank verlieren.  Mit diesem Befehl führen Sie auch Datenbanken zusammen (wenn Sie zwei Datenbanken haben, können Sie die Dateien von einer der Datenbanken importieren).
locate	Sucht und zeigt den vollständigen Pfad einer Sitzungsaufzeichnungsdatei an. Als Kriterium wird die Datei-ID verwendet. Mit diesem Befehl suchen Sie den Speicherort einer Sitzungsaufzeichnungsdatei.  Außerdem können Sie mit einer bestimmten Datei prüfen, ob die Datenbank aktuell ist.
remove	Entfernt die Verweise auf Sitzungsaufzeichnungsdateien aus der Datenbank. Mit diesem Befehl bereinigen Sie die Datenbank (verwenden Sie diesen Befehl mit Vorsicht).

Befehl	Beschreibung
	Geben Sie den Speicherzeitraum als Kriterium an. Sie können auch die zugeordnete physische Datei löschen.
removeall	Entfernt alle Verweise auf Sitzungsaufzeichnungsdateien aus der Datenbank für die Sitzungsaufzeichnung und setzt diese auf den Originalzustand zurück. Die physische Datei wird nicht gelöscht. Sie können diese Dateien jedoch nicht im Sitzungsaufzeichnungsplayer suchen. Mit diesem Befehl bereinigen Sie die Datenbank (verwenden Sie diesen Befehl mit Vorsicht). Gelöschte Verweise können nur von einer Sicherungskopie wieder hergestellt werden.
version	Zeigt die Schemaversion der Datenbank für die Sitzungsaufzeichnung an.
/l	Protokolliert die Ergebnisse und Fehler im Windows-Ereignisprotokoll.
/f	Erzwingt die Ausführung des Befehls ohne Aufforderungen.
/s	Unterdrückt die Copyright-Nachricht.
/?	Zeigt die Hilfe für die Befehle an.

# Hinweise von Dritten

May 10, 2016

Die Sitzungsaufzeichnung kann Softwarekomponenten von Drittanbietern enthalten, für die die folgenden Lizenzbedingungen gelten: Diese Liste gilt zu dem angegebenen Datum korrekt. Diese Liste kann sich bei bestimmten Versionen des Produkts ändern und ist möglicherweise nicht komplett. Die Liste wird ohne Mängelgewähr zur Verfügung gestellt. IN DEM NACH DEM ANWENDBAREN RECHT ZULÄSSIGEN UMFANG GEBEN CITRIX UND SEINE LIEFERANTEN KEINE AUSDRÜCKLICHE ODER STILLSCHWEIGENDEN ZUSICHERUNGEN ODER GEWÄHRLEISTUNGEN BEZÜGLICH DER LISTE, DEREN RICHTIGKEIT ODER VOLLSTÄNDIGKEIT ODER DEN AUS DER NUTZUNG ODER VERTEILUNG DER LISTE ERGEHENDEN FOLGEN. DURCH NUTZUNG ODER VERTEILUNG DER LISTE STIMMEN SIE ZU, DASS CITRIX UNTER KEINEN UMSTÄNDEN FÜR SPEZIELLE, DIREKTE, INDIREKTE, ODER FOLGESCHÄDEN ODER JEDGLICHE ANDEREN SCHÄDEN HAFTET, DIE AUS DER NUTZUNG ODER VERTEILUNG DIESER LISTE ERWACHSEN.

MMC .NET-Bibliothek

Lizenziert unter Common Public License, Version 1.0

# Personal vDisk

May 10, 2016

Sie können ein Diagnosetool zum Überwachen der Änderungen verwenden, die von Benutzern an beiden Bereichen persönlicher vDisks (Benutzerdaten- und Anwendungsbereiche) vorgenommen werden. Zu diesen Änderungen gehören Anwendungen, die von Benutzern installiert wurden, und von ihnen geänderte Dateien. Die Änderungen werden in einer Reihe von Berichten erfasst.

1. Führen Sie auf der zu überwachenden Maschine C:\Programme\Citrix\Personal vDisk\bin\CtxPvdDiag.exe aus.
2. Navigieren Sie zu einem Speicherort, an dem die Berichte und Protokolle gespeichert werden sollen, legen Sie fest, welche Berichte generiert werden sollen, und klicken Sie auf OK. Die folgenden Berichte sind verfügbar:

Bericht oder Protokoll	Generierte Dateien	Überwachte Änderungen
Softwarestrukturbericht	Software.Dat.Report.txt, Software.Dat.delta.txt	Software.Dat.Report.txt erfasst Änderungen, die vom Benutzer an der Struktur HKEY_LOCAL_MACHINE\Software vorgenommen werden. Der Bericht besteht aus den folgenden Abschnitten:      • <b>List of Applications installed on the base:</b> Anwendungen, die in Ebene 0 installiert wurden.    • <b>List of user installed software:</b> Anwendungen, die vom Benutzer im Anwendungsbereich der vDisk installiert wurden.    • <b>List of software uninstalled by user:</b> Ursprünglich in Ebene 0 installierte Anwendungen, die vom Benutzer entfernt wurden.      Weitere Informationen zu Software.Dat.delta.txt finden Sie im Strukturdeltabericht.
Systemstrukturbericht	SYSTEM.CurrentControlSet.DAT.Report.txt	In dieser Datei werden die Änderungen erfasst, die vom Benutzer an der Struktur HKEY_LOCAL_MACHINE\System vorgenommen werden. Der Bericht besteht aus den folgenden Abschnitten:      • <b>List of user installed services:</b> Vom Benutzer installierte Dienste und Treiber.    • <b>Startup of following services were changed:</b> Dienste und Treiber, deren Starttyp vom Benutzer geändert wurde.
Sicherheitsstrukturbericht	SECURITY.DAT.Report.txt	Diese Datei dient zur Überwachung aller Änderungen, die der Benutzer in der

Bericht oder Protokoll	Generierte Dateien	Struktur HKEY_LOCAL_MACHINE\Security <b>Überwachte Änderungen</b> vornimmt.
Strukturbericht für die Sicherheitskontenverwaltung (SAM)	SAM.DAT.Report.txt	Diese Datei dient zur Überwachung aller Änderungen, die der Benutzer in der Struktur HKEY_LOCAL_MACHINE\SAM vornimmt.
Strukturdeltabericht	Software.Dat.delta.txt	In dieser Datei werden alle Registrierungsschlüssel und Werte erfasst, die vom Benutzer in der Struktur HKEY_LOCAL_MACHINE\Software hinzugefügt oder entfernt wurden, sowie alle geänderten Werte.
Personal vDisk-Protokolle	Pvd-IvmSupervisor.log, PvdActivation.log, PvdSvc.log, PvdWMI.log, SysVol-IvmSupervisor.log, vDeskService-<#>.log	Diese Dateien werden standardmäßig in P:\Users\Local\Temp\PVDLOGS generiert und dann aber an den ausgewählten Speicherort verschoben.
Windows-Betriebssystemprotokoll	EvtLog_App.xml, EvtLog_System.xml, setupapi.app.log, setuperr.log, setupapi.dev.log, msinfo.txt	EvtLog_App.xml und EvtLog_System.xml sind die Anwendungs- und Systemereignisprotokolle im XML-Format aus dem persönlichen vDisk-Volumen.   Die Protokolle Setupapi.app.log und setuperr.log enthalten Protokollmeldungen bezüglich der Ausführung von msixec.exe bei der Installation von Personal vDisk.   Setupapi.dev.log enthält Protokollmeldungen über die Geräteinstallation.   Msinfo.txt enthält die Ausgabe von msinfo32.exe. Informationen zu dieser Ausgabe finden Sie in der Microsoft-Dokumentation.
Dateisystembericht	FileSystemReport.txt	In dieser Datei werden die Änderungen erfasst, die vom Benutzer am Dateisystem vorgenommen werden. Der Bericht besteht aus den folgenden Abschnitten:       • <b>Files Relocated:</b> Dateien in Ebene 0, die vom Benutzer zur vDisk verschoben wurden. Dateien der Ebene 0 sind Dateien, die aus dem Masterimage von der Maschine geerbt wurden, der die persönliche vDisk zugeordnet ist.     • <b>Files Removed:</b> Dateien in Ebene 0, die

Bericht oder Protokoll	Generierte Dateien	Überwachte Änderungen
		durch eine Benutzeraktion (z. B. Entfernen einer Anwendung) verborgen wurden.       • <b>Files Added (MOF,INF,SYS):</b> Dateien mit der Erweiterung "mof", "inf" oder "sys", die vom Benutzer der vDisk hinzugefügt wurden (z. B. bei der Installation einer Anwendung wie Visual Studio 2010, bei der eine MOF-Datei für die automatische Wiederherstellung registriert wird).     • <b>Files Added Other:</b> Andere Dateien, die der vDisk vom Benutzer hinzugefügt wurden (z. B. beim Installieren einer Anwendung).     • <b>Base Files Modified But Not Relocated:</b> Dateien in Ebene 0, die vom Benutzer geändert wurden, jedoch nicht von den Personal vDisk-Kernelmodustreibern in der vDisk erfasst wurden.

# Konfigurationsprotokollierung

May 10, 2016

Die Konfigurationsprotokollierung dient zum Erfassen der Sitekonfigurationsänderungen und Administratoraktivitäten in einer Datenbank. Sie können den protokollierten Inhalt folgendermaßen verwenden:

- Diagnose und Problembehandlung nach der Durchführung von Konfigurationsänderungen; das Protokoll liefert eine Breadcrumbspur.
- Hilfe beim Änderungsmanagement und der Nachverfolgung von Konfigurationen
- Bericht über Administratoraktivitäten

Zum Festlegen der Einstellungen für die Konfigurationsprotokollierung, zum Anzeigen der Konfigurationsprotokolle und zum Generieren von HTML- und CSV-Berichten verwenden Sie Citrix Studio. Sie können die Anzeige des Konfigurationsprotokolls durch Datumsbereiche und durch Ergebnisse der Volltextsuche filtern. Ist die verbindliche Protokollierung aktiviert, verhindert sie, dass Änderungen an der Konfiguration vorgenommen werden, es sei denn diese können protokolliert werden. Mit der entsprechenden Berechtigung können Sie Einträge aus dem Konfigurationsprotokoll löschen. Sie können das Feature der Konfigurationsprotokollierung nicht zum Bearbeiten des Inhalts von Protokollen verwenden.

Die Konfigurationsprotokollierung verwendet ein PowerShell 2.0-SDK und den Konfigurationsprotokollierungsdienst. Der Konfigurationsprotokollierungsdienst wird auf allen Controllern in der Site ausgeführt. Wenn ein Controller ausfällt, übernimmt automatisch der Dienst auf einem anderen Controller die Verarbeitung von Protokollanforderungen.

Standardmäßig ist die Konfigurationsprotokollierung aktiviert und verwendet die Datenbank, die zusammen mit der Site erstellt wurde (die Sitekonfigurationsdatenbank). Citrix empfiehlt dringend, den Speicherort der Datenbank für die Konfigurationsprotokollierung so schnell wie möglich nach der Erstellung einer Site zu ändern. Die Konfigurationsprotokollierungsdatenbank unterstützt dieselben Features für hohe Verfügbarkeit wie die Sitekonfigurationsdatenbank.

Der Zugriff auf die Konfigurationsprotokollierung wird über die delegierte Administration mit den Einstellungen "Protokollierungseinstellungen bearbeiten" und "Konfigurationsprotokolle anzeigen" gesteuert.

Konfigurationsprotokolle werden bei der Erstellung lokalisiert. Beispiel: Ein in Englisch erstelltes Protokoll wird unabhängig vom Gebietsschema des Lesers in Englisch gelesen.

Konfigurationsänderungen und Administratoraktivitäten, die von Studio, Director und PowerShell-Skripts ausgehen, werden protokolliert. Beispiele protokollierter Konfigurationsänderungen sind Arbeiten (Erstellen, Bearbeiten, Löschen, Zuweisen) mit:

- Maschinenkataloge
- Bereitstellungsgruppen (einschließlich Ändern der Energieverwaltungseinstellungen)
- Administratorrollen und Geltungsbereiche
- Hostressourcen und Verbindungen
- Citrix Richtlinien über Studio

Beispiele protokollierter Administratoraktivitäten:

- Energieverwaltung für eine virtuelle Maschine oder einen Benutzerdesktop
- Senden einer Nachricht an einen Benutzer von Studio oder Director aus

Die folgenden Vorgänge werden nicht protokolliert:

- Autonome Vorgänge wie das Einschalten virtueller Maschinen per Poolverwaltung.
- Über die Gruppenrichtlinien-Verwaltungskonsole implementierte Richtlinienaktionen; verwenden Sie Microsoft-Tools, um Protokolle dieser Aktionen anzuzeigen.
- Über die Registrierung vorgenommene Änderungen, direkter Zugriff von der Datenbank oder von anderen Quellen als Studio, Director oder PowerShell.
- Wenn die Bereitstellung initialisiert wird, steht die Konfigurationsprotokollierung ab dem Zeitpunkt zur Verfügung, zu dem die erste Instanz des Konfigurationsprotokollierungsdiensts sich beim Konfigurationsdienst registriert. Daher werden die frühen Phasen der Konfiguration nicht protokolliert (z. B., wenn das Datenbankschema bei der Initialisierung eines Hypervisors abgerufen und angewendet wird).



# Verwalten der Konfigurationsprotokollierung

May 10, 2016

Standardmäßig wird für die Konfigurationsprotokollierung die Datenbank verwendet, die zusammen mit einer Site erstellt wird (die Sitekonfigurationsdatenbank). Citrix empfiehlt, den Speicherort der Datenbank für die Konfigurationsprotokollierung (und der Datenbank für den Überwachungsdienst, von dem die Sitekonfigurationsdatenbank auch standardmäßig verwendet wird) aus folgenden Gründen nach der Erstellung einer Site zu ändern:

- Die Backupstrategie für die Konfigurationsprotokollierungsdatenbank unterscheidet sich wahrscheinlich von der Backupstrategie für die Sitekonfigurationsdatenbank.
- Die Menge der für die Konfigurationsprotokollierung (und den Überwachungsdienst) gesammelten Daten kann den für die Sitekonfigurationsdatenbank verfügbaren Speicherplatz zu stark limitieren.
- Eine einzelne Fehlerquelle für die drei Datenbanken wird beseitigt (d. h. aufgeteilt).

Hinweis: Produkteditionen, die keine Konfigurationsprotokollierung unterstützen, haben keinen Knoten namens "Protokollierung" in Studio. Weitere Informationen finden Sie unter [XenDesktop 7.6 and XenApp 7.6 Features and Entitlements](#).

Standardmäßig ist die Konfigurationsprotokollierung aktiviert und die verbindliche Protokollierung ist deaktiviert.

1. Wählen Sie im Studio-Navigationsbereich Protokollierung aus.
2. Wählen Sie im Aktionsbereich Einstellungen aus. Das Dialogfeld "Konfigurationsprotokollierung" enthält die Datenbankinformationen und Angaben dazu, ob Konfigurationsprotokollierung und verbindliche Protokollierung aktiviert oder deaktiviert sind.
  - Zum Aktivieren der Konfigurationsprotokollierung wählen Sie das Optionsfeld Protokollierung aktivieren. Dies ist die Standardeinstellung. Wenn nicht in die Datenbank geschrieben werden kann, werden die Informationen verworfen, der Vorgang wird jedoch fortgesetzt.
  - Zum Deaktivieren der Konfigurationsprotokollierung wählen Sie das Optionsfeld Protokollierung deaktivieren. Wenn die Protokollierung zuvor aktiviert war, können bereits vorhandene Protokolle weiterhin mit dem PowerShell-SDK gelesen werden.
  - Zum Aktivieren der verbindlichen Protokollierung deaktivieren Sie das Kontrollkästchen Änderungen zulassen, wenn die Verbindung mit der Datenbank getrennt ist. Es wird dann keine Konfigurationsänderung oder administrative Aktivität, die normalerweise protokolliert würde, zugelassen, es sei denn, sie kann in die Datenbank für die Konfigurationsprotokollierung geschrieben werden.

Sie können die verbindliche Protokollierung nur aktivieren, wenn die Konfigurationsprotokollierung aktiviert ist, d. h. wenn das Optionsfeld "Protokollierung aktivieren" für die Konfigurationsprotokollierung aktiviert ist. Tritt bei dem Dienst für die Konfigurationsprotokollierung ein Fehler auf, und die hohe Verfügbarkeit wird nicht verwendet, beginnt die verbindliche Protokollierung. In solchen Fällen werden Vorgänge, die normalerweise protokolliert würden, nicht ausgeführt.
  - Zum Deaktivieren der verbindlichen Protokollierung aktivieren Sie das Kontrollkästchen Änderungen zulassen, wenn die Verbindung mit der Datenbank getrennt ist. Konfigurationsänderungen und administrative Aktivitäten sind dann zulässig, selbst wenn kein Zugriff auf die Datenbank für die Konfigurationsprotokollierung besteht. Dies ist die Standardeinstellung.

Hinweis: Sie können den Speicherort der Datenbank nicht ändern, wenn die verbindliche Protokollierung aktiviert ist, da bei der Standortänderung eine kurze Trennung verursacht wird, die nicht protokolliert werden kann.

1. Erstellen Sie einen Datenbankserver mit einer unterstützten SQL Server-Version.
2. Wählen Sie im Studio-Navigationsbereich Protokollierung aus.
3. Wählen Sie im Aktionsbereich Einstellungen aus.
4. Klicken Sie im Dialogfeld "Protokollierungseinstellungen" auf Protokollierungsdatenbank ändern.
5. Geben Sie im Dialogfeld "Protokollierungsdatenbank ändern" den Speicherort des Servers mit dem neuen Datenbankserver ein (in einem der Formate aus der folgenden Tabelle) sowie den Namen der Datenbank.

Datenbanktyp	Eingabe	Auswirkungen dieser Datenbankkonfiguration
Eigenständig oder gespiegelt	servername	Die Standardinstanz wird verwendet und SQL Server verwendet den Standardport.
	servername\INSTANCENAME	Eine benannte Instanz wird verwendet und SQL Server verwendet den Standardport.
	servername,port-number	Die Standardinstanz wird verwendet und SQL Server verwendet einen benutzerdefinierten Port. (Das Komma ist erforderlich.)
Sonstiges	cluster-name	Eine geclusterte Datenbank.
	availability-group-listener	Eine stets aktivierte Datenbank.

- Damit die Datenbank von Studio erstellt wird, klicken Sie auf OK. Wenn Sie dazu aufgefordert werden, klicken Sie auf OK und die Datenbank wird automatisch erstellt. Studio versucht, mit den Anmeldeinformationen des aktuellen Studio-Benutzers auf die Datenbank zuzugreifen; wenn dies fehlschlägt, werden Sie zur Eingabe der Anmeldeinformationen des Datenbankbenutzers aufgefordert. Das Datenbankschema wird dann von Studio in die Datenbank hochgeladen. (Die Anmeldeinformationen werden nur während der Datenbankerstellung gespeichert.)
- Zum manuellen Erstellen der Datenbank klicken Sie auf Datenbankskript erstellen. Das generierte Skript enthält Anweisungen zum manuellen Erstellen der Datenbank. Stellen Sie vor dem Hochladen des Schemas sicher, dass die Datenbank leer ist und dass mindestens ein Benutzer Zugriffs- bzw. Änderungsberechtigung für die Datenbank hat.

Die Daten der Konfigurationsprotokollierung aus der älteren Datenbank werden nicht in die neue Datenbank importiert. Die Protokolle beider Datenbanken können beim Abrufen von Protokollen nicht aggregiert werden. Der erste Protokolleintrag in der neuen Datenbank für die Konfigurationsprotokollierung gibt an, dass eine Datenbankänderung stattfand; die vorherige Datenbank wird jedoch nicht identifiziert.

Beim Initiieren von Konfigurationsänderungen und bei Verwaltungsaktivitäten werden die von Studio und Director bewirkten High-Level-Operationen im oberen mittleren Bereich von Studio angezeigt. Eine High-Level-Operation führt zu mindestens einem Dienst- und SDK-Aufruf, bei dem es sich um eine Low-Level-Operation handelt. Wenn Sie eine High-Level-Operation im oberen mittleren Bereich auswählen, werden im unteren mittleren Bereich die Low-Level-Operationen angezeigt.

Schlägt eine Operation vor der Beendigung fehl, kann die Protokollierung evtl. in der Datenbank nicht abgeschlossen werden, sodass z. B. ein Startdatensatz keinen entsprechenden Stoppsatz hat. In solchen Fällen wird im Protokoll

angezeigt, dass Informationen fehlen. Wenn Sie Protokolle auf Zeitbereichsbasis anzeigen, werden unvollständige Protokolle angezeigt, wenn die Daten in den Protokollen mit den Kriterien übereinstimmen. Beispiel: Wenn alle Protokolle für die letzten fünf Tage angefordert werden und ein Protokoll eine in den letzten fünf Tagen gelegene Startzeit aber keine Endzeit hat, wird dieses ebenfalls angezeigt.

Wenn Sie bei Verwendung eines Skripts zum Aufrufen von PowerShell-Cmdlets eine Low-Level-Operation erstellen ohne die übergeordnete High-Level-Operation anzugeben, wird von der Konfigurationsprotokollierung eine Ersatz-High-Level-Operation erstellt.

Zum Anzeigen des Inhalts des Konfigurationsprotokolls wählen Sie im Studio-Navigationsbereich Protokollierung aus. Standardmäßig wird im mittleren Bereich der Protokollinhalt chronologisch (neueste Einträge zuerst), angezeigt, wobei die Einträge durch das Datum getrennt sind.

Zum Filtern der Anzeige nach folgenden Kriterium	Führen Sie folgenden Schritt aus
Suchergebnisse	Geben Sie Text in das Feld "Suchen" oben im mittleren Bereich ein. Die gefilterte Anzeige enthält die Anzahl der Suchergebnisse. Zum Zurückkehren zur Standardprotokollanzeige löschen Sie den Text im Feld Suchen.
Spaltenüberschrift	Klicken Sie auf eine Spaltenüberschrift, um die Anzeige nach dem entsprechenden Feld zu sortieren.
Datumsbereich	Wählen Sie aus der Dropdownliste neben dem Feld "Suchen" oben im mittleren Bereich ein Intervall aus.

Sie können CSV- und HTML-Berichte mit Konfigurationsprotokolldaten generieren.

- Der CSV-Bericht enthält alle Protokolldaten aus einem angegebenen Zeitintervall. Die hierarchischen Daten in der Datenbank werden in eine einzelne CSV-Tabelle vereinfacht. Kein Aspekt der Daten hat Vorrang in der Datei. Es wird keine Formatierung verwendet und keine Lesbarkeit angenommen. Die Datei (unter dem Namen "MyReport") enthält lediglich die Daten in einem allgemein verwendbaren Format. CSV-Dateien werden oft für die Archivierung oder als Datenquelle für ein Tool zur Bearbeitung von Berichten oder Daten (z. B. Microsoft Excel) verwendet.
- Der HTML-Bericht enthält Protokolldaten aus einem angegebenen Zeitintervall in lesbarem Format. Er bietet eine strukturierte Ansicht für die Prüfung auf Änderungen, durch die navigiert werden kann. Der HTML-Bericht umfasst zwei Dateien: Zusammenfassung und Details. Die Zusammenfassung enthält High-Level-Operationen mit Informationen zu Zeitpunkt, Auslöser und Ergebnis. Durch Klicken auf den Link Details neben jedem Vorgang navigieren Sie zu den Low-Level-Operationen in der Datei Details, die zusätzliche Informationen bietet.

Zum Generieren eines Konfigurationsprotokollierungsberichts wählen Sie im Studio-Navigationsbereich Protokollierung und dann im Aktionsbereich Benutzerdefinierten Bericht erstellen aus.

- Wählen Sie den Datumsbereich für den Bericht.
- Wählen Sie das Berichtsformat: CSV, HTML oder beides.
- Navigieren Sie zu dem Speicherort, an dem der Bericht gespeichert werden soll.

Zum Löschen des Konfigurationsprotokolls müssen Sie über bestimmte Rechte der delegierten Administration und

Berechtigungen für die SQL Server-Datenbank verfügen.

- **Delegierte Administration:** Sie müssen eine Rolle der delegierten Administration haben, mit der die Bereitstellungsconfiguration gelesen werden kann. Die integrierte Volladministratorrolle hat diese Berechtigung. Für eine benutzerdefinierte Rolle muss für die Kategorie "Andere Berechtigungen" "Lesen" oder "Verwalten" aktiviert sein. Wenn Sie die Konfigurationsprotokolldaten vor dem Löschen sichern möchten, muss die benutzerdefinierte Rolle in der Kategorie der Protokollierungsberechtigungen Lese- oder Verwaltungsberechtigung haben.
- **SQL Server-Datenbank:** Sie müssen einen Anmeldenamen für SQL Server haben und zum Löschen von Datensätzen aus der Datenbank berechtigt sein. Dies kann mit zwei Möglichkeiten erreicht werden:
  - Verwenden Sie zur Anmeldung für die SQL Server-Datenbank die Serverrolle "sysadmin", mit der Sie beliebige Aktivitäten auf dem Datenbankserver durchführen können. Auch die Serverrollen "serveradmin" oder "setupadmin" sind zum Löschen von Vorgängen berechtigt.
  - Wenn Ihre Bereitstellung zusätzliche Sicherheit erfordert, verwenden Sie Anmeldeinformationen einer anderen Rolle als "sysadmin", die einem Datenbankbenutzer zugeordnet sind, der zum Löschen von Datensätzen aus der Datenbank berechtigt ist.
    1. Erstellen Sie in SQL Server Management Studio eine SQL Server-Anmeldung mit einer anderen Serverrolle (nicht "sysadmin").
    2. Ordnen Sie diese Anmeldung einem Benutzer in der Datenbank zu; SQL Server erstellt automatisch einen Benutzer in der Datenbank mit dem gleichen Namen wie die Anmeldung.
    3. Geben Sie für die Datenbankrollen-Mitgliedschaft mindestens eines der Rollenmitglieder für den Datenbankbenutzer an: ConfigurationLoggingSchema\_ROLE oder dbowner.Weitere Informationen finden Sie in der Dokumentation zu SQL Server Management Studio.

So löschen Sie die Konfigurationsprotokolle:

1. Wählen Sie im Studio-Navigationsbereich Protokollierung aus.
2. Wählen Sie im Aktionsbereich Protokolle löschen aus.
3. Sie haben nun die Möglichkeit, vor dem Löschen ein Backup der Protokolle anzulegen. Wenn Sie eine Backupdatei erstellen, navigieren Sie zu dem Speicherort, an dem diese gespeichert werden soll. Das Backup wird als CSV-Datei erstellt.

Nach dem Löschen der Konfigurationsprotokolle wird das Löschen des Protokolls als erste Aktivität im leeren Protokoll erfasst. Dieser Eintrag enthält Details darüber, wann und von wem die Protokolle gelöscht wurden.

# Überwachungsdienst-OData-API

May 10, 2016

Zusätzlich zur Anzeige von historischen Daten in der Citrix Director-Konsole können Sie Daten mit der Überwachungsdienst-OData-API abfragen. Sie können die API für Folgendes verwenden:

- Analysieren historischer Trends für die Planung
- Durchführen einer eingehenden Problembehandlung bei Verbindungs- und Maschinenfehlern
- Extrahieren von Informationen für die Eingabe in andere Tools und Prozesse, z. B. wenn PowerPivot-Tabellen in Microsoft Excel für die Anzeige von Daten auf verschiedene Weise verwendet werden
- Erstellen einer benutzerdefinierten Benutzeroberfläche auf der Basis der von der API bereitgestellten Daten

Die Überwachungsdienst-OData-API verwendet das Open Data-Protokoll (OData), ein Webprotokoll für die Abfrage und Aktualisierung von Daten, das auf Webtechnologien wie HTTP aufbaut. Weitere Informationen über das OData-Protokoll finden Sie unter <http://www.odata.org>.

Die Überwachungsdienst-API setzt mit den Windows Communication Foundation (WCF) Data Services auf SQL Server-Datenbanken auf, die während der Verarbeitung und Konsolidierung aufgefüllt werden. Bei Verwendung von WCF mit wsHttpBinding werden zwei Endpunkte verfügbar gemacht. Die Basisadresse ist `http://{dc-host}/Citrix/Monitor/OData/v2`. Sie können Endpunkte auch mit SSL schützen (weitere Informationen finden Sie unter [Schützen von Endpunkten mit SSL](#)).

1. Der Endpunkt "Data" gestattet direkten Lesezugriff auf die Datenbankentitäten, Zugriff ist mit der Abfragesprache OData möglich. Dieser Endpunkt gestattet einen sehr flexiblen Zugriff im Hinblick auf Filter- und Spaltenauswahl. Die URI der Data-API ist: `http://{dc-host}/Citrix/Monitor/OData/v2/Data`. Weitere Informationen über den Zugriff auf Monitor Service-Daten finden Sie unter [Zugriff auf Daten mit der API](#).
2. Der Endpunkt "Methods" macht Dienstvorgänge verfügbar, die von Citrix Director für das Abrufen von Daten verwendet werden, die eine komplexe Gruppierung und hohe Leistungsstandards erfordern (z. B. Abfragen für das Dashboard und Trend-Seiten). Die URI der Methods-API ist: `http://{dc-host}/Citrix/Monitor/OData/v2/Methods`. Methoden werden nur in Director selbst verwendet, d. h. nicht von der Mehrheit der Citrix Kunden. Sie werden daher hier nicht dokumentiert.

Die in XenApp und XenDesktop 7.6 enthaltene API-Version bietet die folgenden neuen Features:

- **Hotfix-Bestand** In der Ansicht "Benutzerdetails" oder „Maschine“ in Director können Sie eine Liste aller Citrix Hotfixes anzeigen, die auf einer Maschine installiert wurden. Sie können mit der API diese Daten extrahieren und benutzerdefinierte Berichte erstellen (z. B. Status der installierten Hotfixes über eine ganze Site) oder sie in eine Analyticsengine ziehen. Neue Klassen wurden eingeführt und die Maschinenklasse unterstützt nun die Nachverfolgung von auf dem Controller und VDAs installierten Citrix Hotfixes.
- **Anonyme Sitzungsproblembehandlung.** Sitzungen können in Form gepoolter lokaler Benutzerkonten ausgeführt werden. Die API hat eine neue `IsAnonymous`-Eigenschaft für die Sitzungsklasse (Standardwert = FALSE).
- **Bericht über die Verwendung gehosteter Anwendungen.** Director bietet neue Kapazitätsberichte zur Verwendung gehosteter Anwendungen über bestimmte Zeiträume. Mit der API können Sie Berichte zu den Details jeder in einer Benutzersitzung ausgeführten Anwendungsinstanz erstellen.

Alle die Daten betreffenden Updates sind in der API-Referenz unter <http://support.citrix.com/help/monitorserviceapi/7.6/> vollständig dokumentiert.

Die Methode "GetSessionSummary" ist seit diesem Release veraltet.

# Zugriff auf Daten mit der API

May 10, 2016

Die folgenden Datentypen sind über die Überwachungsdienst-API verfügbar:

- Daten zu Verbindungsfehlern
- Maschinen in fehlerhaftem Zustand
- Sitzungsnutzung
- Anmeldedauer
- Daten zum Lastausgleich
- Auf eine Maschine angewendete Hotfixes
- Verwendung gehostete Anwendung

Eine komplette Beschreibung der Datenobjekte finden Sie unter <http://blogs.citrix.com/2013/08/27/xendesktop-7-monitor-service-what-data-is-available/>.

Zur Verwendung der Überwachungsdienst-OData-API müssen Sie XenApp- bzw. XenDesktop-Administrator sein. Zum Aufrufen der API ist Lesezugriff erforderlich. Welche Daten zurückgegeben werden, hängt von den Rollen und Berechtigungen des XenApp- bzw. XenDesktop-Administrators ab. Administratoren von Bereitstellungsgruppen können beispielsweise die Überwachungsdienst-API aufrufen, aber die zurückgegebenen Daten hängen von dem mit Citrix Studio eingerichteten Bereitstellungsgruppenzugriff ab. Weitere Informationen über XenApp-/XenDesktop-Administratorrollen und -berechtigungen finden Sie unter [Delegierte Administration](#).

## Abfragen der Daten

Die Überwachungsdienst-API ist eine REST-basierte API, auf die mit einem OData-Consumer zugegriffen werden kann. OData-Consumer sind Anwendungen, die mit dem OData-Protokoll verfügbar gemachte Daten nutzen. OData-Consumer reichen vom einfachen Webbrowser bis zur benutzerdefinierten Anwendung, die sämtliche Features des OData-Protokolls nutzt. Weitere Informationen über OData-Consumer finden Sie unter <http://www.odata.org/ecosystem#consumers>.

Jeder Teil des Überwachungsdienst-Datenmodells ist zugänglich und kann über die URL gefiltert werden. OData bietet eine Abfragesprache im URL-Format zum Abrufen von Einträgen aus einem Dienst. Weitere Informationen finden Sie unter <http://msdn.microsoft.com/en-us/library/ff478141.aspx>.

Die Abfrage wird serverseitig verarbeitet und kann clientseitig mit dem OData-Protokoll weiter gefiltert werden.

Die modellierten Daten fallen in drei Hauptkategorien: Aggregierte Daten (die Zusammenfassungstabellen), aktueller Zustand von Objekten (Maschinen, Sitzungen usw.) und Protokolldaten, wobei es sich um historische Ereignisse handelt (z. B. Verbindungen).

Hinweis: Enumerationen werden vom OData-Protokoll nicht unterstützt, stattdessen werden ganze Zahlen verwendet. Weitere Informationen zum Bestimmen der von der Überwachungsdienst-OData-API zurückgegebenen Werte finden Sie unter <http://support.citrix.com/help/monitorserviceapi/7.6/>.

## Aggregation der Datenwerte

Der Überwachungsdienst erfasst diverse Daten über Benutzersitzungsnutzung, Benutzeranmeldeleistung, Sitzungslastausgleich und zu Fehlern bei Verbindungen und Maschinen. Die Daten werden je nach Kategorie unterschiedlich aggregiert. Zum Interpretieren der Daten sind Kenntnisse über die Aggregation der mit den OData-Methoden-APIs abgerufenen Datenwerte unverzichtbar. Beispiel:

- Fehler bei verbundenen Sitzungen und Maschinen treten in einem bestimmten Zeitraum auf und werden daher als Höchstwerte in einem Zeitraum verfügbar gemacht.
- Die Anmeldedauer ist ein Zeitlängenwert und wird daher als Durchschnitt in einem Zeitraum verfügbar gemacht.
- Die Anzahl der Anmeldungen und Verbindungsfehler repräsentieren eine Anzahl von Vorkommen in einem bestimmten Zeitraum und werden als Summen in einem Zeitraum gemacht.

## Auswertung gleichzeitiger Daten

Sitzungen müssen sich überschneiden, um als gleichzeitig angesehen zu werden. Bei einem Zeitintervall von 1 Minute werden jedoch alle Sitzungen in dieser Minute (mit oder ohne Überschneidung) als gleichzeitig angesehen, d. h. das Intervall ist so klein, dass sich der Mehraufwand für die Berechnung der Genauigkeit nicht lohnt. Finden die Sitzungen in der gleichen Stunde, aber nicht in der gleichen Minute statt, werden sie als einander nicht überschneidend angesehen.

## Korrelation der Zusammenfassungstabellen mit Rohdaten

Das Datenmodell stellt Metriken auf zwei verschiedene Arten dar:

- Die Zusammenfassungstabellen zeigen aggregierte Ansichten der Metriken in Granularitäten pro Minute, Stunde und Tag an.
- Die Rohdaten stehen für einzelne Ereignisse oder den aktuellen Zustand, der bzw. die für eine Sitzung, Verbindung, Anwendung und andere Objekte protokolliert werden.

Wenn Sie versuchen, Daten über API-Aufrufe hinweg oder innerhalb des Datenmodells selbst zu korrelieren, sollten Sie die folgenden Konzepte und Einschränkungen kennen:

- **Keine Zusammenfassungsdaten für Teilintervalle:** Die Zusammenfassungen von Metriken erfüllen die Anforderungen von historischen Trends über lange Zeiträume hinweg. Diese Metriken werden für vollständige Intervalle in der Zusammenfassungstabelle aggregiert. Für Teilintervalle am Anfang (die ältesten verfügbaren Daten) und am Ende der Datensammlung gibt es keine Zusammenfassungsdaten. Beim Anzeigen der Aggregation eines Tages (Intervall=1440) bedeutet dies, dass der erste Tag und der aktuelle unvollständige Tag keine Daten aufweisen. Obwohl für diese Teilintervalle u. U. Rohdaten vorhanden sind, werden sie nie zusammengefasst. Sie können das früheste und letzte Aggregationsintervall für eine bestimmte Datengranularität festlegen, indem Sie die Mindest- und Höchstwerte für "SummaryDate" aus einer bestimmten Zusammenfassungstabelle nehmen. Die Spalte "SummaryDate" stellt den Start des Intervalls dar. Die Spalte "Granularity" steht für die Länge des Intervalls der aggregierten Daten.
- **Korrelation nach Zeit:** Metriken werden, wie oben beschrieben, für vollständige Intervalle in der Zusammenfassungstabelle aggregiert. Sie können für historische Trends verwendet werden, aber rohe Ereignisdaten stellen möglicherweise einen aktuelleren Zustand dar als die Zusammenfassung für die Trendanalyse. Bei zeitbasierten Vergleichen zwischen der Zusammenfassung und den Rohdaten muss beachtet werden, dass es keine Zusammenfassungsdaten für Teilintervalle gibt, die am Anfang und Ende des Zeitraums auftreten.
- **Verpasste und latente Ereignisse:** Wenn Ereignisse verpasst werden oder während des Aggregationszeitraums latent sind, sind die für die Zusammenfassungstabelle aggregierten Metriken möglicherweise ungenau. Obwohl der Überwachungsdienst versucht, einen genauen aktuellen Zustand zu erhalten, wird die Aggregation für verpasste oder latente Ereignisse nicht im Nachhinein neu für die Zusammenfassungstabellen berechnet.
- **Hochverfügbare Verbindungen:** Bei hoher Verfügbarkeit von Verbindungen entstehen in den Zusammenfassungsdaten für aktuelle Verbindungen Lücken, aber die Sitzungsinstanzen werden dennoch in den Rohdaten ausgeführt.
- **Beibehaltungszeitraum für Daten:** Daten werden in den Zusammenfassungstabellen basierend auf einem anderen Bereinigungszeitplan beibehalten als Rohdaten von Ereignissen. Daten fehlen möglicherweise, weil die Zusammenfassungstabellen oder die unformatierten Tabellen bereinigt wurde. Beibehaltungszeiträume können



unterschiedliche Granularitäten für Zusammenfassungsdaten aufweisen. Daten basierend auf niedrigerer Granularität (Minuten) werden schneller bereinigt als Daten, die auf höherer Granularität (Tage) basieren. Wenn Daten bereinigt wurden und in einer Granularitätskategorie fehlen, sind sie möglicherweise in einer höheren Granularitätskategorie. API-Aufrufe geben nur Daten für die angeforderte Granularität zurück. Wenn für eine Granularität keine Daten zurückgegeben werden, sind möglicherweise für den gleichen Zeitraum Daten für eine höhere Granularität vorhanden.

- **Zeitzonen:** Metriken werden mit UTC-Zeitstempeln gespeichert. Zusammenfassungstabellen werden basierend auf stündlichen Zeitzonengrenzen aggregiert. Bei Zeitzonen, die nicht in diese stündlichen Grenzen fallen, gibt es möglicherweise Unstimmigkeiten beim Ort der Datenaggregation.

## Datengranularität und -beibehaltung

Die Granularität der aggregierten Daten, die von Director abgerufen werden, ist eine Funktion des angeforderten Zeitraums (T). Folgende Regeln gelten:

- $0 < T \leq 1$  Stunde: minutengenaue Granularität wird verwendet
- $0 \leq T \leq 30$  Tage: stundengenaue Granularität wird verwendet
- $T > 31$  Tage: tagesgenaue Granularität wird verwendet

Angeforderte Daten, die nicht von aggregierten Daten stammen, stammen von den rohen Sitzungs- und Verbindungsinformationen. Diese Menge dieser Daten nimmt schnell zu, daher haben sie eine eigene Bereinigungseinstellung. Bereinigung gewährleistet, dass nur relevante Daten langfristig gespeichert werden. Damit wird eine bessere Leistung sichergestellt, während die für die Berichterstellung erforderliche Granularität beibehalten werden kann. Bei Kunden, die keine Lizenz für die Platinum Edition haben, beginnt die Bereinigung am 8. Tag, unabhängig vom Standardbereinigungszeitraum. Platinum-Kunden können den Beibehaltungszeitraum auf die gewünschte Anzahl an Tagen einstellen, sonst wird der Standardwert verwendet.

Mit den folgenden Einstellungen wird die Bereinigung gesteuert:

Einstellungsname	Betroffene Bereinigung	Standardwert (Tage)	Zugriff mit
GroomSessionsRetentionDays	Sitzungs- und Sitzungsdetaileinträge	7 für Nicht-Platinum-Benutzer, 90 für Platinum	Cmdlet (set/get-monitorconfiguration)
GroomSummariesRetentionDays	Einträge für DesktopGroupSummary, FailureLogSummary und LoadIndexSummary. Aggregierte Daten, tägliche Granularität	7 für Nicht-Platinum-Benutzer, 90 für Platinum	Cmdlet (set/get-monitorconfiguration)
GroomHourlyRetentionDays	Aggregierte Daten - stundengenaue Granularität	32 Tage	Monitor.Configuration Database Table. Siehe Hinweis unten.
GroomMinuteRetentionDays	Aggregierte Daten - minutengenaue Granularität	3 Tage	Monitor.Configuration Database Table. Siehe Hinweis unten.



GroomFailuresRetentionDays Einstellungsname	Betroffene Bereinigung MachineFailureLog und ConnectionFailureLog	Standardwert (Page) 7 für Nicht- Platinum- Benutzer, 90 für Platinum	Zugriff mit Cmdlet (set/get- monitorconfiguration)
GroomLoadIndexesRetentionDays	Einträge für LoadIndex	7 für Nicht- Platinum- Benutzer, 90 für Platinum	Cmdlet (set/get- monitorconfiguration)
GroomDeletedRetentionDays	Maschinen-, Katalog-, Desktopgruppen- und Hypervisorentitäten, die einen LifecycleState von "Deleted" haben. Dadurch werden auch zugehörige Einträge für Sitzung, Sitzungsdetail, Zusammenfassung, Fehler oder LoadIndex gelöscht.	7 für Nicht- Platinum- Benutzer, 90 für Platinum	Cmdlet (set/get- monitorconfiguration)
GroomMachineHotfixHistoryRetentionDays	Auf VDA und Controllermaschinen angewendete Hotfixes	90 für Nicht- Platinum- und Platinum- Benutzer	Cmdlet (set/get- monitorconfiguration)

Achtung: Nach dem Ändern von Werten auf der Überwachungsdienstdatenbank ist ein Neustart des Diensts erforderlich, damit die neuen Werte wirksam werden. Führen Sie Änderungen an der Überwachungsdienstdatenbank nur mit Anleitung vom Citrix Support durch.

Das Beibehalten von Daten über lange Zeiträume hinweg hat die folgenden Auswirkungen auf die Größe von Tabellen:

- **Stundengenaue Daten:** Wenn Sie stundengenaue Daten bis zu zwei Jahre lang in der Datenbank speichern, wächst die Datenbank einer Site mit 1000 Bereitstellungsgruppen ungefähr wie folgt an:  
 $1000 \text{ Bereitstellungsgruppen} \times 24 \text{ Stunden/Tag} \times 365 \text{ Tage/Jahr} \times 2 \text{ Jahre} = 17.520.000 \text{ Datenreihen}$ . Diese große Datenmenge in den Aggregationstabellen hat beträchtliche Auswirkungen auf die Leistung. Wenn man bedenkt, dass die Dashboarddaten aus dieser Tabelle gezogen werden, sind die Anforderungen an den Datenbankserver möglicherweise riesig. Übermäßig viele Daten können dramatische Auswirkungen auf die Leistung haben.
- **Sitzungs- und Ereignisdaten:** Diese Daten werden jedes Mal gesammelt, wenn eine Sitzung gestartet und eine Verbindung/Wiederverbindung hergestellt wird. Bei einer großen Site (100.000 Benutzer) nimmt die Menge dieser Daten sehr schnell zu. Beispielsweise entsprechen die über zwei Jahre gespeicherten Tabellen mehr als 1 TB Daten und erfordern eine High-End-Unternehmensdatenbank.

# Schützen von Endpunkten mit SSL

May 10, 2016

In diesem Dokument wird erläutert, wie Sie die Endpunkte der Überwachungsdienst-OData-API mit SSL schützen. Wenn Sie SSL verwenden, müssen Sie SSL auf allen Delivery Controllern in der Site konfigurieren. Sie können keine Mischung aus Controllern mit und ohne SSL verwenden.

Zum Schützen der Endpunkte des Überwachungsdiensts mit SSL müssen Sie die folgende Konfiguration vornehmen. Einige Schritte müssen nur einmal pro Site vorgenommen werden, andere hingegen auf jeder Maschine, auf der der Überwachungsdienst gehostet wird. Die Schritte werden nachfolgend beschrieben.

## Teil 1: Zertifikatregistrierung im System

1. Erstellen Sie mit einem vertrauenswürdigen Zertifikat-Manager ein Zertifikat. Das Zertifikat muss dem Port auf der Maschine zugeordnet werden, den Sie für OData SSL verwenden möchten.
2. Konfigurieren Sie den Überwachungsdienst zur Verwendung dieses Ports für SSL-Kommunikation. Die Schritte hängen von der jeweiligen Umgebung ab und davon, wie diese mit Zertifikaten funktioniert. Das folgende Beispiel zeigt das Konfigurieren von Port 449:
  - Ordnen Sie das Zertifikat einem Port zu:  
`netsh http add sslcert ipport=0.0.0.0:449 certhash=97bb629e50d556c80528f4991721ad4f28fb74e9appid='{00000000-0000-0000-0000-000000000000}'`  
Tipp: Stellen Sie in einem PowerShell-Befehlsfenster sicher, dass Sie die GUID in der Anwendungs-ID mit einfachen Anführungszeichen wie oben dargestellt einschließen, da der Befehl sonst nicht funktioniert. Beachten Sie, dass diesem Beispiel eine Zeile für bessere Lesbarkeit hinzugefügt wurde.

## Teil 2: Ändern der Konfigurationseinstellungen des Überwachungsdiensts

1. Führen Sie auf jedem Delivery Controller in der Site folgende PowerShell-Befehle einmal aus. Damit wird die Registrierung des Überwachungsdiensts beim Konfigurationsdienst aufgehoben.  
`asnp citrix.* $ServiceGroup = get-configregisteredServiceInstance -servicetype Monitor | Select -First 1 ServiceGroupUid remove-configServiceGroup -ServiceGroupUid $ServiceGroup.ServiceGroupUic`
2. Führen Sie den folgenden Schritt auf allen Controllern in der Site aus:
  - Verwenden Sie eine Eingabeaufforderung zum Suchen des Verzeichnisses des installierten Citrix Überwachungsdiensts (in der Regel C:\Programme\Citrix\Monitor\Service). Führen Sie in diesem Verzeichnis Folgendes aus:  
`Citrix.Monitor.Exe -CONFIGUREFIREWALL -ODATAPort 449 -RequireODATAssl`
  - Führen Sie die folgenden PowerShell-Befehle aus:  
`asnp citrix.* (falls dies noch nicht in diesem Fenster ausgeführt wird) get-MonitorServiceInstance | register-ConfigServiceInstance Get-ConfigRegisteredServiceInstance -ServiceType Config | Reset-M`

# Beispiele

May 10, 2016

Die folgenden Beispiele zeigen, wie Daten mit der Überwachungsdienst-OData-API exportiert werden.

## Beispiel 1 – unformatiertes XML

1. Geben Sie die URL für jede Datengruppe in einen Webbrowser ein, der mit den für die XenApp- bzw. XenDesktop-Site erforderlichen Administratorrechten ausgeführt wird. Citrix empfiehlt die Verwendung des Chrome-Browsers mit dem Add-In Advanced Rest Client.
2. Zeigen Sie die Quelle an.

## Beispiel 2 – PowerPivot für Excel

Bei diesen Anweisungen wird davon ausgegangen, dass Sie Microsoft Excel und PowerPivot bereits installiert haben.

Öffnen Sie Excel mit den für die XenApp- bzw. XenDesktop-Site erforderlichen Administratorrechten.

Bei Verwendung von Excel 2010:

1. Klicken Sie auf der Registerkarte "PowerPivot".
2. Klicken Sie auf "PowerPivot-Fenster".
3. Klicken Sie im Menüband auf **Aus Datenfeeds**.
4. Wählen Sie einen Anzeigenamen für die Verbindung aus (z. B. XenDesktop-Überwachungsdaten) und geben Sie die Datenfeed-URL ein: `http://{dc-host}/Citrix/Monitor/OData/v1/Data` (oder "https:", falls Sie SSL verwenden).
5. Klicken Sie auf **Weiter**.
6. Wählen Sie die Tabellen aus, die Sie in Excel importieren möchten, und klicken Sie auf **Fertig stellen**. Die Daten werden abgerufen.
7. Sie können die Daten jetzt über PowerPivot mit PivotTables und PivotCharts anzeigen und analysieren. Weitere Informationen finden Sie im Learning Center: <http://www.microsoft.com/en-us/bi/LearningCenter.aspx>.

Bei Verwendung von Excel 2013:

1. Klicken Sie auf die Registerkarte "Daten".
2. Wählen Sie "Aus anderen Quellen > Aus OData-Datenfeed".
3. Geben Sie die URL des Datenfeeds ein: `http://{dc-host}/Citrix/Monitor/OData/v1/Data` (oder "https:", wenn Sie SSL verwenden) und klicken Sie auf **Weiter**.
4. Wählen Sie die Tabellen aus, die Sie in Excel importieren möchten, und klicken Sie auf **Weiter**.
5. Akzeptieren Sie die Standardnamen oder passen Sie Namen an und klicken Sie auf **Fertig stellen**.
6. Wählen Sie **Nur Verbindung** oder **Pivot Report**. Die Daten werden abgerufen.
7. Sie können die Daten jetzt über PowerPivot mit PivotTables und PivotCharts anzeigen und analysieren. Weitere Informationen finden Sie im Learning Center: <http://www.microsoft.com/en-us/bi/LearningCenter.aspx>.

## Beispiel 3 – LINQPad

Bei diesen Anweisungen wird davon ausgegangen, dass Sie LINQPad bereits installiert haben.

1. Führen Sie LinqPad mit den für die XenApp- bzw. XenDesktop-Site erforderlichen Administratorrechten aus.  
Tipp: Die einfachste Methode ist Download, Installation und Ausführung auf dem Delivery Controller.
2. Klicken Sie auf den Link "Add connection".

3. Wählen Sie "WCF Data Services 5.1 (OData 3)" und klicken Sie auf **Weiter**.
4. Geben Sie die URL des Datenfeeds ein: `http://{dc-host}/Citrix/Monitor/OData/v2/Data` (oder "https:", wenn Sie SSL verwenden). Falls erforderlich, geben Sie den Benutzernamen und das Kennwort für den Zugriff auf den Delivery Controller ein. Klicken Sie auf **OK**.
5. Sie können jetzt LINQ-Abfragen für den Datenfeed ausführen und die Daten nach Bedarf exportieren. Beispiel: Klicken Sie mit der rechten Maustaste auf "Catalogs" und wählen Sie **Catalogs.Take(100)** aus. Damit werden die ersten 100 Kataloge in der Datenbank zurückgegeben. Wählen Sie "Export > Export to Excel with formatting" aus.

Weitere funktionierende Beispiele für die Verwendung der API mit LINQPad finden Sie unter <http://blogs.citrix.com/2014/01/14/creating-director-custom-reports-for-monitoring-xendesktop/>.

# XenApp- und XenDesktop-SDK

May 10, 2016

Für XenApp und XenDesktop gibt es ein SDK, das auf zahlreichen Microsoft Windows PowerShell Version 3.0-Snap-Ins basiert, mit dem Sie die gleichen Tasks wie mit der Citrix Studio-Konsole und andere Tasks ausführen können, die in Studio allein nicht möglich sind.

Ab Version 7.5 bieten XenApp und XenDesktop mit FlexCast Management Architecture eine einheitliche Architektur und Verwaltung. Daher stehen nun in XenApp viele Features bereit, die es zuvor nur in XenDesktop gab. Elemente des SDKs für gemeinsame Features gelten also nun sowohl für XenApp als auch für XenDesktop, selbst wenn die Befehle sich nur auf XenDesktop beziehen.

- **Neues SDK auf höherer Ebene:** XenDesktop 7 stellt ein neues SDK auf höherer Ebene bereit, mit dem Sie schnell und mühelos ein Skript erstellen und die Siteerstellung und -wartung automatisieren. Das SDK auf höherer Ebene isoliert Sie von der Komplexität der SDKs auf niedrigerer Ebene, sodass Sie für das Erstellen einer neuen Site nur zwei Cmdlets ausführen müssen.
- **Neue SDKs auf niedriger Ebene:** Einzelne SDKs auf niedriger Ebene werden für die neuen XenDesktop 7-Dienste bereitgestellt, u. a. ein dediziertes und verbessertes SDK für den Dienst für die delegierte Administration (DAS), das vorher Teil des Broker-SDKs in XenDesktop 5 war. Es gibt auch SDKs für neue Features, u. a. Überwachungsdienst, Umgebungstest und Konfigurationsprotokollierung.
- **Kataloge und Bereitstellungsgruppen für Windows-Serverbetriebssystemmaschine:** Mit dem XenDesktop 7-SDK stellen Sie kostengünstig Anwendungen und Desktops bereit, die auf Serverbetriebssystemen gehostet werden.
- **Anwendungen für Desktopbetriebssystemmaschinen:** Anwendungen für Desktopbetriebssystemmaschinen haben sich auf der SDK-Ebene wesentlich geändert. Wenn Sie vorhandene Skripts für die Ausführung von Anwendungen auf Desktopbetriebssystemen haben, müssen Sie diese Skripts für XenDesktop 7 aktualisieren, da es fast keine Rückwärtskompatibilität gibt.
- **Anwenden von Einstellungen auf Maschinen in Bereitstellungsgruppen:** In XenDesktop 7 können Sie mit Konfigurationsslots Einstellungen auf Maschinen in einer bestimmten Bereitstellungsgruppe statt auf alle Maschinen in einer Site anwenden. So können Sie für eine bestimmte Bereitstellungsgruppe konfigurieren, welche Einstellungen für diese Gruppe gelten. Zahlreiche vordefinierte Konfigurationsslots werden bereitgestellt, die unterschiedliche Typen von Einstellungen enthalten, u. a. Einstellungen für StoreFront-Adressen, die mit Speicherorten von Receiver- oder App-V-Veröffentlichungserver verwendet werden. Eine Sammlung von Einstellungen kann sich nur auf eine bestimmte Bereitstellungsgruppe auswirken, und eine andere Sammlung von Einstellungen von dem gleichen Slot wirkt sich auf eine andere Bereitstellungsgruppe aus. Sie können Namen verwenden, die für Ihre Bereitstellung geeignet sind, z. B. "Vertriebsabteilungsrichtlinie".
- **Ersetzte Katalogtypen:** In XenDesktop 7 wurden Katalogtypen durch Kataloge mit individuellen Eigenschaften ersetzt. Für die Rückwärtskompatibilität können Sie weiterhin vorhandene Skripts nutzen, die Katalogtypen verwenden, z. B. ein Image (gepoolt) und Thin Clone (dediziert) usw., intern werden diese jedoch in Gruppen von Eigenschaften konvertiert.  
Achtung: Die Rückwärtskompatibilität mit XenDesktop 5-Katalogtypen wurde beibehalten, wenn es möglich und praktikabel war. Verwenden Sie beim Schreiben neuer Skripte jedoch keine Katalogtypen sondern geben Sie Kataloge mit individuellen Eigenschaften an.
- **Ersetztes Desktopobjekte:** In XenDesktop 5 ist das Desktopobjekt einer der Haupttypen des SDK-Objekts, das in Broker-SDK-Skripts verwendet wird. Das Desktopobjekt beschreibt sowohl die Maschine als auch die Sitzung auf der

Maschine. In XenDesktop 7 wird dieses Objekt durch das Sitzungsobjekt und das Maschinenobjekt ersetzt, die beide erweitert wurden und jetzt die Funktion des Desktopobjekts übernehmen. Für die Rückwärtskompatibilität können Sie jedoch weiterhin vorhandene Skripts verwenden, die das Desktopobjekt nutzen.

Achtung: Die Rückwärtskompatibilität mit XenDesktop 5 wurde beibehalten, wenn es möglich und praktikabel war. Verwenden Sie beim Schreiben neuer Skripte jedoch nicht das Desktopobjekt sondern geben Sie Sitzungs- und Maschinenobjekte an.

Die Richtlinienregeln im SDK und der Studio-Konsole sind nicht identisch. Anspruchs- und Zuordnungsrichtlinienregeln sind unabhängige Entitäten im SDK, und in der Konsole. Diese Entitäten sind nicht sichtbar, da sie nahtlos mit der Bereitstellungsgruppe zusammengeführt werden. Weiterhin sind Richtlinienregeln im SDK weniger restriktiv.

Das SDK enthält mehrere PowerShell-Snap-Ins, die automatisch vom Installationsassistenten installiert werden, wenn Sie den Controller oder Studio installieren.

So greifen Sie auf die Cmdlets zu:

1. Starten Sie eine Shell in PowerShell 3.0.

Klicken Sie zum Starten einer Shell über die Konsole auf **Studio**, wählen Sie die Registerkarte "PowerShell" aus und klicken Sie auf **PowerShell starten**.

Sie müssen die Shell oder das Skript mit einer ID ausführen, die über Citrix Administratorrechte verfügt. Obwohl die Mitglieder der lokalen Administratorgruppe auf dem Controller automatisch über Volladministratorprivilegien verfügen, um XenDesktop zu installieren, empfiehlt Citrix, dass Sie für den normalen Betrieb Citrix Administratoren mit den entsprechenden Rechten erstellen und nicht das lokale Administratorkonto verwenden. Wenn Sie Windows Server 2008 ausführen, müssen Sie die Shell oder das Skript als Citrix Administrator und nicht als Mitglied der lokalen Administratorgruppe ausführen.

2. Legen Sie die Ausführungsrichtlinie in PowerShell fest, um SDK-Cmdlets in Skripten zu verwenden.  
Weitere Informationen zur PowerShell-Ausführungsrichtlinie finden Sie in der Dokumentation von Microsoft.
3. Fügen Sie mit dem Befehl **Add -PSSnapin** in der Windows PowerShell-Konsole die Snap-Ins hinzu, die Sie in der PowerShell-Umgebung benötigen. V1 und V2 geben die Version des Snap-Ins an (XenDesktop 5-Snap-Ins sind Version 1; XenDesktop 7-Snap-Ins sind Version 2). Geben Sie zum Beispiel Folgendes ein:

**Add-PSSnapin Citrix.ADIdentity.Admin.V2**

Geben Sie zum Importieren aller Cmdlets Folgendes ein:

**Add-PSSnapin Citrix.\*.Admin.V\***

Nach dem Import haben Sie Zugriff auf die Cmdlets und die zugehörige Hilfe.

Ein Beispiel eines typischen Anwendungsfalls finden Sie unter [Erste Schritte mit dem SDK](#).

Tipp: Eine vollständige Liste der gesamten Hilfe für die Cmdlets finden Sie unter [Hilfe zu PowerShell-Cmdlets](#).

Mit dem Citrix Group Policy SDK können Sie Einstellungen und Filter für Gruppenrichtlinien anzeigen und konfigurieren. Es verwendet einen PowerShell-Anbieter, um einen virtuellen Datenträger zu erstellen, der mit den Maschinen- und Benutzereinstellungen und -filtern übereinstimmt. Der Anbieter wird als Erweiterung zu New-PSDrive angezeigt. Für die

Verwendung des Group Policy SDKs muss Studio oder das XenApp- bzw. XenDesktop-SDK installiert sein.

### Hinzufügen des Group Policy SDKs

1. Zum Hinzufügen des Gruppenrichtlinien-SDKs geben Sie Folgendes ein:

`Add-PSSnapin citrix.common.grouppolicy`

2. Geben Sie Folgendes ein, um auf die Hilfe zuzugreifen:

`help New-PSDrive -path localgpo:/`

### Verwenden des Group Policy SDKs

1. Zum Erstellen eines virtuellen Laufwerks und zum Laden der Einstellungen geben Sie Folgendes ein:

`New-PSDrive [-PSProvider] CitrixGroupPolicy`

*— -Controller*

**New-PSDrive [-PSProvider] CitrixGroupPolicy**

*— -Controller*

wobei

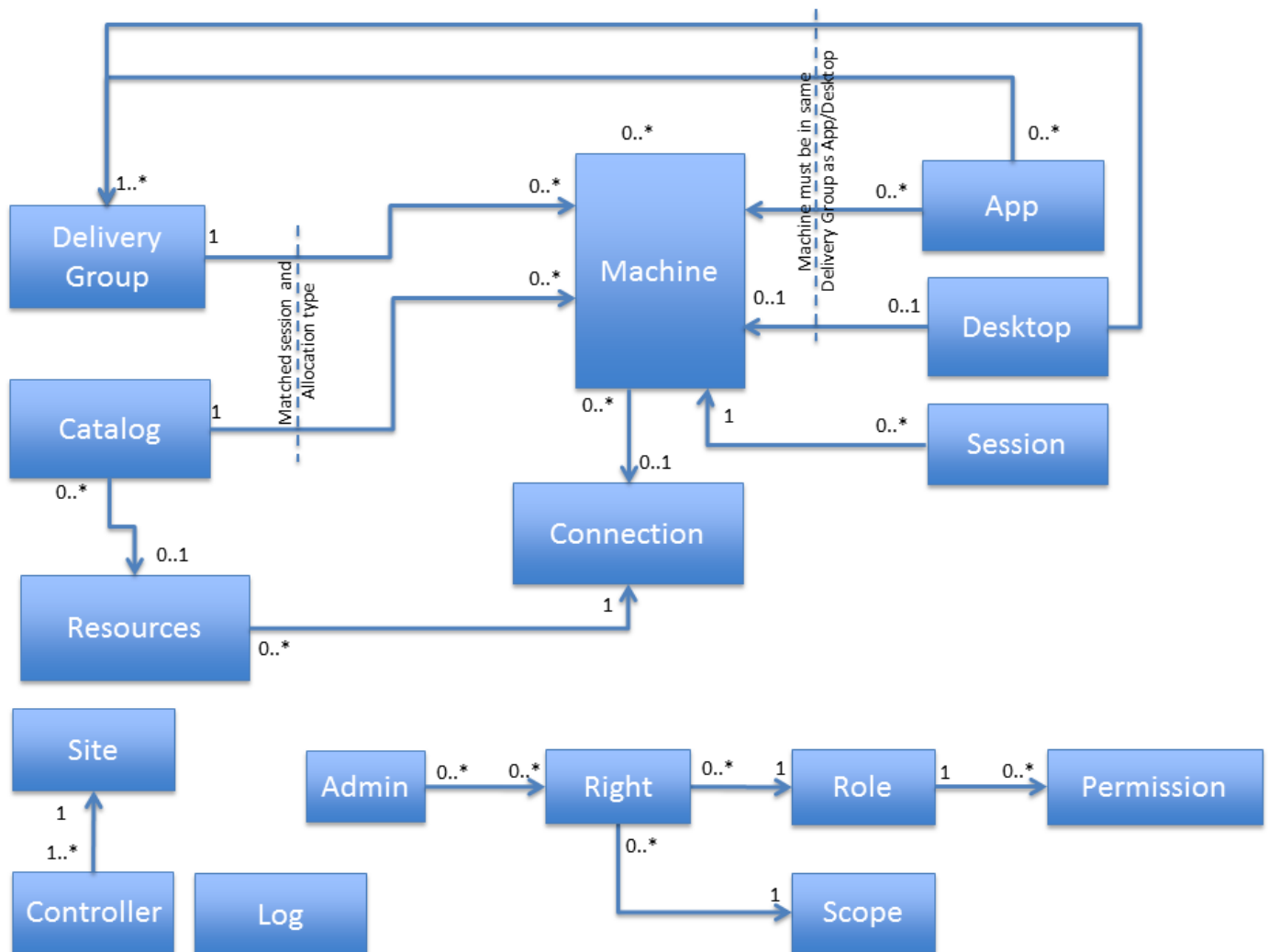
*— -Controller*

der vollqualifizierte Domänenname eines Controllers in der Site ist, mit der Sie eine Verbindung herstellen und aus der Sie Einstellungen laden möchten.

# Grundlegendes zum XenDesktop-Verwaltungsmodell

May 10, 2016

Das Verwaltungsmodell von XenDesktop 7 ist durch Objekte und deren Verbindungen untereinander definiert, wie in der folgenden Abbildung dargestellt.



In den folgenden Abschnitten werden die Objekte, ihre Zuständigkeiten, Merkmale und grundlegenden Eigenschaften beschrieben.

## Verbindung (Connection)

Eine Verbindungsentität bietet die erforderlichen Informationen zum Erstellen einer Verbindung mit dem Verwaltungspunkt einer Virtualisierungsplattform. Mit diesen Informationen werden die erforderlichen Energieverwaltungs- und Provisioning-Funktionen bereitgestellt. Die Verbindung definiert Folgendes:

- Identität der Verbindung, die u. a. einen Namen und einen internen ID-Wert (GUID) umfasst.
- Typ: XenServer, SCVMM oder vCenter.
- Benutzername: der Kontoname für Verbindungen mit dem Hypervisor.
- Kennwort: das Kennwort für Verbindungen mit dem Hypervisor.
- Zustand der Verbindung und ob mit dem Hypervisor kommuniziert werden kann.
- Alle vom Hypervisor ausgehenden aktuellen Warnungen.



Das Kennwort kann nicht zurückgelesen werden, wenn es einmal festgelegt und verschlüsselt in der Datenbank gespeichert ist.

Im Modell gibt es eine dieser Entitäten für jede Plattformdefinition. Zur Laufzeit werden jedoch viele Verbindungen mit der Virtualisierungsplattform hergestellt, um die Anforderungen der XenDesktop-Bereitstellung zu bedienen. Diese Verbindungen können von vielen Ressourcenentitäten verwendet werden (weitere Informationen siehe unten).

Hinweis: Beim Einrichten einer Verbindung mit XenServer geben Sie einen Host oder XenServer-Pool an. Beim Einrichten einer Verbindung mit Microsoft Hyper-V oder VMware geben Sie Informationen für die Verwaltungsserver an (SCVMM oder vCenter). Außerdem müssen Sie beim Konfigurieren einer Verbindung den Benutzernamen im richtigen Format für die verwendete Virtualisierungslösung angeben. Wenn Sie eine Verbindung mit XenServer einrichten, geben Sie einen einfachen Benutzernamen an, z. B. "Benutzername". Für eine Verbindung mit VMware oder Hyper-V müssen Sie einen Domänenbenutzernamen, z. B. "Domäne\Benutzername", angeben.

## Ressourcen

Hinweis: Dies gilt nur für die Erstellung neuer Maschinen mit Maschinenerstellungsdienste (MCS) oder Provisioning Services (PVS).

Mit dieser Einstellung wird ein Satz Ressourcen in der Virtualisierungsplattform definiert, die das Produkt verbrauchen kann. Dadurch hat der Administrator die Möglichkeit, die Bereitstellung auf einen Teil der von der Plattform bereitgestellten Ressourcen zu beschränken und verschiedene Ressourcensätze für unterschiedliche Zwecke einzurichten. Hiermit werden dann die Speicher- und Netzwerknutzung für in der Site bereitgestellte Maschinen gesteuert. Die Ressourcenentität definiert Folgendes:

- **Netzwerke:** Die Netzwerke, die zum Provisioning von neuen Maschinen in der Virtualisierungsplattform verfügbar sind. (Für PVS ist dies das Streamingnetzwerk. Nur eins kann ausgewählt werden oder das erste wird verwendet.)
- **Speicher:** Der Speicher, der zum Provisioning von neuen Maschinen in der Virtualisierungsplattform verfügbar ist.

Mindestens eine Ressourceneinstellung muss für jedes Verbindungsobjekt definiert werden, wenn eine Bereitstellungsmethode verwendet werden soll. Mit einer einzelnen Verbindung können viele Ressourceneinstellungen verknüpft werden.

Die Verbindungsdetails werden von der Verbindungsentität bereitgestellt.

Hinweis: Beim Festlegen von Ressourcen für eine Verbindung mit SCVMM oder vCenter müssen Sie den Host oder Cluster sowie den Speicher und das Netzwerk auswählen.

## Administrator

Hiermit wird die Person definiert, die das Produkt verwaltet. Es kann verschiedene Administratoren für das Produkt geben, wobei jeder verschiedene Funktionen im Hinblick auf das Produkt hat. Der Administrator definiert Folgendes:

- Das Active Directory-Konto, über das der Administrator definiert wird, definiert als sein Name und SID-Wert. Dies kann ein einzelner Benutzer oder eine Sicherheitsgruppe sein.
- Die Rechte, die der Administrator hat (d. h. welche Rollen für welche Geltungsbereiche verfügbar sind).

## Maschine

Eine Maschine ist eine Repräsentation eines virtuellen oder physischen Computers, mit der innerhalb einer Site Sitzungen für Benutzer bereitgestellt werden können. Die Maschine definiert Folgendes:

- Identität der Maschine, z. B. SAM-Name, DNS-Name, Hostname (Hypervisor), IP-Adresse, SID, Lizenz-ID, usw.
  - Status der Maschine, einschließlich Energiezustand, Registrierungszustand, Zustand von Personal vDisk, Lastindex, usw.
- Viele dieser Werte werden in einen einzigen Zustandswert aggregiert, dem Zusammenfassungszustand.

- Informationen zur Umgebung und Konfiguration der Maschine, z. B. die Versionsnummern der installierten Betriebssysteme und Citrix Komponenten, einschließlich der Funktionsebene der Maschine.
- Daten zu den aktuellen Aktivitäten der Maschinen, z. B. die Ursache für die letzte Aufhebung der Registrierung, die letzte auf der Maschine ausgeführte Energieaktion, der letzte Verbindungsfehler, usw.
- Wartungsmodus und die Zustände für "WindowsConnectionSetting" zum Steuern des Verhaltens der Maschine beim Aktivieren/Deaktivieren/Entlasten.
- Sichtbare Benutzerressourceneinstellungen für die Maschine für den zugewiesenen Desktopfall, z. B. Symbol, veröffentlichte Namen, usw.
- Bei mehreren Sitzungsmaschinen (Serverbetriebssystemmaschinen) werden Informationen über die ausgeführten Sitzungen zusammengefasst, z. B. die Anzahl der aktiven oder ausstehenden Sitzungen, usw.
- Einer zugewiesenen Maschine zugeordnete Werte für Tags.
- Provisioninginformationen zum Image, wie der Pfad zum Masterimage, das zum Erstellen der Maschine verwendete Provisioningschema und ob für die Maschine ein aktualisiertes Image bis zum nächsten Neustart aussteht.

Die meisten Maschinenwerte werden vom Broker Service SDK verfügbar gemacht, aber Elemente werden auch von anderen Diensten, z. B. den Maschinenerstellungsdiensten, angegeben. Andere Werte, die pro Maschine angezeigt werden, wurden vom Katalog oder von der Bereitstellungsgruppe (falls vorhanden) geerbt, denen die Maschine zugeordnet ist, oder vom Provisioningschema, das für die Erstellung verwendet wurde. Ein Beispiel ist der Maschinentyp, der auf Katalogebene durch eine Kombination von Faktoren definiert wird, z. B. physisch oder virtuell, statisch oder zufällig, wie Benutzeränderungen beibehalten werden, und andere. Bei Maschinen mit Einzelsitzungen (Desktopbetriebssystemmaschinen) sind auch die Informationen über die Sitzung (falls vorhanden), die auf der Maschine ausgeführt wird, der Maschine zugeordnet. Dazu gehören u. a. die Identität verbundener Benutzer, der Sitzungszustand, das verwendete Protokoll, usw. Maschinen können im Modell nur bestehen, wenn sie als Teil eines Katalogs definiert sind. Außerhalb dieses Konzepts können sie nicht bestehen.

## Katalog

Ein Katalog definiert eine Gruppe von Maschinen, die normalerweise äquivalent sind, aber nicht immer. Eine Bereitstellung kann mehrere Kataloge enthalten, sodass verschiedene Maschinengruppen erstellt und für unterschiedliche Zwecke gespeichert werden können. Ein Katalog definiert Folgendes:

- Katalogtyp, der durch die Werte der folgenden Eigenschaften definiert wird:
  1. Bereitstellungsmethode für die Maschinen (Maschinenerstellungsdienste, PVS oder manuell)
  2. Wie die Maschinen im Katalog Benutzern zugeteilt werden: statisch und Benutzern permanent zugewiesen oder jedes Mal nach dem Zufallsprinzip, wenn ein Benutzer eine Ressource anfordert
  3. Ob es sich um Maschinen mit Einzelsitzung (Desktopbetriebssystemmaschinen) oder mehreren Sitzungen (Serverbetriebssystemmaschinen) handelt
  4. Ob es sich um physische oder virtuelle Maschinen handelt
  5. Handhabung von Benutzeränderungen an Maschinen: ob sie nach der Abmeldung des Benutzers verworfen werden oder lokal auf der Maschine oder mit Personal vDisk beibehalten werden
- Ob der Katalog für Remote-PC-Benutzer verwendet wird und wenn dies der Fall ist, die dem Katalog zuzuordnenden Active Directory-Organisationseinheiten und Bereitstellungsgruppen.
- Details für das Provisioning mit Maschinenerstellungsdienste (wenn MCS verwendet wird):
  1. Masterimage für die Maschinen
  2. Speichergröße und Anzahl der CPUs
  3. Datenträgergröße, Laufwerksbuchstabe und Speicherbelegung für Personal vDisk
  4. Active Directory-Kontobenennungsschema und -Organisationseinheiten für die erstellten Maschinen und eine Liste der bereits erstellten Konten

- Details der PVS-Server, die dem Katalog zugeordnet sind
- Die für die Maschinen erwartete Funktionsebene; Maschinen niedrigerer Funktionsebenen dürfen sich nicht bei der Site registrieren.

Außerdem werden auf Katalogebene auch einige Nutzungswerte für Maschinen angezeigt, die im Katalog als verwendet oder verfügbar angegeben sind.

## Bereitstellungsgruppe

Die Bereitstellungsgruppe enthält Details zu einer Sammlung von Maschinen, die zur Bereitstellung von Desktops und/oder Anwendungen für einen Endbenutzer verwendet werden. Mehrere Bereitstellungsgruppen können mit demselben Katalog verknüpft werden, sodass die Maschinen eines Katalogs je nach den Anforderungen unterschiedlicher Benutzergruppen verteilt werden können. Die Bereitstellungsgruppe definiert Folgendes:

- Zuteilungstyp der Maschinen in der Gruppe: Gibt an, ob Maschinen für Benutzer freigegeben (zufällig) oder permanent einem oder mehreren Benutzern zugewiesen sind (statisch) und ob es sich um Maschinen mit Einzelsitzungen (Desktopbetriebssystem) oder mehreren Sitzungen (Serverbetriebssystem) handelt.
- Bereitstellungstyp der Gruppe: Gibt an, ob die Gruppe nur Anwendungen, nur Desktops oder eine Mischung aus Anwendungen und Desktops für Benutzer bereitstellt.
- Steuerungseinstellungen für die Energieverwaltung der Maschinen in der Gruppe, u. a.:
  1. Welche Stunden an unterschiedlichen Wochentagen als "Spitzenzeiten" gelten
  2. Wie viele Maschinen zu verschiedenen Zeiten an unterschiedlichen Wochentagen als zufällige/nicht zugewiesene Maschinen ausgeführt werden, einschließlich Puffergrößen
  3. Die zu verwendende Zeitzone zum Auswerten der täglichen Stunden für die oben aufgeführten Einstellungen
  4. Ob und wie die Energieverwaltung für zugewiesene Maschinen erfolgt
  5. Ob und wie das Herunterfahren oder Anhalten von Maschinen nach auslösenden Ereignissen erfolgt, z. B. in Bezug auf das Trennen der Verbindung für Benutzer oder deren Abmeldung
  6. Ob Maschinen durch die auf ihnen ausgeführten Sitzungen als beschädigt gelten und ein Neustart erzwungen wird, um sie nach jeder Verwendung wieder in einen sauberen, bekannten Zustand zu versetzen
- Wie Desktopressourcen der Gruppe Endbenutzern angezeigt werden, einschließlich verwendeter Symbole, Farbtiefe und Namen. Außerdem wird die erforderliche Sicherheitsstufe für die ICA-Verbindungen für die Maschinen in der Gruppe definiert. Für Desktopressourcen die Anzahl der Desktops, die ein Benutzer gleichzeitig aus der Gruppe beanspruchen kann.
- Regeln, die die Verfügbarkeit dieser Maschinen für die Endbenutzer festlegen. Regeln können nicht nur die Identität des Benutzers berücksichtigen, sondern auch von wo aus und wie der Benutzer eine Verbindung herstellt, den Status des Clientgeräts und die unterstützten Remoting-Protokolle.
- Ob die Gruppe aktiviert oder deaktiviert ist, einschließlich der Einstellung für den Wartungsmodus.
- Ob Endbenutzer die Maschinen selber zurücksetzen dürfen (z. B. mit StoreFront).
- Tagwerte für eine zufällige/freigegebene Gruppe.
- Die für die Maschinen der Gruppe erwartete Funktionsebene; Maschinen niedrigerer Funktionsebenen dürfen sich nicht bei der Site registrieren.
- Ob die Gruppe für Remote-PC-Benutzer verwendet werden kann und welche Remote-PC-Kataloge der Gruppe zugeordnet sind.
- Ein Zeitplan zum regelmäßigen Neustarten von Maschinen mit mehreren Sitzungen (Serverbetriebssystem) zu einer bestimmten Zeit und an einem bestimmten Tag sowie Steuerungseinstellungen für die Ausführung des Neustarts.

Mit Bereitstellungsgruppen sind darüber hinaus Einstellungen für Features verknüpft, wie Profilverwaltungs- und StoreFront-URL-Einstellungen, die separat als "Maschinenkonfigurationsobjekte" definiert werden und einer oder mehreren Bereitstellungsgruppen zugeordnet sind.

## Anwendung

Hiermit können die Charakteristiken einer Seamlessanwendung (d. h. ein unverankertes Fenster, separat vom Desktop) für Endbenutzer festgelegt werden. Normalerweise ist jede Anwendung nur einer Bereitstellungsgruppe zugeordnet, aber Anwendungsdefinitionen können bei Bedarf für mehrere Gruppen freigegeben werden. Anwendungen können entweder auf einer Remotemaschine ausgeführt und auf dem lokalen Clientdesktop angezeigt werden oder sie können auf der lokalen Clientmaschine installiert sein und dort ausgeführt werden, wobei Windows auf einem Remotedesktop überlagert wird. Die Anwendung definiert Folgendes:

- Typ der Anwendung, "HostedOnDesktop" (Auf Desktop gehostet) oder "InstalledOnClient" (Auf Client installiert).
- Für auf Desktops gehostete Anwendungen:
  1. Den Pfad zur ersten ausführbaren Datei für die Anwendung, die auf der VDA-Maschine ausgeführt wird, und die Befehlszeilenparameter, die ggf. beim Start der Anwendung angegeben werden
  2. Eine bestimmte Gruppe von Benutzern mit Zugriff auf die Anwendung als eine Untergruppe der Benutzer, die Zugriff auf die Bereitstellungsgruppe(n) haben (optional)
  3. Die anwendungsspezifischen Einstellungen, die auf den Anwendungsprozess anzuwenden sind, u. a. eine CPU-Prioritätsstufe, ob die Anwendung auf die Erstellung von Proxy-Druckern warten soll, usw.
- Für auf Clients installierte Anwendungen:
  1. Ein Flag, das angibt, dass das Symbol für die Anwendung vom Clientgerät abzurufen ist
  2. Ein Flag, das festlegt, dass mit den für die Anwendung bereitgestellten Argumenten zusätzliche Sicherheitsmaßnahmen verwendet werden
- Wie die Anwendungsressource dem Endbenutzer angezeigt wird, einschließlich verwendeter Symbole, Standardspeicherort auf dem Clientgerät, Name und ob die Verknüpfung im Startmenü oder auf dem Desktop oder beidem angezeigt wird.
- Alle Dateitypuordnungen für die Anwendung durch Zuordnung von Dateierweiterungen zur Anwendung.
- Die Bereitstellungsgruppen, denen die Anwendung zugeordnet ist, und ein optionaler Prioritätswert für die Wahl zwischen mehreren Gruppen.
- Ob die Anwendung aktiviert ist und, davon unabhängig, ob sie Endbenutzern angezeigt wird.

## Desktop

In XenDesktop 7 wird das Desktopobjekt, das die Maschine und die Sitzung auf der Maschine beschreibt, durch das Sitzungsobjekt und das Maschinenobjekt ersetzt, die beide erweitert wurden und jetzt die Funktion des Desktopobjekts übernehmen.

## Sitzung

Hiermit werden die Details einer Windows-Sitzung festgelegt, die auf einer von der Site gesteuerten Maschine ausgeführt wird. Die Sitzung kann von XenDesktop oder XenApp initiiert oder auf andere Art erstellt werden, z. B. durch einen Benutzer, der sich über die Konsole oder über RDP direkt bei der Maschine anmeldet.

Die Sitzung definiert Folgendes:

- Die Identität der Maschine, auf der die Sitzung ausgeführt wird, einschließlich Maschinen-DNS-Name, IP-Adresse, NetBios-Name, SID, usw.
- Die Identität des Benutzers, der die Sitzung ausführt, einschließlich SAM-Name, UPN, SID, usw.
- Die Identität des Benutzers, der die Verbindung mit der Sitzung vermittelt hat, einschließlich SAM-Name, SID, usw.
- Die Identität der Endpunkt-Clientmaschine, die für die Verbindung mit der Sitzung verwendet wurde, einschließlich Gerätenamen, IP-Adresse und ID.
- Die Identität der Maschine, die den Start angefordert hat, d. h. der Webserver, von dem der Start ausgeführt wurde. Einschließlich Name und IP-Adresse.

- Die Identität der Maschine, die als Gateway für die Sitzungsverbindung fungiert, einschließlich Maschinen-DNS-Name und IP-Adresse.
- Details wichtiger Ereignisse in der Sitzung, einschließlich Startzeit, die Zeit, zu der zuletzt eine Verbindung zur Sitzung hergestellt bzw. vermittelt wurde, usw.
- Die Dauer von Aspekten der aktuellen Sitzungserstellung oder -verbindung, u. a. die erforderliche Zeit zum Vermitteln der Sitzung, die erforderliche Zeit zum Erstellen der Sitzung, usw.
- Den aktuellen Status der Sitzung, einschließlich dem Gesamtsitzungszustand, ob die ICA-Verbindung gesichert ist und das verwendete Protokoll.
- Den aktuellen Zustand der Maschine, auf der die Sitzung ausgeführt wird, einschließlich einem Zusammenfassungszustand, Energiezustand, usw.
- Details darüber, wie die Sitzungsverbindung hergestellt wurde, z. B. ob die Sitzung autonom vermittelt oder verbunden wurde sowie die verwendeten Sitzungskontexttags für "SmartAccess".
- Ob die Sitzung ausgeblendet ist. Sitzungen können ausgeblendet werden, wenn bestimmte Arten von Problemen auftreten, während ein Benutzer eine Anwendung oder einen Desktop startet.
- Die Liste der vermittelten Anwendungen, die in der Sitzung ausgeführt werden.

Das Sitzungs-SDK-Objekt hat zudem Informationen von den zugehörigen Maschinen-, Bereitstellungsgruppen- und Katalog-SDK-Objekten. Zu diesen Informationen gehören Identitätsinformationen, grundlegende Konfiguration- und Statusinformationen.

## Controller

Hiermit werden Details zu den einzelnen Delivery Controller-Maschinen in der Site festgelegt. Die meisten Daten sind dynamische Zustandsdaten aus der ausgeführten Site und nicht Konfigurationseinstellungen. Der Controller zeigt Folgendes an:

- Die aktiven Sitedienste für jeden Controller.
- Die Version der Controller-Komponenten.
- Die Identität der Controller-Maschine als vollständig qualifizierter DNS-Name, SAM-Name, SID, usw.
- Typ und Version des Betriebssystems der Controller-Maschine.
- Aktueller Status von Controller und Dienst sowie aktuelle Aktivitätszeiten.
- Anzahl der beim Controller registrierten Maschinen.
- Die dem Controller zugeordneten Hypervisorverbindungen zum Zweck der Sitedienstidentifizierung.

Die meisten Controller-Werte werden vom Broker Service SDK verfügbar gemacht, aber Elemente werden auch von anderen Diensten, z. B. den Maschinenerstellungsdiensten, angegeben.

## Site

Die Site ist eine logische Darstellung der XenDesktop-Site auf der obersten Ebene, und zwar aus der Sicht der in der Site ausgeführten Konfigurationsdienste. Die Site enthält u. a. Lizenzierungsinformationen, Sitemetadaten und den Sitenamen.

Eine XenDesktop-Installation umfasst nur eine einzige Konfigurationssiteinstanz. Das Siteobjekt hat die folgenden Eigenschaften:

- Name
- Controller: Liste der Controller in der Site
- Datenbanken: Liste der von der Site verwendeten Datenbanken
- DefaultIconUid
- LicenseInformation

- Metadaten

## Protokoll

Es enthält Details zu den gesammelten Konfigurationsprotokollen, die die Administratoraktivitäten für die Site seit der Aktivierung der Protokollierung beschreiben. Lesevorgänge des Administrators werden nicht protokolliert, aber alle Administratoraktionen, die die Konfiguration oder den Status der Site ändern, werden im Protokoll erfasst. Sie können das Protokoll auf einer von drei Stufen anzeigen: Protokolle auf hoher Stufe, Protokolle auf niedriger Stufe und Vorgangsdetails. Protokolle der niedrigen Stufe sind Teil eines größeren Vorgangs auf hoher Stufe, während die Vorgangsdetails Elemente innerhalb eines Vorgangs auf niedriger Stufe beschreiben. Protokollobjekte zeigen Folgendes an:

- Die Identität des Administrators, der den Vorgang durchgeführt hat, einschließlich der IP-Adresse der Maschine, auf der er ausgeführt wurde.
- Anfangs- und Endzeit der Ausführung des Vorgangs.
- Erfolg oder Fehler beim Ausführen des Vorgangs sowie Grund.
- Eine Beschreibung des Vorgangs als Textzeichenfolge mit Vorgangstyp, Quelltyp, Zieltyp, usw.
- Details der für den Vorgang bereitgestellten Parameter.
- Alle übergeordneten/untergeordneten Beziehungen in der Protokollhierarchie.
- Bei Konfigurationsänderungen der Wert vor und nach der Änderung für die geänderten Objekte.

Weitere Informationen darüber, was protokolliert wird, finden Sie in der Dokumentation unter [Konfigurationsprotokollierung](#).

## Recht

Mit einem Recht wird eine Kombination aus Rolle und dem zulässigen Geltungsbereich für die Rollenberechtigungen definiert. Die in der Rolle definierten Berechtigungen können von dem angegebenen Administrator ausgeführt werden, aber nur für Objekte, die direkt oder indirekt dem angegebenen Geltungsbereich zugeordnet sind. Ein Recht definiert Folgendes:

- Die Rolle mit den zulässigen Berechtigungen.
- Den Geltungsbereich mit den zulässigen Rollenberechtigungen.

Rechte sind keine SDK-Objekte, die separat bearbeitet werden können. Sie sind immer einem bestimmten Administratorobjekt zugeordnet. Ein einzelner Administrator kann mehrere Berechtigungen haben, wobei die Gesamtfunktion des Administrators die Summe aller individuellen Rechte ist.

## Geltungsbereich

Ein Geltungsbereich definiert eine benannte Gruppierung von Objekten, wobei Objekte in der Gruppierung Administratorrechte für Objekte haben, die basierend auf einzelnen Rollen gesteuert werden. SDK-Objekten verschiedener Typen können Geltungsbereiche direkt zugeordnet sein, z. B. Kataloge, Bereitstellungsgruppen, Verbindungen, Ressourcen, usw. Diese SDK-Objekte haben Eigenschaften, die die Geltungsbereiche auflisten, die dem Objekt zugeordnet wurden. Anderen SDK-Objekten werden durch sekundäre Zuordnung zu Objekten, denen Geltungsbereiche direkt zugeordnet sind, Administratorrechte gewährt. Beispiel: Maschinenobjekte übernehmen die Geltungsbereichszuordnungen von den Maschinenkatalogen und Bereitstellungsgruppen, denen sie angehören; Sitzungsobjekte übernehmen die Geltungsbereichszuordnungen von den Maschinen, auf denen die Sitzung ausgeführt wird. Einige Geltungsbereiche sind vordefiniert (in der Praxis ist zurzeit nur der Geltungsbereich "Alle" integriert). Sie können aber andere Geltungsbereiche erstellen und so geeignete Gruppierungsdefinitionen für Ihre Bereitstellung festlegen.

Ein Geltungsbereich definiert Folgendes:

- Die Identität des Geltungsbereichs, die u. a. einen Namen und einen internen ID-Wert (GUID) umfasst.
- Ob der Geltungsbereich integriert ist.

## Rolle

Eine Rolle definiert einen Satz Berechtigungen, die ein Administrator ausführt. Rollen werden Administratoren immer mit einem zugeordneten Geltungsbereich gewährt. Sie alleine verleihen keine Rechte, obwohl einige Rollen allgemeine Berechtigungen für Objekte haben können, denen keine Geltungsbereiche zugeordnet sind. Einige Rollen sind vordefiniert, aber Sie können benutzerdefinierte Rollen erstellen und so geeignete Berechtigungen für Ihre Bereitstellung festlegen. Eine Rolle definiert Folgendes:

- Die Identität der Rolle, die u. a. einen Namen und einen internen ID-Wert (GUID) umfasst.
- Ob die Rolle integriert ist.
- Die Berechtigungen, die die Rolle ausmachen.

Integrierte Rollen:

- Volladministrator: Kann alle Aufgaben und Vorgänge ausführen.
- Lesezugriffadministrator: Kann alle Objekte in den angegebenen Geltungsbereichen sowie globale Informationen anzeigen, aber nicht ändern.
- Maschinenkatalogadministrator: Kann Maschinenkataloge erstellen und verwalten sowie Maschinenprovisioning durchführen.
- Bereitstellungsgruppenadministrator: Kann Anwendungen, Desktops und Maschinen bereitstellen sowie die mit ihnen verbundenen Sitzungen verwalten. Kann Bereitstellungsgruppen und Anwendungen erstellen und verwalten.
- Helpdeskadministrator: Kann Bereitstellungsgruppen anzeigen und die diesen Gruppen zugeordneten Sitzungen und Maschinen verwalten. Kann zur Problembehandlung von Endbenutzerproblemen Endbenutzerressourcen anzeigen und eingeschränkte Zustandsänderungen durchführen, darf aber kaum Konfigurationsänderungen vornehmen.
- Hostadministrator: Kann Hostverbindungen und ihnen zugeordnete Ressourceneinstellungen verwalten.

## Berechtigung

Eine Berechtigung definiert eine Aufgabe oder einen Vorgang auf Konsolenebene, die/der zulässig ist, wenn die Berechtigung in einer Rolle eingeschlossen ist. Jede Berechtigung kann mehrere SDK-Vorgänge (Cmdlets) auf niedriger Ebene zulassen und ein SDK-Vorgang auf einer besonders niedrigen Ebene kann durch verschiedene verwandte Berechtigungen gewährt werden. Eine Berechtigung definiert Folgendes:

- Die Identität der Berechtigung, die u. a. einen Namen und einen internen ID-Wert (GUID) umfasst.
- Die Gruppenmitgliedschaft der Berechtigung. In Berechtigungsgruppen sind Berechtigungen für einen bestimmten Funktionsbereich zusammengefasst.
- Den Satz SDK-Vorgänge auf niedriger Ebene, den die Berechtigung umfasst.



# Erste Schritte mit dem SDK

May 10, 2016

Führen Sie folgende Schritte für das Erstellen eines Skripts aus:

1. Führen Sie in Citrix Studio den Vorgang aus, für den Sie ein Skript erstellen möchten, z. B. Erstellen eines Katalogs für eine Gruppe von Maschinenerstellungsdienste-Maschinen.
2. Sammeln Sie das Protokoll der SDK-Vorgänge, die Studio für das Durchführen der Task ausgeführt hat.
3. Überprüfen Sie das Skript, um die Funktion jedes Teils zu verstehen. Dies hilft Ihnen bei der Anpassung des eigenen Skripts. Weitere Informationen finden Sie im Beispiel eines Anwendungsfalls, in dem die Skriptvorgänge ausführlich beschrieben werden.
4. Konvertieren und passen Sie das Studio-Skriptfragment an, um es in ein besser verwendbares Skript zu verwandeln. Gehen Sie hierzu folgendermaßen vor:
  - Verwenden Sie Variablen. Für einige Cmdlets können Parameter angegeben werden, z. B. TaskId. Allerdings kann es nicht eindeutig sein, woher der in diesen Parametern verwendete Wert stammt, da Studio Werte von den Ergebnisobjekten früherer Cmdlets verwendet.
  - Entfernen Sie alle Befehle, die nicht erforderlich sind.
  - Fügen Sie einige Schritte in einer Schleife hinzu, damit sie einfach zu steuern sind. Beispiel: Fügen Sie die Maschinenerstellung in einer Schleife hinzu, damit Sie die Anzahl der erstellten Maschinen steuern können.

## Beispiele

Hinweis: Um beim Erstellen eines Skripts sicherzustellen, dass Sie immer die aktuellen Verbesserungen und Fixes erhalten, empfiehlt Citrix, dass Sie den oben erläuterten Schritten folgen, anstatt das Beispielskript zu kopieren und einzufügen.

Beispiele	Beschreibung
<a href="#">Create catalog</a>	Skript: Erstellen eines Katalogs für eine Gruppe von Maschinenerstellungsdienste-Maschinen
<a href="#">Beispiel: Erstellen und Konfigurieren eines Hosts</a>	Skript: Erstellen und Konfigurieren eines Hosts
<a href="#">Beispiel: Erstellen eines PvD-Desktops</a>	Skript: Erstellen einer Bereitstellungsgruppe, die PvD-Desktops (Personal vDisk) enthält
<a href="#">Beispiel: Abrufen von Lastausgleichsinformationen</a>	Anzeigen der Lastindexwerte für Serverbetriebssystemmaschinen



# Beispiel: Erstellen eines Katalogs

May 10, 2016

Das folgende Beispiel zeigt, wie ein Katalog für eine Gruppe von Maschinenerstellungsdienste-Maschinen erstellt wird.

Bevor Sie beginnen, stellen Sie sicher, dass Sie die in **Erste Schritte mit dem SDK** aufgeführten Schritte ausführen. In diesem Dokument wird dargestellt, wie Sie Studio für die Durchführung des Vorgangs verwenden, für den Sie ein Skript erstellen möchten (in diesem Fall das Erstellen eines Katalogs für eine Gruppe von Maschinenerstellungsdienste-Maschinen) und das Sammeln des Protokolls der SDK-Vorgänge, die Studio für das Ausführen der Aufgabe vorgenommen hat. Die Ausgabe kann dann so angepasst werden, dass ein Skript für die Automatisierung der Katalogerstellung erstellt wird.

Hinweis: Um sicherzustellen, dass Sie immer die aktuellen Verbesserungen und Fixes erhalten, empfiehlt Citrix, dass Sie den in diesem Dokument beschriebenen Schritten folgen, anstatt das Beispielskript zu kopieren und einzufügen. Zeilennummern und Zeilenumbrüche wurden dem Skript für eine bessere Lesbarkeit hinzugefügt.

## Inhalt des Skripts

Im folgenden Abschnitt wird die Funktion jedes Teils des Skripts erläutert, das Studio erstellt. Dies hilft Ihnen bei der Anpassung des eigenen Skripts. Zeilennummern wurden für eine bessere Lesbarkeit hinzugefügt.

1. **Start-LogHighLevelOperation** -AdminAddress 'ddc.dumdev.internal.citrix.com:80' -Source 'Studio' -StartTime 29/05/2013 14:43:08 -Text 'Create Machine Catalog "ExampleMachines"'  
Startet einen protokollierten Vorgang und gibt eine Protokoll-ID zurück, die an nachfolgende Vorgänge übergeben wird, um sie der größeren Task zuzuordnen.
2. **New-BrokerCatalog** -AdminAddress 'ddc.dumdev.internal.citrix.com:80' -AllocationType 'Permanent' -Description 'Example Machines' -IsRemotePC \$False -LoggingId f39a2792-064a-43eb-97c7-397cc1238e46  
Erstellt einen Brokerkatalog. Dieser Katalog enthält Maschinen, die Sie gerade erstellen.
3. **New-AcctIdentityPool** -AdminAddress 'ddc.dumdev.internal.citrix.com:80' -AllowUnicode -Domain 'dumdev.internal.citrix.com' -IdentityPoolName 'ExampleMachines' -LoggingId f39a2792-064a-43eb-97c7-397cc1238e46  
Definiert den Mechanismus zum Erstellen von Active Directory-Computerkonten. Dies wird zu einem Container für Active Directory-Konten, die für die Maschinen erstellt werden, die Sie erstellen.
4. **Set-BrokerCatalogMetadata** -AdminAddress 'ddc.dumdev.internal.citrix.com:80' -CatalogId 1 -LoggingId f39a2792-064a-43eb-97c7-397cc1238e46 -Name 'Citrix\_DesktopStudio\_IdentityPoolUid' -Value 'f39a2792-064a-43eb-97c7-397cc1238e46'  
Stellt Metadaten im Brokerkatalog ein, die Details des Identitätspool enthalten. Dies ist nicht unbedingt erforderlich.
5. **Test-ProvSchemeNameAvailable** -AdminAddress 'ddc.dumdev.internal.citrix.com:80' -ProvisioningSchemeName @( 'ExampleMachines' )  
Überprüft, ob der gewünschte Name verfügbar ist. Dies ist nicht unbedingt erforderlich.
6. **New-ProvScheme** -AdminAddress 'ddc.dumdev.internal.citrix.com:80' -CleanOnBoot -HostingUnitName 'SharedNFS' -IdentityPoolName 'ExampleMachines' -LoggingId f39a2792-064a-43eb-97c7-397cc1238e46  
Erstellt ein Bereitstellungsschema-Objekt. Dies ist eine Vorlage für die Maschinen, die Sie erstellen möchten. Sie gibt den Hypervisor, das Netzwerk, den Speicher, den Speicherplatz, die Anzahl der CPUs usw. an. Sie übernimmt Parameter vom bereits eingerichteten System, u. a. den Namen des Hostingeräts und den Pfad zum VM-Snapshot, die für die Maschinen verwendet werden, die Sie erstellen. Dieser Befehl erstellt eine konsolidierte Kopie des verwendeten VM-Snapshots, daher kann der Abschluss des Prozesses länger dauern.

In diesem Beispiel gibt das Studio-Skript das Flag -RunAsynchronous in diesem Befehl an. Dies bedeutet, dass der Befehl vor dem Abschluss die Steuerung an den Administrator zurückgibt; Sie müssen daher auf den Abschluss warten, bevor Sie Vorgänge durchführen, für die der Befehl abgeschlossen sein muss. Wenn dieses Flag nicht angegeben ist, wird der Befehl synchron intern ausgeführt und die Steuerung wird nicht zurückgegeben, bis der Befehl abgeschlossen ist (erfolgreich oder fehlerhaft). Sie können den Status einer asynchronen Task mit dem Cmdlet Get-ProvTask überprüfen. Geben Sie die Task-ID an, die der Vorgang zurückgegeben hat, der die Task gestartet hat, in diesem Fall das Cmdlet New-ProvScheme.

7. **Set-BrokerCatalog** -AdminAddress 'ddc.dumdev.internal.citrix.com:80' -LoggingId f39a2792-064a-43eb-97c7-397cc1238e46 -Name 'ExampleMachines' -ProvisioningSchemeId 76125e3a-9001-4993-8000-000000000000  
Aktualisiert den Brokerkatalog mit der eindeutigen ID des oben erstellten Provisioningschemas.
8. **Add-ProvSchemeControllerAddress** -AdminAddress 'ddc.dumdev.internal.citrix.com:80' -ControllerAddress @( 'DDC.dumdev.internal.citrix.com' ) -LoggingId f39a2792-064a-43eb-97c7-397cc1238e46  
Fügt dem Provisioningschema-Objekt eine Gruppe von Controlleradressen hinzu. Dies ist eine Liste von Adressen, die die erstellten Maschinen für die Registrierung bei einem Controller (Broker) verwenden können, wenn sie bereitgestellt werden. Die Registrierungsadressen der Maschinen können mit mehreren Methoden angegeben werden. Diese Informationen sind jedoch erforderlich, wenn der Administrator im VDA-Installationsprogramm die Adressen mit den Maschinenerstellungsdiensten angeben möchte. Änderungen an dieser Liste wirken sich nur auf Maschinen aus, die nach der Änderung erstellt werden, nicht auf vorhandene Maschinen.
9. **Get-AcctADAccount** -AdminAddress 'ddc.dumdev.internal.citrix.com:80' -IdentityPoolId b99aee6d-8772-4dbc-978b-8eb9a26e2407 -Lock \$False -MaxRecordCount 2147483647 -State 'Available'  
Studio erhält eine Liste der verfügbaren Maschinen-IDs vom Identitätspool, damit vorhandene Konten, die nicht verwendet werden, verbraucht werden können statt neue Konten zu erstellen. Hinweis: Dies wird nicht im Skript benötigt, da neue Konten erstellt werden können, wenn das Skript in einem Kontext ausgeführt wird, der entsprechende Berechtigungen hat. Wenn das Skript jedoch keine Berechtigungen zum Erstellen von Konten hat, ändern Sie das Skript so, dass verfügbare Konten verbraucht werden (ein eigenständiger Prozess wird benötigt, um ein Pool der Konten dem Identitätspool bereitzustellen, bevor das Skript ausgeführt wird).
10. **New-AcctADAccount** -AdminAddress 'ddc.dumdev.internal.citrix.com:80' -Count 2 -IdentityPoolId b99aee6d-8772-4dbc-978b-8eb9a26e2407 -LoggingId f39a2792-064a-43eb-97c7-397cc1238e46  
Erstellt die erforderlichen Active Directory-Computerkonten in Active Directory. Das Skript erstellt ein Konto und kann, falls erforderlich, weitere Konten mit dem Parameter "Count" des Befehls erstellen. Die Konten werden in der Organisationseinheit erstellt, die im obigen Provisioningschema festgelegt wurde.
11. **New-ProvVM** -ADAccountName @( 'DUMDEV\Example-0001\$', 'DUMDEV\Example-0002\$' ) -AdminAddress 'ddc.dumdev.internal.citrix.com:80' -LoggingId f39a2792-064a-43eb-97c7-397cc1238e46  
Erstellt virtuelle Maschinen basierend auf der Vorlagendefinition im oben erstellten Provisioningschema. Der Abschluss dieses Prozesses kann länger dauern.
12. **Lock-ProvVM** -AdminAddress 'ddc.dumdev.internal.citrix.com:80' -LoggingId f39a2792-064a-43eb-97c7-397cc1238e46 -ProvisioningSchemeName 'ExampleMachines' -Tag 'Brokered' -VMID @( '0710bb77-d01f-d006-4d67-5472e5cd349f' )  
Sperrt die bereitgestellten virtuellen Maschinen und verhindert versehentliche Modifikationen an der virtuellen Maschine. Consumer des SDKs können hiermit angeben, dass die virtuelle Maschine verwendet wird und den Grund für die Sperrung. Das Skript sperrt die VM mit einem Tag "Brokered". Dies gibt an, dass die virtuelle Maschine erstellt und dem Brokerkatalog hinzugefügt wurde und erst nach dem Entfernen aus dem Katalog gelöscht werden kann. Sie können den Tagnamen beliebig angeben.
13. **New-BrokerMachine** -AdminAddress 'ddc.dumdev.internal.citrix.com:80' -CatalogId 1 -HostedMachineId '0710bb77-d01f-d006-4d67-5472e5cd349f' -HypervisorConnectionId 1 -LoggingId f39a2792-064a-43eb-97c7-397cc1238e46  
Erstellt ein Broker-Maschinenobjekt. Dies sind im Katalog gespeicherte Objekte, die die bereitgestellte Maschine mit dem Katalog zusammenführen.
14. **Start-BrokerMachinePvdImagePrepare** -AdminAddress 'ddc.dumdev.internal.citrix.com:80' -InputObject @(2) -LoggingId f39a2792-064a-43eb-97c7-397cc1238e46  
Fordert den Brokerdienst auf, einen Vorbereitungsvorgang für Personal vDisk auszulösen. Dies ist erforderlich, damit die Maschine den Speicher für Personal vDisk initialisieren kann.
15. **Stop-LogHighLevelOperation** -AdminAddress 'ddc.dumdev.internal.citrix.com:80' -HighLevelOperationId f39a2792-064a-43eb-97c7-397cc1238e46 -IsSuccessful \$true  
Beendet die Protokollierung, die im ersten Schritt begonnen wurde, und gibt einen Erfolg an.

## Anpassen des Skripts

Im folgenden Abschnitt wird erläutert, wie Sie die Studio-Ausgabe in ein Skript konvertieren und anpassen, das besser verwendbar ist. Zusätzlich zur Verwendung von Variablen und dem Entfernen von nicht benötigten Befehlen erfahren Sie, wie Sie die Maschinenerstellung einer Schleife hinzufügen, damit Sie die Anzahl der erstellten Maschinen steuern können. Zeilennummern wurden für eine bessere Lesbarkeit hinzugefügt.

```
1. [CmdletBinding()] param ([Parameter(Mandatory=$true)] [string] $hostingUnitPath, [Parameter(Mandatory=$true)] [string] $catalogName, [string] $catalogDescription, [Parameter(Mandatory=$true)] [int]
```

# Beispiel: Erstellen und Konfigurieren eines Hosts

May 10, 2016

Das folgende Beispiel zeigt das Erstellen und Konfigurieren eines Hosts.

Bevor Sie beginnen, führen Sie die in **Erste Schritte mit dem SDK** erläuterten Schritte aus. Sie beschreiben, wie Sie mit Studio den Vorgang durchführen, für den Sie ein Skript erstellen möchten (in diesem Fall das Erstellen eines Hosts), und wie Sie das Protokoll der SDK-Vorgänge sammeln, die Studio für die Aufgabe ausgeführt hat. Die Ausgabe kann dann so angepasst werden, dass ein Skript für die Automatisierung der Hosterstellung erstellt wird.

Hinweis: Um sicherzustellen, dass Sie immer die aktuellen Verbesserungen und Fixes erhalten, empfiehlt Citrix, dass Sie den in diesem Dokument beschriebenen Schritten folgen, anstatt das Beispielskript zu kopieren und einzufügen. Zeilennummern und Zeilenumbrüche wurden dem Skript für eine bessere Lesbarkeit hinzugefügt.

## Inhalt des Skripts

Im folgenden Abschnitt wird die Funktion jedes Teils des Skripts erläutert, das Studio erstellt. Dies hilft Ihnen bei der Anpassung des eigenen Skripts. Zeilennummern wurden für eine bessere Lesbarkeit hinzugefügt.

1. Get-LogSite -AdminAddress 'mycontroller.example.com:80'

Führt eine Abfrage beim Konfigurationsprotokollierungsdienst durch, um Informationen zur Sitekonfiguration zu erhalten.

2. Start-LogHighLevelOperation -AdminAddress 'mycontroller.example.com:80' -Source 'Studio' -StartTime 14/08/2013 14:30:28 -Text 'Create Connection "Example XenServer"'

Startet einen allgemeinen Protokollierungsvorgang bei der Konfigurationsprotokollierung, in der die restlichen Befehle bestehen. Gibt eine Protokoll-ID zurück, die an nachfolgende Vorgänge übergeben wird.

3. Set-HypAdminConnection -AdminAddress 'mycontroller.example.com:80'

Legt den Speicherort des Hostdienstes fest, der von den Konfigurations-Cmdlets verwendet wird. Da der Hostdienst einen PowerShell-Anbieter offenlegt, können nicht alle Cmdlets eine Adresse für den Dienst annehmen. Daher legt dieses Cmdlet einen Standardspeicherort fest.

4. New-Item -ConnectionType 'XenServer' -HypervisorAddress @('http://xenhost1.example.com') -LoggingId e355ce51-8cbb-400a-ae81-1fdc567239cb -Path @('XDHyp:\Connections\Example XenServer')  
Erstellt eine Verbindung zu XenServer (xenhost1.beispiel.com). Dies ist eine nicht-persistente Verbindung und ist nur für diesen PowerShell-Runspace verfügbar.

5. Stop-LogHighLevelOperation -AdminAddress 'mycontroller.example.com:80' -EndTime 14/08/2013 14:30:29 -HighLevelOperationId 'e355ce51-8cbb-400a-ae81-1fdc567239cb' -IsSuccessful \$True  
Beendet die zuvor begonnene Protokollierung und gibt einen Erfolg an.

6. Get-LogSite -AdminAddress 'mycontroller.example.com:80'

Führt eine Abfrage beim Konfigurationsprotokollierungsdienst durch, um Informationen zur Sitekonfiguration zu erhalten.

7. Start-LogHighLevelOperation -AdminAddress 'mycontroller.example.com:80' -Source 'Studio' -StartTime 14/08/2013 14:30:30 -Text 'Update Connection "Example XenServer"'

Startet einen neuen allgemeinen Protokollierungsvorgang.

8. Set-HypAdminConnection -AdminAddress 'mycontroller.example.com:80'

Legt erneut die Adressangaben des Hostdienstes fest (diese Wiederholung wurde in dem optimierten Skript unten entfernt).

9. Set-Item -HypervisorAddress @('http://xenhost1.example.com','http://xenhost2.example.com') -LoggingId 44e15629-6906-4840-a36c-984aaf67be6d -PassThru -Path @('XDHyp:\Connections\Example XenServer')  
Aktualisiert die in Schritt 4 erstellte Verbindung. Da sich mehr als ein XenServer im Pool befindet, werden alle Adressen angegeben, damit hohe Verfügbarkeit möglich ist.

10. Stop-LogHighLevelOperation -AdminAddress 'mycontroller.example.com:80' -EndTime 14/08/2013 14:30:31 -HighLevelOperationId '44e15629-6906-4840-a36c-984aaf67be6d' -IsSuccessful \$True  
Beendet den in Schritt 7 begonnenen Protokollierungsvorgang.

11. Get-LogSite -AdminAddress 'mycontroller.example.com:80'

Führt eine Abfrage beim Konfigurationsprotokollierungsdienst durch, um Informationen zur Sitekonfiguration zu erhalten.

12. Start-LogHighLevelOperation -AdminAddress 'mycontroller.example.com:80' -Source 'Studio' -StartTime 14/08/2013 14:31:03 -Text 'Create Resources "Example Resources" and Persist Connection'

Startet einen neuen Protokollierungsvorgang.

13. Set-HypAdminConnection -AdminAddress 'mycontroller.example.com:80'

Legt erneut die Adressangaben des Hostdienstes fest.

14. Get-ChildItem -Path @('XDHyp:\Connections')

Füllt die Dialogfelder des Assistenten mit den Inhalten der Hostverbindung aus.

15. Set-HypAdminConnection -AdminAddress 'mycontroller.example.com:80'

Legt erneut die Adressangaben des Hostdienstes fest.

16. Remove-Item -LoggingId 76caa3f4-df93-4cb2-b78d-6a8824766314 -Path @('XDHyp:\Connections\Example XenServer')

Entfernt die im Assistenten erstellte temporäre Verbindung.

17. Set-HypAdminConnection -AdminAddress 'mycontroller.example.com:80'

Legt erneut die Adressangaben des Hostdienstes fest.

18. New-Item -ConnectionType 'XenServer' -HypervisorAddress @('http://xenhost1.example.com','http://xenhost2.example.com') -LoggingId 76caa3f4-df93-4cb2-b78d-6a8824766314 -Path @('XDHyp:\Connections\Example XenServer')  
Erstellt die Verbindung erneut als persistente Verbindung, die in die Datenbank geschrieben wird und somit anderen PowerShell-Runspace zur Verfügung steht.

19. New-BrokerHypervisorConnection -AdminAddress 'mycontroller.example.com:80' -HypHypervisorConnectionUid a14096ba-5074-44ff-b596-371e345c0449 -LoggingId 76caa3f4-df93-4cb2-b78d-6a8824766314  
Fügt die Hostverbindung zum Brokerdienst hinzu.

20. Set-HypAdminConnection -AdminAddress 'mycontroller.example.com:80'

Legt erneut die Adressangaben des Hostdienstes fest.

21. New-Item -HypervisorConnectionName 'Example XenServer' -LoggingId 76caa3f4-df93-4cb2-b78d-6a8824766314 -NetworkPath @('XDHyp:\Connections\Example XenServer\Network 0.network') -Path @('XDHyp:\Connections\Example XenServer\Network 0.network')  
Erstellt die HostingUnit (in Studio als "Ressourcen" bezeichnet) mit den in Schritt 14 gesammelten Informationen.

22. Set-HypAdminConnection -AdminAddress 'mycontroller.example.com:80'

Legt erneut die Adressangaben des Hostdienstes fest.

23. Get-Item -Path @('XDHyp:\Connections\Example XenServer')

Ruft ein neu erstelltes Objekt ab.

Stop-LogHighLevelOperation -AdminAddress 'mycontroller.example.com:80' -EndTime 14/08/2013 14:31:07 -HighLevelOperationId '76caa3f4-df93-4cb2-b78d-6a8824766314' -IsSuccessful \$True  
Beendet die zuvor begonnene Protokollierung und gibt an, ob sie erfolgreich war.

#### Anpassen des Skripts

Im folgenden Abschnitt wird erläutert, wie Sie die Studio-Ausgabe in ein Skript konvertieren und anpassen, das besser verwendbar ist. Das folgende Skript wurde vereinfacht, sodass anstelle der temporären Hostverbindung, die mit dem Studio-Skript erstellt wurde, um die Assistenten mit Informationen zu versorgen, eine persistente Verbindung erstellt wird. Aus dieser Verbindung heraus werden dann Informationen für die Erstellung der HostingUnit (Ressourcen) abgefragt. Hinweis: Die LoggingId- und HypHyperConnectionUid-Details sind unterschiedlich.

Für bessere Lesbarkeit wurden Zeilennummern hinzugefügt. Jedes nummerierte Element steht für einen PowerShell-Befehl.

1. Start-LogHighLevelOperation -AdminAddress 'mycontroller.example.com:80' -Source 'Studio' -Text 'Create Connection "Example XenServer"' 2. Set-HypAdminConnection -AdminAddress 'mycontrolle

# Beispiel: Erstellen eines PvD-Desktops

May 10, 2016

Mit dem Skriptbeispiel in diesem Dokument kann eine Bereitstellungsgruppe erstellt werden, die PvD-Desktops (Personal vDisk) enthält.

Bevor Sie beginnen, führen Sie die unter [Erste Schritte mit dem SDK](#) erläuterten Schritte aus. Sie beschreiben, wie Sie Studio für die Durchführung des Vorgangs verwenden, für den Sie ein Skript erstellen möchten, und für das Sammeln des Protokolls der SDK-Vorgänge, die Studio für die Ausführung der Aufgabe vorgenommen hat. Die Ausgabe kann so angepasst werden, dass ein Skript für die Automatisierung der Aufgabe erstellt wird.

**Hinweis:** Um sicherzustellen, dass Sie immer die aktuellen Verbesserungen und Fixes erhalten, empfiehlt Citrix, dass Sie den in diesem Dokument beschriebenen Schritten folgen, anstatt das Beispielskript zu kopieren und einzufügen.

## Inhalt des Skripts

Im folgenden Abschnitt wird die Funktion jedes Teils des Skripts erläutert, das Studio erstellt. Dies hilft Ihnen bei der Anpassung des eigenen Skripts. Zeilennummern und Zeilenumbrüche wurden dem Skript für eine bessere Lesbarkeit hinzugefügt.

- 1. Start-LogHighLevelOperation -AdminAddress 'test-ddc.mydomain.com:80' -Source 'Studio' -StartTime 31/07/2013 10:08:58 -Text 'Create Delivery Group 'Win7 PvD Desktops''**  
Startet einen protokollierten Vorgang und gibt eine Protokoll-ID zurück, die an nachfolgende Vorgänge übergeben wird, um sie der größeren Aufgabe zuzuordnen.
- 2. New-BrokerDesktopGroup -AdminAddress 'test-ddc.mydomain.com:80' -ColorDepth 'TwentyFourBit' -DeliveryType 'DesktopsOnly' -DesktopKind 'Private' -InMaintenanceMode \$False -IsRemotePC \$False**  
Erstellt eine neue Bereitstellungsgruppe mit Optionen, die vom Studio-Assistenten gesammelt wurden.
- 3. Add-BrokerMachinesToDesktopGroup -AdminAddress 'test-ddc.mydomain.com:80' -Catalog 'win7-pvd' -Count 2 -DesktopGroup 'Win7 PvD Desktops' -LoggingId 846f2d42-a994-4bce-ab58-be05c8d73b99'**  
Fügt die vom benannten Katalog angeforderte Anzahl an Maschinen zur neuen Bereitstellungsgruppe hinzu.
- 4. Set-Variable -Name 'brokerUsers' -Value @('S-1-5-21-3291547628-200264090-930806513-1104';'S-1-5-21-3291547628-200264090-930806513-1105')** Get-BrokerUser -AdminAddress 'test-ddc.mydomain.com:80'  
Die oben aufgeführten Befehle sind nicht erforderlich, da Benutzer von Studio verifiziert werden.
- 5. Test-BrokerAssignmentPolicyRuleNameAvailable -AdminAddress 'test-ddc.mydomain.com:80' -Name @('Win7 PvD Desktops')**  
Studio prüft, ob der Richtlinienzuordnungsname zur Verfügung steht.
- 6. New-BrokerAssignmentPolicyRule -AdminAddress 'test-ddc.mydomain.com:80' -DesktopGroupUid 41 -Enabled \$True -IncludedUserFilterEnabled \$False -LoggingId 846f2d42-a994-4bce-ab58-be05c8d73b99'**  
Erstellt die neue Richtlinienzuordnungsregel für die Bereitstellungsgruppe. Es sind keine Benutzer angegeben, daher erfolgt die Steuerung durch die Zugriffsrichtlinienregel.
- 7. Set-Variable -Name 'brokerUsers' -Value @('S-1-5-21-3291547628-200264090-930806513-1104';'S-1-5-21-3291547628-200264090-930806513-1105')** Get-BrokerUser -AdminAddress 'test-ddc.mydomain.com:80'  
Die oben aufgeführten Befehle sind nicht erforderlich, da Studio weitere Prüfungen durchführt.
- 8. Test-BrokerAccessPolicyRuleNameAvailable -AdminAddress 'test-ddc.mydomain.com:80' -Name @('Win7 PvD Desktops\_Direct')**  
Studio stellt sicher, dass der Regelname für die Zugriffsrichtlinie zur Verfügung steht.
- 9. New-BrokerAccessPolicyRule -AdminAddress 'test-ddc.mydomain.com:80' -AllowedConnections 'NotViaAG' -AllowedProtocols @('HDX';'RDP') -AllowRestart \$True -DesktopGroupUid 41 -Enabled \$True**  
Erstellt die Zugriffsrichtlinienregel für den neuen Desktop für Verbindungen, die nicht über NetScaler Gateway erfolgen.
- 10. Test-BrokerAccessPolicyRuleNameAvailable -AdminAddress 'test-ddc.mydomain.com:80' -Name @('Win7 PvD Desktops\_AG')** New-BrokerAccessPolicyRule -AdminAddress 'test-ddc.mydomain.com:80'  
Studio wiederholt diesen Vorgang für Verbindungen über NetScaler Gateway.
- 11. Test-BrokerPowerTimeSchemeNameAvailable -AdminAddress 'test-ddc.mydomain.com:80' -Name @('Win7 PvD Desktops\_Weekdays')** New-BrokerPowerTimeScheme -AdminAddress 'test-ddc.mydomain.com:80'  
Studio prüft, ob die Namen für die (optionalen) Wochentag- und Wochenend-Energieschemas verfügbar sind und fügt sie hinzu.
- 12. Stop-LogHighLevelOperation -AdminAddress 'test-ddc.mydomain.com:80' -EndTime 31/07/2013 10:09:05 -HighLevelOperationId '846f2d42-a994-4bce-ab58-be05c8d73b99' -IsSuccessful \$True**  
Beendet die Protokollierung, die in Schritt 1 begonnen wurde, und gibt einen Erfolg an.

## Anpassen des Skripts

In diesem Abschnitt wird erläutert, wie Sie die Studio-Ausgabe in ein besser verwendbares Skript konvertieren und anpassen.

Das Skript erstellt eine Bereitstellungsgruppe, die PvD-Desktops enthält. Der in den Parametern angegebene Katalog muss bereits vorhanden sein und die richtigen Daten enthalten (den Zuteilungstyp "Static" und einen PvD-Datenträger). Das Skript ist für die Ausführung von einer PowerShell-Befehlszeile konzipiert, wobei Sie als Citrix Administrator angemeldet sein müssen. Es werden keine Berechtigungen überprüft. Das Skript schlägt fehl, wenn der Benutzer nicht über die entsprechenden Berechtigungen verfügt.

<# Sample usage: .\CreatePvDGroup.ps1 -GroupName "Win7 PvD Desktops" -SrcCatalog "win7-pvd" -NumDesktops 2 -Users @('mydomain\user1';'mydomain\user2') -#> Param( [Parameter(Mandatory=\$true)] \$GroupName, [Parameter(Mandatory=\$true)] \$SrcCatalog, [Parameter(Mandatory=\$true)] \$NumDesktops, [Parameter(Mandatory=\$true)] \$Users )

In der Tabelle werden die im Skript verwendeten Parameter erläutert.

Parameter	Beschreibung
GroupName	Der Name der neuen PvD-Desktopgruppe.
SrcCatalog	Der Name des Katalogs, der zum Erstellen des PvD-Desktops verwendet wird. Verwenden Sie beim Erstellen des Katalogs den Zuteilungstyp "Static". Auch Maschinen müssen PvD-Datenträger haben.
NumDesktops	Die Anzahl der Maschinen, die der PvD-Desktopgruppe hinzugefügt werden. Wenn nicht genug Maschinen verfügbar sind, werden so viele wie möglich hinzugefügt.
Benutzer	Die Benutzer, die auf die Gruppe zugreifen können. Dies ist eine Liste von Benutzern oder Gruppen, z. B. @('mydomain\Domain Users') oder @('mydomain\user1';'mydomain\user2')

Set-HypAdminConnection -AdminAddress \$adminAddress

Gibt die zu verwendende Hypervisor-Adminverbindung an. Damit entfällt für einige Befehle der Bedarf von -AdminAddress.

\$peakPoolSize = 2 \$weekendPoolSizeByHour = new-object int[] 24 \$weekdayPoolSizeByHour = new-object int[] 24 9..17 | %{ \$weekdayPoolSizeByHour[\$\_] = \$peakPoolSize } \$peakHours = (0..23 | %{

Hiermit werden 24 Elementarrays mit einer 1 oder einer 0 in jedem Eintrag erstellt. Auf diese Weise können Sie für die Bereitstellungsgruppen die Spitzenzeiten für die Energiezeitpläne festlegen. Die

Elemente 9 bis 17 (Stunden beginnend von 09:00 bis 17:00) für Wochentage sind auf 1 eingestellt, während andere auf 0 festgelegt sind. Während der Spitzenzeiten werden zwei nicht zugewiesene Maschinen gestartet (wenn verfügbar).

```
$logId = Start-LogHighLevelOperation -Text "Create PvD desktop group" -Source "Create PvD Desktop Group Script"
```

Start einer neuen Protokollierung. Hierbei wird eine Protokoll-ID zurückgegeben, die an nachfolgende Vorgänge übergeben wird, um sie der Aufgabe zum Erstellen von Gruppen zuzuordnen.

```
$grp = New-BrokerDesktopGroup -DesktopKind 'Private' -DeliveryType 'DesktopsOnly' -LoggingId $logId -Name $GroupName -PublishedName $GroupName -SessionSupport 'SingleSession'
```

Erstellt die neue Bereitstellungsgruppe und stellt private Desktops bereit. Der verwendete Katalog muss passende Maschinen enthalten (permanent mit einem PvD-Datenträger). PublishedName ist der Name, der Endbenutzern angezeigt wird. Im Folgenden wird der Name als Gruppenname verwendet.

```
New-BrokerAssignmentPolicyRule -DesktopGroupUid $grp.Uid -IncludedUserFilterEnabled $False -LoggingId $logId -MaxDesktops 1 -Name ($GroupName + '_AssignRule') | Out-Null
```

Zugewiesene Desktops benötigen eine Zuordnungsrichtlinie. Deaktivieren Sie den Benutzerfilter, damit der Zugriff vollständig von Zugriffsrichtlinienregeln gesteuert wird.

```
New-BrokerAccessPolicyRule -AllowedConnections 'NotViaAG' -AllowedProtocols @('HDX','RDP') -AllowRestart $True -DesktopGroupUid $grp.Uid -IncludedSmartAccessFilterEnabled $True -Name ($GroupName + '_AccessRule')
```

Geben Sie Zugriffsbeschränkungen an: Ermöglichen Sie direkten Zugriff über NetScaler Gateway mit HDX- und RDP-Protokollen. Der Benutzer kann bei Bedarf einen Neustart des Desktops anfordern.

```
New-BrokerPowerTimeScheme -DaysOfWeek 'Weekdays' -DesktopGroupUid $grp.Uid -DisplayName 'Weekdays' -LoggingId $logId -Name ($GroupName + '_Weekdays') -PeakHours $peakHours
```

Optional: Geben Sie Energiezeitpläne an.

```
Stop-LogHighLevelOperation -HighLevelOperationId $logId -IsSuccessful $True
```

Beendet die Konfigurationsprotokollierung und gibt an, ob der Vorgang erfolgreich war oder nicht.

# Beispiel: Abrufen von Lastausgleichinformationen

May 10, 2016

Sie können mit Serverbetriebssystemmaschinen kostengünstig auf Serverbetriebssystemen gehostete Anwendungen und Desktops mehreren Benutzern bereitstellen.

Der Lastausgleich für Serverbetriebssystemmaschinen in einer Bereitstellung erfolgt mit Citrix Richtlinien. Es gibt mehrere Lastausgleichsrichtlinieneinstellungen zum Aktivieren und Konfigurieren der Lastverwaltung zwischen Servern, die Windows-Serverbetriebssystemmaschinen bereitstellen. Weitere Informationen finden Sie in der Referenzdokumentation zu den Richtlinieneinstellungen für die Lastverwaltung. Sie arbeiten mit Richtlinien in Studio oder in der Gruppenrichtlinien-Verwaltungskontrolle in Windows. Weitere Informationen finden Sie in der Dokumentation zu Richtlinien.

Sie zeigen die Last entweder mit Citrix Director oder mit den Studio-Konsolen oder mit dem PowerShell SDK an. Im folgenden Beispiel wird beschrieben, wie Sie die Last mit dem PowerShell SDK anzeigen.

Hinweis: Wenn Sie frühere Versionen von XenDesktop verwendet haben, sind Sie sicherlich mit dem Befehl `qfarm/load` vertraut. Dieses Tool steht nicht mehr zur Verfügung. Mit PowerShell können Sie jedoch eine ähnliche Ausgabe anzeigen, wie im Beispiel unten dargestellt.

## Beispiel: Abrufen von Lastindexwerten

Folgen Sie den Anweisungen unten zum Anzeigen einer Liste von Maschinen mit den berechneten bzw. gemessenen Lastindexwerten und der Anzahl der auf den Maschinen ausgeführten Sitzungen:

1. Starten Sie eine Shell in PowerShell. Weitere Informationen finden Sie unter [XenApp- und XenDesktop-SDK](#).
2. Geben Sie Folgendes ein:

```
Get-BrokerMachine -SessionSupport MultiSession -Property 'DnsName','LoadIndex','SessionCount'
```

Hinweis: Lastindexwerte können bis zu 10.000 betragen. Sie geben VDA-Maschinenlast an, die aus den konfigurierten Quellen, wie Anzahl von Sitzungen, berechnet wurde. Der Wert 10.000 bedeutet, die VDA-Maschine ist vollständig ausgelastet. Der Broker sendet keine weitere Benutzersitzung an die Maschine.

Weitere Informationen und Beispiele finden Sie in der Cmdlet-Hilfe für das Cmdlet `get-brokmachine` und in den Info-Abschnitten, u. a. [about\\_Broker\\_filtering-xd7.html](#). Siehe [Hilfe zu PowerShell-Cmdlets](#).

# Hilfe zu PowerShell-Cmdlets

May 10, 2016

Dieser Abschnitt enthält die PowerShell-Hilfe für alle Cmdlets.

- Citrix.AdIdentity.Admin.V2
- Citrix.AppV.Admin.V1
- Citrix.Broker.Admin.V2
- Citrix.Configuration.Admin.V2
- Citrix.ConfigurationLogging.Admin.V1
- Citrix.DelegatedAdmin.Admin.V1
- Citrix.EnvTest.Admin.V1
- Citrix.Host.Admin.V2
- Citrix.MachineCreation.Admin.V2
- Citrix.Monitor.Admin.V1
- Citrix.Storefront.Admin.V1



# Citrix.AdIdentity.Admin.V2

May 10, 2016

Name	Description
<a href="#">AcctAdIdentitySnapin</a>	The Active Directory Identity Service PowerShell snap-in provides
<a href="#">Acct Filtering</a>	Describes the common filtering options for XenDesktop cmdlets.

Name	Description
<a href="#">Add-AcctADAccount</a>	Import Active Directory computer accounts from Active Directory for use in the AD Identity Service.
<a href="#">Add-AcctIdentityPoolScope</a>	Add the specified IdentityPool(s) to the given scope(s).
<a href="#">Copy-AcctIdentityPool</a>	Copies an Identity Pool and its associated Identities to a new IdentityPool
<a href="#">Get-AcctADAccount</a>	Gets the AD accounts stored in the AD Identity Service.
<a href="#">Get-AcctDBConnection</a>	Gets the database string for the specified data store used by the AdIdentity Service.
<a href="#">Get-AcctDBSchema</a>	Gets a script that creates the AdIdentity Service database schema for the specified data store.
<a href="#">Get-AcctDBVersionChangeScript</a>	Gets a script that updates the AdIdentity Service database schema.
<a href="#">Get-AcctIdentityPool</a>	Gets existing identity pools.
<a href="#">Get-AcctInstalledDBVersion</a>	Gets a list of all available database schema versions for the AdIdentity Service.
<a href="#">Get-AcctScopedObject</a>	Gets the details of the scoped objects for the AdIdentity Service.
<a href="#">Get-AcctService</a>	Gets the service record entries for the AdIdentity Service.
<a href="#">Get-AcctServiceAddedCapability</a>	Gets any added capabilities for the AdIdentity Service on the controller.

Name	Description
<a href="#">Get-AcctServiceInstance</a>	Gets the service instance entries for the AdIdentity Service.
<a href="#">Get-AcctServiceStatus</a>	Gets the current status of the AdIdentity Service on the controller.
<a href="#">New-AcctADAccount</a>	Creates AD computer accounts in the specified identity pool.
<a href="#">New-AcctIdentityPool</a>	Creates a new identity pool.
<a href="#">Remove-AcctADAccount</a>	Removes AD computer accounts from an identity pool.
<a href="#">Remove-AcctIdentityPool</a>	Removes identity pools.
<a href="#">Remove-AcctIdentityPoolMetadata</a>	Removes metadata from the given IdentityPool.
<a href="#">Remove-AcctIdentityPoolScope</a>	Remove the specified IdentityPool(s) from the given scope(s).
<a href="#">Remove-AcctServiceMetadata</a>	Removes metadata from the given Service.
<a href="#">Rename-AcctIdentityPool</a>	Renames an identity pool.
<a href="#">Repair-AcctADAccount</a>	Resets the Active Directory machine password for the given accounts.
<a href="#">Reset-AcctServiceGroupMembership</a>	Reloads the access permissions and configuration service locations for the AdIdentity Service.
<a href="#">Set-AcctDBConnection</a>	Configures a database connection for the AdIdentity Service.
<a href="#">Set-AcctIdentityPool</a>	Update parameters of an identity pool.
<a href="#">Set-AcctIdentityPoolMetadata</a>	Adds or updates metadata on the given IdentityPool.
<a href="#">Set-AcctServiceMetadata</a>	Adds or updates metadata on the given Service.
<a href="#">Test-AcctDBConnection</a>	Tests a database connection for the AdIdentity Service.
<a href="#">Test-AcctIdentityPoolNameAvailable</a>	Checks to ensure that the proposed name for an identity pool is unused.
<a href="#">Unlock-AcctADAccount</a>	Unlocks AD accounts within the AD Identity Service.

Name	Description
<a href="#">Unlock-AcctIdentityPool</a>	Unlocks identity pools.
<a href="#">Update-AcctADAccount</a>	Refreshes the AD computer account state stored in the AD Identity Service.

# about\_AcctAdIdentitySnapin

May 10, 2016

about\_AcctADIdentityServiceSnapin

The Active Directory Identity Service PowerShell snap-in provides administrative functions for the Active Directory Identity Service.

All commands in this snap-in have 'Acct' in their name.

The Active Directory Identity Service PowerShell snap-in enables both local and remote administration of the Active Directory Identity Service. It provides facilities to store details about Active Directory computer accounts that the Machine Creation Service can use.

The snap-in provides two main entities:

## Identity

A representation of an Active Directory computer account that reflects the state of the account within the context of the Machine Creation Service. When an account is created by or imported into the Active Directory Identity Service, the account password is stored. Once the account is consumed by the Machine Creation Service, the password is discarded. For accounts registered with the Active Directory Identity Service, identities hold the following additional state information.

### Available

The Active Directory account is registered with the service, the password for the account is known, and the account is available to be consumed by another service. Accounts that are successfully created with the New-AcctADAccount command or imported using the Add-ADAccount command, are initially assigned this state.

### InUse

The Active Directory account is registered and has been consumed by another service. The password for the account is no longer known to the service.

### Error

The Active Directory account is registered, but is missing,

disabled, or locked within Active Directory. Accounts that are not successfully created with the New-AcctADAccount command or imported using the Add-ADAccount command appear in this state. Use the Update-AcctADAccount and Repair-AcctADAccount commands to resolve issues with accounts in this state.

#### Tainted

The Active Directory account is registered and has been released by all the consuming services, but cannot be made available for use as the password is no longer known. Use the Repair-AcctADAccount command to reset account passwords and restore the account state to 'Available'.

Identities can also be marked as 'Locked' by the Machine Creation Service to indicate that they are in use and must not be changed. These services are also responsible for unlocking the Active Directory accounts when they no longer require exclusive access. Use the Unlock-AcctADAccount command to allow the lock to be overridden, if necessary.

#### Identity Pool

Containers for identities that can be configured with all the information required for new Active Directory accounts to be created. Alternatively, identity pools can be populated by importing accounts that already exist in Active Directory. All accounts registered with the Active Directory Identity Service must be placed into one of these containers. An identity can belong to more than one identity pool, but the state of the identity cannot be different in each pool. For example, an identity that is in use will be marked 'InUse' in all the identity pools of which it is part.

To avoid conflicting changes, identity pools can also be marked as 'Locked'

operations are also responsible for unlocking the identity pool. Use the unlock-AcctIdentityPool command to allow the lock to be overridden, if necessary.

#### Account Creation (using the New-AcctADAccount command)

To use PowerShell to create new Active Directory accounts, the runspace

must be run using an account with sufficient permissions in the required Active Directory container (specified by the identity pool organizational unit parameter) for accounts to be created.

Import Accounts (using the Add-AcctADAccount command)

There are two modes for this operation: situations where the Active Directory account passwords are known and situations where the passwords are not known.

If the account passwords are known, the accounts can be imported without the need for administrative permissions in Active Directory. The accounts are imported and the password provided is used to change the existing password.

If the passwords are not known, the runspace must be run using an account that has permissions to reset the password for the accounts.

# about\_Acct\_Filtering

May 10, 2016

## XenDesktop - Advanced Dataset Filtering

Describes the common filtering options for XenDesktop cmdlets.

Some cmdlets operate on large quantities of data and, to reduce the overhead of sending all of that data over the network, many of the Get- cmdlets support server-side filtering of the results.

The conventional way of filtering results in PowerShell is to pipeline them into Where-Object, Select-Object, and Sort-Object, for example:

```
Get-<Noun> | Where { $_.Size = 'Small' } | Sort 'Date' | Select -First 10
```

However, for most XenDesktop cmdlets the data is stored remotely and it would be slow and inefficient to retrieve large amounts of data over the network and then discard most of it. Instead, many of the Get- cmdlets provide filtering parameters that allow results to be processed on the server, returning only the required results.

You can filter results by most object properties using parameters derived from the property name. You can also sort results or limit them to a specified number of records:

```
Get-<Noun> -Size 'Small' -SortBy 'Date' -MaxRecordCount 10
```

You can express more complex filter conditions using a syntax and set of operators very similar to those used by PowerShell expressions.

Those cmdlets that support filtering have the following common parameters:

-MaxRecordCount <int>

Specifies the maximum number of results to return.  
For example, to return only the first nine results use:

```
Get-<Noun> -MaxRecordCount 9
```

If not specified, only the first 250 records are returned, and if more are available, a warning is produced:

WARNING: Only first 250 records returned. Use -MaxRecordCount to retrieve more.

You can suppress this warning by using -WarningAction or by specifying a value for -MaxRecordCount.

To retrieve all records, specify a large number for -MaxRecordCount. As the value is an integer, you can use the following:

```
Get-<Noun> -MaxRecordCount [int]::MaxValue
```

-ReturnTotalRecordCount [<SwitchParameter>]

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. For example:

```
Get-<Noun> -MaxRecordCount 9 -ReturnTotalRecordCount
```

....

```
Get-<Noun> : Returned 9 of 10 items
```

```
At line:1 char:18
```

```
+ Get-<Noun> <<<< -MaxRecordCount 9 -ReturnTotalRecordCount
```

```
+ CategoryInfo : OperationStopped: (:) [Get-<Noun>], PartialDataException
```

```
+ FullyQualifiedErrorId : PartialData,Citrix.<SDKName>.SDK.Get<Noun>
```

The count can be accessed using the TotalAvailableResultCount property:

```
$count = $error[0].TotalAvailableResultCount
```

-Skip <int>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

-SortBy <string>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order, respectively. Ascending order is assumed if no prefix is present.



Sorting occurs before -MaxRecordCount and -Skip parameters are applied. For example, to sort by Name and then by Count (largest first) use:

```
-SortBy 'Name,-Count'
```

By default, sorting by an enumeration property uses the numeric value of the elements. You can specify a different sort order by qualifying the name with an ordered list of elements or their numeric values, or <null> to indicate the placement of null values.

Elements not mentioned are placed at the end in their numeric order. For example, to sort by two different enums and then by the object id:

```
-SortBy 'MyState(StateC,<null>,StateA,StateB),Another(0,3,2,1),Id'
```

-Filter <String>

This parameter lets you specify advanced filter expressions, and supports combination of conditions with -and and -or, and grouping with braces. For example:

```
Get-<Noun> -Filter 'Name -like "High*" -or (Priority -eq 1 -and Severity -ge 2)'
```

The syntax is close enough to PowerShell syntax that you can use script blocks in most cases. This can be easier to read as it reduces quoting:

```
Get-<Noun> -Filter { Count -ne $null }
```

The full -Filter syntax is provided below.

Filtering by strings performs a case-insensitive wildcard match.

Separate parameters are combined with an implicit -and operator.

Normal PowerShell quoting rules apply, so you can use single or double quotes, and omit the quotes altogether for many strings. The order of parameters does not make any difference. The following are equivalent:

```
Get-<Noun> -Company Citrix -Product Xen*
Get-<Noun> -Company "citrix" -Product '[X]EN*'
Get-<Noun> -Product "Xen*" -Company "CITRIX"
Get-<Noun> -Filter { Company -eq 'Citrix' -and Product -like 'Xen*' }
```

See [about\\_Quoting\\_Rules](#) and [about\\_Wildcards](#) for details about PowerShell handling of quotes and wildcards.

To avoid wildcard matching or include quote characters, you can escape the wildcards using the normal PowerShell escape mechanisms (see [about\\_Escape\\_Characters](#)), or switch to a filter expression and the `-eq` operator:

```
Get-<Noun> -Company "Abc[*]" # Matches Abc*
Get-<Noun> -Company "Abc`*" # Matches Abc*
Get-<Noun> -Filter { Company -eq "Abc*" } # Matches Abc*
Get-<Noun> -Filter { Company -eq "A`"B`C`" } # Matches A"B'C
```

Simple filtering by numbers, booleans, and TimeSpans perform direct equality comparisons, although if the value is nullable you can also search for null values. Here are some examples:

```
Get-<Noun> -Uid 123
Get-<Noun> -Enabled $true
Get-<Noun> -Duration 1:30:40
Get-<Noun> -NullableProperty $null
```

More comparisons are possible using advanced filtering with `-Filter`:

```
Get-<Noun> -Filter 'Capacity -ge 10gb'
Get-<Noun> -Filter 'Age -ge 20 -and Age -lt 40'
Get-<Noun> -Filter 'VolumeLevel -like "[123]"'
Get-<Noun> -Filter 'Enabled -ne $false'
Get-<Noun> -Filter 'NullableProperty -ne $null'
```

You can check boolean values without an explicit comparison operator, and you can also combine them with `-not`:

```
Get-<Noun> -Filter 'Enabled' # Equivalent to 'Enabled -eq $true'
Get-<Noun> -Filter '-not Enabled' # Equivalent to 'Enabled -eq $false'
```

See [about\\_Comparison\\_Operators](#) for an explanation of the operators, but note that only a subset of PowerShell operators are supported (`-eq`, `-ne`, `-gt`, `-ge`, `-lt`, `-le`, `-like`, `-notlike`, `-in`, `-notin`, `-contains`, `-notcontains`).

Enumeration values can either be specified using typed values or the string name of the enumeration value:

```
Get-<Noun> -Shape [Shapes]::Square
Get-<Noun> -Shape Circle
```

With filter expressions, typed values can be specified with simple variables or quoted strings. They also support enumerations with wildcards:

```
$s = [Shapes]::Square
Get-<Noun> -Filter { Shape -eq $s -or Shape -eq "Circle" }
Get-<Noun> -Filter { Shape -like 'C*' }
```

By their nature, floating point values, DateTime values, and TimeSpan values are best suited to relative comparisons rather than just equality. DateTime strings are converted using the locale and time zone of the user device, but you can use ISO8601 format strings (YYYY-MM-DDThh:mm:ss.sTZD) to avoid ambiguity. You can also use standard PowerShell syntax to create these values:

```
Get-<Noun> -Filter { StartTime -ge "2010-08-23T12:30:00.0Z" }
$d = [DateTime]"2010-08-23T12:30:00.0Z"
Get-<Noun> -Filter { StartTime -ge $d }
$d = (Get-Date).AddDays(-1)
Get-<Noun> -Filter { StartTime -ge $d }
```

Relative times are quite common and, when using filter expressions, you can also specify DateTime values using a relative format:

```
Get-<Noun> -Filter { StartTime -ge '-2' } # Two days ago
Get-<Noun> -Filter { StartTime -ge '-1:30' } # Hour and a half ago
Get-<Noun> -Filter { StartTime -ge '-0:0:30' } # 30 seconds ago
```

## ARRAY PROPERTIES

When filtering against list or array properties, simple parameters perform a case-insensitive wildcard match against each of the members. With filter expressions, you can use the -contains and -notcontains operators. Unlike PowerShell, these perform wildcard matching on strings.

Note that for array properties the naming convention is for the returned property to be plural, but the parameter used to search for any match is singular. The following are equivalent (assuming Users is an array property):

```
Get-<Noun> -User Fred*
Get-<Noun> -Filter { User -like "Fred*" }
Get-<Noun> -Filter { Users -contains "Fred*" }
```

You can also use the singular form with -Filter to search using other operators:

```
Match if any user in the list is called "Frederick"
Get-<Noun> -Filter { User -eq "Frederick" }
Match if any user in the list has a name alphabetically below 'F'
Get-<Noun> -Filter { User -lt 'F' }
```

## COMPLEX EXPRESSIONS

When matching against multiple values, you can use a sequence of comparisons joined with -or operators, or you can use -in and -notin:

```
Get-<Noun> -Filter { Shape -eq 'Circle' -or Shape -eq 'Square' }
$shapes = 'Circle','Square'
Get-<Noun> -Filter { Shape -in $shapes }
$sides = 1..4
Get-<Noun> -Filter { Sides -notin $sides }
```

Braces can be used to group complex expressions, and override the default left-to-right evaluation of -and and -or. You can also use -not to invert the sense of any sub-expression:

```
Get-<Noun> -Filter { Size -gt 4 -or (Color -eq 'Blue' -and Shape -eq 'Circle') }
Get-<Noun> -Filter { Sides -lt 5 -and -not (Color -eq 'Blue' -and Shape -eq 'Circle') }
```

## PAGING

The simplest way to page through data is to use the -Skip and -MaxRecordCount parameters. So, to read the first three pages of data with 10 records per page, use:

```
Get-<Noun> -Skip 0 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 10 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 20 -MaxRecordCount 10 <other filtering criteria>
```

You must include the same filtering criteria on each call, and ensure that the data is sorted consistently.

The above approach is often acceptable, but as each call performs an independent query, data changes can result in records being skipped or appearing twice. One approach to improve this is to sort by a unique id field and then start the search for the next page at the unique id after the last unique id of the previous page. For example:

```
Get the first page
Get-<Noun> -MaxRecordCount 10 -SortBy SerialNumber

SerialNumber ...

A120004
A120007
... 7 other records ...
A120900

Get the next page
Get-<Noun> -MaxRecordCount 10 -Filter { FirstName -gt 'A120900' }

SerialNumber ...
```

-----  
A120901  
B220000  
...

<Filter> ::= <ScriptBlock> | <ComponentList>

<ScriptBlock> ::= "{" <ComponentList> "}"

<ComponentList> ::= <Component> <AndOrOperator> <ComponentList> |

<Component>

<Component> ::= <NotOperator> <Factor> |

<Factor>

<Factor> ::= "(" <ComponentList> ")" |

<PropertyName> <ComparisonOperator> <Value> |  
<PropertyName>

<AndOrOperator> ::= "-and" | "-or"

<NotOperator> ::= "-not" | "!"

<ComparisonOperator>

::= "-eq" | "-ne" | "-le" | "-ge" | "-lt" | "-gt" |  
"-like" | "-notlike" | "-contains" | "-notcontains" |  
"-in" | "-notin"

<PropertyName> ::= <simple name of property>

<Value> ::= <string literal> | <numeric literal> |

<scalar variable> | <array variable> |  
"\$null" | "\$true" | "\$false"

Numeric literals support decimal and hexadecimal literals, with optional multiplier suffixes (kb, mb, gb, tb, pb).

Dates and times can be specified as string literals. The current culture determines what formats are accepted. To avoid any ambiguity, use strings formatted to the ISO8601 standard. If not specified, the current time zone is used.

Relative date-time string literals are also supported, using a minus sign followed by a TimeSpan. For example, "-1:30" means 1 hour and 30 minutes ago.

# Add-AcctADAccount

May 10, 2016

Import Active Directory computer accounts from Active Directory for use in the AD Identity Service.

```
Add-AcctADAccount [-IdentityPoolName] <String> -ADAccountName <String[]> [-Password <String>] [-SecurePassword
<SecureString>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Add-AcctADAccount -IdentityPoolUid <Guid> -ADAccountName <String[]> [-Password <String>] [-SecurePassword <SecureString>
[-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

Provides the ability for Active Directory computer accounts that already exist in Active Directory to be used by the Citrix AD Identity Service and the other Citrix Machine Creation Services.

All aspects of this command that need to make modifications to the accounts in Active Directory will do so using the account that the PowerShell runspace is using. This means that if the passwords need resetting for the accounts, the user performing the operation in PowerShell must have sufficient privileges in Active Directory for this operation to complete successfully.

The following rules apply to the importing of Active Directory accounts; If the current account password is supplied, the cmdlet will attempt to change the password so that it is known only to the Citrix Identity Service. This uses password change operations and does not need AD account administration permissions. If the current password is not supplied, the cmdlet will attempt to reset the password for the Active Directory account so that it is known only to the Citrix Identity Service. This requires the cmdlet to have enough privileges in Active Directory for the accounts' password reset to be available. Imported accounts in a disabled or locked state in Active Directory are imported with the account marked in an error state. If the identity pool into which the account is being imported does not have a domain set, it assumes the domain of the first account imported into it.

[New-AcctADAccount](#)

[Remove-AcctADAccount](#)

[Repair-AcctADAccount](#)

[Get-AcctADAccount](#)

[Update-AcctADAccount](#)

[Unlock-AcctADAccount](#)

**-IdentityPoolName**<String>

The identity pool name into which to add the imported accounts.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-ADAccountName**<String[]>

The Active Directory account name to be imported.

Active Directory accounts are accepted in the following formats: Fully qualified DN e.g.

CN=MyComputer,OU=Computers,DC=MyDomain,DC=Com; UPN format e.g MyComputer@MyDomain.Com; Domain qualified e.g MyDomain\MyComputer.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-IdentityPoolUid**<Guid>

The unique identifier for the identity pool to which imported accounts are to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Password**<String>

The current password for the computer account.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecurePassword**<SecureString>

The current password for the account (provided in a Secure String class).

Required?	false
Default Value	
Accept Pipeline Input?	



Accept Pipeline Input?	false
------------------------	-------

#### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### **Citrix.AdIdentity.Sdk.AccountOperationDetailedSummary**

The Add-AcctADAccount returns an object that contains the following parameters;

SuccessfulAccountsCount <int>

The number of accounts that were added successfully

FailedAccountsCount <int>

The number of accounts that were not added.

FailedAccounts <Citrix.XDPowerShell.AccountError[]>

The list of accounts that failed to be added. Each one has the following properties;

ADAccountName <string>

ADAccountSid <String>

ErrorReason <AdIdentityStatus>

This can be one of the following

UnableToConvertDomain

UnableToAccessAccountProperties

IdentityNotLocatedInDomain

UnableToAccessAccountProperties

IdentityDuplicateObjectExists

IdentityObjectLocked

IdentityObjectInUse

FailedToConnectToDomainController

FailedToExecuteSearchInAD

FailedToAccessComputerAccountInAD

FailedToSetPasswordInAD

FailedToChangePasswordInAD

ADServiceDatabaseError

ADServiceDatabaseNotConfigured

ADServiceStatusInvalidDb

DiagnosticInformation <Exception>

Any other error information

SuccessfulAccounts <Citrix.AdIdentity.Sdk.Identity[]>

The list of accounts that were successfully added. Each one has the following properties;

ADAccountSid <string>

The AD account SID for the imported account.

ADAccountName <string>

The AD account name for the imported account.

Domain <string>

The domain for the imported account.

State <Citrix.AdIdentity.Sdk.ADIdentityState>

The state for the account. This can be;

Available

The account is not used.

InUse

The account is in use.

#### Error

The account is in error (i.e. the account is locked or disabled in AD).

#### Tainted

The account is no longer used, but the password is no longer known.

#### Lock <Boolean>

The account is locked (in the database not in AD).

To maintain maximum security when using the command programmatically, Citrix recommends you use the 'SecurePassword' property instead of the 'Password' property.

In the case of failure, the following errors can result.

#### Error Codes

-----

##### IdentityPoolAlreadyLocked

The specified identity pool was locked by another operation.

##### IdentityPoolNotFound

The specified identity pool was not found.

##### IdentityDuplicateObjectExists

The specified AD account already exists for the identity pool.

##### PermissionDenied

The user does not have administrative rights to perform this operation.

##### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

##### DatabaseError

An error occurred in the service while attempting a database operation.

##### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

##### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

##### CommunicationError

An error occurred while communicating with the service.

ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used, or examine the XenDesktop logs.

----- EXAMPLE 1 -----

C:\PS>Add-AcctADAccount -IdentityPoolName MyPool -ADAccountName "Domain\account","Domain\account2" -OutVariable result

SuccessfulAccounts	SuccessfulAccountsCount	FailedAccountsCount	FailedAccounts
-----	-----	-----	-----
{domain\account, domain\account2}	2	0	{}

```
$result[0].SuccessfulAccounts

ADAccountSid : S-1-5-21-1315084875-1285793635-2418178940-2644
ADAccountName : domain\account
Domain : Domain.com
State : Available
Lock : False

ADAccountSid : S-1-5-21-1315084875-1285793635-2418178940-2645
ADAccountName : domain\account2
Domain : Domain.com
State : Available
Lock : False
```

Import the two accounts (account and account2) from AD into the identity Pool called "MyPool"

# Add-AcctIdentityPoolScope

May 10, 2016

Add the specified IdentityPool(s) to the given scope(s).

```
Add-AcctIdentityPoolScope [-Scope] <String[]> -InputObject <IdentityPool[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Add-AcctIdentityPoolScope [-Scope] <String[]> -IdentityPoolUid <Guid[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Add-AcctIdentityPoolScope [-Scope] <String[]> -IdentityPoolName <String[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

The AddAcctIdentityPoolScope cmdlet is used to associate one or more IdentityPool objects with given scope(s).

There are multiple parameter sets for this cmdlet, allowing you to identify the IdentityPool objects in different ways:

- IdentityPool objects can be piped in or specified by the InputObject parameter
- The IdentityPoolUid parameter specifies objects by IdentityPoolUid
- The IdentityPoolName parameter specifies objects by IdentityPoolName (supports wildcards)

To add a IdentityPool to a scope you need permission to change the scopes of the IdentityPool and permission to add objects to all of the scopes you have specified.

If the IdentityPool is already in a scope, that scope will be silently ignored.

[Remove-AcctIdentityPoolScope](#)

[Get-AcctScopedObject](#)

**-Scope**<String[]>

Specifies the scopes to add the objects to.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-InputObject<IdentityPool[]>**

Specifies the IdentityPool objects to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

### **-IdentityPoolUid<Guid[]>**

Specifies the IdentityPool objects to be added by IdentityPoolUid.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

### **-IdentityPoolName<String[]>**

Specifies the IdentityPool objects to be added by IdentityPoolName.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

None

If the command fails, the following errors can be returned.

### Error Codes

-----

#### UnknownObject

One of the specified objects was not found.

#### ScopeNotFound

One of the specified scopes was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command with the specified objects or scopes.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

### ----- EXAMPLE 1 -----

```
c:\PS>Add-AcctIdentityPoolScope Finance -IdentityPoolUid 6702C5D0-C073-4080-A0EE-EC74CB537C52
```

Adds a single IdentityPool to the 'Finance' scope.

### ----- EXAMPLE 2 -----

```
c:\PS>Add-AcctIdentityPoolScope Finance,Marketing -IdentityPoolUid 6702C5D0-C073-4080-A0EE-EC74CB537C52
```

Adds a single IdentityPool to the multiple scopes.

### ----- EXAMPLE 3 -----

```
c:\PS>Get-AcctIdentityPool | Add-AcctIdentityPoolScope Finance
```

Adds all visible IdentityPool objects to the 'Finance' scope.

### ----- EXAMPLE 4 -----

```
c:\PS>Add-AcctIdentityPoolScope Finance -IdentityPoolName A*
```

Adds IdentityPool objects with a name starting with an 'A' to the 'Finance' scope.



# Copy-AcctIdentityPool

May 10, 2016

Copies an Identity Pool and its associated Identities to a new IdentityPool

Copy-AcctIdentityPool [-IdentityPoolName] <String> [-NewIdentityPoolName] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Copy-AcctIdentityPool -IdentityPoolUid <Guid> [-NewIdentityPoolName] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Provides the ability to copy an IdentityPool.

The new IdentityPool will contain all the accounts that were in the original pool and will have the same domain and OU set. The naming scheme will be unset and the StartCount will be set to 1.

[New-AcctIdentityPool](#)

[Get-AcctIdentityPool](#)

[Remove-AcctIdentityPool](#)

**-IdentityPoolName**<String>

The name of the identity pool that is to be copied.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-IdentityPoolUid**<Guid>

The unique identifier for the identity pool that is to be copied.

Required?	true
Default Value	

Accept Pipeline Input?	false
------------------------	-------

**-NewIdentityPoolName<String>**

The name for the new IdentityPool.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

**Citrix.AdIdentity.Sdk.IdentityPool**

This object provides details of the new identity pool and contains the following information:

**IdentityPoolName <string>**

The name of the identity pool.

IdentityPoolUid <Guid>

The unique identifier for the identity pool.

NamingScheme <string>

The naming scheme for the identity pool.

NamingSchemeType <Citrix.XDInterServiceTypes.ADIdentityNamingScheme>

The naming scheme type for the identity pool. This can be one of the following:

Numeric - naming scheme uses numeric indexes

Alphabetic - naming scheme uses alphabetic indexes

StartCount <int>

The next index to be used when creating an identity from the identity pool.

OU <string>

The Active Directory distinguished name for the OU in which accounts created from this identity pool will be created.

Domain <string>

The Active Directory domain that accounts in the pool belong to.

Lock <Boolean>

Indicates whether the identity pool is locked.

In the case of failure, the following errors can result.

Error Codes

-----

IdentityPoolDuplicateObjectExists

An identity pool with the same name exists already.

IdentityPoolObjectNotFound

The identity pool to be modified could not be located.

PermissionDenied

The user does not have administrative rights to perform this operation.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

# Get-AcctADAccount

May 10, 2016

Gets the AD accounts stored in the AD Identity Service.

## Syntax

```
Get-AcctADAccount [-IdentityPoolName <String>] [-ADAccountSid <String>] [-Domain <String>] [-State <ADIdentityState>] [-Lock <Boolean>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-AcctADAccount [-IdentityPoolUid <Guid>] [-ADAccountSid <String>] [-Domain <String>] [-State <ADIdentityState>] [-Lock <Boolean>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to locate the AD accounts stored within the AD Identity Service and view the state of the accounts.

## Related topics

[New-AcctADAccount](#)

[Add-AcctADAccount](#)

[Remove-AcctADAccount](#)

[Unlock-AcctADAccount](#)

[Update-AcctADAccount](#)

[Repair-AcctADAccount](#)

## Parameters

**-ADAccountSid**<String>

The AD Account SID of the account.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Domain**<String>

The domain of the account (this is in dns format).

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-State**<ADIdentityState>

The current state of the identity stored in the AD Identity Service for the AD account.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Lock**<Boolean>

Indicates if the account is locked in the AD Identity Service.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

See about\_Acct\_Filtering for details.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

See about\_Acct\_Filtering for details.

Required?	false

Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

See about\_Acct\_Filtering for details.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

See about\_Acct\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Filter**<String>

See about\_Acct\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

#### **-IdentityPoolName**<String>

The name of the identity pool to which the account is registered.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-IdentityPoolUid**<Guid>

The unique identifier for the identity pool that the account is registered to.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### Return Values

Citrix.AdIdentity.Sdk.IdentityInPool

The Get-AcctADAccount returns an object that contains the following parameters

ADAccountSid <string>

The AD account SID for the retrieved account.

ADAccountName <string>

The AD account name for the retrieved account.

Domain <string>

The domain for the imported account.

State <Citrix.XDInterServiceTypes.ADIdentityState>

The state for the account. This can be;

Available

The account is not used.



## InUse

The account is in use.

## Error

The account is in error (i.e. the account is locked or disabled in AD).

## Tainted

The account is no longer used, but the password is no longer known.

## Lock <Boolean>

The account is locked (in the database not in AD).

## IdentityPoolName <System.String>

The name of the containing identity pool.

## IdentityPoolUid <System.Guid>

The GUID identifying the containing identity pool.

## Notes

In the case of failure the following errors can result.

## Error Codes

-----

## PartialData

Only a subset of the available data was returned.

## CouldNotQueryDatabase

The query required to get the database was not defined.

## PermissionDenied

The user does not have administrative rights to perform this operation.

## ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

## CommunicationError

An error occurred while communicating with the service.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

## ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

C:\>Get-AcctADAccount

Return all the AD accounts that are registered in the AD Identity Service.

### ----- EXAMPLE 2 -----

C:\>Get-AcctADAccount -IdentityPoolName MyPool -Lock \$false

Return all the AD accounts that are registered in the AD Identity Service in the identity pool called "MyPool" that are also locked.

### ----- EXAMPLE 3 -----

C:\>Get-AcctADAccount -Filter {IdentityPoolName -Like "p\*" -or IdentityPoolName -eq "MyPool"}

Return all the AD accounts that are registered in the AD Identity Service in the identity pool called "MyPool" or in an identity pool that has a name that starts with a 'p'. For full details of the advanced filtering aspects of this command see about\_Acct\_Filtering.

# Get-AcctDBConnection

May 10, 2016

Gets the database string for the specified data store used by the AdIdentity Service.

## Syntax

```
Get-AcctDBConnection [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the database connection string for the specified data store.

If the returned string is blank, no valid connection string has been specified. In this case the service is running, but is idle and awaiting specification of a valid connection string.

## Related topics

[Get-AcctServiceStatus](#)

[Set-AcctDBConnection](#)

[Test-AcctDBConnection](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

system.string

The database connection string configured for the AdIdentity Service.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

NoDBConnections

The database connection string for the AdIdentity Service has not been specified.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Get-AcctDBConnection

Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True

Get the database connection string for the AdIdentity Service.

# Get-AcctDBSchema

May 10, 2016

Gets a script that creates the AdIdentity Service database schema for the specified data store.

## Syntax

```
Get-AcctDBSchema [-DatabaseName <String>] [-ServiceGroupName <String>] [-ScriptType <ScriptTypes>] [-LocalDatabase]
[-Sid <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets SQL scripts that can be used to create a new AdIdentity Service database schema, add a new AdIdentity Service to an existing site, remove a AdIdentity Service from a site, or create a database server logon for a AdIdentity Service. If no Sid parameter is provided, the scripts obtained relate to the currently selected AdIdentity Service instance, otherwise the scripts relate to AdIdentity Service instance running on the machine identified by the Sid provided. When obtaining the Evict script, a Sid parameter must be supplied. The current service instance is that on the local machine, or that explicitly specified by the last usage of the -AdminAddress parameter to a AdIdentity SDK cmdlet. The service instance used to obtain the scripts does not need to be a member of a site or to have had its database connection configured. The database scripts support only Microsoft SQL Server, or SQL Server Express, and require Windows integrated authentication to be used. They can be run using SQL Server's SQLCMD utility, or by copying the script into an SQL Server Management Studio (SSMS) query window and executing the query. If using SSMS, the query must be executed in 'SMDCMD mode'. The ScriptType parameter determines which script is obtained. If ScriptType is not specified, or is FullDatabase, the script contains:

- o Creation of service schema
- o Creation of database server logon
- o Creation of database user
- o Addition of database user to AdIdentity Service roles

If ScriptType is Instance, the returned script contains:

- o Creation of database server logon
- o Creation of database user
- o Addition of database user to AdIdentity Service roles

If ScriptType is Evict, the returned script contains:

- o Removal of AdIdentity Service instance from database
- o Removal of database user

If ScriptType is Login, the returned script contains:

- o Creation of database server logon only

If the service uses two data stores they can exist in the same database. You do not need to configure a database before using this command.

## Related topics

[Set-AcctDBConnection](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database for which the schema will be generated.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ServiceGroupName**<String>

Specifies the name of the service group to be used when creating the database schema. The service group is a collection of all the AdIdentity services that share the same database instance and are considered equivalent; that is, all the services within a service group can be used interchangeably.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ScriptType**<ScriptTypes>

Specifies the type of database script returned. Available script types are:

### Database

Returns a full database script that can be used to create a database schema for the AdIdentity Service in a database instance that does not already contain a schema for this service. The DatabaseName and ServiceGroupName parameters must be specified to create a script of this type.

### Instance

Returns a permissions script that can be used to add further AdIdentity services to an existing database instance that already contains the full AdIdentity service schema, associating the services to the Service Group. The Sid parameter can optionally be specified to create a script of this type.

### Login

Returns a database logon script that can be used to add the required logon accounts to an existing database instance that contains the AdIdentity Service schema. This is used primarily when creating a mirrored database environment. The

DatabaseName parameter must be specified to create a script of this type.

#### Evict

Returns a script that can be used to remove the specified AdIdentity Service from the database entirely. The DatabaseName and Sid parameters must be specified to create a script of this type.

Required?	false
Default Value	Database
Accept Pipeline Input?	false

#### -LocalDatabase<SwitchParameter>

Specifies whether the database script is to be used in a database instance run on the same controller as other services in the service group. Including this parameter ensures the script creates only the required permissions for local services to access the database schema for AdIdentity services.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -Sid<String>

Specifies the SID of the controller on which the AdIdentity Service instance to remove from the database is running.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

#### -AdminAddress<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

### System.string

A string containing the required SQL script for application to a database.

## Notes

The scripts returned support Microsoft SQL Server Express Edition, Microsoft SQL Server Standard Edition, and Microsoft SQL Server Enterprise Edition databases only, and are generated on the assumption that integrated authentication will be used.

If the ScriptType parameter is not included or set to 'FullDatabase', the full database script is returned, which will:

Create the database schema.

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist).

If the ScriptType parameter is set to 'Instance', the script will:

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist) and associate it with a user.

If the ScriptType parameter is set to 'Login', the script will:

Create the logon (providing the schema does not already exist) and associate it with a pre-existing user of the same name.

If the LocalDatabase parameter is included, the NetworkService account will be added to the list of accounts permitted to access the database. This is required only if the database is run on a controller.

If the command fails, the following errors can be returned.

## Error Codes

-----

### GetSchemasFailed

The database schema could not be found.

### ActiveDirectoryAccountResolutionFailed

The specified Active Directory account or Group could not be found.

### DatabaseError

An error occurred in the service while attempting a database operation.



#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Get-AcctDBSchema -DatabaseName MyDB -ServiceGroupName MyServiceGroup > c:\AcctSchema.sql
```

Get the full database schema for site data store of the AdIdentity Service and copy it to a file called 'c:\AcctSchema.sql'.

This script can then be used to create the schema in a pre-existing database named 'MyDB' that does not already contain a AdIdentity Service site schema.

##### ----- EXAMPLE 2 -----

```
c:\PS>Get-AcctDBSchema -DatabaseName MyDB -scriptType Login > c:\AdIdentityLogins.sql
```

Get the logon scripts for the AdIdentity Service.

# Get-AcctDBVersionChangeScript

May 10, 2016

Gets a script that updates the AdIdentity Service database schema.

## Syntax

```
Get-AcctDBVersionChangeScript -DatabaseName <String> -TargetVersion <Version> [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Returns a database script that can be used to upgrade or downgrade the site or secondary schema for the AdIdentity Service from the current schema version to a different version.

## Related topics

[Get-AcctInstalledDBVersion](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database instance to which the update applies.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-TargetVersion**<Version>

Specifies the version of the database you want to update to.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.Management.Automation.PSObject

A PSObject containing the required SQL script for application to a database.

## Notes

The PSObject returned by this cmdlet contains the following properties:

-- Script The raw text of the SQL script to apply the update, or null in the case when no upgrade path to the specified target version exists.

-- NeedExclusiveAccess Indicates whether all services in the service group must be shut down during the update or not.

-- CanUndo Indicates whether the generated script allows the updated schema to be reverted to the state prior to the update.

Scripts to update the schema version are stored in the database so any service in the service group can obtain these scripts. Extreme caution should be exercised when using update scripts. Citrix recommends backing up the database before attempting to upgrade the schema. Database update scripts may require exclusive use of the schema and so may not be able to execute while any AdIdentity services are running. However, this depends on the specific update being carried out.

After a schema update has been carried out, services that require the previous version of the schema may cease to operate. The ServiceState parameter reported by the Get-AcctServiceStatus command provides information about service compatibility. For example, if the schema has been upgraded to a more recent version that a service cannot use, the service reports "DBNewerVersionThanService".

If the command fails, the following errors can be returned.

## Error Codes

-----

### NoOp

The operation was successful but had no effect.

### NoDBConnections

The database connection string for the AdIdentity Service has not been specified.

### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> $update = Get-AcctDBVersionChangeScript -DatabaseName MyDb -TargetVersion 1.0.75.0
```

```
C:\PS> $update.Script > update_75.sql
```

Gets an SQL update script to update the current schema to version 1.0.75.0. The resulting update\_75.sql script is suitable for direct use with the SQL Server SQLCMD utility.

# Get-AcctIdentityPool

May 10, 2016

Gets existing identity pools.

## Syntax

```
Get-AcctIdentityPool [-IdentityPoolName <String>] [-IdentityPoolUid <Guid>] [-Lock <Boolean>] [-Scopeld <Guid>] [-ScopeName <String>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to locate existing identity pools.

## Related topics

[New-AcctIdentityPool](#)

[Remove-AcctIdentityPool](#)

[Rename-AcctIdentityPool](#)

[Set-AcctIdentityPool](#)

## Parameters

**-IdentityPoolName**<String>

The name of the identity pool.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IdentityPoolUid**<Guid>

The unique identifier for the identity pool.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Lock**<Boolean>

Whether the identity pool is locked or not.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ScopeId**<Guid>

Gets only results with a scope matching the specified scope identifier.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ScopeName**<String>

Gets only results with a scope matching the specified scope name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount**<SwitchParameter>

See about\_Acct\_Filtering for details.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-MaxRecordCount**<Int32>

See about\_Acct\_Filtering for details.

Required?	false
Default Value	250
Accept Pipeline Input?	false

**-Skip**<Int32>

See about\_Acct\_Filtering for details.

Required?	false
Default Value	0
Accept Pipeline Input?	false

**-SortBy**<String>

See about\_Acct\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Filter**<String>

See about\_Acct\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host

name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Return Values

`Citrix.AdIdentity.Sdk.IdentityPool`

This object provides details of the identity pool and contains the following information:

`IdentityPoolName <string>`

The name of the identity pool.

`IdentityPoolUid <Guid>`

The unique identifier for the identity pool.

`NamingScheme <string>`

The naming scheme for the identity pool.

`NamingSchemeType <Citrix.XDInterServiceTypes.ADIdentityNamingScheme>`

The naming scheme type for the identity pool. This can be one of the following:

Numeric - naming scheme uses numeric indexes

Alphabetic - naming scheme uses alphabetic indexes

`StartCount <int>`

The next index to be used when creating an identity from the identity pool.

`OU <string>`

The Active Directory distinguished name for the OU in which accounts created from this identity pool will be created.

`Domain <string>`

The Active Directory domain that accounts in the pool belong to.

`Lock <Boolean>`

Indicates if the identity pool is locked.

## Notes

In the case of failure, the following errors can result.



## Error Codes

-----

### PartialData

Only a subset of the available data was returned.

### CouldNotQueryDatabase

The query required to get the database was not defined.

### PermissionDenied

The user does not have administrative rights to perform this operation.

### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

### CommunicationError

An error occurred while communicating with the service.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

### InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS>Get-AcctIdentityPool
```

```
IdentityPoolName : MyPool
```

```
IdentityPoolUid : 22072d9e-6a8f-494b-a5bc-2ef18ca4b915
```

```
NamingScheme : Acc####
```

```
NamingSchemeType : Numeric
```

```
StartCount : 1
```

```
OU :
```

```
Domain : mydomain.com
```

```
Lock : True
```

```
IdentityPoolName : MyPool2
```

```
IdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16
```

NamingScheme : Test####  
NamingSchemeType : Alphabetic  
StartCount : 1  
OU :  
Domain : mydomain.com  
Lock : False  
Gets all the identity pools.

----- **EXAMPLE 2** -----

C:\PS>Get-AcctIdentityPool -IdentityPoolName M\*

IdentityPoolName : MyPool  
IdentityPoolUid : 22072d9e-6a8f-494b-a5bc-2ef18ca4b915  
NamingScheme : Acc####  
NamingSchemeType : Numeric  
StartCount : 1  
OU :  
Domain : mydomain.com  
Lock : True  
Gets all the identity pools beginning with the character 'M'.

# Get-AcctInstalledDBVersion

May 10, 2016

Gets a list of all available database schema versions for the AdIdentity Service.

## Syntax

Get-AcctInstalledDBVersion [-Upgrade] [-Downgrade] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Returns the current version of the AdIdentity Service database schema, if no flags are set, otherwise returns versions for which upgrade or downgrade scripts are available and have been stored in the database.

## Related topics

## Parameters

**-Upgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be updated should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Downgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be reverted should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

### System.Version

The Get-AcctInstalledDbVersion command returns objects containing the new definition of the AdIdentity Service database schema version.

Major <Integer>

Minor <Integer>

Build <Integer>

Revision <Integer>

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

Both the Upgrade and Downgrade flags were specified.

#### NoOp

The operation was successful but had no effect.

#### NoDBConnections

The database connection string for the AdIdentity Service has not been specified.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-AcctInstalledDBVersion
```

Major	Minor	Build	Revision
-------	-------	-------	----------

-----	-----	-----	-----
-------	-------	-------	-------

5	6	0	0
---	---	---	---

Get the currently installed version of the AdIdentity Service database schema.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-AcctInstalledDBVersion -Upgrade
```

Major	Minor	Build	Revision
-------	-------	-------	----------

-----	-----	-----	-----
-------	-------	-------	-------

6	0	0	0
---	---	---	---

Get the versions of the AdIdentity Service database schema for which upgrade scripts are supplied.

# Get-AcctScopedObject

May 10, 2016

Gets the details of the scoped objects for the AdIdentity Service.

## Syntax

```
Get-AcctScopedObject [-ScopeId <Guid>] [-ScopeName <String>] [-ObjectType <ScopedObjectType>] [-ObjectId <String>] [-ObjectName <String>] [-Description <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns a list of directly scoped objects including the names and identifiers of both the scope and object as well as the object description for display purposes.

There will be at least one result for every directly scoped object. When an object is associated with multiple scopes the output contains one result per scope duplicating the object details.

No records are returned for the All scope, though if an object is not in any scope a result with a null ScopeId and ScopeName will be returned.

## Related topics

### Parameters

**-ScopeId**<Guid>

Gets scoped object entries for the given scope identifier.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ScopeName**<String>

Gets scoped object entries with the given scope name.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ObjectType<ScopedObjectType>**

Gets scoped object entries for objects of the given type.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ObjectId<String>**

Gets scoped object entries for objects with the specified object identifier.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ObjectName<String>**

Gets scoped object entries for objects with the specified object identifier.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Description<String>**

Gets scoped object entries for objects with the specified description.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Acct\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

**-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false



### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Acct\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.AdIdentity.Sdk.ScopedObject

The Get-AcctScopedObject command returns an object containing the following properties:

ScopeId <Guid?>

Specifies the unique identifier of the scope.

ScopeName <String>

Specifies the display name of the scope.

ObjectType <ScopedObjectType>

Type of the object this entry relates to.

ObjectId <String>

Unique identifier of the object.

ObjectName <String>

Display name of the object

Description <String>

Description of the object (possibly \$null if the object type does not have a description).

Notes

If the command fails, the following errors can be returned.

Error Codes

-----

UnknownObject

One of the specified objects was not found.

PartialData

Only a subset of the available data was returned.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

CouldNotQueryDatabase

The query required to get the database was not defined.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-AcctScopedObject -ObjectType Scheme
```

ScopelId : eff6f464-f1ee-4442-add3-99982e0cec01

ScopeName : Sales

ObjectType : Scheme

Objectld : cd4174ee-9e4b-4e57-b126-9dbf757fe493

ObjectName : MyExampleScheme

Description : Test scheme

ScopelId : 304e0fa7-d390-47f0-a94f-7e956a324c41

ScopeName : Finance

ObjectType : Scheme

Objectld : cd4174ee-9e4b-4e57-b126-9dbf757fe493

ObjectName : MyExampleScheme

Description : Test scheme

ScopelId :

ScopeName :

ObjectType : Scheme

Objectld : 5062e46b-71bc-4ac9-901a-30fe6797e2f6

ObjectName : AnotherScheme

Description : Another scheme in no scopes

Gets all of the scoped objects with type Scheme. The example output shows a scheme object (MyExampleScheme) in two scopes Sales and Finance, and another scheme (AnotherScheme) that is not in any scope. The ScopelId and ScopeName values returned are null in the final record.

# Get-AcctService

May 10, 2016

Gets the service record entries for the AdIdentity Service.

## Syntax

```
Get-AcctService [-Metadata <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns instances of the AdIdentity Service that the service publishes. The service records contain account security identifier information that can be used to remove each service from the database.

A database connection for the service is required to use this command.

## Related topics

### Parameters

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Acct\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-Sort By**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Acct\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.AdIdentity.Sdk.Service

The Get-AcctServiceInstance command returns an object containing the following properties.

Uid <Integer>

Specifies the unique identifier for the service in the group. The unique identifier is an index number.

ServiceHostId <Guid>

Specifies the unique identifier for the service instance.

DNSName <String>

Specifies the domain name of the host on which the service runs.

MachineName <String>

Specifies the short name of the host on which the service runs.

CurrentState <Citrix.Fma.Sdk.ServiceCore.ServiceState>

Specifies whether the service is running, started but inactive, stopped, or failed.

LastStartTime <DateTime>

Specifies the date and time at which the service was last restarted.

LastActivityTime <DateTime>

Specifies the date and time at which the service was last stopped or restarted.

OSType

Specifies the operating system installed on the host on which the service runs.

OSVersion

Specifies the version of the operating system installed on the host on which the service runs.

ServiceVersion

Specifies the version number of the service instance. The version number is a string that reflects the full build version of the service.

DatabaseUserName <string>

Specifies for the service instance the Active Directory account name with permissions to access the database. This will be either the machine account or, if the database is running on a controller, the NetworkService account.

Sid <string>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

ActiveSiteServices <string[]>

Specifies the names of active site services currently running in the service. Site services are components that perform long-running background processing in some services. This field is empty for services that do not contain site services.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

UnknownObject

One of the specified objects was not found.

PartialData

Only a subset of the available data was returned.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

CouldNotQueryDatabase

The query required to get the database was not defined.

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-AcctService
```

```
Uid : 1
ServiceHostId : aef6f464-f1ee-4042-a523-66982e0cecd0
DNSName : MyServer.company.com
MachineName : MYSERVER
CurrentState : On
LastStartTime : 04/04/2011 15:25:38
LastActivityTime : 04/04/2011 15:33:39
OSType : Win32NT
OSVersion : 6.1.7600.0
ServiceVersion : 5.1.0.0
DatabaseUserName : NT AUTHORITY\NETWORK SERVICE
SID : S-1-5-21-2316621082-1546847349-2782505528-1165
ActiveSiteServices : {MySiteService1, MySiteService2...}
Get all the instances of the AdIdentity Service running in the current service group.
```



# Get-AcctServiceAddedCapability

May 10, 2016

Gets any added capabilities for the AdIdentity Service on the controller.

## Syntax

```
Get-AcctServiceAddedCapability [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables updates to the AdIdentity Service on the controller to be detected.

You do not need to configure a database connection before using this command.

## Related topics

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.String

String containing added capabilities.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-AcctServiceAddedCapability
```

Get the added capabilities of the AdIdentity Service.

# Get-AcctServiceInstance

May 10, 2016

Gets the service instance entries for the AdIdentity Service.

## Syntax

```
Get-AcctServiceInstance [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns service interfaces published by the instance of the AdIdentity Service. Each instance of a service publishes multiple interfaces with distinct interface types, and each of these interfaces is represented as a ServiceInstance object. Service instances can be used to register the service with a central configuration service so that other services can use the functionality.

You do not need to configure a database connection to use this command.

## Related topics

## Parameters

**-AdminAddress** <String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.AdIdentity.Sdk.ServiceInstance

The Get-AcctServiceInstance command returns an object containing the following properties.

ServiceGroupUid <Guid>

Specifies the unique identifier for the service group of which the service is a member.

ServiceGroupName <String>

Specifies the name of the service group of which the service is a member.

ServiceInstanceUID <Guid>

Specifies the unique identifier for registered service instances, which are service instances held by and obtained from a

central configuration service. Unregistered service instances do not have unique identifiers.

ServiceType <String>

Specifies the service instance type. For this service, the service instance type is always Acct.

Address

Specifies the address of the service instance. The address can be used to access the service and, when registered in the central configuration service, can be used by other services to access the service.

Binding

Specifies the binding type that must be used to communicate with the service instance. In this release of XenDesktop, the binding type is always 'wcf\_HTTP\_kerb'. This indicates that the service provides a Windows Communication Foundation endpoint that uses HTTP binding with integrated authentication.

Version

Specifies the version of the service instance. The version number is used to ensure that the correct versions of the services are used for communications.

ServiceAccount <String>

Specifies the Active Directory account name for the machine on which the service instance is running. The account name is used to provide information about the permissions required for interservice communications.

ServiceAccountSid <String>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

InterfaceType <String>

Specifies the interface type. Each service can provide multiple service instances, each for a different purpose, and the interface defines the purpose. Available interfaces are:

SDK - for PowerShell operations

InterService - for operations between different services

Peer - for communications between services of the same type

Metadata <Citrix.AdIdentity.Sdk.Metadata[]>

The collection of metadata associated with registered service instances, which are service instances held by and obtained from a central configuration service. Metadata is not stored for unregistered service instances.

Notes

If the command fails, the following errors can be returned.

Error Codes

-----

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-AcctServiceInstance
```

```
Address : http://MyServer.com:80/Citrix/AdIdentityService
Binding : wcf_HTTP_kerb
InterfaceType : SDK
Metadata :
MetadataMap :
ServiceAccount : ENG\MyAccount$
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164
ServiceGroupName : MyServiceGroup
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000
ServiceType : Acct
Version : 1

Address : http://MyServer.com:80/Citrix/AdIdentityService/IServiceApi
Binding : wcf_HTTP_kerb
InterfaceType : InterService
Metadata :
MetadataMap :
```

ServiceAccount : ENG\MyAccount  
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164  
ServiceGroupName : MyServiceGroup  
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d  
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000  
ServiceType : Acct  
Version : 1

Get all instances of the AdIdentity Service running on the specified machine. For remote services, use the AdminAddress parameter to define the service for which the interfaces are required. If the AdminAddress parameter has not been specified for the runspace, service instances running on the local machine are returned.

# Get-AcctServiceStatus

May 10, 2016

Gets the current status of the AdIdentity Service on the controller.

## Syntax

```
Get-AcctServiceStatus [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables the status of the AdIdentity Service on the controller to be monitored. If the service has multiple data stores it will return the overall state as an aggregate of all the data store states. For example, if the site data store status is OK and the secondary data store status is DBUnconfigured then it will return DBUnconfigured.

## Related topics

[Set-AcctDBConnection](#)

[Test-AcctDBConnection](#)

[Get-AcctDBConnection](#)

[Get-AcctDBSchema](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Get-AcctServiceStatus command returns an object containing the status of the AdIdentity Service together with extra diagnostics information.

DBUnconfigured

The AdIdentity Service does not have a database connection configured.

DBRejectedConnection

The database rejected the logon attempt from the AdIdentity Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

#### InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the AdIdentity Service schema has not been added to the database.

#### DBNotFound

The specified database could not be located with the configured connection string.

#### DBNewerVersionThanService

The version of the AdIdentity Service currently in use is incompatible with the version of the AdIdentity Service schema on the database. Upgrade the AdIdentity Service to a more recent version.

#### DBOlderVersionThanService

The version of the AdIdentity Service schema on the database is incompatible with the version of the AdIdentity Service currently in use. Upgrade the database schema to a more recent version.

#### DBVersionChangeInProgress

A database schema upgrade is currently in progress.

#### OK

The AdIdentity Service is running and is connected to a database containing a valid schema.

#### Failed

The AdIdentity Service has failed.

#### Unknown

(0) The service status cannot be determined.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException



An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-AcctServiceStatus
```

DBUnconfigured

Get the current status of the AdIdentity Service.

# New-AcctADAccount

May 10, 2016

Creates AD computer accounts in the specified identity pool.

## Syntax

```
New-AcctADAccount [-IdentityPoolName] <String> -ADAccountName <String[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
New-AcctADAccount [-IdentityPoolName] <String> -Count <Int32> [-StartCount <Int32>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
New-AcctADAccount -IdentityPoolUid <Guid> -Count <Int32> [-StartCount <Int32>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
New-AcctADAccount -IdentityPoolUid <Guid> -ADAccountName <String[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to create new AD computer accounts and register them in an already existing identity pool.

The accounts are created using the information stored in the identity pool. This provides the account name (via the Naming Scheme property and Start Count), domain, and OU.

The runspace used for this command must have sufficient privileges in Active Directory to create the new computer accounts.

The AD account names will pad the index to use all the space specified in the identity pool naming scheme (e.g. "acc###" will become "acc001"). However, if the index overflows the available space the cmdlet expands the format to use the next incremental number (e.g. "acc###" will become "acc1000" if the index is 10000, which cannot fit into the three '#' placeholders). If this expanded name exceeds the 15 character name limit, the accounts are not created.

There can be only one creation process running for a specific identity pool at any one time. Attempting to start another account creation process while an existing one is executing results in an error being returned.

## Related topics

[Add-AcctADAccount](#)

[Remove-AcctADAccount](#)

[Get-AcctADAccount](#)

[Repair-AcctADAccount](#)

[Unlock-AcctADAccount](#)

[Update-AcctADAccount](#)

## Parameters

#### **-IdentityPoolName**<String>

The name of the identity pool in which to create the accounts.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-ADAccount Name**<String[]>

The AD account names to be created. These are just the simple machine account names e.g. MyVM001

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Count**<Int32>

The number of accounts to create.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-IdentityPoolUid**<Guid>

The unique identifier for the identity pool in which the accounts will be created.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Start Count**<Int32>

The start index for the create process.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

Citrix.AdIdentity.Sdk.AccountOperationDetailedSummary

The Add-AcctADAccount returns an object that contains the following parameters;

SuccessfulAccountsCount <int>

The number of accounts that were added successfully

FailedAccountsCount <int>

The number of accounts that were not added.

FailedAccounts <Citrix.AdIdentity.Sdk.AccountError[]>

The list of accounts that failed to be added. Each one has the following parameters;

ADAccountName <string>

ADAccountSid <String>

ErrorReason <string>

This can be one of the following

IdentityDuplicateObjectExists

An identity with the same SID already exists.

ADServiceDatabaseError

An error occurred in the service while attempting a database operation.

ADServiceDatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

ADServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

FailedToConnectToDomainController

Contacting Active Directory failed.

FailedToGetOrganizationUnitInAD

Failed to access the OU in Active Directory.

FailedToGetDefaultComputerContainerInAD

Failed to access the default computers container in Active Directory.

FailedToCreateComputerAccountInAD

Failed to create the computer account in Active Directory.

FailedToAccessComputerAccountInAD

Failed to read the newly created computer account in Active Directory.

FailedToGetSidFromAD

Failed to get the SID for the created account from Active Directory.

FailedToSetSamAccountNameInAD

Failed to set the SAM account name in Active Directory for the account created.

FailedToSetUserAccountControlInAD

Failed to set the user account controller properties for the account created in Active Directory.

FailedToSaveChangeInAD

Failed to save the changes made to the created computer account in Active Directory.

FailedToSetPasswordInAD

Failed to set the password for the created computer account in Active Directory.

FailedToEnableAccountInAD

Failed to enable the newly created computer account in Active Directory.

ComputerNameAlreadyInUseInAD

The computer name for the computer to create is in use in Active Directory.

FailedToGetDistinguishedNameInAD

Failed to get the distinguished name for the created computer account in ActiveDirectory.

FailedToSetDnsHostNameInAD

Failed to set the Dns Host Name property for the created computer account in ActiveDirectory.

FailedToSetDisplayNameInAD

Failed to set the DisplayName property for the created computer account in ActiveDirectory.

FailedToWriteServicePrincipalNameInAD

Failed to set the ServicePrincipalName property for the created computer account in ActiveDirectory.

DiagnosticInformation <Exception>

Any other error information

SuccessfulAccounts <Citrix.AdIdentity.Sdk.Identity[]>

The list of accounts that were successfully added. Each object provides details of the identity and contains the following information:

ADAccountSID <string>

The Sid of the identity.

ADAccountName <string>

The account name for the identity.

Domain <string>

The domain name that the account was created in.

State <string>

The current state of the AD account. This can be one of the following:

Error

The account is locked or disabled in AD.

Available

The account is in AD and available to be consumed by the other Machine Creation Services.

InUse

The account is in AD and is being consumed by the other Machine Creation Services.

Tainted

The account is in AD and no longer consumed by other Machine Creation Services. However, the password is no longer known so cannot be reused without 'Repairing' the account. See repair-AcctADAccount for details.

Lock <Boolean>

Indicates if the identity pool is locked.

Notes

In the case of failure, the following errors can result.

Error Codes

-----

NamingSchemeNotSpecifiedForIdentityPool

No naming scheme is defined in the specified identity pool.

IdentityPoolObjectNotFound

The specified identity pool was not located.

IdentityPoolAlreadyLocked

The specified identity pool is locked.

PermissionDenied

The user does not have administrative rights to perform this operation.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## CommunicationError

An error occurred while communicating with the service.

## ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>New-AcctADAccount -IdentityPoolName MyPool -Count 2 -OutVariable result
```

SuccessfulAccounts	SuccessfulAccountsCount	FailedAccountsCount	FailedAccounts
{MyDomain\ACC001, MyDomain\ACC002}	2	0	{}

```
$result[0].SuccessfulAccounts
```

```
ADAccountSid : S-1-5-21-1315084875-1285793635-2418178940-2684
```

```
ADAccountName : MyDomain\ACC001
```

```
Domain : MyDomain.com
```

```
State : Available
```

```
Lock : False
```

```
ADAccountSid : S-1-5-21-1315084875-1285793635-2418178940-2685
```

```
ADAccountName : MyDomain\ACC002
```

```
Domain : MyDomain.com
```

```
State : Available
```

```
Lock : False
```

Creates two new AD accounts and registers them in the identity pool called "MyPool".

### ----- EXAMPLE 2 -----

```
c:\PS>New-AcctADAccount -IdentityPoolName MyPool -Count 2 -StartCount 50 -OutVariable result
```

SuccessfulAccounts	SuccessfulAccountsCount	FailedAccountsCount	FailedAccounts
--------------------	-------------------------	---------------------	----------------



```

{MyDomain\ACC050, MyDomain\ACC051} 2 0 {}
```

```
$result[0].SuccessfulAccounts
```

```
ADAccountSid : S-1-5-21-1315084875-1285793635-2418178940-2686
```

```
ADAccountName : MyDomain\ACC050
```

```
Domain : MyDomain.com
```

```
State : Available
```

```
Lock : False
```

```
ADAccountSid : S-1-5-21-1315084875-1285793635-2418178940-2687
```

```
ADAccountName : MyDomain\ACC051
```

```
Domain : MyDomain.com
```

```
State : Available
```

```
Lock : False
```

Creates two new AD accounts and registers them in the identity pool called "MyPool", starting from an index of 50.

# New-AcctIdentityPool

May 10, 2016  
Creates a new identity pool.

## Syntax

New-AcctIdentityPool -IdentityPoolName <String> [-NamingScheme <String>] [-NamingSchemeType <ADIdentityNamingScheme>] [-OU <String>] [-Domain <String>] [-AllowUnicode] [-StartCount <Int32>] [-Scope <String[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Provides the ability to create identity pools that can be used to store AD computer accounts.

The naming scheme, naming scheme type, and domain must be specified if the identity pool is to be used to create new accounts.

Each identity pool is tied to a single domain. All the identities in an identity pool belong to the same domain. If the domain is not specified by a parameter to this command the domain will be set when an account is imported into it.

## Related topics

- [Remove-AcctIdentityPool](#)
- [Rename-AcctIdentityPool](#)
- [Set-AcctIdentityPool](#)
- [Test-AcctIdentityPoolNameAvailable](#)
- [New-AcctADAccount](#)

## Parameters

**-IdentityPoolName**<String>

The name of the identity pool. This must not contain any of the following characters \/:;#.\*?=<>|[]()""

Required?	true
Default Value	
Accept Pipeline Input?	false

**-NamingScheme**<String>

Defines the template name for AD accounts created in the identity pool. The scheme can consist of fixed characters and a variable part defined by '#' characters. There can be only one variable region defined. The number of '#' characters defines the minimum length of the variable region. For example, a naming scheme of H#### could create accounts called H0001, H0002 (for a numeric scheme type) or HAAAA, HAAAB (for an alphabetic type).

Citrix recommends that you define a naming scheme that will not clash with accounts which already exist in AD; account creation will fail if a clash occurs.

There are restrictions on what constitutes a valid computer name: Minimum name length: 2 (DNS) Maximum name length: 15 bytes (NetBIOS) The following characters are not allowed in a computer name: backslash (\) slash mark (/) colon (:) asterisk (\*) question mark (?)

quotation mark (") less than sign (<) greater than sign (>) vertical bar (|) comma (,) tilde (~) exclamation point (!) at sign (@) number sign (#) dollar sign (\$) percent (%) caret (^) ampersand (&) apostrophe (') parenthesis (()) braces ({}), underscore (\_) The following names are reserved and must not be used at the end of the naming scheme: -GATEWAY -GW -TAC Names must not start with a period (NetBIOS). Names must not be composed entirely of numbers (NetBIOS). Names must not contain a blank or space characters (DNS).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-NamingSchemeType**<ADIdentityNamingScheme>

The type of naming scheme. This can be Numeric or Alphabetic. This defines the format of the variable part of the AD account names that will be created.

Required?	false
Default Value	Numeric
Accept Pipeline Input?	false

#### **-OU**<String>

The OU that computer accounts will be created into. If this is not specified, accounts are created into the default account container specified by AD. This is the 'Computers' container for out-of-the-box installations of AD. The OU must be a valid AD container and of the domain specified for the pool;

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Domain**<String>

The AD domain name for the pool. Specify this in FQDN format; for example, MyDomain.com.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AllowUnicode**<SwitchParameter>

Allow the naming scheme to have characters other than alphanumeric characters.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-StartCount**<Int32>

Defines the next number that will be used if creating new AD accounts for the identity pool.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Scope**<String[]>

The administration scopes to be applied to the new identity pool.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

Citrix.AdIdentity.Sdk.IdentityPool

This object provides details of the identity pool and contains the following information:

IdentityPoolName <string>

The name of the identity pool.

IdentityPoolUid <Guid>

The unique identifier for the identity pool.

NamingScheme <string>

The naming scheme for the identity pool.

NamingSchemeType <Citrix.XDInterServiceTypes.ADIdentityNamingScheme>

The naming scheme type for the identity pool. This can be one of the following:

Numeric - naming scheme uses numeric indexes

Alphabetic - naming scheme uses alphabetic indexes

StartCount <int>

The next index to be used when creating an identity from the identity pool.

OU <string>

The Active Directory distinguished name for the OU in which accounts created from this identity pool will be created.

Domain <string>

The Active Directory domain that accounts in the pool belong to.

Lock <Boolean>

Indicates if the identity pool is locked.

## Notes

In the case of failure, the following errors can result.

### Error Codes

-----

UnableToConvertDomainName

Unable to convert domain name to DNS format.

NamingSchemeNotEnoughCharacters

Naming scheme does not have enough characters specified.

NamingSchemeTooManyCharacters

Naming scheme has too many characters specified.

NamingSchemeIllegalCharacter

Naming scheme contains illegal characters.

NamingSchemeMayNotStartWithPeriod

Naming scheme starts with a period (.) character.

NamingSchemeMayNotBeAllNumbers

Naming scheme contains only numbers.

NamingSchemeMissingNumericSpecifications

Naming scheme does not contain any variable specification (i.e. no '#' characters are specified).

NamingSchemeHasMoreThanOneSetOfHashes

Naming scheme has more than one variable region (i.e. there are '#' characters separated by other characters).

IdentityPoolDuplicateObjectExists

An identity pool with the same name already exists.

IdentityPoolOUInvalid

Identity Pool OU invalid as it does not exist.

IdentityPoolOUOfWrongDomain

IdentityPool OU invalid as it refers to a different domain to the domain specified for the pool.

InvalidIdentityPoolParameterCombination

Caused by either of the following validation errors:

- \* If an OU is specified then a domain must also be specified.

- \* NamingScheme, NamingSchemeType and Domain must all be present if any of these are specified.

NamingSchemeIllegalComputerName

The naming scheme supplied is not valid.

PermissionDenied

The user does not have administrative rights to perform this operation.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

CommunicationError

An error occurred while communicating with the service.

ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>New-AcctIdentityPool -IdentityPoolName MyPool -NamingScheme Acc#### -Domain MyDomain.com -NamingSchemeType Numeric
```

```
IdentityPoolName : MyPool
IdentityPoolUid : 22072d9e-6a8f-494b-a5bc-2ef18ca4b915
NamingScheme : Acc####
NamingSchemeType : Numeric
StartCount : 1
OU :
Domain : MyDomain.com
Lock : False
```

Create a new identity pool from which accounts can be created in the domain called "MyDomain.com" using a numeric scheme of the format Acc####. The first account that will be created from this identity pool definition is Acc0001.

New AD accounts can be imported into this pool too, but must be from the Domain "MyDomain.com".

# Remove-AcctADAccount

May 10, 2016  
Removes AD computer accounts from an identity pool.

## Syntax

```
Remove-AcctADAccount [-IdentityPoolName] <String> -ADAccountName <String[]> [-RemovalOption
<ADIdentityRemoveAccountOption>] [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Remove-AcctADAccount [-IdentityPoolName] <String> -ADAccountSid <String[]> [-RemovalOption
<ADIdentityRemoveAccountOption>] [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Remove-AcctADAccount -IdentityPoolUid <Guid> -ADAccountSid <String[]> [-RemovalOption <ADIdentityRemoveAccountOption>] [-
Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Remove-AcctADAccount -IdentityPoolUid <Guid> -ADAccountName <String[]> [-RemovalOption <ADIdentityRemoveAccountOption>]
[-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to remove AD accounts from an identity pool. This removes the AD account from the Citrix service management scope. This process provides the options for removing the account in AD (or disabling it) if required.

All aspects of this command that need to make modifications to the accounts in AD will use the account that the runspace is using. This means that if an account is to be removed from AD or disabled, the user performing the operation in PowerShell must have sufficient privileges in AD for this operation to complete successfully.

If the option to remove the account from AD or to disable it in AD is specified, the AD operation must succeed for the account to be removed from the Citrix AD Identity Service database. Use caution when using the Force parameter because this allows removal of accounts that are in the 'inUse' state, which may result in the machines becoming unusable.

## Related topics

- [New-AcctADAccount](#)
- [Add-AcctADAccount](#)
- [Repair-AcctADAccount](#)
- [Unlock-AcctADAccount](#)
- [Update-AcctADAccount](#)
- [Get-AcctADAccount](#)

## Parameters

**-IdentityPoolName**<String>

The identity pool that the accounts are to be removed from.

Required?	true
Default Value	



Accept Pipeline Input?	false
------------------------	-------

#### **-ADAccountName**<String[]>

The AD account name to be removed. AD accounts are accepted in the following formats: Fully qualified DN e.g. CN=MyComputer,OU=Computers,DC=MyDomain,DC=Com; UPN format e.g MyComputer@MyDomain.Com; Domain qualified e.g MyDomain\MyComputer.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-ADAccountSid**<String[]>

The Active Directory Account SID for the account to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-IdentityPoolUid**<Guid>

The unique identifier for the identity pool that the accounts are to be removed from.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-RemovalOption**<ADIdentityRemoveAccountOption>

Defines the behavior relating to the AD account.

None - Do not attempt to remove the account from AD Delete - Attempt to remove the account from AD Disable - Attempt to disable the account in AD

Required?	false
Default Value	None

Accept Pipeline Input?	false
------------------------	-------

#### **-Force**<SwitchParameter>

Indicates if accounts that are marked as 'in-use' can be removed.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

Citrix.AdIdentity.Sdk.AccountOperationSummary

The remove-AcctADAccount command returns an object with the following parameters;

SuccessfulAccountsCount <int>

The number of accounts that were removed successfully.

FailedAccountsCount <int>

The number of accounts that were not removed.

FailedAccounts <Citrix.AdIdentity.Sdk.AccountError[]>

The list of accounts that failed to be removed. Each one has the following parameters:

ADAccountName <string>

ADAccountSid <String>

ErrorReason <AdIdentityStatus>

This can be one of the following

UnableToConvertDomain

IdentityNotLocatedInDomain

IdentityNotInIdentityPool

IdentityObjectInUse

IdentityObjectLocked

ADServiceDatabaseError

ADServiceDatabaseNotConfigured

ADServiceStatusInvalidDb

FailedToConnectToDomainController

FailedToDisableAccountInAD

FailedToDeleteAccountInAD

FailedToExecuteSearchInAD

FailedToAccessComputerAccountInAD

DiagnosticInformation <Exception>

Any other error information

Notes

In the case of failure, the following errors can result.

Error Codes

-----

IdentityPoolNotFound

The specified identity pool was not found.

IdentityPoolAlreadyLocked

The specified identity pool was locked by another operation.

PermissionDenied

The user does not have administrative rights to perform this operation.

## ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## CommunicationError

An error occurred while communicating with the service.

## ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS>Remove-AcctADAccount -IdentityPool MyPool -ADAccountName "Domain\account","domain\account2"
```

SuccessfulAccountsCount	FailedAccountsCount	FailedAccounts
2	0	{}

Removes two accounts (account and account2) from the identity pool called "MyPool", leaving the AD accounts untouched.

### ----- EXAMPLE 2 -----

```
C:\PS>Remove-AcctADAccount -IdentityPool MyPool -RemovalOption Delete -ADAccountName "Domain\account","domain\account2"
```

SuccessfulAccountsCount	FailedAccountsCount	FailedAccounts
2	0	{}

Removes two accounts (account and account2) from the identity pool called "MyPool" (and from Active Directory).

### ----- EXAMPLE 3 -----

```
C:\PS>Remove-AcctADAccount -IdentityPool MyPool -ADAccountName "Domain\account","domain\account2" -Force
```

SuccessfulAccountsCount	FailedAccountsCount	FailedAccounts
2	0	{}

Removes two accounts (account and account2) from the identity pool called "MyPool", leaving the AD accounts untouched. The

accounts are removed regardless of whether they are in the 'inUse' state or not.

----- **EXAMPLE 4** -----

```
C:\PS>Remove-AcctADAccount -IdentityPool MyPool -ADAccountName "Domain\account","domain\account2" -OutVariable result
```

SuccessfulAccountsCount	FailedAccountsCount	FailedAccounts
-----	-----	-----
1	1	{account2}

```
C:\PS>$result[0].FailedAccounts
```

ADAccountName	ADAccountSid	ErrorReason
-----	-----	-----
Domain\account2	account2	IdentityObjectLocked

Shows failure of removal of one of two accounts and how to retrieve the failure reason.

----- **EXAMPLE 5** -----

```
C:\PS>Remove-AcctADAccount -IdentityPool MyPool -ADAccountSid S-1-5-21-1315084875-1285793635-2418178940-2685
```

SuccessfulAccountsCount	FailedAccountsCount	FailedAccounts
-----	-----	-----
1	0	{}

Removes one account (S-1-5-21-1315084875-1285793635-2418178940-2685) from the identity pool called "MyPool", leaving the AD accounts untouched.

# Remove-AcctIdentityPool

May 10, 2016

Removes identity pools.

## Syntax

```
Remove-AcctIdentityPool [-IdentityPoolName] <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AcctIdentityPool -IdentityPoolUid <Guid> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to remove identity pools. The identity pool must be emptied of any AD accounts that it contains before it can be removed.

## Related topics

[New-AcctIdentityPool](#)

[Rename-AcctIdentityPool](#)

[Set-AcctIdentityPool](#)

[Unlock-AcctIdentityPool](#)

## Parameters

**-IdentityPoolName**<String>

The name of the identity pool to remove.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-IdentityPoolUid**<Guid>

The unique identifier for the identity pool to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Notes

In the case of failure the following errors can be produced.

#### Error Codes

-----

#### IdentityPoolObjectNotFound

The specified identity pool could not be located.

#### UnableToRemoveDueToAssociatedAccounts

The identity pool is not empty.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### ExceptionThrown

An unexpected error occurred. To locate more details see the Windows event logs on the controller being used, or examine the XenDesktop logs.

#### Examples

##### ----- **EXAMPLE 1** -----

`c:\Remove-AcctIdentityPool -IdentityPoolName MyPool`

Removes the identity pool called "MyPool".



# Remove-AcctIdentityPoolMetadata

May 10, 2016  
Removes metadata from the given IdentityPool.

## Syntax

```
Remove-AcctIdentityPoolMetadata [-IdentityPoolUid] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Remove-AcctIdentityPoolMetadata [-IdentityPoolUid] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Remove-AcctIdentityPoolMetadata [-IdentityPoolName] <String> -Name <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Remove-AcctIdentityPoolMetadata [-IdentityPoolName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Remove-AcctIdentityPoolMetadata [-InputObject] <IdentityPool[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Remove-AcctIdentityPoolMetadata [-InputObject] <IdentityPool[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given IdentityPool.

## Related topics

[Set-AcctIdentityPoolMetadata](#)

## Parameters

**-IdentityPoolUid**<Guid>

Id of the IdentityPool

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-IdentityPoolName**<String>

Name of the IdentityPool

--	--

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-InputObject**<IdentityPool[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-

LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

##### InvalidParameterCombination

The cmdlet parameters are inconsistent.

##### UnknownObject

One of the specified objects was not found.

##### DatabaseError

An error occurred in the service while attempting a database operation.

##### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

##### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

##### PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-AcctIdentityPool | % { Remove-AcctIdentityPoolMetadata -Map $_.MetadataMap }
```

Remove all metadata from all IdentityPool objects.

# Remove-AcctIdentityPoolScope

May 10, 2016

Remove the specified IdentityPool(s) from the given scope(s).

## Syntax

```
Remove-AcctIdentityPoolScope [-Scope] <String[]> -InputObject <IdentityPool[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-AcctIdentityPoolScope [-Scope] <String[]> -IdentityPoolUid <Guid[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-AcctIdentityPoolScope [-Scope] <String[]> -IdentityPoolName <String[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The RemoveAcctIdentityPoolScope cmdlet is used to remove one or more IdentityPool objects from the given scope(s).

There are multiple parameter sets for this cmdlet, allowing you to identify the IdentityPool objects in different ways:

- IdentityPool objects can be piped in or specified by the InputObject parameter
- The IdentityPoolUid parameter specifies objects by IdentityPoolUid
- The IdentityPoolName parameter specifies objects by IdentityPoolName (supports wildcards)

To remove a IdentityPool from a scope you need permission to change the scopes of the IdentityPool.

If the IdentityPool is not in a specified scope, that scope will be silently ignored.

## Related topics

[Add-AcctIdentityPoolScope](#)

[Get-AcctScopedObject](#)

## Parameters

**-Scope**<String[]>

Specifies the scopes to remove the objects from.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-InputObject**<IdentityPool[]>

Specifies the IdentityPool objects to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

### **-IdentityPoolUid**<Guid[]>

Specifies the IdentityPool objects to be removed by IdentityPoolUid.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

### **-IdentityPoolName**<String[]>

Specifies the IdentityPool objects to be removed by IdentityPoolName.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

None

### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

##### UnknownObject

One of the specified objects was not found.

##### ScopeNotFound

One of the specified scopes was not found.

##### DatabaseError

An error occurred in the service while attempting a database operation.

##### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

##### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

##### PermissionDenied

You do not have permission to execute this command with the specified objects or scopes.

##### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Remove-AcctIdentityPoolScope Finance -IdentityPoolUid 6702C5D0-C073-4080-A0EE-EC74CB537C52
```

Removes a single IdentityPool from the 'Finance' scope.

### ----- EXAMPLE 2 -----

```
c:\PS>Remove-AcctIdentityPoolScope Finance,Marketing -IdentityPoolUid 6702C5D0-C073-4080-A0EE-EC74CB537C52
```

Removes a single IdentityPool from multiple scopes.

### ----- EXAMPLE 3 -----

```
c:\PS>Get-AcctIdentityPool | Remove-AcctIdentityPoolScope Finance
```

Removes all visible IdentityPool objects from the 'Finance' scope.

### ----- EXAMPLE 4 -----

```
c:\PS>Remove-AcctIdentityPoolScope Finance -IdentityPoolName A*
```

Removes IdentityPool objects with a name starting with an 'A' from the 'Finance' scope.



# Remove-AcctServiceMetadata

May 10, 2016

Removes metadata from the given Service.

## Syntax

```
Remove-AcctServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AcctServiceMetadata [-ServiceHostId] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AcctServiceMetadata [-InputObject] <Service[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AcctServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Service.

## Related topics

[Set-AcctServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name<String>**

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-AcctService | % { Remove-AcctServiceMetadata -Map $_.MetadataMap }
```

Remove all metadata from all Service objects.

# Rename-AcctIdentityPool

May 10, 2016

Renames an identity pool.

## Syntax

```
Rename-AcctIdentityPool [-IdentityPoolName] <String> [-NewIdentityPoolName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Rename-AcctIdentityPool -IdentityPoolUid <Guid> -NewIdentityPoolName <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to change the name of an existing identity pool.

## Related topics

[Get-AcctIdentityPool](#)

[Set-AcctIdentityPool](#)

[Remove-AcctIdentityPool](#)

[Test-AcctIdentityPoolNameAvailable](#)

## Parameters

**-IdentityPoolName**<String>

The name of the identity pool to be renamed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-IdentityPoolUid**<Guid>

The unique identifier for the identity pool to be renamed.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-NewIdentityPoolName**<String>

The new name for the identity pool. This must be a name which is not used by an existing identity pool, and it must not contain any of the following characters \/:#.\*?=<>|[]()''''

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-PassThru**<SwitchParameter>

Defines whether or not the command returns a result showing the new state of the updated provisioning scheme.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

Citrix.AdIdentity.Sdk.IdentityPool

This object provides details of the identity pool and contains the following information:

IdentityPoolName <string>

The name of the identity pool.

IdentityPoolUid <Guid>

The unique identifier for the identity pool.

NamingScheme <string>

The naming scheme for the identity pool.

NamingSchemeType <Citrix.XDInterServiceTypes.ADIdentityNamingScheme>

The naming scheme type for the identity pool. This can be one of the following:

Numeric - naming scheme uses numeric indexes

Alphabetic - naming scheme uses alphabetic indexes

StartCount <int>

The next index to be used when creating an identity from the identity pool.

OU <string>

The Active Directory distinguished name for the OU in which accounts created from this identity pool will be created.

Domain <string>

The Active Directory domain that accounts in the pool belong to.

Lock <Boolean>

Indicates if the identity pool is locked.

## Notes

In the case of failure the following errors can result.

## Error Codes

-----

IdentityPoolObjectNotFound

The specified identity pool could not be located.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

#### Examples

##### ----- **EXAMPLE 1** -----

**C:\PS>Rename-AcctIdentityPool -IdentityPoolName oldName -NewIdentityPoolName newName**

Renames an existing identity pool called "oldName" to be called "newName".



# Repair-AcctADAccount

May 10, 2016

Resets the Active Directory machine password for the given accounts.

## Syntax

```
Repair-AcctADAccount -ADAccountName <String[]> [-Password <String>] [-SecurePassword <SecureString>] [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Repair-AcctADAccount -ADAccountSid <String[]> [-Password <String>] [-SecurePassword <SecureString>] [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This provides the ability to synchronize the account password stored in Active Directory with the password stored in the AD Identity Service. If successful, this results in the AD Identity Service account state being reset to 'available' so it can be consumed by other Machine Creation Services.

If the current account password is not supplied using the Password or SecurePassword Parameters, this requires the user who initiated the runspace to have the required permissions in Active Directory to reset the Active Directory Account password.

If the current account password is supplied then this command will use the password change operation which does not require any elevated permissions in Active Directory.

## Related topics

[New-AcctADAccount](#)

[Add-AcctADAccount](#)

[Remove-AcctADAccount](#)

[Unlock-AcctADAccount](#)

[Update-AcctADAccount](#)

[Get-AcctADAccount](#)

## Parameters

**-ADAccount Name**<String[]>

The Active Directory account name(s) that are to be repaired. Active Directory accounts are accepted in the following formats: Fully qualified DN e.g. CN=MyComputer,OU=Computers,DC=MyDomain,DC=Com; UPN format e.g. MyComputer@MyDomain.Com; Domain qualified e.g. MyDomain\MyComputer.

Required?	true
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ADAccountSid**<String[]>

The Active Directory account SID(s) that are to be repaired.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Password**<String>

The current password for the computer account.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecurePassword**<SecureString>

The current password for the account (provided in a Secure String class).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Force**<SwitchParameter>

Indicates whether accounts that are marked as 'in-use' can be repaired or not.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.AdIdentity.Sdk.AccountOperationSummary

The Repair-AcctADAccout command returns an object with the following parameters:

SuccessfulAccountsCount <int>

The number of accounts that were repaired successfully.

FailedAccountsCount <int>

The number of accounts that were not repaired.

FailedAccounts <Citrix.AdIdentity.Sdk.AccountError[]>

The list of accounts that failed to be repaired. Each one has the following parameters:

ADAccountName <string>

ADAccountSid <String>

ErrorReason <AdIdentityStatus>

This can be one of the following

UnableToConvertDomain

IdentityNotLocatedInDomain

IdentityNotFound

IdentityObjectInUse

IdentityObjectLocked

ADServiceDatabaseError

ADServiceDatabaseNotConfigured

ADServiceStatusInvalidDb

FailedToConnectToDomainController

FailedToExecuteSearchInAD

FailedToAccessComputerAccountInAD

FailedToSetPasswordInAD

FailedToChangePasswordInAD

DiagnosticInformation <Exception>

Any other error information

## Notes

In the case of failure, the following errors can result.

### Error Codes

-----

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with database failed for various reasons.

## CommunicationError

An error occurred while communicating with the service.

## ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS>Repair-AcctADAccount -ADAccountName "Domain\account","domain\account2"
```

SuccessfulAccountsCount	FailedAccountsCount	FailedAccounts
2	0	{}

# Reset-AcctServiceGroupMembership

May 10, 2016

Reloads the access permissions and configuration service locations for the AdIdentity Service.

## Syntax

```
Reset-AcctServiceGroupMembership [-ConfigServiceInstance] <ServiceInstance[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Enables you to reload AdIdentity Service access permissions and configuration service locations. The Reset-AcctServiceGroupMembership command must be run on at least one instance of the service type (Acct) after installation and registration with the configuration service. Without this operation, the AdIdentity services will be unable to communicate with other services in the XenDesktop deployment. When the command is run, the services are updated when additional services are added to the deployment, provided that the configuration service is not stopped. The Reset-AcctServiceGroupMembership command can be run again to refresh this information if automatic updates do not occur when new services are added to the deployment. If more than one configuration service instance is passed to the command, the first instance that meets the expected service type requirements is used.

## Related topics

### Parameters

**-ConfigServiceInstance**<ServiceInstance[]>

Specifies the configuration service instance object that represents the service instance for the type 'InterService' that references a configuration service for the deployment.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

Citrix.AdIdentity.Sdk.ServiceInstance[] Service instances containing a ServiceInstance object that refers to the central configuration service interservice interface can be piped to the Reset-AcctServiceGroupMembership command.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### NoSuitableServiceInstance

None of the supplied service instance objects were suitable for resetting service group membership.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config | Reset-AcctServiceGroupMembership
```

Reset the service group membership for a service in a deployment where the configuration service is configured and running on the same machine as the service.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config -AdminAddress OtherServer.example.com | Reset-AcctServiceGroupmembership
```

Reset the service group membership for a service in a deployment where the configuration service that is configured and running on a machine named 'OtherServer.example.com'.



# Set-AcctDBConnection

May 10, 2016  
Configures a database connection for the AdIdentity Service.

## Syntax

Set-AcctDBConnection [-DBConnection] <String> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Configures a connection to a database in which the AdIdentity Service can store its state. The service will attempt to connect and start using the database immediately after the connection is configured. The database connection string is updated to the specified value regardless of whether it is valid or not. Specifying an invalid connection string prevents a service from functioning until the error is corrected.

After a connection is configured, you cannot alter it without first clearing it (by setting the connection to \$null).

You do not need to configure a database connection to use this command.

## Related topics

[Get-AcctServiceStatus](#)

[Get-AcctDBConnection](#)

[Test-AcctDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be used by the AdIdentity Service. Passing in \$null will clear any existing database connection configured.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Force**<SwitchParameter>

If present, allows the local administrator to set the connection string to null when there are problems contacting the database or other services.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and

Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

**Return Values**

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Set-AcctDBConnection command returns an object containing the status of the AdIdentity Service together with extra diagnostics information.

**DBUnconfigured**

The AdIdentity Service does not have a database connection configured.

**DBRejectedConnection**

The database rejected the logon attempt from the AdIdentity Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

**InvalidDBConfigured**

The expected stored procedures are missing from the database. This may be because the AdIdentity Service schema has not been added to the database.

**DBNotFound**

The specified database could not be located with the configured connection string.

**DBNewerVersionThanService**

The version of the AdIdentity Service currently in use is incompatible with the version of the AdIdentity Service schema on the database. Upgrade the AdIdentity Service to a more recent version.

**DBOlderVersionThanService**

The version of the AdIdentity Service schema on the database is incompatible with the version of the AdIdentity Service currently in use. Upgrade the database schema to a more recent version.

**DBVersionChangeInProgress**

A database schema upgrade is currently in progress.

OK

The AdIdentity Service is running and is connected to a database containing a valid schema.

Failed

The AdIdentity Service has failed.

Unknown

The status of the AdIdentity Service cannot be determined.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

InvalidDBConnectionString

The database connection string has an invalid format.

DatabaseConnectionDetailsAlreadyConfigured

There was already a database connection configured. After a configuration is set, it can only be set to \$null.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Set-AcctDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Configures a database connection string for the AdIdentity Service.

----- **EXAMPLE 2** -----

```
c:\PS>Set-AcctDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Configures an invalid database connection string for the AdIdentity Service.

# Set-AcctIdentityPool

May 10, 2016

Update parameters of an identity pool.

## Syntax

```
Set-AcctIdentityPool [-IdentityPoolName] <String> [-NamingScheme <String>] [-NamingSchemeType
<ADIdentityNamingScheme>] [-OU <String>] [-Domain <String>] [-AllowUnicode] [-PassThru] [-StartCount <Int32>] [-
LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-AcctIdentityPool -IdentityPoolUid <Guid> [-NamingScheme <String>] [-NamingSchemeType
<ADIdentityNamingScheme>] [-OU <String>] [-Domain <String>] [-AllowUnicode] [-PassThru] [-StartCount <Int32>] [-
LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to modify the parameters of an identity pool.

Note: When changing a naming scheme or naming scheme type, the index is not reset to 0; it continues to avoid AD account name clashes with existing accounts. If required, use the New-AcctAdAccount command to change the index when creating further accounts.

## Related topics

[New-AcctIdentityPool](#)

[Get-AcctIdentityPool](#)

[Remove-AcctIdentityPool](#)

## Parameters

**-IdentityPoolName**<String>

The name of the identity pool that is to be modified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-IdentityPoolUid**<Guid>

The unique identifier for the identity pool that is to be updated.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	false

#### **-NamingScheme**<String>

The new naming scheme that is to be used for the identity pool.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-NamingSchemeType**<ADIdentityNamingScheme>

The new naming scheme type that is to be used for the identity pool. This can be Numeric or Alphabetic.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OU**<String>

The new OU to be used for the Identity Pool. All accounts created after this is set are created in this AD container. This will not move any of the existing accounts. The OU must be a valid AD container and of the domain specified for the pool.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Domain**<String>

The new Active Directory domain that is to be used for the identity pool. All new accounts will be created in this domain, but this will not impact any of the existing accounts. The domain can be specified in either long or short form (i.e. domain or domain.com).

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AllowUnicode**<SwitchParameter>

Updates the definition of the allowed characters in a naming scheme.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PassThru**<SwitchParameter>

Defines whether the command returns the new state of the identity pool or not.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### **-StartCount**<Int32>

The start index for the next create operation

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.AdIdentity.Sdk.IdentityPool

This object provides details of the identity pool and contains the following information:

IdentityPoolName <string>

The name of the identity pool.

IdentityPoolUid <Guid>

The unique identifier for the identity pool.

NamingScheme <string>

The naming scheme for the identity pool.

NamingSchemeType <Citrix.XDInterServiceTypes.ADIdentityNamingScheme>

The naming scheme type for the identity pool. This can be one of the following:

Numeric - naming scheme uses numeric indexes

Alphabetic - naming scheme uses alphabetic indexes

StartCount <int>

The next index to be used when creating an identity from the identity pool.

OU <string>

The Active Directory distinguished name for the OU in which accounts created from this identity pool will be created.

Domain <string>

The Active Directory domain that accounts in the pool belong to.

Lock <Boolean>

Indicates whether the identity pool is locked.

## Notes

In the case of failure, the following errors can result.

## Error Codes

-----

### InvalidIdentityPoolParameterCombination

Caused by either of the following validation errors:

- \* If an OU is specified then a domain must also be specified.

- \* NamingScheme, NamingSchemeType and Domain must all be present if any of them are specified.

### NamingSchemeIllegalComputerName

The naming scheme supplied is not valid.

### UnableToConvertDomainName

Unable to convert domain name to DNS format.

### NamingSchemeNotEnoughCharacters

Naming scheme does not have enough characters specified.

### NamingSchemeTooManyCharacters

Naming scheme has too many characters specified.

### NamingSchemeIllegalCharacter

Naming scheme contains illegal characters.

### NamingSchemeMayNotStartWithPeriod

Naming scheme starts with a period (.) character.

### NamingSchemeMayNotBeAllNumbers

Naming scheme contains only numbers.

### NamingSchemeMissingNumericSpecifications

Naming scheme does not contain any variable specification (i.e. no '#' characters are specified).



#### NamingSchemeHasMoreThanOneSetOfHashes

Naming scheme has more than one variable region (i.e. there are '#' characters separated by other characters).

#### IdentityPoolDuplicateObjectExists

An identity pool with the same name exists already.

#### IdentityPoolObjectNotFound

The identity pool to be modified could not be located.

#### IdentityPoolOUInvalid

Identity Pool OU invalid as it does not exist.

#### IdentityPoolOUOfWrongDomain

IdentityPool OU invalid as it refers to a different domain to the domain specified for the pool.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS>Set-AcctIdentityPool -IdentityPoolName poolName -StartCount 100 -NamingScheme AC####
```

Changes the start count, and the naming scheme, so that the next account generated will be AC0100 (assuming that account does not already exist)

# Set-AcctIdentityPoolMetadata

May 10, 2016  
Adds or updates metadata on the given IdentityPool.

## Syntax

```
Set-AcctIdentityPoolMetadata [-IdentityPoolUid] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Set-AcctIdentityPoolMetadata [-IdentityPoolUid] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Set-AcctIdentityPoolMetadata [-IdentityPoolName] <String> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Set-AcctIdentityPoolMetadata [-IdentityPoolName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Set-AcctIdentityPoolMetadata [-InputObject] <IdentityPool[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Set-AcctIdentityPoolMetadata [-InputObject] <IdentityPool[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability for additional custom data to be stored against given IdentityPool objects.

## Related topics

[Remove-AcctIdentityPoolMetadata](#)

## Parameters

**-IdentityPoolUid**<Guid>

Id of the IdentityPool

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-IdentityPoolName**<String>

Name of the IdentityPool

Required?	true

Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-InputObject**<IdentityPool[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the IdentityPool specified. The property cannot contain any of the following characters \;#.\*?=<>|[]{}"

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	

Accept Pipeline Input?	false
------------------------	-------

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-AcctIdentityPoolMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

### Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Set-AcctIdentityPoolMetadata -IdentityPoolUid 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
-----	-------

---	-----
-----	-------

property	value
----------	-------

Add metadata with a name of 'property' and a value of 'value' to the IdentityPool with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Set-AcctServiceMetadata

May 10, 2016

Adds or updates metadata on the given Service.

## Syntax

```
Set-AcctServiceMetadata [-ServiceHostId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-AcctServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-AcctServiceMetadata [-InputObject] <Service[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-AcctServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Allows you to store additional custom data against given Service objects.

## Related topics

[Remove-AcctServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the Service specified. The property cannot contain any of the following characters \/:;#.\*?=<>|[]{}"

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	



Accept Pipeline Input?	false
------------------------	-------

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-AcctServiceMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Set-AcctServiceMetadata -ServiceHostId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
---	----
property	value

Add metadata with a name of 'property' and a value of 'value' to the Service with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Test-AcctDBConnection

May 10, 2016

Tests a database connection for the AdIdentity Service.

## Syntax

```
Test-AcctDBConnection [-DBConnection] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Tests a connection to the database in which the AdIdentity Service can store its state. The service will attempt to connect to the database without affecting the current connection to the database.

You do not have to clear the connection to use this command.

## Related topics

[Get-AcctServiceStatus](#)

[Get-AcctDBConnection](#)

[Set-AcctDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be tested by the AdIdentity Service.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

### Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Test-AcctDBConnection command returns an object containing the status of the AdIdentity Service if the connection string of the specified data store were to be set to the string being tested, together with extra diagnostics information for the specified connection string.

#### DBRejectedConnection

The database rejected the logon attempt from the AdIdentity Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

#### InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the AdIdentity Service schema has not been added to the database.

#### DBNotFound

The specified database could not be located with the configured connection string.

#### DBNewerVersionThanService

The version of the AdIdentity Service currently in use is incompatible with the version of the AdIdentity Service schema on the database. Upgrade the AdIdentity Service to a more recent version.

#### DBOlderVersionThanService

The version of the AdIdentity Service schema on the database is incompatible with the version of the AdIdentity Service currently in use. Upgrade the database schema to a more recent version.

#### DBVersionChangeInProgress

A database schema upgrade is currently in progress.

#### OK

The Set-AcctDBConnection command would succeed if it were executed with the supplied connection string.

#### Failed

The AdIdentity Service has failed.

#### Unknown

The status of the AdIdentity Service cannot be determined.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidDBConnectionString

The database connection string has an invalid format.

#### PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Test-AcctDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Tests a database connection string for the AdIdentity Service.

----- **EXAMPLE 2** -----

```
c:\PS>Test-AcctDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Tests an invalid database connection string for the AdIdentity Service.

# Test-AcctIdentityPoolNameAvailable

May 10, 2016

Checks to ensure that the proposed name for an identity pool is unused.

## Syntax

```
Test-AcctIdentityPoolNameAvailable [-IdentityPoolName] <String[]> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Checks to ensure that the proposed name for an identity pool is unused. This check is done without regard for scoping of existing identity pools, so the names of inaccessible pools are also checked.

## Related topics

[New-AcctIdentityPool](#)

[Rename-AcctIdentityPool](#)

## Parameters

**-IdentityPoolName**<String[]>

The name or names of the identity pool(s) to be tested.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

object[]

An array of PSObjects that pair the name and availability of the name

## Notes

In the case of failure, the following errors can result.

### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

Test-AcctIdentityPoolNameAvailable -IdentityPoolName \$NewPoolName

This tests whether the value of \$NewPoolName is unique or not, and can be used to create a new provisioning scheme or rename an existing one without failing. True is returned if the name is good.

# Unlock-AcctADAccount

May 10, 2016

Unlocks AD accounts within the AD Identity Service.

## Syntax

```
Unlock-AcctADAccount -ADAccountName <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Unlock-AcctADAccount -ADAccountSid <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to unlock the AD Identity Service identity item that references a specified AD account. An AD account is marked as locked in the AD Identity Service while the Machine Creation Services (MCS) are processing tasks relating to the account. If these tasks are forcibly stopped, an account can remain locked despite no longer being processed. This command resolves this issue, but use it with caution because unlocking an account that MCS expects to be locked can result in an MCS operation being cancelled. Use this command only when MCS has locked an account for use in a provisioning operation, and where this operation has failed without unlocking the account.

Note: This command does NOT make any changes to the account information stored in Active Directory.

## Related topics

[Get-AcctADAccount](#)

[New-AcctADAccount](#)

[Add-AcctADAccount](#)

[Repair-AcctADAccount](#)

[Remove-AcctADAccount](#)

[Update-AcctADAccount](#)

[Unlock-AcctADAccount](#)

## Parameters

**-ADAccountName**<String>

The AD account name to be unlocked. AD account name is accepted in the following formats: Fully qualified DN e.g. CN=MyComputer,OU=Computers,DC=MyDomain,DC=Com; UPN format e.g MyComputer@MyDomain.Com; Domain qualified e.g MyDomain\MyComputer.

Required?	true
Default Value	
Accept Pipeline Input?	false



### **-ADAccountSid**<String>

The AD account SID that represents the account to be unlocked.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Input Type

Citrix.AdIdentity.Sdk.IdentityInPool You can pipe an object containing a parameter called 'ADAccountSID' to unlock-AcctADAccount.

### Notes

In the case of failure, the following errors can result.

### Error Codes

-----

#### IdentityNotLocatedInDomain

The specified AD account could not be located in Active Directory.

#### IdentityObjectNotFound

The identity could not be found.

#### IdentityAlreadyUnlocked

The identity is not locked.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

### Examples

#### ----- EXAMPLE 1 -----

```
c:\PS>Unlock-AcctADAccount -ADAccountName Domain\account
Unlocks the AD account called "Domain\account".
```

#### ----- EXAMPLE 2 -----

```
c:\PS>Get-AcctADAccount -Filter {lock -eq $true} | Unlock-AcctADAccount
Unlocks all the locked AD accounts.
```



# Unlock-AcctIdentityPool

May 10, 2016

Unlocks identity pools.

## Syntax

```
Unlock-AcctIdentityPool [-IdentityPoolName] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Unlock-AcctIdentityPool -IdentityPoolUid <Guid> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to unlock the specified identity pool. Identity pools are locked automatically when being updated (e.g. when new accounts are being created into them). The pool must never be left in a locked state; this command allows recovery from an error should this ever occur. Use this command with caution, as unlocking an identity pool which is supposed to be locked may result in unexpected behavior.

## Related topics

[New-AcctIdentityPool](#)

[Remove-AcctIdentityPool](#)

[Set-AcctIdentityPool](#)

## Parameters

**-IdentityPoolName**<String>

The name of the identity pool to unlock.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-IdentityPoolUid**<Guid>

The unique identifier for the identity pool to be unlocked.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Input Type

Citrix.AdIdentity.Sdk.IdentityPool You can pipe an object containing a parameter called 'IdentityPoolName' to unlock-AcctIdentityPool.

### Notes

In the case of failure, the following errors can result.

#### Error Codes

-----

#### IdentityPollObjectNotFound

The specified identity pool could not be located.

#### IdentityPoolAlreadyUnlocked

The specified identity pool is not locked.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS>Unlock-AcctIdentityPool -IdentityPool MyPool
```

Unlocks the identity pool called "MyPool".

##### ----- EXAMPLE 2 -----

```
C:\PS>Get-AcctIdentityPool -Filter {Lock -eq $true} | Unlock-AcctIdentityPool
```

Unlocks all the locked identity pools.

# Update-AcctADAccount

May 10, 2016

Refreshes the AD computer account state stored in the AD Identity Service.

## Syntax

```
Update-AcctADAccount [-IdentityPoolName] <String> [-AllAccounts] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Update-AcctADAccount -IdentityPoolUid <Guid> [-AllAccounts] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to synchronize the state of the AD accounts stored in the AD Identity Service with the AD accounts themselves. By default, this checks all accounts marked as 'error' to determine if accounts are still in an error state (i.e. disabled or locked). If you specify the 'AllAccounts' option, it checks accounts not in error state and updates the status of these accounts too.

## Related topics

[New-AcctADAccount](#)

[Add-AcctADAccount](#)

[Remove-AcctADAccount](#)

[Repair-AcctADAccount](#)

[Unlock-AcctADAccount](#)

[Get-AcctADAccount](#)

## Parameters

**-IdentityPoolName**<String>

The name of the identity pool for the accounts to be refreshed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-IdentityPoolUid**<Guid>

The unique identifier for the identity pool of the AD accounts that are to be refreshed.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-AllAccounts**<SwitchParameter>

Indicates if all accounts should be refreshed or only the ones marked as in error.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Notes

In the case of failure, the following errors can result.



## Error Codes

-----

### PermissionDenied

The user does not have administrative rights to perform this operation.

### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with database failed for various reasons.

### CommunicationError

An error occurred while communicating with the service.

### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

c:\PS>Update-AcctADAccount -IdentityPoolName MyPool

Checks the status of all accounts in the Identity Pool MyPool that are currently mark as in the Error state and checks if the account is now available.

### ----- EXAMPLE 2 -----

c:\PS>Update-AcctADAccount -IdentityPoolName MyPool -AllAccounts

Checks the status of all accounts in the Identity Pool MyPool marking them as Available, InUse, Tainted or Error as appropriate.

# Citrix.AppV.Admin.V1

May 10, 2016

## Cmdlets

Name	Description
<a href="#">ConvertTo-CtxAppVLauncherArg</a>	Returns a string containing information to send to the App-V Launcher. You can plug this string directly into the Virtual Delivery Agent (VDA) to launch App-V applications.
<a href="#">Get-CtxAppVApplication</a>	Enumerates all published App-V applications for a given Management server.
<a href="#">Get-CtxAppVApplicationInfo</a>	Enumerates information for a given application in a given package for a given Management server.
<a href="#">Get-CtxAppVServer</a>	Returns URLs for App-V Publishing and Management servers contained in a Citrix App-V policy. Returned values are in string format.
<a href="#">Get-CtxAppVServerSetting</a>	Returns settings for the specified App-V Publishing Server.
<a href="#">New-CtxAppVServer</a>	Creates a new Citrix App-V policy containing the specified App-V Management and Publishing Server URLs.
<a href="#">Set-CtxAppVServerSetting</a>	Specifies the App-V Publishing server settings to use on the VDA. These settings determine whether or not the App-V Client can automatically initiate a publishing refresh on certain events such as user logon or at specified intervals.
<a href="#">Test-CtxAppVServer</a>	Tests the given URL for the presence of App-V Management and Publishing servers.

# ConvertTo-CtxAppVLauncherArg

May 10, 2016  
Returns a string containing information to send to the App-V Launcher. You can plug this string directly into the Virtual Delivery Agent (VDA) to launch App-V applications.

Syntax

ConvertTo-CtxAppVLauncherArg [-AppVPublishingServer] <string> [[-PackageId] <String>] [-AppId] <String> -SeqLoc <String> -TargetInPackage <boolean> [<CommonParameters>]

ConvertTo-CtxAppVLauncherArg [-LauncherPath] <SwitchParameter> [<CommonParameters>]

Detailed Description

Returns a string containing information to send to the App-V Launcher. You can plug this string directly into the Virtual Delivery Agent (VDA) to launch App-V applications.

Related topics

Parameters

**-AppVPublishingServer**<>

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-PackageId**<>

The Package Id of the application.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AppId**<>

The App Id of the application.

Required?	True
Default Value	
Accept Pipeline Input?	false

**-SeqLoc**<>

The sequence location of the package.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-TargetInPackage**<>

Pass this as \$true if TargetInPackage field is "true" in the Manifest file for the particular App Id.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LauncherPath**<>

Gets the Citrix Launcher path. The Citrix Launcher is a component installed on the VDA.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

### Examples

#### ----- EXAMPLE 1 -----

ConvertTo-CtxAppVLauncherArg -AppVPublishingServer "http://appv-2k82-srv.blrstrm.com:8082" -PackageId "1bcb6993-10b1-4659-b9d4-e809e10cecdf\_5" -Appld "[[ProgramFilesX86]]Beyond Compa  
 Converts the arguments provided into a cmdlet to launch Beyond Compare from http://appv-server on the VDA.

#### ----- EXAMPLE 2 -----

ConvertTo-CtxAppVLauncherArg -LauncherPath  
 Gets the Citrix App-V Launcher path.

# Get-CtxAppVApplication

May 10, 2016

Enumerates all published App-V applications for a given Management server.

## Syntax

```
Get-CtxAppVApplication [-AppVManagementServer] <string> [<CommonParameters>]
```

## Detailed Description

Queries a given Management server and fetches all published applications for that server. Applications that are published but have no user Entitlements are not displayed by this cmdlet.

## Related topics

## Parameters

**-AppVManagementServer**<String>

App-V Management Server

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

## Examples

----- **EXAMPLE 1** -----

```
Get-CtxAppVApplication -AppVManagementServer "xmas-demo-appv"
```

Displays all published applications for the Management server "xmas-demo-appv".

# Get-CtxAppVApplicationInfo

May 10, 2016  
Enumerates information for a given application in a given package for a given Management server.

### Syntax

```
Get-CtxAppVApplicationInfo [-AppVManagementServer] <string> [-Appld] <String> [-PackageId] <string> [[-Property] <string[]>] [<CommonParameters>]
Get-CtxAppVApplicationInfo [-AppVManagementServer] <String> [[-InputObject] <AppVServeApplications[]>] [[-Property] <string[]>] [<CommonParameters>]
```

### Detailed Description

Queries a given Management server and fetches the requested information for a given application.

### Related topics

### Parameters

**-AppvManagementServer**<string>

Machine name of the App-V Management server. The name does not need to be specified as a Fully Qualified Domain Name (FQDN).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Appld**<string>

The App Id of the given application.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PackageId**<string>

The Package Id of the given application.

Required?	false
Default Value	
Accept Pipeline Input?	false

### Return Values

Citrix.VirtApp.Studio.PowerShellManager.AppVAppData

### Examples

----- **EXAMPLE 1** -----

Get-CtxAppVApplicationInfo -AppVManagementServer "xmas-demo-appv" -Appld "[[ProgramFilesX86]]Beyond Compare 3\BCompare.exe" -PackageId "1bcb6993-10b1-4659-b9d4-e809e10cecdf\_5"  
Fetches all application information for the Beyond Compare 3 application from the xmas-demo-appv Management server.

----- **EXAMPLE 2** -----

Get-CtxAppVApplicationInfo -AppVManagementServer "xmas-demo-appv" -Appld "[[ProgramFilesX86]]Beyond Compare 3\BCompare.exe" -PackageId "1bcb6993-10b1-4659-b9d4-e809e10cecdf\_5" -f  
Fetches only information about FTA and User Entitlements for the Beyond Compare 3 application from the xmas-demo-appv Management server.

----- **EXAMPLE 3** -----

Get-CtxAppVApplication -AppVManagementServer "xmas-demo-appv" | Get-CtxAppVApplicationInfo -AppVManagementServer "xmas-demo-appv"  
Pipelines the output of Get-CtxAppVApplication to Get-CtxAppVApplicationInfo to get the application properties of all the published applications on the xmas-demo-appv Management server.

# Get-CtxAppVServer

May 10, 2016

Returns URLs for App-V Publishing and Management servers contained in a Citrix App-V policy. Returned values are in string format.

## Syntax

```
Get-CtxAppVServer -ByteArray <byte[]> [-ConsumedByStudio <bool>] [<CommonParameters>]
```

## Detailed Description

Returns URLs for App-V Publishing and Management servers contained in a Citrix App-V policy. Returned values are in string format.

## Related topics

## Parameters

### **-ByteArray**<>

Specifies the Citrix App-V policy created using the New-CtxAppVServer cmdlet.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

### **-ConsumedByStudio**<>

If set to "true", outputs URLs for the App-V Publishing and Management servers. If set to "false", outputs both the URL and settings for the App-V Publishing server.

Required?	false
Default Value	
Accept Pipeline Input?	false

## Examples

### ----- EXAMPLE 1 -----

```
$config = Get-BrokerMachineConfiguration -Name appv*
```

```
Get-CtxAppVServer -ByteArray $config[0].Policy
```

Returns Publishing Server URL , Management Server URL configured in given policy





# Get-CtxAppVServerSetting

May 10, 2016

Returns settings for the specified App-V Publishing Server.

## Syntax

```
Get-CtxAppVServerSetting -AppVPublishingServer <string> [<CommonParameters>]
```

## Detailed Description

Returns settings for the specified App-V Publishing Server. For more information about these settings, see the App-V documentation on the Microsoft website at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

## Related topics

## Parameters

**-AppVPublishingServer<>**

URL of the Publishing Server for which settings are to be returned.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

## Return Values

Publishing Server settings. These settings are : GlobalRefreshEnabled;  
GlobalRefreshOnLogon;GlobalRefreshInterval;GlobalRefreshIntervalUnit;UserRefreshEnabled;  
UserRefreshOnLogon;UserRefreshInterval;UserRefreshIntervalUnit;

## Examples

----- **EXAMPLE 1** -----

```
Get-CtxAppVServerSetting -AppVPublishingServer http://AppV50Publishing:8082
```

This example returns settings associated with http://AppV50PublishingServer:8082 in the following format:

GlobalRefreshEnabled: false

GlobalRefreshOnLogon: false

GlobalRefreshInterval: 0

GlobalRefreshIntervalUnit: Day

UserRefreshEnabled: true

UserRefreshOnLogon: false

UserRefreshInterval: 0

UserRefreshIntervalUnit: Hour

# New-CtxAppVServer

May 10, 2016  
Creates a new Citrix App-V policy containing the specified App-V Management and Publishing Server URLs.

## Syntax

New-CtxAppVServer -PublishingServer <string> -ManagementServer <String> [-UserRefreshEnabled [<Boolean>]] [-UserRefreshOnLogon <Boolean>] [-UserRefreshInterval <int>] [-UserRefreshIntervalUnit <Enum>] [-GlobalRefreshEnabled [<Boolean>]] [-GlobalRefreshOnLogon <Boolean>] [-GlobalRefreshInterval <Int>] [-GlobalRefreshIntervalUnit <Enum>] [-CommonParameters< >]

## Detailed Description

Creates a new Citrix App-V policy containing the specified App-V Management and Publishing server URLs. Additionally, accepts Publishing Server settings that control how and when automatic refresh occurs on the VDA.

## Related topics

## Parameters

### -PublishingServer<>

URL of the Publishing server to add to the Citrix policy.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

### -ManagementServer<>

URL of the Management server to add to the Citrix policy.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

### -UserRefreshEnabled<>

Enables a refresh of packages published to user groups either at user logon or at a specified interval. For more information, see the App-V 5.0 documentation at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	false
Default Value	
Accept Pipeline Input?	false

### -UserRefreshOnLogon<>

Specifies whether or not to initiate a refresh of packages published to user groups on every user logon. For more information, see the App-V 5.0 documentation at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	false
Default Value	
Accept Pipeline Input?	false

### -UserRefreshInterval<>

Specifies the frequency at which to initiate a refresh of packages published to user groups. This can be either days or hours, as specified by the UserRefreshIntervalUnit setting. For more information, see the App-V 5.0 documentation at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	false
Default Value	
Accept Pipeline Input?	

Accept Pipeline Input?	false
------------------------	-------

#### -UserRefreshInterval<>

Specifies the frequency at which to initiate a refresh of packages published to user groups. This can be either days or hours, as specified by the UserRefreshIntervalUnit setting. For more information, see the App-V 5.0 documentation at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -UserRefreshIntervalUnit<>

Specifies the unit for the UserRefreshInterval setting. This can be set to either Hours (0) or Days (1). For more information, see the App-V 5.0 documentation at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -GlobalRefreshEnabled<>

Enables a refresh of packages published to machine groups either at user logon or at a specified interval. For more information, see the App-V 5.0 documentation at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -GlobalRefreshOnLogon<>

Specifies whether or not to initiate a refresh of packages published to machine groups on every user logon. For more information, see the App-V 5.0 documentation at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -GlobalRefreshInterval<>

Specifies the frequency at which to initiate a refresh of packages published to machine groups. This can be either days or hours, as specified by the GlobalRefreshIntervalUnit setting. Please refer to App-V 5.0 documentation for details. <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -GlobalRefreshIntervalUnit<>

Specifies the unit for the GlobalRefreshInterval setting. This can be set to either Hours (0) or Days (1). Please refer to App-V 5.0 documentation for details. <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	false
Default Value	
Accept Pipeline Input?	false

## Examples

### EXAMPLE 1

New-CtxAppVServer -ManagementServer http://AppV-Mgmt-Server:8080 -PublishingServer http://appV-Pub-Server:8082

Creates a new Citrix Policy for Management Server AppV-Mgmt-Server:8080 & Publishing Server: AppV-Mgmt-Server:8082. Default Publishing Server setting is used for http://AppV-Mgmt-Server:8082

Default values for publishing server settings used for Http://AppV-Mgmt-Server:8082 are:

GlobalRefreshEnabled = false; GlobalLogonRefresh = false ; GlobalIntervalRefreshInterval = 0; GlobalIntervalRefreshUnit = Day

UserRefreshEnabled = true; UserLogonRefresh = true ; UserIntervalRefreshInterval = 0; GlobalIntervalRefreshUnit = Day

### EXAMPLE 1

New-CtxAppVServer -ManagementServer http://AppV-Mgmt-Server:8080 -PublishingServer http://appV-Pub-Server:8082 -GlobalRefreshEnabled \$true -GlobalLogonRefresh \$true -GlobalRefreshInterval  
Creates a new Citrix Policy for Management Server AppV-Mgmt-Server:8080 & Publishing Server: AppV-Mgmt-Server:8082. User specified Publishing Server settings are used for AppV-Mgmt-Server:8082

Following values are used to configure Publishing Server : http://AppV-Mgmt-Server:8082

GlobalRefreshEnabled = True; GlobalLogonRefresh = True ; GlobalIntervalRefreshInterval = 2; GlobalIntervalRefreshUnit = Hour

UserRefreshEnabled = true; UserLogonRefresh = true ; UserIntervalRefreshInterval = 3; GlobalIntervalRefreshUnit = Hour

# Set-CtxAppVServerSetting

May 10, 2016

Specifies the App-V Publishing server settings to use on the VDA. These settings determine whether or not the App-V Client can automatically initiate a publishing refresh on certain events such as user logon or at specified intervals.

## Syntax

Set-CtxAppVServerSetting -AppVPublishingServer <string> [-UserRefreshEnabled [<Boolean>]] [-UserRefreshOnLogon <Boolean>] [-UserRefreshInterval <int>] [-UserRefreshIntervalUnit <Enum>] [-GlobalRefreshEnabled [<Boolean>]] [-GlobalRefreshOnLogon <Boolean>] [-GlobalRefreshInterval <int>] [-GlobalRefreshIntervalUnit <Enum>] [<CommonParameters>]

## Detailed Description

Specifies the App-V Publishing server settings to use on the VDA. These settings determine whether or not the App-V Client can automatically initiate a publishing refresh on certain events such as user logon or at specified intervals. Please refer to Microsoft App-V 5.0 documentation for more details on these settings : <http://technet.microsoft.com/en-us/library/jj687745.aspx>

## Related topics

## Parameters

### -AppVPublishingServer<>

URL of the Publishing Server for which settings are to be set.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

### -UserRefreshEnabled<>

Enables a refresh of packages published to user groups either at user logon or at a specified interval. For more information, see the App-V documentation on the Microsoft website at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	True
Default Value	
Accept Pipeline Input?	false

### -UserRefreshOnLogon<>

Specifies whether or not to initiate a refresh of packages published to user groups on every user logon. For more information, see the App-V documentation on the Microsoft website at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	true
Default Value	
Accept Pipeline Input?	false

### -UserRefreshInterval<>

Specifies the frequency at which to initiate a refresh of packages published to user groups. This can be either days or hours, as specified by the UserRefreshIntervalUnit setting. For more information, see the App-V documentation on the Microsoft website at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	true
Default Value	
Accept Pipeline Input?	false

### -UserRefreshIntervalUnit<>

Specifies the unit for the UserRefreshInterval setting. This can be set to either Hours (0) or Days (1). For more information, see the App-V documentation on the Microsoft website at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	true
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-GlobalRefreshEnabled<>**

Enables a refresh of packages published to machine groups either at user logon or at a specified interval. For more information, see the App-V documentation on the Microsoft website at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	True
Default Value	
Accept Pipeline Input?	false

#### **-GlobalRefreshOnLogon<>**

Specifies whether or not to initiate a refresh of packages published to machine groups on every user logon. For more information, see the App-V documentation on the Microsoft website at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-GlobalRefreshInterval<>**

Specifies the frequency at which to initiate a refresh of packages published to machine groups. This can be either days or hours, as specified by the GlobalRefreshIntervalUnit setting. For more information, see the App-V documentation on the Microsoft website at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-GlobalRefreshIntervalUnit<>**

Specifies the unit for the GlobalRefreshInterval setting. This can be set to either Hours (0) or Days (1). For more information, see the App-V documentation on the Microsoft website at <http://technet.microsoft.com/en-us/library/jj687745.aspx>

Required?	true
Default Value	
Accept Pipeline Input?	false

#### Examples

##### ----- EXAMPLE 1 -----

Set-CtxAppVServerSetting -AppVPublishingServer http://AppV50Publishing:8082 -GlobalRefreshEnabled \$true -GlobalRefreshOnLogon \$true -GlobalRefreshInterval 2 -GlobalRefreshIntervalUnit Hour -

# Test-CtxAppVServer

May 10, 2016

Tests the given URL for the presence of App-V Management and Publishing servers.

## Syntax

Test-CtxAppVServer [-AppVManagementServer] <string> [<CommonParameters>]

Test-CtxAppVServer [-AppVPublishingServer] <string> [<CommonParameters>]

## Detailed Description

Tests the given URL for the presence of App-V Management and Publishing servers.

## Related topics

## Parameters

**-AppVManagementServer**<String>

Machine name of the App-V Management server.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-AppVPublishingServer**<String>

Machine name of the App-V Publishing server.

Required?	false
Default Value	
Accept Pipeline Input?	false

## Return Values

Microsoft.AppvAgent.AppvClientPackage

## Examples

----- **EXAMPLE 1** -----

Test-CtxAppVServer -AppVManagementServer "xmas-demo-appv"



Tests whether "xmas-demo-appv" is a Management server or not. The name does not need to be specified as a Fully Qualified Domain Name (FQDN).

----- **EXAMPLE 2** -----

Test-CtxAppVServer -AppVPublishingServer "http://appvb2refserver.blrstrm.com:8082"

Tests whether "http://appvb2refserver.blrstrm.com:8082" is a Publishing server or not. Specify the full address, including FQDN and port number, of the Publishing server.

# Citrix.Broker.Admin.V2

May 10, 2016

## Overview

Name	Description
<a href="#">Broker AccessPolicy</a>	Controls client-connection-based access to desktop groups.
<a href="#">Broker Applications</a>	Describes how to publish and manage hosted applications.
<a href="#">Broker AssignmentPolicy</a>	Controls the automatic, permanent assignment of machines to users.
<a href="#">Broker Concepts</a>	Overview of the Citrix Broker.
<a href="#">Broker ConfigurationSlots</a>	Overview of assigning a collection of related settings to a desktop group.
<a href="#">Broker ControllerDiscovery</a>	Describes the way that machines providing published resources discover
<a href="#">Broker Desktops</a>	Describes desktop concepts and usage.
<a href="#">Broker EntitlementPolicy</a>	Controls end-user entitlement to desktop and application sessions provided
<a href="#">Broker ErrorHandling</a>	Describes broker errors generated by cmdlets and how to access them.
<a href="#">Broker Filtering</a>	Describes the common filtering options for XenDesktop cmdlets.
<a href="#">Broker Licensing</a>	Overview of broker licensing configuration.
<a href="#">Broker Machines</a>	Describes machine concepts and usage.
<a href="#">Broker Policies</a>	Overview of the site policies that control users' access to desktop and
<a href="#">Broker PostInstallPreConfiguration</a>	Describes how to configure the Citrix Broker Service port numbers, URL
<a href="#">Broker PowerManagement</a>	Describes power management of machines used for desktops and applications.
<a href="#">Broker RemotePC</a>	Overview of the Remote PC feature.

## Cmdlets

Name	Description
<a href="#">Add-BrokerApplication</a>	Adds applications to a desktop group.
<a href="#">Add-BrokerDesktopGroup</a>	Associate Remote PC desktop groups with the specified Remote PC catalog.
<a href="#">Add-BrokerMachine</a>	Adds one or more machines to a desktop group.
<a href="#">Add-BrokerMachineConfiguration</a>	Adds a machine configuration to a desktop group.
<a href="#">Add-BrokerMachinesToDesktopGroup</a>	Adds machines from a catalog to a desktop group.
<a href="#">Add-BrokerScope</a>	Add the specified catalog/desktop group to the given scope(s).
<a href="#">Add-BrokerTag</a>	Associate a tag with another object.
<a href="#">Add-BrokerUser</a>	Creates an association between a user and another broker object
<a href="#">Disconnect-BrokerSession</a>	Disconnect a session.
<a href="#">Export-BrokerDesktopPolicy</a>	Gets the site wide Citrix Group Policy settings.
<a href="#">Get-BrokerAccessPolicyRule</a>	Gets rules from the site's access policy.
<a href="#">Get-BrokerAdminFolder</a>	Get the admin folders in this site.
<a href="#">Get-BrokerAppAssignmentPolicyRule</a>	Gets application rules from the site's assignment policy.
<a href="#">Get-BrokerAppEntitlementPolicyRule</a>	Gets application rules from the site's entitlement policy.
<a href="#">Get-BrokerApplication</a>	Get the applications published on this site.
<a href="#">Get-BrokerApplicationInstance</a>	Gets the running applications on the desktops.
<a href="#">Get-BrokerAssignmentPolicyRule</a>	Gets desktop rules from the site's assignment policy.
<a href="#">Get-BrokerCatalog</a>	Gets catalogs configured for this site.

Name	Description
<a href="#">Get-BrokerConfigurationSlot</a>	Gets configuration slots configured for this site.
<a href="#">Get-BrokerConfiguredFTA</a>	Gets any file type associations configured for an application.
<a href="#">Get-BrokerConnectionLog</a>	Get entries from the site's session connection log.
<a href="#">Get-BrokerController</a>	Gets Controllers running broker services in the site.
<a href="#">Get-BrokerDBConnection</a>	Gets the database connection string for the specified data store used by the Broker Service.
<a href="#">Get-BrokerDBSchema</a>	Gets SQL scripts to create or maintain the database schema for the Citrix Broker Service.
<a href="#">Get-BrokerDBVersionChangeScript</a>	Gets an SQL service schema update script for the Citrix Broker Service.
<a href="#">Get-BrokerDelayedHostingPowerAction</a>	Gets power actions that are executed after a delay.
<a href="#">Get-BrokerDesktop</a>	Gets desktops configured for this site.
<a href="#">Get-BrokerDesktopGroup</a>	Gets broker desktop groups configured for this site.
<a href="#">Get-BrokerDesktopUsage</a>	Get usage history of desktop groups.
<a href="#">Get-BrokerEntitlementPolicyRule</a>	Gets desktop rules from the site's entitlement policy.
<a href="#">Get-BrokerHostingPowerAction</a>	Gets power actions queued for machines.
<a href="#">Get-BrokerHypervisorAlert</a>	Gets hypervisor alerts recorded by the controller.
<a href="#">Get-BrokerHypervisorConnection</a>	Gets hypervisor connections matching the specified criteria.
<a href="#">Get-BrokerIcon</a>	Get stored icons.
<a href="#">Get-BrokerImportedFTA</a>	Gets the imported file type associations.
<a href="#">Get-BrokerInstalledDbVersion</a>	Gets a list of all available database schema versions for the Broker Service.
<a href="#">Get-BrokerLease</a>	Gets stored leases.

Name	Description
<a href="#">Get-BrokerMachine</a>	Gets machines belonging to this site.
<a href="#">Get-BrokerMachineCommand</a>	Get the list of commands queued for delivery to a desktop.
<a href="#">Get-BrokerMachineConfiguration</a>	Gets machine configurations defined for this site.
<a href="#">Get-BrokerMachineStartMenuShortcutIcon</a>	Retrieves a Start Menu Shortcut icon from the specified machine.
<a href="#">Get-BrokerMachineStartMenuShortcuts</a>	Retrieves the Start Menu Shortcuts from the specified machine.
<a href="#">Get-BrokerPowerTimeScheme</a>	Gets power management time schemes for desktop groups.
<a href="#">Get-BrokerPrivateDesktop</a>	Get private desktops configured for this site.
<a href="#">Get-BrokerRebootCycle</a>	Gets one or more reboot cycles.
<a href="#">Get-BrokerRebootSchedule</a>	Gets one or more reboot schedules.
<a href="#">Get-BrokerRemotePCAccount</a>	Get RemotePCAccount entries configured for this site.
<a href="#">Get-BrokerResource</a>	Gets resources that a user can broker connections to.
<a href="#">Get-BrokerScopedObject</a>	Gets the details of the scoped objects for the Broker Service.
<a href="#">Get-BrokerServiceAddedCapability</a>	Gets any added capabilities for the Broker Service on the controller.
<a href="#">Get-BrokerServiceInstance</a>	Gets the service instance entries for the Broker Service.
<a href="#">Get-BrokerServiceStatus</a>	Gets the current state of the Broker Service on the controller.
<a href="#">Get-BrokerSession</a>	Gets a list of sessions.
<a href="#">Get-BrokerSessionLinger</a>	Gets one or more session lingering settings.
<a href="#">Get-BrokerSessionPreLaunch</a>	Gets one or more session pre-launch settings.
<a href="#">Get-BrokerSharedDesktop</a>	Get shared desktops configured for this site.

<b>Get-BrokerSite</b> <b>Name</b>	<b>Description</b>
Get-BrokerTag	Gets one or more tags.
Get-BrokerUnconfiguredMachine	Gets machines that have registered but are not yet configured in this site.
Get-BrokerUser	Gets broker users configured for this site.
Group-BrokerDesktop	Groups and counts desktops with the same value for a specified property.
Group-BrokerMachine	Groups and counts machines with the same value for a specified property.
Group-BrokerSession	Groups and counts sessions with the same value for a specified property.
Import-BrokerDesktopPolicy	Sets the site wide Citrix Group Policy settings for the site.
Move-BrokerAdminFolder	Moves a folder to another place in the hierarchy, optionally renaming it
Move-BrokerApplication	Move a published application from one admin folder to another
New-BrokerAccessPolicyRule	Creates a new rule in the site's access policy.
New-BrokerAdminFolder	Creates a new admin folder.
New-BrokerAppAssignmentPolicyRule	Creates a new application rule in the site's assignment policy.
New-BrokerAppEntitlementPolicyRule	Creates a new application rule in the site's entitlement policy.
New-BrokerApplication	Creates a new published application.
New-BrokerAssignmentPolicyRule	Creates a new desktop rule in the site's assignment policy.
New-BrokerCatalog	Adds a new catalog to the site.
New-BrokerConfigurationSlot	Creates a new configuration slot.
New-BrokerConfiguredFTA	Creates a file type association with a published application.

Name	Description
New-BrokerDelayedHostingPowerAction	Causes a power action to be queued after a delay.
New-BrokerDesktopGroup	Create a new desktop group for managing the brokering of groups of desktops.
New-BrokerEntitlementPolicyRule	Creates a new desktop rule in the site's entitlement policy.
New-BrokerHostingPowerAction	Creates a new action in the power action queue.
New-BrokerHypervisorConnection	Creates a new hypervisor connection.
New-BrokerIcon	Creates a new icon.
New-BrokerMachine	Adds a machine that can be used to run desktops and applications.
New-BrokerMachineCommand	Creates a new command to deliver to a desktop.
New-BrokerMachineConfiguration	Creates a new machine configuration associated with an existing configuration slot.
New-BrokerPowerTimeScheme	Creates a new power time scheme for a desktop group.
New-BrokerRebootSchedule	Creates a new reboot schedule for a desktop group.
New-BrokerRemotePCAccount	Create a new RemotePCAccount.
New-BrokerSessionLinger	Creates a new session linger setting for a desktop group.
New-BrokerSessionPreLaunch	Creates a new session pre-launch setting for a desktop group.
New-BrokerTag	Creates a new tag.
New-BrokerUser	Creates a new broker user object
Remove-BrokerAccessPolicyRule	Deletes a rule from the site's access policy.
Remove-BrokerAccessPolicyRuleMetadata	Deletes AccessPolicyRule Metadata from the AccessPolicyRule objects
Remove-BrokerAdminFolder	Removes an admin folder.

Name	Description
<a href="#">Remove-BrokerAdminFolderMetadata</a>	Deletes AdminFolder Metadata from the AdminFolder objects
<a href="#">Remove-BrokerAppAssignmentPolicyRule</a>	Deletes an application rule from the site's assignment policy.
<a href="#">Remove-BrokerAppEntitlementPolicyRule</a>	Deletes an application rule from the site's entitlement policy.
<a href="#">Remove-BrokerApplication</a>	Deletes one or more applications, or an association of an application.
<a href="#">Remove-BrokerApplicationInstanceMetadata</a>	Deletes ApplicationInstance Metadata from the ApplicationInstance objects
<a href="#">Remove-BrokerApplicationMetadata</a>	Deletes Application Metadata from the Application objects
<a href="#">Remove-BrokerAssignmentPolicyRule</a>	Deletes a desktop rule from the site's assignment policy.
<a href="#">Remove-BrokerAssignmentPolicyRuleMetadata</a>	Deletes AssignmentPolicyRule Metadata from the AssignmentPolicyRule objects
<a href="#">Remove-BrokerCatalog</a>	Removes catalogs from the site.
<a href="#">Remove-BrokerCatalogMetadata</a>	Deletes Catalog Metadata from the Catalog objects
<a href="#">Remove-BrokerConfigurationSlot</a>	Removes a configuration slot.
<a href="#">Remove-BrokerConfigurationSlotMetadata</a>	Deletes ConfigurationSlot Metadata from the ConfigurationSlot objects
<a href="#">Remove-BrokerConfiguredFTA</a>	Deletes one or more configured file type associations.
<a href="#">Remove-BrokerControllerMetadata</a>	Deletes Controller Metadata from the Controller objects
<a href="#">Remove-BrokerDelayedHostingPowerAction</a>	Cancels one or more delayed power actions.
<a href="#">Remove-BrokerDesktopGroup</a>	Remove desktop groups from the system or remove them from a Remote PC catalog.
<a href="#">Remove-BrokerDesktopGroupMetadata</a>	Deletes DesktopGroup Metadata from the DesktopGroup objects
<a href="#">Remove-BrokerEntitlementPolicyRule</a>	Deletes a desktop rule from the site's entitlement policy.
<a href="#">Remove-BrokerEntitlementPolicyRuleMetadata</a>	Deletes EntitlementPolicyRule Metadata from the



Name	Description
<a href="#">Remove-BrokerHostingPowerAction</a>	EntitlementPolicyRule objects Cancel one or more pending power actions.
<a href="#">Remove-BrokerHostingPowerActionMetadata</a>	Deletes HostingPowerAction Metadata from the HostingPowerAction objects
<a href="#">Remove-BrokerHypervisorAlertMetadata</a>	Deletes HypervisorAlert Metadata from the HypervisorAlert objects
<a href="#">Remove-BrokerHypervisorConnection</a>	Removes a hypervisor connection from the system.
<a href="#">Remove-BrokerHypervisorConnectionMetadata</a>	Deletes HypervisorConnection Metadata from the HypervisorConnection objects
<a href="#">Remove-BrokerIcon</a>	Remove an icon.
<a href="#">Remove-BrokerIconMetadata</a>	Deletes Icon Metadata from the Icon objects
<a href="#">Remove-BrokerImportedFTA</a>	Deletes one or more imported file type associations.
<a href="#">Remove-BrokerLease</a>	Remove the specified lease in the Database.
<a href="#">Remove-BrokerLeaseMetadata</a>	Deletes Lease Metadata from the Lease objects
<a href="#">Remove-BrokerMachine</a>	Removes one or more machines from its desktop group or catalog.
<a href="#">Remove-BrokerMachineCommand</a>	Cancel a pending command queued for delivery to a desktop.
<a href="#">Remove-BrokerMachineCommandMetadata</a>	Deletes MachineCommand Metadata from the MachineCommand objects
<a href="#">Remove-BrokerMachineConfiguration</a>	Deletes a machine configuration from the site or removes the association from a desktop group.
<a href="#">Remove-BrokerMachineConfigurationMetadata</a>	Deletes MachineConfiguration Metadata from the MachineConfiguration objects
<a href="#">Remove-BrokerMachineMetadata</a>	Deletes Machine Metadata from the Machine objects
<a href="#">Remove-BrokerPowerTimeScheme</a>	Deletes an existing power time scheme.
<a href="#">Remove-BrokerPowerTimeSchemeMetadata</a>	Deletes PowerTimeScheme Metadata from the PowerTimeScheme objects

<b>Remove-BrokerRebootCycleMetadata</b> Name	Description
Remove-BrokerRebootSchedule	Removes the reboot schedule.
Remove-BrokerRemotePCAccount	Delete RemotePCAccounts from the system.
Remove-BrokerScope	Remove the specified catalog/desktop group from the given scope(s).
Remove-BrokerSessionLinger	Removes a session linger setting.
Remove-BrokerSessionMetadata	Deletes Session Metadata from the Session objects
Remove-BrokerSessionPreLaunch	Removes a session pre-launch setting.
Remove-BrokerSiteMetadata	Deletes Site Metadata from the Site objects
Remove-BrokerTag	Removes a tag from the system or clears a tag to object association.
Remove-BrokerTagMetadata	Deletes Tag Metadata from the Tag objects
Remove-BrokerUser	Remove broker user objects from another broker object
Rename-BrokerAccessPolicyRule	Renames a rule in the site's access policy.
Rename-BrokerAdminFolder	Renames a folder
Rename-BrokerAppAssignmentPolicyRule	Renames an application rule in the site's assignment policy.
Rename-BrokerAppEntitlementPolicyRule	Renames an application rule in the site's entitlement policy.
Rename-BrokerApplication	Renames an application.
Rename-BrokerAssignmentPolicyRule	Renames a desktop rule in the site's assignment policy.
Rename-BrokerCatalog	Renames a catalog.
Rename-BrokerDesktopGroup	Renames a desktop group.
Rename-BrokerEntitlementPolicyRule	Renames a desktop rule in the site's entitlement policy.

Name	Description
<a href="#">Rename-BrokerMachineConfiguration</a>	Renames a machine configuration.
<a href="#">Rename-BrokerPowerTimeScheme</a>	Changes the name of an existing power time scheme.
<a href="#">Rename-BrokerTag</a>	Rename one or more tags.
<a href="#">Reset-BrokerLicensingConnection</a>	Resets the broker's license server connection.
<a href="#">Reset-BrokerServiceGroupMembership</a>	Reloads the access permissions and configuration service locations for the Broker Service.
<a href="#">Send-BrokerSessionMessage</a>	Sends a message to a session.
<a href="#">Set-BrokerAccessPolicyRule</a>	Modifies an existing rule in the site's access policy.
<a href="#">Set-BrokerAccessPolicyRuleMetadata</a>	Creates/Updates metadata key-value pairs for AccessPolicyRule
<a href="#">Set-BrokerAdminFolderMetadata</a>	Creates/Updates metadata key-value pairs for AdminFolder
<a href="#">Set-BrokerAppAssignmentPolicyRule</a>	Modifies an existing application rule in the site's assignment policy.
<a href="#">Set-BrokerAppEntitlementPolicyRule</a>	Modifies an existing application rule in the site's entitlement policy.
<a href="#">Set-BrokerApplication</a>	Changes the settings of an application to the value specified in the command.
<a href="#">Set-BrokerApplicationInstanceMetadata</a>	Creates/Updates metadata key-value pairs for ApplicationInstance
<a href="#">Set-BrokerApplicationMetadata</a>	Creates/Updates metadata key-value pairs for Application
<a href="#">Set-BrokerAssignmentPolicyRule</a>	Modifies an existing desktop rule in the site's assignment policy.
<a href="#">Set-BrokerAssignmentPolicyRuleMetadata</a>	Creates/Updates metadata key-value pairs for AssignmentPolicyRule
<a href="#">Set-BrokerCatalog</a>	Sets the properties of a catalog.
<a href="#">Set-BrokerCatalogMetadata</a>	Creates/Updates metadata key-value pairs for Catalog
<a href="#">Set-BrokerConfigurationSlotMetadata</a>	Creates/Updates metadata key-value pairs for ConfigurationSlot

Name	Description
<a href="#">Set-BrokerControllerMetadata</a>	Creates/Updates metadata key-value pairs for Controller
<a href="#">Set-BrokerDBConnection</a>	Configures a database connection for the Broker Service.
<a href="#">Set-BrokerDesktopGroup</a>	Adjusts the settings of a broker desktop group.
<a href="#">Set-BrokerDesktopGroupMetadata</a>	Creates/Updates metadata key-value pairs for DesktopGroup
<a href="#">Set-BrokerEntitlementPolicyRule</a>	Modifies an existing desktop rule in the site's entitlement policy.
<a href="#">Set-BrokerEntitlementPolicyRuleMetadata</a>	Creates/Updates metadata key-value pairs for EntitlementPolicyRule
<a href="#">Set-BrokerHostingPowerAction</a>	Changes the priority of one or more pending power actions.
<a href="#">Set-BrokerHostingPowerActionMetadata</a>	Creates/Updates metadata key-value pairs for HostingPowerAction
<a href="#">Set-BrokerHypervisorAlertMetadata</a>	Creates/Updates metadata key-value pairs for HypervisorAlert
<a href="#">Set-BrokerHypervisorConnection</a>	Sets the properties of a hypervisor connection.
<a href="#">Set-BrokerHypervisorConnectionMetadata</a>	Creates/Updates metadata key-value pairs for HypervisorConnection
<a href="#">Set-BrokerIconMetadata</a>	Creates/Updates metadata key-value pairs for Icon
<a href="#">Set-BrokerLeaseMetadata</a>	Creates/Updates metadata key-value pairs for Lease
<a href="#">Set-BrokerMachine</a>	Sets properties on a machine.
<a href="#">Set-BrokerMachineCatalog</a>	Moves one or more machines into a different catalog.
<a href="#">Set-BrokerMachineCommandMetadata</a>	Creates/Updates metadata key-value pairs for MachineCommand
<a href="#">Set-BrokerMachineConfiguration</a>	Sets the properties of a machine configuration.
<a href="#">Set-BrokerMachineConfigurationMetadata</a>	Creates/Updates metadata key-value pairs for MachineConfiguration
<a href="#">Set-BrokerMachineMaintenanceMode</a>	Sets whether the specified machine(s) are in maintenance mode.

Name	Description
<a href="#">Set-BrokerMachineMetadata</a>	Creates/Updates metadata key-value pairs for Machine
<a href="#">Set-BrokerPowerTimeScheme</a>	Modifies an existing power time scheme.
<a href="#">Set-BrokerPowerTimeSchemeMetadata</a>	Creates/Updates metadata key-value pairs for PowerTimeScheme
<a href="#">Set-BrokerPrivateDesktop</a>	Change the settings of a private desktop.
<a href="#">Set-BrokerRebootCycleMetadata</a>	Creates/Updates metadata key-value pairs for RebootCycle
<a href="#">Set-BrokerRebootSchedule</a>	Updates the values of one or more desktop group reboot schedules.
<a href="#">Set-BrokerRemotePCAccount</a>	Modify one or more RemotePCAccounts.
<a href="#">Set-BrokerSession</a>	Sets properties of a session.
<a href="#">Set-BrokerSessionLinger</a>	Updates the values of one or more desktop group session linger settings.
<a href="#">Set-BrokerSessionMetadata</a>	Creates/Updates metadata key-value pairs for Session
<a href="#">Set-BrokerSessionPreLaunch</a>	Updates the values of one or more desktop group session pre-launch settings.
<a href="#">Set-BrokerSharedDesktop</a>	Change the settings of a shared desktop.
<a href="#">Set-BrokerSite</a>	Changes the overall settings of the current XenDesktop broker site.
<a href="#">Set-BrokerSiteMetadata</a>	Creates/Updates metadata key-value pairs for Site
<a href="#">Set-BrokerTagMetadata</a>	Creates/Updates metadata key-value pairs for Tag
<a href="#">Start-BrokerCatalogPvdImagePrepare</a>	Start the PVD Image prepare process in the Broker for the machines in the specified catalog(s).
<a href="#">Start-BrokerMachinePvdImagePrepare</a>	Start the PVD Image prepare process in the Broker for the specified machine(s).
<a href="#">Start-BrokerNaturalRebootCycle</a>	Reboots all machines from the specified catalog when they are not in use.

<b>Name</b>	<b>Description</b>
Start-BrokerRebootCycle	Creates and starts a reboot cycle for each desktop group that contains machines from the specified catalog.
Stop-BrokerRebootCycle	Cancels the specified reboot cycle.
Stop-BrokerSession	Stop or log off a session.
Test-BrokerAccessPolicyRuleNameAvailable	Determine whether the proposed AccessPolicyRule Name is available for use.
Test-BrokerAppAssignmentPolicyRuleNameAvailable	Determine whether the proposed AppAssignmentPolicyRule Name is available for use.
Test-BrokerAppEntitlementPolicyRuleNameAvailable	Determine whether the proposed AppEntitlementPolicyRule Name is available for use.
Test-BrokerApplicationNameAvailable	Determine whether the proposed Application Name is available for use.
Test-BrokerAssignmentPolicyRuleNameAvailable	Determine whether the proposed AssignmentPolicyRule Name is available for use.
Test-BrokerCatalogNameAvailable	Determine whether the proposed Catalog Name is available for use.
Test-BrokerDBConnection	Tests whether a database is suitable for use by the Citrix Broker Service.
Test-BrokerDesktopGroupNameAvailable	Determine whether the proposed DesktopGroup Name is available for use.
Test-BrokerEntitlementPolicyRuleNameAvailable	Determine whether the proposed EntitlementPolicyRule Name is available for use.
Test-BrokerLicenseServer	Tests whether or not a license server can be used by the broker.
Test-BrokerMachineNameAvailable	Determine whether the proposed Machine MachineName is available for use.
Test-BrokerPowerTimeSchemeNameAvailable	Determine whether the proposed PowerTimeScheme Name is available for use.
Test-BrokerRemotePCAccountNameAvailable	Determine whether the proposed RemotePCAccount OU is available for use.
Update-BrokerImportedFTA	Imports or updates all of the file type associations for the specified worker.
Update-BrokerLocalLeaseCache	Flushes the local lease cache.

Name	Description
e-BrokerNameCache	Administrative operations on the user and machine name cache.

# about\_Broker\_AccessPolicy

May 10, 2016

## TOPIC

Citrix Broker SDK - Access Policy

## SHORT DESCRIPTION

Controls client-connection-based access to desktop groups.

## LONG DESCRIPTION

The site's access policy defines rules controlling a user's access to desktop groups. Access checks are based on details of the user's connection from their user device to the site. Think of the access policy informally as a connection-based firewall.

The access policy comprises a set of rules. Each rule:

- o Relates to a single desktop group.
- o Contains a set of connection filters and access right controls.

Multiple rules can apply to the same desktop group.

By default, users have no access to any desktop group within a site. A user gains access to a group when their connection's details match the connection filters of one or more rules in the access policy.

The access policy also grants control rights over desktop and application sessions. For example, it can specify which protocols are allowed for a connection from a given endpoint, and whether the user can restart their machine.

To use a resource published by a site, the user must have both access to the desktop group that contains it, and an entitlement to use the resource. Entitlements are typically granted by the site entitlement and assignment policies; see [about\\_Broker\\_Policies](#) for more information.

## ACCESS POLICY RULES

A single access policy rule relates to a single specified desktop group and comprises a set of connection filters and access right grants as described below.

Each rule can be individually enabled or disabled. A disabled rule is ignored when the access policy is evaluated.

## CONNECTION FILTERS OVERVIEW

The connection filters in an access policy rule comprise the following:

- o Local/remote client (SmartAccess) filters
- o Client IP address filters
- o Client name filters
- o User filters



All filters have an include and exclude form that can be individually enabled or disabled. For a rule to be considered when the access policy is evaluated, at least one connection include filter must be enabled. By default, all filters, both include and exclude, are disabled.

The detailed behavior of connection filters is covered later.

## ACCESS RIGHT CONTROLS OVERVIEW

The access right controls in an access policy rule comprise the following:

- o Allowed protocols
- o Whether machine restart, or programmatic session logoff, is allowed

The detailed behavior of access right controls is covered later.

## DETAILS OF CONNECTION FILTERS

To gain access to a desktop group the user's connection must match the filter criteria of at least one access policy rule for that group.

To match a rule, a connection must match all the rule's enabled include connection filters and must not match any of the rule's enabled exclude filters. That is, entries in exclude filters take priority.

Because all rules are evaluated independently, if an exclude filter match prevents a connection gaining access to a desktop group through one rule, the connection may still gain access to the same group through a different rule.

The filters are described in pairs below, but within a single rule a match against any exclude filter prevents a connection from gaining access through that rule irrespective of which include filters within the rule were also matched.

## SMARTACCESS FILTERS

SmartAccess filters allow filtering based on whether the client is directly connected (for example over a local area network (LAN)) or through Access Gateway. For connections through Access Gateway further filtering can be performed based on the tags supplied from Access Gateway itself. The key properties of SmartAccess filters are:

- o AllowedConnections (include filter: Filtered, NotViaAG, ViaAG)

This property controls the behavior of the include filter. The default value is Filtered. The possible values are as follows:

-- Filtered (default)

The filter matches any user connection not through Access Gateway. In addition, the filter may match user connections through Access Gateway subject to the following: if the IncludedSmartAccessTags property is empty, any such connection matches. However, if the property is not empty at least one SmartAccess tag from the filter property must match a SmartAccess tag supplied with the user's

connection.

#### -- NotViaAG

The filter matches only user connections not through Access Gateway. The contents of the IncludedSmartAccessTags property are ignored.

#### -- ViaAG

The filter matches only user connections through Access Gateway. If the IncludedSmartAccessTags property is empty, any such connection matches. However, if the property is not empty at least one SmartAccess tag from the filter property must match a SmartAccess tag supplied with the user's connection.

The IncludedSmartAccessTags property referred to above forms part of the include filter and is used if AllowedConnections is set to Filtered or ViaAG. It comprises a simple list of Access Gateway tags that are matched against those provided in the user's connection details.

#### o ExcludedSmartAccessTags (exclude filter)

A simple list of Access Gateway tags that are matched against those provided in the user's connection details. If any tag in the list matches one supplied with the user's connection, the user's connection does not match the access policy rule containing the filter.

The exclude filter has no setting corresponding to the AllowedConnections property so its behavior is determined only by the ExcludedSmartAccessTags property.

SmartAccess filters are typically used to control local (through a LAN) and remote (through Access Gateway) access to a site. A common model is to define two access policy rules for a group, one for local access and one for remote. The remote rule might impose restrictions on the user device having appropriate antivirus software installed, and potentially exclude certain user groups who would be allowed access over the corporate LAN (see USER FILTERS below).

## CLIENT IP FILTERS

Client IP filters allow filtering based on the IP address of the user's device. The key properties of client IP filters are:

#### o IncludedClientIPs (include filter)

A simple list of numeric IP address ranges that are matched against the user device. The filter matches if the device address falls within any of the ranges in the list.

- o ExcludedClientIPs (exclude filter)

A simple list of numeric IP address ranges that are matched against the user device. If any entry matches the device address, the user's connection does not match the access policy rule containing the filter.

An IP address range in these filters can be specified as a simple IP address or as a range using a conventional subnet mask.

## CLIENT NAME FILTERS

Client name filters allow filtering based on the name of the user's device. The key properties of client name filters are:

- o IncludedClientNames (include filter)

A simple list of names that are matched against the user device. The filter matches if the device name matches any value in the list.

- o ExcludedClientNames (exclude filter)

A simple list of device names that are matched against the user device. If any entry matches the device name, the user's connection does not match the access policy rule containing the filter.

Note: The form of the device name presented to the site depends on the site configuration. For example, by default in these filters you cannot use the form of the name presented by Web Interface.

## USER FILTERS

User filters allow filtering based on the identity of the user. The key properties of user filters are:

- o AllowedUsers (include filter: Filtered, AnyAuthenticated, Any)

This property controls the behavior of the include filter. The default value is Filtered. The possible values are as follows:

- Filtered (default)

The filter matches if the user's logon token contains one or more users or user groups matching those specified in the IncludedUsers property. The IncludedUsers property is a simple list of users or user groups and is used only when the AllowedUsers property is set to Filtered.

- AnyAuthenticated

The filter matches any authenticated Microsoft Windows user. The

contents of the IncludedUsers property are ignored.

-- Any

The filter matches any user. The contents of the IncludedUsers property are ignored. In the current implementation this value is handled in the same way as AnyAuthenticated.

- o ExcludedUsers (exclude filter)

A simple list of users or user groups. If any entry matches one in the user's logon token, the user's connection does not match the access policy rule containing the filter.

The exclude filter has no setting corresponding to the AllowedUsers property so its behavior is determined only by the ExcludedUsers property.

## DETAILS OF ACCESS RIGHT CONTROLS

The access right controls of an access policy rule determine rights that the user has over any desktop or application session that they obtain from the rule's desktop group.

The rights apply only if the user's connection matches the connection filters of a rule, and only if the user also has an entitlement to a desktop or application session from the associated desktop group.

The following properties define the access rights:

- o AllowedProtocols

A simple list of communication protocols over which connections can be made to resources published by the desktop group. For example, use this to restrict protocols with high bandwidth requirements to connections originating from a LAN.

- o AllowRestart

For single-session power-managed machines, allows the user to restart the machine (the machine is powered off using the capabilities of its hypervisor). For multi-session machines the user's session is simply logged off.

For a given connection, if multiple rules result in access being granted to a session from a desktop group, the user's rights are the combined rights of all the rules that matched for that group. The allowed protocol lists from all the rules are combined, and the user is granted restart rights if any one rule has AllowRestart set.

SEE ALSO

about\_Broker\_Policies  
about\_Broker\_AssignmentPolicy  
about\_Broker\_EntitlementPolicy  
New-BrokerAccessPolicyRule  
Get-BrokerAccessPolicyRule  
Set-BrokerAccessPolicyRule  
Rename-BrokerAccessPolicyRule  
Remove-BrokerAccessPolicyRule  
New-BrokerAssignmentPolicyRule  
New-BrokerEntitlementPolicyRule  
New-BrokerAppAssignmentPolicyRule  
New-BrokerAppEntitlementPolicyRule  
Add-BrokerUser

# about\_Broker\_Applications

May 10, 2016

## TOPIC

Citrix Broker SDK - Applications

## SHORT DESCRIPTION

Describes how to publish and manage hosted applications.

## LONG DESCRIPTION

Published applications allow your users to launch applications as if they were installed on their devices when in fact they are hosted remotely. The applications are normally launched in a seamless window, meaning that users see only the application window and not an additional desktop.

Published applications are hosted on either desktop operating systems or server operating systems. Applications are published to users and desktop groups. As such, conceptually they exist "on top" of desktop groups, which are themselves built on top of catalogs. See about\_Broker\_Concepts for more information on catalogs.

There are two types of applications:

- o HostedOnDesktop - application runs on a remote machine and is displayed on the local client desktop.
- o InstalledOnClient - application is installed and run on the local client machine and has its window overlaid on to a remote desktop.

## HOSTING APPLICATIONS

There are three main ways of hosting an application: using a private single-session VDI desktop, a shared single-session VDI desktop, and on shared multi-session server operating systems.

An application hosted on a shared single-session VDI desktop, when launched, is hosted on a random machine within the desktop group. An application hosted on a private single-session VDI desktop ensures that when a user launches the application it is always hosted on the same machine. An application hosted on shared multi-session server operating systems ensures that when a user launches the application it is always hosted on one of the least loaded servers in the desktop group.

You control how the application is hosted based on the kind of desktop group that you choose. Shared desktop groups randomly select a machine, and these desktop groups can only be created from catalogs with a Random AllocationType. On the other hand, private desktop groups host the application on the same machine for that user every time, and these desktop group types can only be created from catalogs with a Permanent AllocationType.

## USER ACCESS & ASSIGNMENT

Users are not assigned an application directly, but instead they are required to first have access to the desktop groups on which the applications are published through access policy rules.

With shared desktop groups, access to a published application also needs an application entitlement policy rule. An application entitlement policy rule defines the set of users who are allowed per-session access to desktops in a specified desktop group.

With private desktop groups, access to a published application also needs an application assignment policy rule. An application assignment policy rule defines the set of users who are allowed access to a single application session in a desktop group.

Users are assigned private machines in one of two ways: pre-assigned or assign-on-first-use. Pre-assigned means that individual user accounts have been specified for the machines within that desktop group. A single machine can only have a single user account (not a group account) assigned to it, and likewise a user can only be assigned a single machine within a desktop group.

Assign-on-first-use requires less configuration. Machines are assigned to users the first time they log on. Rather than allocating users directly to machines, instead users are assigned to the private desktop group, either through individual user accounts or through user group accounts. Then, when a user assigned to the desktop group logs on, a machine is automatically allocated to them.

## USER VISIBILITY

Users can be further filtered by configuring the visibility filter on top of the application. This restricts the application to a subset of the users that were granted access by the access policy and entitlement/assignment policy rules on the desktop group.

## MULTIPLE DESKTOP GROUPS

You can publish an application to multiple desktop groups, which have to be of the same kind. Generally, an application that is published only to private desktop groups is referred to as a "private application". An application that is only published to shared desktop groups is referred to as a "shared application".

## LICENSING

Hosted applications need the appropriate licenses to exist on the license server to be functional.

## NAMING

Applications have three names that identify them: the Name, BrowserName and the PublishedName. The Name is unique for each application, is not visible to the user, and is primarily used for administrative purpose. The BrowserName is unique across the entire site, and is primarily used internally. The PublishedName is not unique and is the name seen by end users who have access to this application.

When creating an application, you only need to specify the Name. If no BrowserName is specified one is automatically generated. If no PublishedName is specified by default it is the same as the Name.

The following special characters are not allowed in the Name, BrowserName or the PublishedName properties: \ / ; : # . \* ? = &lt; &gt; | [ ] ( ) " ' .

In addition the ` character is not allowed in the Name property.

To change the `PublishedName` or `BrowserName` of an application you must use the `Set-BrokerApplication` cmdlet.

To change the Name of an application you must use the `Rename-BrokerApplication` cmdlet.

## OPTIONAL

You can configure file-type associations for applications, so that if a user double-clicks a document icon on their device, a published application automatically starts. For more information, see the help for `New-BrokerConfiguredFTA`.

You can apply tags to applications as another convenient way to further organize (and search for) applications. For more information, see the help for `New-BrokerTag`.

## CMDLETS

### `New-BrokerApplication`

Creates an application for publishing after the needed desktop groups, access policy and entitlement/assignment policy rules have been created.

### `Add-BrokerApplication`

Adds one or more applications to a desktop group.

### `Get-BrokerApplication`

Gets one or more applications.

### `Remove-BrokerApplication`

Deletes one or more applications or it can be used to delete just the association of an application to a desktop group.

### `Rename-BrokerApplication`

Changes the Name of an application.

### `Set-BrokerApplication`

Changes the settings of application objects, except their names.

## EXAMPLES



## SHARED APPLICATION (SingleSession)

You have created a catalog with a Random AllocationType and SingleSession SessionSupport. It contains two machines called worker1 and worker2, both in the ACME domain. You publish an application with shared hosting as follows:

```
C:\PS> Write-Host "Create a desktop group, and add machines to it"
C:\PS> $dg = New-BrokerDesktopGroup "Shared Application Group"
-DesktopKind Shared -DeliveryType AppsOnly -SessionSupport 'SingleSession'
C:\PS> $m1 = Get-BrokerMachine -MachineName "ACME\worker1"
C:\PS> $m2 = Get-BrokerMachine -MachineName "ACME\worker2"
C:\PS> Add-BrokerMachine $m1 -DesktopGroup $dg
C:\PS> Add-BrokerMachine $m2 -DesktopGroup $dg
C:\PS> Write-Host "Create access policy rule for desktop group"
C:\PS> New-BrokerAccessPolicyRule -Name "Shared Application Group - Allow Everyone Access"
-Enabled $true -DesktopGroupUid $dg.Uid -IncludedUserFilterEnabled $true
-AllowedProtocols @("HDX") -AllowedUsers AnyAuthenticated
C:\PS> Write-Host "Create application entitlement policy for desktop group"
C:\PS> New-BrokerAppEntitlementPolicyRule -Name "Shared Application Group - App Entitlement"
-IncludedUsers 'ACME\Domain Users' -DesktopGroupUid $dg.Uid -Enabled $true
C:\PS> Write-Host "Create an application"
C:\PS> New-BrokerApplication -Name "Notepad" -PublishedName "Notepad"
-CommandLineExecutable "notepad.exe" -DesktopGroup $dg.Uid
```

In turn, this set of commands: creates a shared single session desktop group for applications delivery; adds two machines (from a catalog with a Random AllocationType and SingleSession SessionSupport) to the desktop group; creates access policy and application entitlement policy rules; creates an application; specifies its name, the executable, and links the application to the desktop group that will host it.

## SHARED APPLICATION (MultiSession)

You have created a catalog with a Random AllocationType and MultiSession SessionSupport. It contains one machine called worker1 in the ACME domain. You publish an application with shared hosting as follows:

```
C:\PS> Write-Host "Create a desktop group, and add machines to it"
C:\PS> $dg = New-BrokerDesktopGroup "Shared Application Group"
-DesktopKind Shared -DeliveryType AppsOnly -SessionSupport 'MultiSession'
C:\PS> $m1 = Get-BrokerMachine -MachineName "ACME\worker1"
C:\PS> Add-BrokerMachine $m1 -DesktopGroup $dg
C:\PS> Write-Host "Create access policy rule for desktop group"
C:\PS> New-BrokerAccessPolicyRule -Name "Shared Application Group - Allow Everyone Access"
-Enabled $true -DesktopGroupUid $dg.Uid -IncludedUserFilterEnabled $true
-AllowedProtocols @("HDX") -AllowedUsers AnyAuthenticated
C:\PS> Write-Host "Create application entitlement policy for desktop group"
C:\PS> New-BrokerAppEntitlementPolicyRule -Name "Shared Application Group - App Entitlement"
-IncludedUsers 'ACME\Domain Users' -DesktopGroupUid $dg.Uid -Enabled $true
C:\PS> Write-Host "Create an application"
```

```
C:\PS> New-BrokerApplication -Name "Notepad" -PublishedName "Notepad"
-CommandLineExecutable "notepad.exe" -DesktopGroup $dg.Uid
```

In turn, this set of commands: creates a shared multi session desktop group for applications delivery; adds one machine (from a catalog with a Random AllocationType and MultiSession SessionSupport) to the desktop group; creates access policy and application entitlement policy rules; creates an application; specifies its name, the executable, and links the application to the desktop group that will host it.

#### PRIVATE PRE-ASSIGNED APPLICATION

You have a catalog with a Permanent AllocationType and SingleSession SessionSupport. It contains two machines called worker1 and worker2, both in the ACME domain. You publish an application with private hosting using pre-assigned machines as follows:

```
C:\PS> Write-Host "Create a desktop group, and add machines to it"
C:\PS> $dg = New-BrokerDesktopGroup "Private App G1" -DesktopKind Private
-DeliveryType AppsOnly -SessionSupport 'SingleSession'
C:\PS> $m1 = Get-BrokerMachine -MachineName "ACME\worker1"
C:\PS> $m2 = Get-BrokerMachine -MachineName "ACME\worker2"
C:\PS> Add-BrokerMachine $m1 -DesktopGroup $dg
C:\PS> Add-BrokerMachine $m2 -DesktopGroup $dg
C:\PS> Write-Host "Setting access policy rule for desktop group"
C:\PS> New-BrokerAccessPolicyRule -Name "Private App G1 - Allow Everyone Access"
-Enabled $true -DesktopGroupUid $dg.Uid -IncludedUserFilterEnabled $true
-AllowedProtocols @("HDX") -AllowedUsers AnyAuthenticated
C:\PS> Write-Host "Pre-Assign users to the machines"
C:\PS> Add-BrokerUser "ACME\user1" -Machine $m1
C:\PS> Add-BrokerUser "ACME\user2" -Machine $m2
C:\PS> Write-Host "Create an application"
C:\PS> New-BrokerApplication -Name "Notepad" -PublishedName "Notepad"
-CommandLineExecutable "notepad.exe" -DesktopGroup $dg.Uid
```

In turn, this set of commands: creates a private desktop group for applications delivery; adds two machines (from a catalog with a Permanent AllocationType and SingleSession SessionSupport) to the desktop group; creates access policy; creates two user objects; pre-assigns a user to each machine; creates the application; assigns users to it; and links the application to the desktop group that will host it.

#### PRIVATE, ASSIGN-ON-FIRST-USE APPLICATION

You have a catalog created with a Permanent AllocationType and SingleSession SessionSupport. It contains two machines called worker1 and worker2, both in the ACME domain. You publish an application with private hosting using assign-on-first-use machines as follows:

```
C:\PS> Write-Host "Create a desktop group, and add machines to it"
```

```

C:\PS> $dg = New-BrokerDesktopGroup "Private App G2" -DesktopKind Private
-DeliveryType AppsOnly -SessionSupport 'SingleSession'
C:\PS> $m1 = Get-BrokerMachine -MachineName "ACME\worker1"
C:\PS> $m2 = Get-BrokerMachine -MachineName "ACME\worker2"
C:\PS> Add-BrokerMachine $m1 -DesktopGroup $dg
C:\PS> Add-BrokerMachine $m2 -DesktopGroup $dg
C:\PS> Write-Host "Setting access policy rule for desktop group"
C:\PS> New-BrokerAccessPolicyRule -Name "Private App G1 - Allow Everyone Access"
-Enabled $true -DesktopGroupUid $dg.Uid -IncludedUserFilterEnabled $true
-AllowedProtocols @("HDX") -AllowedUsers AnyAuthenticated
C:\PS> Write-Host "Create application assignment policy for desktop group"
C:\PS> New-BrokerAppAssignmentPolicyRule -Name "Private App G2 - App Assignment"
-IncludedUsers 'ACME\Domain Users' -DesktopGroupUid $dg.Uid -Enabled $true
C:\PS> Write-Host "Create an application"
C:\PS> New-BrokerApplication -Name "Notepad" -PublishedName "Notepad"
-CommandLineExecutable "notepad.exe" -DesktopGroup $dg.Uid

```

In turn, this set of commands: creates a private single session desktop group for applications delivery; adds two machines (from a catalog with a Permanent AllocationType and SingleSession SessionSupport) to the desktop group; creates access policy and application assignment policy rules; creates the application, and links the application to the desktop group that will host it.

## SEE ALSO

[about\\_Broker\\_Concepts](#)

[about\\_Broker\\_Desktops](#)

[New-BrokerApplication](#)

[Add-BrokerApplication](#)

[Remove-BrokerApplication](#)

[Rename-BrokerApplication](#)

[Set-BrokerApplication](#)

[New-BrokerAppAssignmentPolicyRule](#)

[Remove-BrokerAppAssignmentPolicyRule](#)

[Set-BrokerAppAssignmentPolicyRule](#)

[New-BrokerAppEntitlementPolicyRule](#)

[Remove-BrokerAppEntitlementPolicyRule](#)

[Set-BrokerAppEntitlementPolicyRule](#)



# about\_Broker\_AssignmentPolicy

May 10, 2016

## TOPIC

Citrix Broker SDK - Assignment Policy

## SHORT DESCRIPTION

Controls the automatic, permanent assignment of machines to users.

## LONG DESCRIPTION

The site's assignment policy defines rules controlling automatic assignment of machines to users in a process referred to as Assign On First Use (AOFU).

In this assignment model, a desktop group is initially populated with machines that have no assignments, and users are granted entitlements to obtain a machine selected at random from the pool by self-service assignment. Once made, the assignment is permanent. The resulting assigned machines can be used to deliver either desktop or application sessions depending on the delivery type of the desktop group.

The assignment policy comprises a set of rules. The policy can be applied only to desktop groups of desktop kind Private.

For assignment of machines to deliver desktop sessions:

- o Multiple rules can apply to the same desktop group.
- o Each rule grants an entitlement to one or more machines.

For assignment of machines to deliver application sessions:

- o Only a single rule can apply to a given desktop group.
- o Each rule grants an entitlement to a single machine.
- o Although only a single application assignment rule can be defined for a group, a user can still launch multiple applications from that group because the applications all run within the same session on the assigned machine.

Once a machine is assigned to a user by an assignment policy rule, the rule plays no further part in controlling access to that machine. The rule can be changed, or even removed, without impacting the user's access to the machine.

Rules for desktop and application machine assignments are distinct. Desktop assignments are managed through the BrokerAssignmentPolicyRule SDK object, and application rules through the BrokerAppAssignmentPolicyRule object.

Desktop machine assignment rules can be created only for desktop groups with delivery type DesktopsOnly, whereas an application machine assignment rule can be created only for delivery type AppsOnly.

Because desktop groups of assigned machines do not allow delivery type DesktopsAndApps, desktop machine assignment

and application machine assignment rules cannot exist for the same desktop group.

Assignment policy rules are also used to configure the RemotePC feature where their detailed usage differs. For more information on the RemotePC feature see "help about\_Broker\_RemotePC".

For an entitlement granted by the assignment policy to be available to a user, the site's access policy must also grant them access to the desktop group.

## ASSIGNMENT POLICY RULES

Each assignment policy rule has the following key properties:

- o The desktop group to which it applies (one rule only ever applies to one group).
- o The users to which machines can be assigned by the rule.

Additionally for desktop assignment rules, the following properties exist:

- o The published name of the entitlement (visible to the user).
- o The number of machines (entitlements) to which the rule grants access.
- o The properties that a newly assigned desktop acquires.

If multiple desktop assignment rules entitle a user to machines from the same desktop group, their total machine entitlement is the sum of those granted by all those rules. The properties of the desktop sessions obtained may differ depending on which of the entitlements the user selects to start a session.

Each rule can be individually enabled or disabled. A disabled rule is ignored when the assignment policy is evaluated.

## USER FILTERS (FULL)

Each rule has two user filters, an include filter and an exclude filter:

- o The include filter contains users and user groups that are granted entitlements to machines.
- o The exclude filter contains users and user groups that are denied entitlements to machines.

If the include filter of a rule contains multiple instances of a user (either explicit or implicit), this does not alter the number of machine entitlements granted to them by the rule.

Entries in the exclude filter take priority, so if a user appears explicitly or implicitly in both filters, access is denied. Typically, you use this filter to exclude a user or group of users who would otherwise gain access because they are members of a user group specified in the include filter.

Because all rules are independently evaluated, the exclude filter can exclude only users who would otherwise gain an entitlement through the same rule's include filter. That is, if a user is in a rule's include filter but not its exclude filter, the rule

is guaranteed to grant that user access to a machine irrespective of whether the user appears in the exclude filter of other rules.

If a filter contains a user group that contains other users and groups, the filter implicitly includes all of those users and groups.

By default the exclude filter is disabled.

To maintain assignment policy rules that can be fully displayed and edited with Citrix Studio, use the simplified user filter model below and do not use the exclude filter.

## USER FILTERS (SIMPLIFIED)

The included user filter described above also supports a simplified usage model where the filter itself is disabled. When this is done, any user who has access to the desktop group through the access policy is implicitly granted an entitlement to a machine through the entitlement policy rule without the need to list the user in the rule's include filter.

This is useful in cases where the access policy for the desktop group already explicitly specifies the users who should have access.

Even if the include filter is disabled, the exclude filter can still be used to deny the entitlement to users who would otherwise gain access through the access policy alone.

This simplified usage cannot be used for RemotePC desktop groups.

## REMOTE PC USAGE

When a desktop group is configured as a RemotePC group, the usage of the assignment policy differs in the following ways:

- o Only a single rule can apply to a given desktop group.
- o The delivery type of the desktop group must be DesktopsOnly.
- o The simplified user filter model cannot be used. Users or user groups must appear explicitly in the included user filter of the assignment policy rule.
- o If the included user filter is disabled then RemotePC assignment for the group is also effectively disabled.

For more information on the RemotePC feature see "help about\_Broker\_RemotePC".

## ADDITIONAL DESKTOP ASSIGNMENT RULE PROPERTIES

Desktop assignment rules specify the following additional properties:

- o PublishedName
- o Description
- o IconUid
- o ColorDepth
- o SecureIcaRequired

The published name, description, and icon UID properties apply to the desktop entitlement itself and determine the way in which the entitlement is presented to the user in, for example, StoreFront.

The color depth and secure ICA properties apply to the desktop session that is obtained when the entitlement is used.

In all cases, these properties can be explicitly specified. However, a null value (the default) means that the corresponding property is taken from the desktop group to which the rule applies. This inheritance from groups is dynamic; if the property of the group changes, the property of the entitlement changes too.

For assignment rules these properties are copied to the newly assigned desktop when the granted entitlement is first used. If the properties on the rule are subsequently changed the properties on the user's assigned desktop do not change. The dynamic inheritance of desktop properties from the desktop group continues after the assignment has occurred if the original rule did not provide explicit property values.

## CALCULATING OVERALL MACHINE ENTITLEMENTS

Assignment rules are modified only by the SDK; they are not modified when used by the system to make automatic assignments. The number of machine entitlements offered to the user is determined by the number of machines (within the desktop group) already assigned to them and by which rules those assignments were made.

For each desktop group, the number of desktop entitlements available to the user is determined as follows:

1. The total number of entitlements for the user to machines in the group,

from all rules, is calculated.

2. The total number of assigned machines (from any source) that the user

already has in the group is determined.

3. The outstanding entitlement value is derived as the difference of the

above two numbers.

4. If the outstanding entitlement value is zero or fewer, no further

entitlements are allowed.

Otherwise, for each applicable rule, the number of entitlements is that defined by the rule itself, minus the number of desktops already assigned to the user by that rule, and capped by the outstanding entitlement value.

Desktop and application machine entitlements both follow the above rules. However, because only a single application assignment rule granting a single entitlement per group can exist, these rules are seldom a consideration for applications.



## EXAMPLES

### Simple case:

A user is entitled to a single machine in a group by a single assignment rule:

1. On first use, the user sees a single desktop entitlement.  
If the user selects this, a new desktop is assigned.
2. On subsequent uses, the user sees the assigned desktop only.  
No other entitlements are displayed.

### Complex case:

A user is entitled to two machines, one from each of two different rules, A and B, both applying to the same desktop group. In addition, the user has a machine explicitly assigned to them by the administrator within that group:

1. On first use, the user sees the administrator-assigned desktop, a single entitlement to a desktop of type A, and a single entitlement to a desktop of type B. If either of the entitlements is selected, a desktop of the appropriate type, A or B, is assigned.
2. On subsequent uses, the user sees the administrator-assigned desktop and the desktop assigned by the selected entitlement.  
No further entitlements are displayed.

## MACHINE ENTITLEMENT PRESENTATION

For desktop machine assignment rules, the user interface determines whether users can visually distinguish between an entitlement to a machine and an actual assigned desktop. This cannot be controlled by Broker SDK cmdlets.

Although the number of entitlements seen by users takes account of all of their currently assigned desktops, the dynamic state of the system can affect the user interaction. This means that entitlements are displayed even where the pool of machines is empty, or the remaining desktops are in maintenance mode or otherwise unavailable. If the user attempts to use an entitlement in these cases, they receive a no-available-desktop error.

For application machine entitlement rules, the entitlements themselves are not presented to the end user; only the applications available to the user are presented. The use of available assignments, or of already assigned machines, is managed automatically.

## NOTES

If an assignment rule grants entitlements to a user group:

- o The machine is assigned to the user who selects the entitlement, thus an assignment policy rule cannot be used to assign a machine to a user group. However, an administrator can assign a machine to a user group using the Add-BrokerUser cmdlet.
- o The number of machine entitlements specified in a desktop assignment rule applies to each member of the user group. For example, if a rule grants a user group access to two machines, every user in the group is entitled to two desktops.

The total number of machine entitlements defined by the assignment policy for a given desktop group may exceed the number of machines present in the group. A user attempting to use an entitlement when the pool of machines is empty receives a no-desktop-available error. (See CALCULATING OVERALL MACHINE ENTITLEMENTS above).

A machine assigned to a user as a result of the assignment policy remains permanently assigned unless the machine assignment itself is removed by the administrator. Such assignments are permanent even if the assignment rule is deleted, when the machine is treated as administrator-assigned.

The assignment policy cannot be used to assign a machine to more than one user. This is only possible using the Add-BrokerUser cmdlet.

#### SEE ALSO

[about\\_Broker\\_Policies](#)

[about\\_Broker\\_AccessPolicy](#)

[about\\_Broker\\_EntitlementPolicy](#)

[about\\_Broker\\_Applications](#)

[New-BrokerAssignmentPolicyRule](#)

[Get-BrokerAssignmentPolicyRule](#)

[Set-BrokerAssignmentPolicyRule](#)

[Rename-BrokerAssignmentPolicyRule](#)

[Remove-BrokerAssignmentPolicyRule](#)

[New-BrokerAppAssignmentPolicyRule](#)

[Get-BrokerAppAssignmentPolicyRule](#)

[Set-BrokerAppAssignmentPolicyRule](#)

[Rename-BrokerAppAssignmentPolicyRule](#)

[Remove-BrokerAppAssignmentPolicyRule](#)

[Add-BrokerUser](#)

# about\_Broker\_Concepts

May 10, 2016

## TOPIC

Citrix Broker - Concepts

## SHORT DESCRIPTION

Overview of the Citrix Broker.

## LONG DESCRIPTION

The Citrix Broker is a Microsoft Windows service running on a delivery controller that responds to desktop/application launch requests from users through StoreFront by selecting a suitable machine, powering it up if necessary, and then returning the address of the selected machine to the user's endpoint system so that a session connection can be made. If required for resilience or scale, additional instances of the Broker may be installed to run on additional delivery controllers in the same delivery site.

In addition to handling launch requests, the Broker also has background responsibilities in the delivery site; these include: maintaining appropriate numbers of unused, powered-up machines to avoid unnecessary delays to users launching desktops/applications; maintaining periodic contact with powered-up machines; and monitoring the state of machines and user sessions.

The Citrix.Broker.Admin PowerShell snap-in provides the cmdlets needed to administer and monitor the behaviour of the Broker service, either on the local system (by default) or on another system (by use of the -AdminAddress command-line parameter).

The cmdlets in the broker SDK manage objects in the following broad functional areas:

## PROVISIONING CONFIGURATION

The Broker must be informed of the machines which are at its disposal. In order to do this, machines are organized in catalogs, created with the New-BrokerCatalog cmdlet; machines are introduced into the system through the use of the New-BrokerMachine cmdlet.

Catalogs define the nature of the machines contained within them, such as the allocation type (that is, static or random), how the machines are actually provisioned (that is, by MCS, PVS or manually), whether they are physical or virtual machines, whether they are single-session or multi-session, etc.

Typically, machines configured from a provisioning standpoint are not associated with specific users (though they may need to be, for example if the machine's disk image was captured from a specific user's physical desktop using a P2V tool); this is normally done when configuring how resources are brokered to users, below.

It is also possible for a catalog to be configured to be populated automatically with end users' existing physical machines using the RemotePC feature. The about\_Broker\_RemotePC topic gives more detail.

Provisioning configuration involves the following SDK objects:

BrokerHypervisorConnection

BrokerCatalog  
BrokerMachine  
BrokerRemotePCAccount  
BrokerUser

For more information, see:

Get-Help about\_Broker\_Machines  
Get-Help about\_Broker\_RemotePC  
Get-Help Get-BrokerHypervisorConnection  
Get-Help Get-BrokerCatalog  
Get-Help Get-BrokerMachine  
Get-Help Get-BrokerUser

## BROKERING CONFIGURATION

In order that resources (that is, desktops and applications) can be used in user sessions, the Broker must be configured to connect incoming user launch requests through StoreFront with the correct machine. This is achieved by adding machines to desktop groups. The grouping of machines in desktop groups need not necessarily match the grouping of the machines within the catalogs that were used for the configuration of provisioning. It is through the desktop group that the configuration of which users can use which machine resources is achieved.

Configuration of the mapping between resources and end users is achieved through a combination of machine assignment and entitlement rules. In addition, access to those resources must also be configured (for example, some resources could be configured only to be accessible to users when they are not connecting remotely through Access Gateway.) The about\_Broker\_Policies topic gives more detail of the rich configuration options available.

It is also possible for a desktop group to be configured to be populated automatically with end users existing physical machines using the RemotePC feature. The about\_Broker\_RemotePC topic gives more detail.

When machines are virtual, the broker can be configured to minimize power usage by switching them off when they are not expected to be in use while still maintaining good response times for end-user launch requests. This is achieved through power policy for desktop groups. This allows separate configuration for peak compared to off-peak times of the week of the number of machines nominally to be powered up, the number of machines to be powered up and idling, in addition to those in use to be used as a "buffer" to ensure prompt servicing of user launch requests, and the behavior required for virtual machines when users disconnect from their sessions for extended periods of time.

It is also possible to issue explicit power commands to machines. The about\_Broker\_PowerManagement topic gives more detail.

Configuration of Brokering involves the following SDK objects:

BrokerDesktopGroup  
BrokerPrivateDesktop  
BrokerSharedDesktop  
BrokerRemotePCAccount

BrokerPowerTimeScheme  
BrokerUser  
BrokerTag  
BrokerAccessPolicyRule  
BrokerAssignmentPolicyRule  
BrokerEntitlementPolicyRule  
BrokerApplication  
BrokerApplicationInstance  
BrokerAppAssignmentPolicyRule  
BrokerAppEntitlementPolicyRule  
BrokerConfiguredFTA  
BrokerImportedFTA

For more information, see:

Get-Help about\_Broker\_Desktops  
Get-Help about\_Broker\_Policies  
Get-Help about\_Broker\_Applications  
Get-Help about\_Broker\_RemotePC  
Get-Help Get-BrokerPrivateDesktop  
Get-Help Get-BrokerSharedDesktop  
Get-Help Get-BrokerPowerTimeScheme  
Get-Help Get-BrokerUser  
Get-Help Get-BrokerTag  
Get-Help Get-BrokerAccessPolicyRule  
Get-Help Get-BrokerAssignmentPolicyRule  
Get-Help Get-BrokerEntitlementPolicyRule

## MONITORING AND ADMINISTRATION

After you have provisioned and configured machines for brokering, use the broker SDK to monitor and administer user sessions and other aspects of the delivery site.

Monitoring and administration involve the following SDK objects:

BrokerServiceStatus  
BrokerHypervisorAlert  
BrokerDesktop  
BrokerDesktopUsage  
BrokerHostingPowerAction  
BrokerSession  
BrokerSessionMessage

For more information, see:

Get-Help about\_Broker\_Desktops  
Get-Help Get-BrokerServiceStatus  
Get-Help Get-BrokerHypervisorAlert  
Get-Help Get-BrokerDesktop  
Get-Help Get-BrokerDesktopUsage  
Get-Help Get-BrokerHostingPowerAction  
Get-Help Get-BrokerSession  
Get-Help Send-BrokerSessionMessage

## SITE MANAGEMENT

The broker must be configured after installation; this is normally performed automatically by the Citrix Studio console. Configuration tasks include selecting the database (and obtaining the SQL scripting to initialize it), selecting the Citrix Configuration Service that holds the site configuration.

Note that some aspects of broker configuration (such as the port number on which the broker listens for SDK connections) cannot be configured with PowerShell cmdlets. These are configured through the use of the Broker Service executable. For more information, see `about_Broker_PostInstallPreConfiguration`.

A further important aspect of site management concerns the way in which machines providing resources identify the delivery controllers to which they belong. For more information, see `about_Broker_ControllerDiscovery`.

Managing XenDesktop sites involves the following SDK objects:

BrokerSite  
BrokerController  
BrokerDBConnection  
BrokerDBSchema  
BrokerDBVersionChangeScript  
BrokerInstalledDbVersion  
BrokerServiceInstance  
BrokerServiceGroupMembership  
BrokerNameCache

For more information, see:

Get-Help about\_Broker\_PostInstallPreConfiguration  
Get-Help about\_Broker\_ControllerDiscovery  
Get-Help Get-BrokerSite  
Get-Help Get-BrokerController  
Get-Help Get-BrokerDBConnection  
Get-Help Get-BrokerDBSchema  
Get-Help Get-BrokerDBVersionChangeScript  
Get-Help Get-BrokerInstalledDbVersion  
Get-Help Get-BrokerServiceInstance

Get-Help Reset-BrokerServiceGroupMembership  
Get-Help Update-BrokerNameCache

# about\_Broker\_ConfigurationSlots

May 10, 2016

## TOPIC

Citrix Broker SDK - Configuration Slots and Machine Configurations

## SHORT DESCRIPTION

Overview of assigning a collection of related settings to a desktop group.

## LONG DESCRIPTION

Collections of related settings may be applied to individual desktop groups through the creation of configuration slots and machine configurations.

A configuration slot defines a collection of related settings that are to be associated with that slot. Each machine configuration is associated with a single configuration slot and provides specific values for settings of that slot.

The SettingsGroup property of the configuration slot determines the particular collection of related settings that are associated with that slot. These groups are defined by Citrix and are not modifiable by administrators. For example, there is a particular group of Profile management specific settings that may be associated with a configuration slot. Because of the close association between a configuration slot and its collection of related settings, the full set of configuration slots is created during the site creation.

Each machine configuration is associated with a single configuration slot. The machine configuration contains policy data that provides specific values for the settings associated with that configuration slot.

Every value set in a machine configuration's policy must belong to the configuration slot's settings group. Therefore the appropriate SDK snap-in must be used to create the policy data. For example, the Profile management snap-in must be used to create the policy data for a machine configuration associated with the Profile management configuration slot.

To have particular policy settings applied to the machines in a desktop group, a machine configuration is associated with that desktop group. A machine configuration may be associated with multiple desktop groups. A desktop group may be associated with multiple machine configurations.

When a machine configuration is associated with a desktop group, the configuration inherits the delegated administration restrictions of the desktop group. Thus, if a machine configuration is associated with multiple desktop groups, an administrator can only modify the policy data of the configuration if the administrator has permission to modify every one of those desktop groups.

For detailed information about defining and assigning machine configurations, see:

[help New-BrokerMachineConfiguration](#)

[help Add-BrokerMachineConfiguration](#)

## SEE ALSO

[New-BrokerConfigurationSlot](#)



New-BrokerMachineConfiguration

Add-BrokerMachineConfiguration

Import-BrokerDesktopPolicy

# about\_Broker\_ControllerDiscovery

May 10, 2016

## TOPIC

Citrix Broker - Configuring Controller Discovery

## SHORT DESCRIPTION

Describes the way that machines providing published resources discover delivery controllers.

## LONG DESCRIPTION

In order for the broker to be able to connect users to desktops and applications, the machines from which they are published must register (that is, establish communication) with the broker on an appropriate delivery controller in the delivery site.

The default operation, whose configuration is described in this topic, is to use information from the registry. This is referred to as registry-based controller discovery. The registry information can be supplied when installing the delivery agent software on each machine or it can be supplied through group-policy.

If machines are provisioned using quick deploy, information about delivery controllers is stored in a special "identity disk" attached to the VM.

Finally, in some deployments, the use of an Organizational Unit (OU) in Active Directory (AD) may be preferred. This is referred to as AD-based controller discovery. In this case, you must configure the GUID of the OU in the machines' registries. Such configuration is not described in this topic.

To perform registry-based controller discovery, run the PowerShell script called Set-ADControllerDiscovery.ps1 that is installed on each controller in the folder:

```
$Env:ProgramFiles\Citrix\Broker\Service\Setup Scripts
```

For more information, run this script with the -help parameter.

# about\_Broker\_Desktops

May 10, 2016

## TOPIC

Citrix Broker SDK - Desktops

## SHORT DESCRIPTION

Describes desktop concepts and usage.

## LONG DESCRIPTION

A desktop is a machine that is able to run a Microsoft Windows desktop environment (with a shell, icons and taskbar) or individual applications (seamlessly integrated with the local desktop). The configuration of the desktop determines whether it can run only desktop environments, only applications, or both desktops and applications. Machines running workstation operating systems are able to run one session at a time (single-session), whereas machines running server operating systems have the ability to run multiple simultaneous sessions (multi-session).

A key aspect of desktops is how they are assigned (or allocated) to users. Two allocation types are supported:

- o Random/Shared - A user is assigned a desktop at random from a pool of shared desktops. Multi-session desktops are able to run sessions to multiple users simultaneously, whereas single-session desktops can only run one session at a time, and are returned to the pool when the user logs off. Single-session shared desktops usually discard user data stored on them after the user logs off. Multi-session shared desktops, however, do not tend to discard user data after a log-off, as this is only possible when the desktop is rebooted by a reboot schedule.
- o Permanent/Private - A private desktop is permanently assigned to a specific user and data stored on it is retained across logons and restarts. A private desktop can have users assigned explicitly or on first use.

## DESKTOP GROUPS

Desktops are collected together in desktop groups, and these provide a flexible grouping mechanism that can be used to associate:

- o Desktops running on a particular type of machine
- o Desktops with particular software installed
- o Desktops for a set of users
- o Desktops accessed in a similar way
- o Desktops configured in a particular way
- o Any combination of the above

Each desktop group can only contain one type of desktop, determined by its `AllocationType` and `SessionSupport` properties.

When assigning shared desktops or assign-on-first-use (AOFU) desktops to users, the set of candidates comes from available desktops in a particular desktop group.

You configure power management policy for single-session desktop groups, including peak and off-peak settings, for each desktop group. See `about_Broker_PowerManagement` for details.

## CREATION OF DESKTOPS

Desktop objects are created automatically when a machine is added to a desktop group. The type of desktop is determined by the `AllocationType` property of the desktop group.

In order for a machine to be added from a catalog, the machine must be compatible with the desktop group. For this to be true, the catalog's `AllocationType` must be compatible, and the `SessionSupport` property must match.

Note: Because the session support and functional level of the machine are determined by the software on the machine (operating system and Citrix VDA, respectively), the `SessionSupport` and `MinimalFunctionalLevel` of the catalog and desktop group may match, but not be compatible with the machine. In this case any attempt of registration by the machine will fail.

You can add machines explicitly using the `Add-BrokerMachine` cmdlet, or a number of free machines can be acquired from a catalog using the `Add-BrokerMachinesToDesktopGroup` cmdlet. A machine can only be associated with one desktop group at a time, and has a `DesktopUid` property that references the corresponding desktop object. You can also associate desktops to machines with their SID properties.

Desktop objects are deleted when the machine is removed from the desktop group (using the `Remove-BrokerMachine` cmdlet). Desktops are also deleted when the desktop group containing them is deleted (using the `Remove-BrokerDesktopGroup` cmdlet). A desktop cannot be removed from a catalog while it is in a desktop group.

## SHARED (RANDOM) DESKTOPS

Shared desktops are published to users using entitlement policy rules. Each entitlement rule allows access to a single session on a desktop machine, selected at random from the available desktops in a desktop group (with a preference for desktops that are powered on). If there are no available desktops, launching the session fails. See `about_Broker_EntitlementPolicy` and `about_Broker_PowerManagement` for details.

## PRIVATE (STATIC) DESKTOPS

You can assign private desktops to users explicitly or automatically, with the AOFU feature. It is also possible to assign private desktops to particular clients (through IP or client name).

You explicitly assign machines or desktops to users with the `Add-BrokerUser` cmdlet. Machines can be assigned to users before the machine has been added to the desktop group (desktop created), but otherwise the effect is the same.

With `Add-BrokerUser` you can assign a desktop to multiple users or user groups. If the desktop has single-session support then the desktop will be visible to multiple users, but only one user can log on to the desktop at any time.

Assignment policy rules allow you to use AOFU to assign desktops to users in a desktop group. When a user specified in an assignment policy rule launches a session, and if the user does not already have an assigned desktop, the broker selects an available desktop at random from the desktop group and permanently assigns it to that user. Once assigned in this way, the

desktop behaves as though the assignment was made explicitly by an administrator. See `about_Broker_AssignmentPolicy` for details.

It is possible to assign more than one desktop to a user. You can achieve this by explicit assignment, multiple assignment policy rules, or the `MaxDesktops` property of an assignment policy rule.

## DESKTOP CONFIGURATION

When presented to the user, desktops are identified by:

- o An icon (`IconUid` property)
- o A name (`PublishedName` property)
- o A description (`Description` property)

When starting the session, you can configure two connection settings:

- o The color depth used at the start of the session (`ColorDepth` property)
- o Whether SecureICA encryption is required (`SecureIcaRequired` property)

Each desktop group provides default values for these settings, but you can override them if the desktop has more specific settings (`PrivateDesktop`), or use an entitlement policy rule with more specific settings (`SharedDesktop`).

AOFU desktops can inherit these settings from the assignment policy rule when the assignment to the user takes place.

## MAINTENANCE MODE

There are times when it is necessary to disable desktops. You can do this by setting the `InMaintenanceMode` property of a desktop to `$true`. This puts it into maintenance mode. The broker excludes single-session desktops in maintenance mode from brokering decisions and does not start new sessions on them. Existing sessions are unaffected. For multi-session desktops in maintenance mode, reconnections to existing sessions are allowed, but no new sessions are created on the machine.

Desktops in maintenance mode are also excluded from automatic power management, although explicit power actions are still performed.

Note that disabling desktop groups, entitlement policy rules, assignment policy rules, or applications are other ways of disabling aspects of brokering.

## DESKTOP STATUS

Once desktops are created, you can query the configuration and state information for different kinds of desktop, or retrieve more information about desktops using the `Get-BrokerMachine` cmdlet. To get details of any sessions running on the desktops, use the `Get-BrokerSession` cmdlet.

You can also group desktops by a specific property, counting the number of desktops with each value using the `Group-BrokerMachine` cmdlet. This can provide useful summary statistics.

## DESKTOP USAGE

Every hour the broker records how many desktops from each desktop group are in use, and the `Get-BrokerDesktopUsage` cmdlet returns this information. Analyze historical usage records to understand desktop usage patterns and help with the choice of idle pool and buffer settings.

## DESKTOP CONDITIONS

CPU usage, ICA latency, and profile logon times of desktops are monitored. When one of these values exceeds a threshold (configured by policy), the condition is flagged in the `DesktopConditions` property of the desktop. When the value drops below the threshold again, the condition is cleared. Use `Get-BrokerMachine` or `Group-BrokerMachines` cmdlets to query this information.

## SEE ALSO

[about\\_Broker\\_Concepts](#)

[about\\_Broker\\_Applications](#)

[about\\_Broker\\_EntitlementPolicy](#)

[about\\_Broker\\_AssignmentPolicy](#)

[Add-BrokerMachine](#)

[Add-BrokerMachinesToDesktopGroup](#)

[Remove-BrokerMachine](#)

[Add-BrokerUser](#)

[Set-BrokerMachine](#)

[Set-BrokerPrivateDesktop](#)

[Get-BrokerMachine](#)

[Group-BrokerMachine](#)

[Get-BrokerDesktopUsage](#)

# about\_Broker\_EntitlementPolicy

May 10, 2016

## TOPIC

Citrix Broker SDK - Desktop and Application Entitlement Policy

## SHORT DESCRIPTION

Controls end-user entitlement to desktop and application sessions provided from a pool of shared machines.

## LONG DESCRIPTION

The site's entitlement policy defines rules controlling users' entitlements to desktop and application sessions from pools of shared machines. Each pool is defined by a desktop group.

The entitlement policy comprises a set of rules. Each rule grants users a single entitlement to a desktop or application session in a specified desktop group. The policy can be applied only to groups of desktop kind Random. For desktop entitlements multiple rules can apply to the same group, however for application entitlements only a single rule can apply to a given group.

When the user starts a session by selecting an entitlement the behavior depends on the session-support property of the desktop group:

- o For single-session groups the user is temporarily assigned a machine selected at random from the group to provide their session. When the session ends, the machine is returned to the pool of available machines.
- o For multi-session groups the user session is provided by the machine that is least loaded within the group when the session is launched.

If multiple desktop entitlement rules for the same group contain the same user, the user can have as many desktop sessions from the group concurrently as they have entitlements.

Although only a single application entitlement rule can be defined for a group, a user can still launch multiple applications from that group because the applications all run within that entitlement's single session.

Rules for desktop and application session entitlements are distinct. Desktop entitlements are managed through the BrokerEntitlementPolicyRule SDK object, and application rules through the BrokerAppEntitlementPolicyRule object.

Desktop entitlement rules can be created only for desktop groups with delivery types DesktopsOnly or DesktopsAndApps, whereas an application entitlement rule can be created only for delivery types AppsOnly or DesktopsAndApps.

For desktop groups with delivery type DesktopsAndApps, typically one or more desktop session entitlement rules together with a single application session entitlement rule exist.

For an entitlement granted by the entitlement policy to be available to a user, the site's access policy must also grant them access to the desktop group.

## ENTITLEMENT POLICY RULES

Each entitlement policy rule has the following key properties:

- o The desktop group to which it applies (one rule only ever applies to one group)
- o The users to whom the entitlement is granted

Additionally for desktop entitlement rules, the following properties exist:

- o The published name of the entitlement (visible to the user)
- o Any properties that a desktop session launched using the entitlement should use that differ from the defaults specified on the desktop group

If multiple desktop entitlements are available to a user from the same group the resultant desktop session properties may differ depending on which entitlement the user selects to start the session.

Each rule can be individually enabled or disabled. A disabled rule is ignored when the entitlement policy is evaluated.

#### USER FILTERS (FULL)

Each rule has two user filters, an include filter and an exclude filter:

- o The include filter contains users and user groups that are granted an entitlement to a session
- o The exclude filter contains users and user groups that are denied an entitlement to a session

If the include filter of a rule contains multiple instances of a user (either explicit or implicit), they get only one entitlement by that rule.

Entries in the exclude filter take priority, so if a user appears explicitly or implicitly in both filters, access is denied. Typically, you use this filter to exclude a user or group of users who would otherwise gain access because they are members of a user group specified in the include filter.

Because all rules are independently evaluated, the exclude filter can only exclude users who would otherwise gain an entitlement through the same rule's include filter. That is, if a user is in a rule's include filter but not its exclude filter, the rule is guaranteed to grant that user a session entitlement irrespective of whether the user appears in the exclude filter of other rules.

If a filter contains a user group that contains other users and groups, the filter implicitly includes all of those users and groups.

By default the exclude filter is disabled.

To maintain entitlement policy rules that can be fully displayed and edited with Citrix Studio, use the simplified user filter model below and do not use the exclude filter.



## USER FILTERS (SIMPLIFIED)

The included user filter described above also supports a simplified usage model where the filter itself is disabled. When this is done, any user who has access to the desktop group through the access policy is implicitly granted an entitlement to a session through the entitlement policy rule without the need to list the user in the rule's include filter.

This is useful in cases where the access policy for the desktop group already explicitly specifies the users who should have access.

Even if the include filter is disabled, the exclude filter can still be used to deny the entitlement from users who would otherwise gain access through the access policy alone.

## ADDITIONAL DESKTOP ENTITLEMENT RULE PROPERTIES

Desktop entitlement rules specify the following additional properties:

- o PublishedName
- o Description
- o IconUid
- o ColorDepth
- o SecureIcaRequired

The published name, description, and icon UID properties apply to the desktop entitlement itself and determine the way in which the entitlement is presented to the user in, for example, StoreFront.

The color depth and secure ICA properties apply to the desktop session that is obtained when the entitlement is used.

In all cases, these properties can be explicitly specified. However, a null value (the default) means that the corresponding property is taken from the desktop group to which the rule applies. This inheritance from groups is dynamic; if the property of the group changes, the property of the entitlement changes too.

## NOTES

If a rule grants an entitlement to a user group, the session entitlement applies to the individual user who selects the entitlement. However, this does not prevent a different user in the same user group from using the same entitlement concurrently. So, a rule that grants an entitlement to a user group containing multiple users allows each user concurrent access to a single session from the desktop group.

The total number of entitlements defined by the policy may exceed the number of machines available, or the maximum allowed sessions, from the desktop group. A user attempting to use an entitlement when no further resources are available receives a no-desktop-available error.

If a session launched through an entitlement is active when the entitlement rule is deleted, the session continues unaffected. However:

- o When the user ends the session, they cannot start a new one if the deleted rule was their only entitlement to a session in that group
- o If the user disconnects the session, they cannot reconnect to it

## SEE ALSO

[about\\_Broker\\_Policies](#)

[about\\_Broker\\_AccessPolicy](#)

[about\\_Broker\\_AssignmentPolicy](#)

[about\\_Broker\\_Applications](#)

[New-BrokerEntitlementPolicyRule](#)

[Get-BrokerEntitlementPolicyRule](#)

[Set-BrokerEntitlementPolicyRule](#)

[Rename-BrokerEntitlementPolicyRule](#)

[Remove-BrokerEntitlementPolicyRule](#)

[New-BrokerAppEntitlementPolicyRule](#)

[Get-BrokerAppEntitlementPolicyRule](#)

[Set-BrokerAppEntitlementPolicyRule](#)

[Rename-BrokerAppEntitlementPolicyRule](#)

[Remove-BrokerAppEntitlementPolicyRule](#)

# about\_Broker\_ErrorHandling

May 10, 2016

## TOPIC

Citrix Broker SDK - Error Handling

## SHORT DESCRIPTION

Describes broker errors generated by cmdlets and how to access them.

## LONG DESCRIPTION

The broker SDK cmdlets report errors through the class `SdkErrorRecord`, which is a subclass of the standard Powershell error record class `System.Management.Automation.ErrorRecord`. `SdkErrorRecord` contains:

- o A short string to describe the error status code. This is implemented as a public property named `Status`.
- o A dictionary of key-value pairs containing additional data specific to the cmdlet. This is implemented as a public property named `ErrorData` of type `Dictionary<string, string>`.

The error status property always has a value. Populating the error data dictionary is optional. The number of entries within the dictionary, the content of the entries, and the exact format of the key and value data is specific to each cmdlet.

You can access an `SdkErrorRecord` object using the standard Powershell cmdlet `ErrorVariable` parameter. The type of object returned by `ErrorVariable` depends on whether the error is terminating or non-terminating.

## NON-TERMINATING ERRORS

For non-terminating errors, each object in the returned `ErrorVariable` array is simply an instance of type `SdkErrorRecord`.

## TERMINATING ERRORS

For terminating errors, the object returned by `ErrorVariable` is of type `System.Management.Automation.CmdletInvocationException`.

For non-terminating errors that are escalated as terminating errors (through the "ErrorAction stop" argument), the object returned by `ErrorVariable` is of type `System.Management.Automation.ActionPreferenceStopException`.

`CmdletInvocationException` and `ActionPreferenceStopException` are subclasses of the base class `System.Management.Automation.RuntimeException`, which exposes the `SdkErrorRecord` object through its `ErrorData` property.

## Class `SdkOperationException`

`SdkErrorRecord`'s `Exception` property holds an instance of custom exception class `SdkOperationException`, which also contains the error status code and data dictionary from `SdkErrorRecord`.

The SDK cmdlets generate errors in response to exceptions generated by the underlying system or by the cmdlet detecting

errors locally and instantiating appropriate exception types. Such exceptions, which represent the original cause of the terminating error, are specified in `SdkOperationException`'s `InnerException` property.

For terminating errors, use Powershell scripts to trap `SdkOperationException` and access additional error information and the originating exception.

## REVIEW OF POWERSHELL ERROR HANDLING BEHAVIOR

Powershell scripts can access error information using the following methods:

- o Read error records from the `$Error` arraylist.
- o Get the cmdlet to return error records using the standard `ErrorVariable` cmdlet parameter.
- o Use trap blocks to catch exceptions for terminating errors.

The type of the error record that Powershell puts in the `$Error` arraylist and returns through the `ErrorVariable` cmdlet parameter depends on whether the error is terminating or non-terminating, and whether the "ErrorAction continue" or "ErrorAction stop" cmdlet parameters are specified (that turn terminating errors into non-terminating ones, and vice versa).

For the combinations of error types (terminating, non-terminating) and error actions (stop, continue), the following statements describe the relationship between:

- o The type of object contained in the Powershell `$Error` arraylist.
- o The type of object returned by the `ErrorVariable` cmdlet parameter (assuming the script can continue after a terminating error).
- o The type of the exception that is thrown (where applicable).

Non-terminating errors with `ErrorAction=Continue`, `$Error` type=`SdkErrorRecord`, and `ErrorVariable` type=`SdkErrorRecord` do not throw an exception.

Terminating errors with `ErrorAction=Stop`, `$Error` type=`ErrorRecord`, and `ErrorVariable` type=`CmdletInvocationException` throw an `SdkOperationException` exception.

Non-terminating errors with `ErrorAction=Stop`, `$Error` type=`ErrorRecord`, and `ErrorVariable` type=`ActionPreferenceOperationException` do not throw an exception.

Terminating errors with `ErrorAction=Continue`, `$Error` type=`ErrorRecord`, and `ErrorVariable` type=`CmdletInvocationException` throw an `SdkOperationException` exception.

`ActionPreferenceOperationException` and `CmdletInvocationException` are subclasses of `System.Management.Automation.RuntimeException`.

## ACCESSING ERROR RECORD DATA

The Powershell code sample below demonstrates how to access error information programmatically with the `ErrorVariable` cmdlet parameter and a trap block.

```
Trap exceptions generated from terminating errors trap [Exception] {
```

```

write ""
write "TRAP BLOCK : BEGIN❖"

if($_.Exception.GetType().Name -eq "SdkOperationException")
{
 $sdkOpEx = $_.Exception

 # show error status
 write $("SdkOperationException.Status = " + $sdkOpEx.Status)

 # show error data dictionary
 write $("SdkOperationException.ErrorData=")

 write $("SdkOperationException.InnerException = " + $sdkOpEx.InnerException)
 $_.Exception.ErrorData
}

continue #could also call break here to halt script execution
write "TRAP BLOCK : END"

}

Run tests 1 to
4, below, in turn to examine terminating and ## non-terminating error behavior.
#####

Test 1: Invoke cmdlet that generates a terminating error: # New-BrokerCatalog throws terminating error if a # catalog
with the supplied name already exists. # #New-BrokerCatalog -Name "AlreadyExists" -AllocationType Random -
ProvisioningType Manual -SessionSupport SingleSession -PersistUserChanges OnLocal -MachinesArePhysical $true -
ErrorVariable ev

Test 2: Force script execution to continue after a terminating error. # #New-BrokerCatalog -Name "AlreadyExists" -
AllocationType Random -ProvisioningType Manual -SessionSupport SingleSession -PersistUserChanges OnLocal -
MachinesArePhysical $true -ErrorVariable ev -ErrorAction continue

Test 3: Invoke cmdlet that generates a non-terminating error: # Get-BrokerCatalog generates a non-terminating error if
a catalog # with the specified name doesn't exist. # #Get-BrokerCatalog -Name "IDontExist" -ErrorVariable ev

Test 4 ❖ Force script execution to halt after a non-terminating error. # #Get-BrokerCatalog -Name "IDontExist" -
ErrorVariable ev -ErrorAction "stop"

write "" write "GET ERROR INFORAMTION: BEGIN"

$SdkErrRecord = $null

if($ev[0].GetType().Name -eq "SdkErrorRecord"){

```

```

 $sdkErrRecord = $ev[0]

}

elseif($ev[0].GetType().BaseType.FullName -eq "System.Management.Automation.RuntimeException") {

 $sdkErrRecord = $ev[0].ErrorRecord

} else {

 write ("UNKNOWN ERROR VARIABLE TYPE:")
 $ev[0].GetType().Name

}

if($sdkErrRecord -ne $null) {

 write ("Have sdkErrRecord:")
 write (" Type Name = " + $sdkErrRecord.GetType().FullName)
 write (" Status = " + $sdkErrRecord.Status)
 write (" Exception type = " + $sdkErrRecord.Exception.GetType().FullName)
 write (" ErrorData = ")
 $sdkErrRecord.ErrorData
 write (" FullyQualifiedErrorId = " + $sdkErrRecord.FullyQualifiedErrorId)

} write "GET ERROR INFORMATION: END"

```

# about\_Broker\_Filtering

May 10, 2016

## TOPIC

XenDesktop - Advanced Dataset Filtering

## SHORT DESCRIPTION

Describes the common filtering options for XenDesktop cmdlets.

## LONG DESCRIPTION

Some cmdlets operate on large quantities of data and, to reduce the overhead of sending all of that data over the network, many of the Get- cmdlets support server-side filtering of the results.

The conventional way of filtering results in PowerShell is to pipeline them into Where-Object, Select-Object, and Sort-Object, for example:

```
Get-<Noun> | Where { $_.Size = 'Small' } | Sort 'Date' | Select -First 10
```

However, for most XenDesktop cmdlets the data is stored remotely and it would be slow and inefficient to retrieve large amounts of data over the network and then discard most of it. Instead, many of the Get- cmdlets provide filtering parameters that allow results to be processed on the server, returning only the required results.

You can filter results by most object properties using parameters derived from the property name. You can also sort results or limit them to a specified number of records:

```
Get-<Noun> -Size 'Small' -SortBy 'Date' -MaxRecordCount 10
```

You can express more complex filter conditions using a syntax and set of operators very similar to those used by PowerShell expressions.

Those cmdlets that support filtering have the following common parameters:

-MaxRecordCount <int>

Specifies the maximum number of results to return.  
For example, to return only the first nine results use:

```
Get-<Noun> -MaxRecordCount 9
```

If not specified, only the first 250 records are returned, and if more are available, a warning is produced:

WARNING: Only first 250 records returned. Use -MaxRecordCount to retrieve more.

You can suppress this warning by using -WarningAction or by specifying a value for -MaxRecordCount.

To retrieve all records, specify a large number for -MaxRecordCount. As the value is an integer, you can use the following:

```
Get-<Noun> -MaxRecordCount [int]::MaxValue
```

-ReturnTotalRecordCount [<SwitchParameter>]

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. For example:

```
Get-<Noun> -MaxRecordCount 9 -ReturnTotalRecordCount
```

....

```
Get-<Noun> : Returned 9 of 10 items
```

```
At line:1 char:18
```

```
+ Get-<Noun> <<<< -MaxRecordCount 9 -ReturnTotalRecordCount
```

```
+ CategoryInfo : OperationStopped: (:) [Get-<Noun>], PartialDataException
```

```
+ FullyQualifiedErrorId : PartialData,Citrix.<SDKName>.SDK.Get<Noun>
```

The count can be accessed using the TotalAvailableResultCount property:

```
$count = $error[0].TotalAvailableResultCount
```

-Skip <int>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

-SortBy <string>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order, respectively. Ascending order is assumed if no prefix is present.



Sorting occurs before -MaxRecordCount and -Skip parameters are applied. For example, to sort by Name and then by Count (largest first) use:

```
-SortBy 'Name,-Count'
```

By default, sorting by an enumeration property uses the numeric value of the elements. You can specify a different sort order by qualifying the name with an ordered list of elements or their numeric values, or <null> to indicate the placement of null values.

Elements not mentioned are placed at the end in their numeric order. For example, to sort by two different enums and then by the object id:

```
-SortBy 'MyState(StateC,<null>,StateA,StateB),Another(0,3,2,1),Id'
```

-Filter <String>

This parameter lets you specify advanced filter expressions, and supports combination of conditions with -and and -or, and grouping with braces. For example:

```
Get-<Noun> -Filter 'Name -like "High*" -or (Priority -eq 1 -and Severity -ge 2)'
```

The syntax is close enough to PowerShell syntax that you can use script blocks in most cases. This can be easier to read as it reduces quoting:

```
Get-<Noun> -Filter { Count -ne $null }
```

The full -Filter syntax is provided below.

## EXAMPLES

Filtering by strings performs a case-insensitive wildcard match.

Separate parameters are combined with an implicit -and operator.

Normal PowerShell quoting rules apply, so you can use single or double quotes, and omit the quotes altogether for many strings. The order of parameters does not make any difference. The following are equivalent:

```
Get-<Noun> -Company Citrix -Product Xen*
Get-<Noun> -Company "citrix" -Product '[X]EN*'
Get-<Noun> -Product "Xen*" -Company "CITRIX"
Get-<Noun> -Filter { Company -eq 'Citrix' -and Product -like 'Xen*' }
```

See [about\\_Quoting\\_Rules](#) and [about\\_Wildcards](#) for details about PowerShell handling of quotes and wildcards.

To avoid wildcard matching or include quote characters, you can escape the wildcards using the normal PowerShell escape mechanisms (see [about\\_Escape\\_Characters](#)), or switch to a filter expression and the `-eq` operator:

```
Get-<Noun> -Company "Abc[*]" # Matches Abc*
Get-<Noun> -Company "Abc`*" # Matches Abc*
Get-<Noun> -Filter { Company -eq "Abc*" } # Matches Abc*
Get-<Noun> -Filter { Company -eq "A`"B`C`" } # Matches A"B'C
```

Simple filtering by numbers, booleans, and TimeSpans perform direct equality comparisons, although if the value is nullable you can also search for null values. Here are some examples:

```
Get-<Noun> -Uid 123
Get-<Noun> -Enabled $true
Get-<Noun> -Duration 1:30:40
Get-<Noun> -NullableProperty $null
```

More comparisons are possible using advanced filtering with `-Filter`:

```
Get-<Noun> -Filter 'Capacity -ge 10gb'
Get-<Noun> -Filter 'Age -ge 20 -and Age -lt 40'
Get-<Noun> -Filter 'VolumeLevel -like "[123]"'
Get-<Noun> -Filter 'Enabled -ne $false'
Get-<Noun> -Filter 'NullableProperty -ne $null'
```

You can check boolean values without an explicit comparison operator, and you can also combine them with `-not`:

```
Get-<Noun> -Filter 'Enabled' # Equivalent to 'Enabled -eq $true'
Get-<Noun> -Filter '-not Enabled' # Equivalent to 'Enabled -eq $false'
```

See [about\\_Comparison\\_Operators](#) for an explanation of the operators, but note that only a subset of PowerShell operators are supported (`-eq`, `-ne`, `-gt`, `-ge`, `-lt`, `-le`, `-like`, `-notlike`, `-in`, `-notin`, `-contains`, `-notcontains`).

Enumeration values can either be specified using typed values or the string name of the enumeration value:

```
Get-<Noun> -Shape [Shapes]::Square
Get-<Noun> -Shape Circle
```

With filter expressions, typed values can be specified with simple variables or quoted strings. They also support enumerations with wildcards:

```
$s = [Shapes]::Square
Get-<Noun> -Filter { Shape -eq $s -or Shape -eq "Circle" }
Get-<Noun> -Filter { Shape -like 'C*' }
```

By their nature, floating point values, DateTime values, and TimeSpan values are best suited to relative comparisons rather than just equality. DateTime strings are converted using the locale and time zone of the user device, but you can use ISO8601 format strings (YYYY-MM-DDThh:mm:ss.sTZD) to avoid ambiguity. You can also use standard PowerShell syntax to create these values:

```
Get-<Noun> -Filter { StartTime -ge "2010-08-23T12:30:00.0Z" }
$d = [DateTime]"2010-08-23T12:30:00.0Z"
Get-<Noun> -Filter { StartTime -ge $d }
$d = (Get-Date).AddDays(-1)
Get-<Noun> -Filter { StartTime -ge $d }
```

Relative times are quite common and, when using filter expressions, you can also specify DateTime values using a relative format:

```
Get-<Noun> -Filter { StartTime -ge '-2' } # Two days ago
Get-<Noun> -Filter { StartTime -ge '-1:30' } # Hour and a half ago
Get-<Noun> -Filter { StartTime -ge '-0:0:30' } # 30 seconds ago
```

## ARRAY PROPERTIES

When filtering against list or array properties, simple parameters perform a case-insensitive wildcard match against each of the members. With filter expressions, you can use the -contains and -notcontains operators. Unlike PowerShell, these perform wildcard matching on strings.

Note that for array properties the naming convention is for the returned property to be plural, but the parameter used to search for any match is singular. The following are equivalent (assuming Users is an array property):

```
Get-<Noun> -User Fred*
Get-<Noun> -Filter { User -like "Fred*" }
Get-<Noun> -Filter { Users -contains "Fred*" }
```

You can also use the singular form with -Filter to search using other operators:

```
Match if any user in the list is called "Frederick"
Get-<Noun> -Filter { User -eq "Frederick" }
Match if any user in the list has a name alphabetically below 'F'
Get-<Noun> -Filter { User -lt 'F' }
```

## COMPLEX EXPRESSIONS

When matching against multiple values, you can use a sequence of comparisons joined with -or operators, or you can use -in and -notin:

```
Get-<Noun> -Filter { Shape -eq 'Circle' -or Shape -eq 'Square' }
$shapes = 'Circle','Square'
Get-<Noun> -Filter { Shape -in $shapes }
$sides = 1..4
Get-<Noun> -Filter { Sides -notin $sides }
```

Braces can be used to group complex expressions, and override the default left-to-right evaluation of -and and -or. You can also use -not to invert the sense of any sub-expression:

```
Get-<Noun> -Filter { Size -gt 4 -or (Color -eq 'Blue' -and Shape -eq 'Circle') }
Get-<Noun> -Filter { Sides -lt 5 -and -not (Color -eq 'Blue' -and Shape -eq 'Circle') }
```

## PAGING

The simplest way to page through data is to use the -Skip and -MaxRecordCount parameters. So, to read the first three pages of data with 10 records per page, use:

```
Get-<Noun> -Skip 0 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 10 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 20 -MaxRecordCount 10 <other filtering criteria>
```

You must include the same filtering criteria on each call, and ensure that the data is sorted consistently.

The above approach is often acceptable, but as each call performs an independent query, data changes can result in records being skipped or appearing twice. One approach to improve this is to sort by a unique id field and then start the search for the next page at the unique id after the last unique id of the previous page. For example:

```
Get the first page
Get-<Noun> -MaxRecordCount 10 -SortBy SerialNumber

SerialNumber ...

A120004
A120007
... 7 other records ...
A120900

Get the next page
Get-<Noun> -MaxRecordCount 10 -Filter { FirstName -gt 'A120900' }

SerialNumber ...
```

----- ---  
A120901  
B220000  
...

## FILTER SYNTAX DEFINITION

<Filter> ::= <ScriptBlock> | <ComponentList>

<ScriptBlock> ::= "{" <ComponentList> "}"

<ComponentList> ::= <Component> <AndOrOperator> <ComponentList> |

<Component>

<Component> ::= <NotOperator> <Factor> |

<Factor>

<Factor> ::= "(" <ComponentList> ")" |

<PropertyName> <ComparisonOperator> <Value> |  
<PropertyName>

<AndOrOperator> ::= "-and" | "-or"

<NotOperator> ::= "-not" | "!"

<ComparisonOperator>

::= "-eq" | "-ne" | "-le" | "-ge" | "-lt" | "-gt" |  
"-like" | "-notlike" | "-contains" | "-notcontains" |  
"-in" | "-notin"

<PropertyName> ::= <simple name of property>

<Value> ::= <string literal> | <numeric literal> |

<scalar variable> | <array variable> |  
"\$null" | "\$true" | "\$false"

Numeric literals support decimal and hexadecimal literals, with optional multiplier suffixes (kb, mb, gb, tb, pb).

Dates and times can be specified as string literals. The current culture determines what formats are accepted. To avoid any ambiguity, use strings formatted to the ISO8601 standard. If not specified, the current time zone is used.

Relative date-time string literals are also supported, using a minus sign followed by a TimeSpan. For example, "-1:30" means 1 hour and 30 minutes ago.

# about\_Broker\_Licensing

May 10, 2016

## TOPIC

Citrix Broker - Licensing

## SHORT DESCRIPTION

Overview of broker licensing configuration.

## LONG DESCRIPTION

As part of the licensing setup for a site, the types of product licenses used by the broker when creating connections to virtual desktops or applications can be configured using the Central Configuration Service SDK.

The following licensing related properties can be specified using the Set-ConfigSite cmdlet:

- o LicensingModel - Sets the licensing model to use. Values can be 'Concurrent' or 'UserDevice'.
- o ProductCode - Specifies which product license is supported by the site. Values can be MPS for XenApp licenses or XDT for XenDesktop licenses.
- o ProductEdition - Sets the licensing edition to use.

A license matching the specified model, product code, and edition must be available within the site's license server in order for the broker to grant licenses.

These properties are part of the site object returned by the Get-ConfigSite cmdlet.

## CONCURRENT LICENSE MODEL

A concurrent license is tied to a XenDesktop session. When a user launches a session, a license is checked out to that session. When a user logs off from a session, the license is checked back in again, making it available for another session.

## USER DEVICE LICENSE MODEL

With user device licensing, the license server automatically assigns licenses to users or devices based on usage:

- o User licensing allows users to access their desktops and applications from multiple devices.
- o Device licensing allows multiple users to access their desktops and applications from a single device.

When users or devices connect to an application or desktop, they consume a license for a 90-day license assignment period. The assignment period begins when a connection is made, is renewed to the full 90 days during the life of the connection, and expires (allowing reassignment) 90 days after the last connection terminates. A license assignment can be manually

ended before the 90-day period elapses using the `udadmin` command line installed on the license server.

## LICENSING STATE

The broker site contains the following properties related to licensing state:

- o `LicensingGracePeriodActive` - Reports if the broker is in licensing grace period.
- o `LicensingOutOfBoxGracePeriodActive` - Reports if the broker is in out-of-box grace period.
- o `LicensingGraceHoursLeft` - The number of grace hours remaining, if the broker is in grace period, else it is null.
- o `LicensedSessionsActive` - The number of active, licensed sessions.
- o `LicenseGraceSessionsRemaining` - The number of grace sessions available, if the broker is in licensing grace period, else it is null.

These properties are part of the site object returned by the `Get-BrokerSite` cmdlet.

## LICENSE SERVER TEST

The broker SDK cmdlet `Test-BrokerLicenseServer` checks whether or not a given server can be used as a license server by the broker.

## RESETTING LICENSE SERVER CONNECTION

The broker SDK cmdlet `Reset-BrokerLicensingConnection` resets the broker's connection to the license server.

## LICENSE BURN-IN DATE

The version of the product that is supported within the site is denoted by a licensing burn-in date. This date can be accessed through the `LicensingBurnInDate` field of the site object returned by the `Get-ConfigSite` cmdlet.

## SEE ALSO

`about_Broker_Site`

[Test-BrokerLicenseServer](#)

[Reset-BrokerLicensingConnection](#)

[Get-BrokerController](#)

`Set-ConfigSite`

`Get-ConfigSite`

[Get-BrokerSite](#)



# about\_Broker\_Machines

May 10, 2016

## TOPIC

Citrix Broker SDK - Machine Object

## SHORT DESCRIPTION

Describes machine concepts and usage.

## LONG DESCRIPTION

A machine represents a physical or virtual machine that can be used to provide a user with one or more desktops, applications or both. When a machine is created, you must assign it to a catalog, which defines how the machine is allocated to a user (static or random), the session support it provides (single-session or multi-session), how the machine's disk images are created and managed (PVS, MCS or manually) and the expected functional level. If the machine is virtual but not provisioned by MCS, you must also assign it to a hypervisor connection, which represents the hypervisor (or pool of linked hypervisors) that runs the virtual machine.

Creating a machine object is the first step in the broker SDK towards configuring a physical or virtual machine to provide desktops and/or applications to users. A machine must be added to a desktop group before it is able to be used (see about\_Broker\_Desktops). To add machines to a desktop group, the Add-BrokerMachine or Add-BrokerMachinesToDesktopGroup cmdlets can be used. This creates a desktop object corresponding to the machine.

## CATALOGS

When a machine is created, you must assign it to a catalog. The catalog defines the behavior of the machine within a site as well as the expected functionality and properties of the machine:

- o Allocation type: The catalog determines how the machines are allocated to the user. Allocation can be static or random. Static allocation is where the machine is permanently assigned to a specific user. Data stored is retained across logons and restarts.

The other type of allocation is random, where a random machine is assigned to a user from a pool when a session is requested. The machine returns to the pool when the user logs off.

- o How the machine is created: The catalog collects together machines that are created in the same way: either with PVS, MCS or manually.
- o Physical or Virtual: A machine that is virtual can have its power state controlled and monitored by the system. Virtual machines must be associated with a hypervisor connection, either directly or, in the case of MCS provisioned machines, indirectly through the provisioning scheme. Single session virtual machines can be managed using power policy to automatically be turned on or off as needed. Machines marked as physical are not monitored or controlled as to their power state.

- o How the users settings are stored: The catalog also determines how the users settings are stored, either on a Citrix Personal vDisk, on the machine's local drive, or if the user settings are discarded.
- o RemotePC: If the catalog is set up as a remote PC catalog, machines are added automatically upon registration based on the site configuration. In order for a catalog to be specified as a RemotePC catalog, the session support must be single session and the catalog must be set up for physical machines.
- o Functional level: The functional level of a machine is determined by the version of the Citrix VDA software it is running. Some features are not supported in machines with lower functional levels. Catalogs can supply a minimal functional level, meaning any machines in the catalog with a lower functional level will be unable to register with the site.
- o Session support: This can either be single-session or multi-session. Single-session machines can have an active session with up to one user at any time, whereas multi-session machines have the capability to have active sessions with multiple users simultaneously. The session support of a machine is determined by the variant of the VDA software component installed on the machine (either with single-session support or multi-session support). The multi-session VDA software may only be installed on server operating systems. The catalog session support must match the session support of the software installed on the machine for the machine to successfully register with the site.

## MACHINE STATUS

After machines are created, you can query the configuration and state information using the Get-BrokerMachine cmdlet. The information the cmdlet can provide includes, but is not limited to, the following:

- o Personal vDisk interactions and lifecycle: The current state of the personal vDisk can be obtained, as well as the configuration options of how the user data is persisted.
- o Session properties: The properties of the current session for single-session machines can be obtained, such as ClientName and ClientAddress. To access session information on multi-session machines, the Get-BrokerSession cmdlet can be used.
- o Application status: If the machine is configured to run applications, information can be found about the published applications running on the machine.
- o Connection information: Information about the time and user of the last

connection to the machine can be found, as well as information about the last deregistration.

For an exhaustive list of the properties of a machine that can be queried, see the `Get-BrokerMachine` cmdlet.

## MACHINE CONFIGURATION

Machine settings can be changed and configured once the machine object has been created, as long as the changes are compatible with the catalog the machine is in. For example, more users can be assigned to the machine than were initially assigned when creating the machine object, this is done with the `Add-BrokerUser` cmdlet.

For a full list of the machine configuration options available, see the `Set-BrokerMachine` command.

## MAINTENANCE MODE

There are times when it is necessary to disable machines. This can be achieved by setting the `InMaintenanceMode` property to `$true`. This puts the machine into maintenance mode. With single-session machines, this means that the broker excludes the machine from brokering decisions and does not start new sessions on them. Existing sessions are unaffected. For multi-session desktops in maintenance mode, reconnections to existing sessions are allowed, but no new sessions are created on the machine.

Machines in maintenance mode are also excluded from automatic power management, although explicit power actions are still performed.

## SEE ALSO

[about\\_Broker\\_PowerManagement](#)

[about\\_Broker\\_Desktops](#)

[New-BrokerMachine](#)

[Add-BrokerMachine](#)

[Add-BrokerMachinesToDesktopGroup](#)

[Remove-BrokerMachine](#)

[Get-BrokerMachine](#)

[Set-BrokerMachine](#)

# about\_Broker\_Policies

May 10, 2016

## TOPIC

Citrix Broker SDK - Access, Entitlement, and Assignment Policies

## SHORT DESCRIPTION

Overview of the site policies that control users' access to desktop and application sessions.

## LONG DESCRIPTION

For an end user to access a desktop or application resource within a site, they must have both an entitlement to use the resource, and have access to the desktop group that contains the resource.

Entitlements to use resources can be granted by one of the following means:

- o The site entitlement policy grants entitlements to launch a shared desktop or application session from a pool of shared machines.
- o The site assignment policy grants entitlements for "self service" permanent assignment of machines to users for running desktop or application sessions, and is referred to as "Assign On First Use" (AOFU)
- o Machines can be permanently assigned ("pre-assigned") to users by the administrator to run either desktop or application sessions.
- o Machines can be configured to allow automatic permanent assignment to their normal user (using the RemotePC feature).

A user must also be granted access to the desktop group that contains the resource. These access rights are controlled by the site's access policy.

The access policy controls access using details of the user's device such as whether it's connected over a local area network (LAN) or connected through Access Gateway, the user device's name, IP address or subnet, and the requested connection protocol. The user's identity can also feed into the access check allowing, for example, certain users access to resources only when locally connected to the site, but others full remote access.

Access and entitlements can be combined to allow rich and fine-grained control over which users have access to site resource from any given user device or location.

Each site has a single access policy, entitlement policy, and assignment policy. Each policy comprises a set of rules. Policies are defined by adding, removing, or changing rules.

Each site policy can also be viewed as a set of distinct policies each relating to a single desktop group. In general a group has one or more policy rules that relate to it, however each rule relates to only a single group. Thus the rules that grant entitlement and access rights to a desktop group define the policy for that group and that group only; changing this policy has no impact on the entitlement and access rights for any other other group in the site.

For detailed information about defining policy rules, see:

help New-BrokerAccessPolicyRule  
help New-BrokerEntitlementPolicyRule  
help New-BrokerAssignmentPolicyRule  
help New-BrokerAppEntitlementPolicyRule  
help New-BrokerAppAssignmentPolicyRule

The mapping of policies to the resources that they make available within a site is described briefly below. For specific information on configuring each category of resource, consult the more detailed help topics listed.

## SHARED DESKTOP AND APPLICATION SESSIONS

To grant access to a group of shared machines, use the access and entitlement policies:

- o The access policy grants access to the desktop group containing the machines to be shared.
- o The entitlement policy grants an entitlement to use one or more machines in the group to specified users or groups of users.

Groups of shared machines can be used to deliver full desktop or seamless application sessions, or both.

For more detailed information about configuring shared machines, see:

help about\_Broker\_AccessPolicy  
help about\_Broker\_EntitlementPolicy

## PRE-ASSIGNED PRIVATE MACHINES

To grant access to private machines, use the access policy and a machine assignment:

- o The access policy grants access to the desktop group containing the machines.
- o The assignment links the desktop to a specified user. You can assign a machine to just one user, multiple users or user groups. However, for single-session machines, only one user can access the machine at a time.

Private machines can be used to deliver full desktop or seamless application sessions (but not both).

For more detailed information about configuring private machines, see:

help about\_Broker\_AccessPolicy  
help Add-BrokerUser

## ASSIGN-ON-FIRST-USE (AOFU) MACHINES

To grant access to a desktop group containing assignable machines, use the access policy and the assignment policy:

- o The access policy grants access to the desktop group containing the pool of machines.
- o The assignment policy grants users a self-service entitlement to pick one or more machines from the pool.

AOFU machines can be used to deliver full desktop or seamless application sessions (but not both from the same desktop group).

For more detailed information about configuring AOFU desktops, see:

`help about_Broker_AccessPolicy`  
`help about_Broker_AssignmentPolicy`

## REMOTE PC MACHINES

The RemotePC feature allows existing physical machines to be assigned automatically to their normal user thus allowing them remote access to their own machine but without the need for the administrator to individually configure access to each machine.

For more detailed information about configuring the Remote PC feature, see:

`help about_Broker_RemotePC`

## SEE ALSO

[about\\_Broker\\_AccessPolicy](#)

[about\\_Broker\\_EntitlementPolicy](#)

[about\\_Broker\\_AssignmentPolicy](#)

[about\\_Broker\\_RemotePC](#)

[New-BrokerAccessPolicyRule](#)

[New-BrokerEntitlementPolicyRule](#)

[New-BrokerAssignmentPolicyRule](#)

[New-BrokerAppEntitlementPolicyRule](#)

[New-BrokerAppAssignmentPolicyRule](#)



# about\_Broker\_PostInstallPreConfiguration

May 10, 2016

## TOPIC

Citrix Broker SDK - Post-Installation Configuration

## SHORT DESCRIPTION

Describes how to configure the Citrix Broker Service port numbers, URL reservations, and Windows Firewall exclusions.

## LONG DESCRIPTION

The XenDesktop installer configures the Citrix Broker Service with information specified during the installation. To change that configuration, use the BrokerService.exe command-line tool on each controller you want to change.

The default installation location of BrokerService.exe is:

```
%ProgramFiles%\Citrix\Broker\Service\BrokerService.exe
```

BrokerService.exe supports the following optional command-line parameters:

-SdkPort <port> (default 80)

Configures the port on which the broker listens for requests from SDK cmdlets. If you change this default value, specify the new value in the AdminAddress parameter on broker cmdlets. For example, if you changed the port to 8080, specify it as follows:

```
Get-BrokerSite -AdminAddress localhost:8080
```

-VdaPort <port> (default 80)

Configures the port on which the broker listens for registration requests from broker machines.

-WiPort <port> (default 80)

Configures the port on which the broker listens for XML requests from StoreFront/Web Interface.

-WiSslPort <port> (default 443)



Configures the port on which the broker listens for SSL (Secure Socket Layer) XML requests from StoreFront/Web Interface.

-ConfigureFirewall

Configures Windows Firewall exclusions for the specified ports.

-Show

Shows the current configuration settings.

-Uninstall

Removes configuration settings, including Windows Firewall exclusions and URL reservations.

-LogFile <fileName>

Configures the file location for logging to a text file. The directory containing the log file must grant write access to the NetworkService account.

-Upgrade

Performs configurations required after an upgrade.

-Quiet

Suppresses console output for status messages.

-? or -Help

Shows usage information for the command.

# about\_Broker\_PowerManagement

May 10, 2016

## TOPIC

Citrix Broker SDK - Machine Power Management

## SHORT DESCRIPTION

Describes power management of machines used for desktops and applications.

## LONG DESCRIPTION

The Citrix Broker Service is in day-to-day control of the power state of the configured desktop and application machines. The Broker Service can control several hypervisors, each hypervisor connection being handled by its own site service, so all Broker Service communication to the hypervisor is through one of the controllers in the site.

## HYPERVISOR CONNECTIONS

Each hypervisor, or pool of linked hypervisors, is described and configured through the XdHyp pseudo-drive and associated Hyp PowerShell commands (cmdlets) which are provided by the host service snap-in. When you have first created the hypervisor connection using the host service cmdlets, you can create a broker equivalent object that references the Hyp instance using a GUID value. Use the broker's HypervisorConnection object to nominate a preferred controller for direct communication with the hypervisor on behalf of all other controllers for day-to-day power actions and status requests.

## POWER ACTIONS AND THROTTLING

The site, through the Broker Service, can control the power state of the machines used by the site for desktops and applications. Power state changes can have a number of causes:

- o Power policy rules, such as requests to shut down or suspend machines when user sessions on those machines end or are disconnected
- o If allowed, user-driven desktop restarts
- o Session launch requests requiring machines to be started
- o Pool size management, which controls the number of running machines
- o Direct administrator request using the SDK or Citrix Studio
- o Reboot schedules and cycles
- o Performing personal vDisk inventory activities
- o Cleaning machines back to the golden master image state after they have been used

The power state changes of machines hosting desktops and applications are controlled using a queuing mechanism. Actions to change the power state are assigned a priority and are sent to the hypervisor according to a throttling mechanism. This avoids overloading the hypervisor.

The queuing and throttling mechanisms take place on a per-hypervisor-connection basis; each hypervisor connection's queue is dealt with independently. You can view the contents of the queues using the Get-BrokerHostingPowerAction cmdlet. This includes recently completed, in-progress, and pending actions (that is, those due to be sent to the hypervisor

based on the throttling settings).

Each power action object comprises:

- o The machine to be acted on (Name, DNS Name, Hosting Name)
- o The action to be performed (TurnOn, TurnOff, Shutdown, Reset, Restart, Suspend, or Resume)
- o The action's priority (Base, the original priority, and Actual, the current priority)
- o The action's state (Pending, Started, Completed, Failed, Canceled, Deleted, or Lost)
- o Time stamps of the action's lifecycle points (when it was created, started, or completed)
- o Any reason the action failed

The throttling of power actions is controlled by three metadata values on the Hyp hypervisor connection object when accessed through the XdHyp pseudo-drive. The four values throttle power actions according to:

- o The maximum absolute number of in-progress power actions
- o The maximum number of in-progress power actions expressed as a percentage of the total number of machines controlled by the hypervisor connection
- o The maximum number of new power actions sent to the hypervisor per minute
- o The maximum number of in-progress PvD inventory activities expressed as a percentage of the total number of machines controlled by the hypervisor connection

You add power actions to the queue using the SDK's New-BrokerHostingPowerAction cmdlet. You cancel power actions in the queue using the Remove-BrokerHostingPowerAction cmdlet. You boost or reduce their priority using the Set-BrokerHostingPowerAction cmdlet.

You can also schedule power actions to be executed in the future, using the New-BrokerDelayedHostingPowerAction command. Only Shutdown and Suspend actions can be scheduled in this way. You can view these delayed power actions using Get-BrokerDelayedHostingPowerAction and cancel them with Remove-BrokerDelayedHostingPowerAction. When a delayed power action is ready to be executed it is deleted, and a corresponding normal power action is placed in the queue described above.

## POWER POLICY

Policy rules associated with a desktop group allow you to change power states at configurable times after session state changes, typically a set number of minutes after session disconnection or session logout.

Note that these policy rules are defined directly as properties of the desktop group.

These policy actions allow the following operations to be specified:

- o A power action to be performed at a defined period after a session is disconnected
- o A power action to be performed at a defined extended period after a session is disconnected
- o A power action to be performed at a defined period after a session is logged off

The two disconnect policy actions are designed to allow multi-stage policies such as initially suspending a machine shortly after a session disconnect occurs, and then later powering-off the machine if the session has not been reconnected.

At the set time after the session state change, the required action is added to the power action queue, and this is then throttled and processed as normal.

## POOL SIZE MANAGEMENT

You can manage flexibly the number of machines running desktops and applications using the pool size. For any given hour of the day and day of the week, this is an absolute number of machines or the percentage of the total number of machines in the desktop group. The pool size specifies the total number of machines that are always running, regardless of whether they are in use or idle. (Note: The number of machines does not depend on their idle status, but this does affect the buffer size value, which is also used to manage pool sizes.)

To start or shut down desktop machines to achieve the desired pool size, the system places power actions in the queue. Standard throttling queue management sends these to the hypervisors. A single desktop group (and its pool) can span multiple hypervisors, so actions to start and shut down machines can be added to multiple queues.

## POWER TIME SCHEMES

Each single-session desktop group can be associated with one or more power time schemes, each scheme covering a number of days of the week. The time schemes specify, for each hour of the day, whether that hour is peak or off-peak. They also specify the number of running unassigned machines maintained by the broker.

You can configure other settings, such as buffer size and any power policy rules differently for peak and off-peak hours. You can define the number of running machines, idle or in use, either as an absolute value or as a percentage of the desktop group size. Machines running desktops and applications are started (or shut down when not in use) to match the required pool size.

Each power time scheme comprises:

- o The name of the scheme
- o The pattern of days of the week covered by the scheme
- o The set of hours considered peak and off peak
- o The set of pool size values (one for each hour of the day)

The hours of the day used by time schemes are the hours in the time zone for the desktop group the scheme is associated with. You cannot associate one desktop group with multiple time schemes covering the same day of the week.

## BUFFER SIZE

In addition to the pool size, you can optionally configure two buffer sizes for each desktop group, one for peak hours and one for off-peak hours. The buffer size defines the minimum number of idle unassigned machines maintained by the broker and is specified as a percentage of the total machines in the group. These are running machines that are not used by any user session. The buffer size on its own never causes machines to shut down. It causes them to start up so a minimum number of idle machine is always available. The buffer size in conjunction with the pool size can cause machines running desktops or applications to be shut down.

## POWER MANAGEMENT OF ASSIGNED MACHINES

Automatic power management for private desktop groups provides the ability to power on all assigned machines at the transition to a peak period and respectively power off all machines at the transition to an off-peak period.

If a machine is shut down during peak hours it will not be automatically powered on again, unless the `AutomaticPowerOnForAssignedDuringPeak` property on the desktop group is also enabled.

Note that all power management facilities apply only to single session machines.

## REBOOT SCHEDULES

Reboot schedules are commonly used after image updates or to perform regular reboots of all machines in a desktop group or catalog to clear down problems resulting from a corrupt state or hung/faulty applications.

Reboot schedules allow distributing the reboot operation of all machines over a provided duration. Individual machine reboots are scheduled in a way that attempts to maintain maximum availability of machines in the group as the reboots occur, and avoid boot storms that overload the underlying infrastructure.

Reboot schedules are the only form of automatic power management that can shut down a machine while users are logged on; however the administrator can provide a warning message to be displayed to end users at a specified period prior to the shutdown taking effect.

## REBOOT CYCLES

Reboot cycles describe the dynamic execution of desktop group or catalog reboot operations. Reboot cycles can be created due to reboot schedules, or by on-demand reboot operations requested via the SDK.

`RebootCycle` objects encapsulate the details of the associated reboot operation and can be queried to show the current status.

## STATUS

You can view the status of the hypervisor connection on the broker hypervisor connection object. You can obtain any hypervisor alerts using the `Get-BrokerHypervisorAlert` cmdlet. You can check the power state of machines running desktops or applications using the relevant `Machine` or `Desktop` objects.

## SEE ALSO

[about\\_Broker\\_Concepts](#)

[about\\_Broker\\_Machines](#)

[about\\_HypHostSnapin](#)

[New-BrokerHypervisorConnection](#)

Get-BrokerHypervisorConnection  
Set-BrokerHypervisorConnection  
Remove-BrokerHypervisorConnection  
New-BrokerHostingPowerAction  
Get-BrokerHostingPowerAction  
Set-BrokerHostingPowerAction  
Remove-BrokerHostingPowerAction  
New-BrokerDelayedHostingPowerAction  
Get-BrokerDelayedHostingPowerAction  
Remove-BrokerDelayedHostingPowerAction  
New-BrokerPowerTimeScheme  
Get-BrokerPowerTimeScheme  
Set-BrokerPowerTimeScheme  
Rename-BrokerPowerTimeScheme  
Remove-BrokerPowerTimeScheme  
Get-BrokerRebootCycle  
Set-BrokerRebootCycleMetadata  
Start-BrokerRebootCycle  
Stop-BrokerRebootCycle  
Get-BrokerRebootSchedule  
Set-BrokerRebootSchedule  
New-BrokerRebootSchedule  
Remove-BrokerRebootSchedule  
New-BrokerDesktopGroup  
Get-BrokerDesktopGroup  
Set-BrokerDesktopGroup  
Add-HypMetadata  
Remove-HypMetadata

# about\_Broker\_RemotePC

May 10, 2016

## TOPIC

Citrix Broker SDK - RemotePC

## SHORT DESCRIPTION

Overview of the Remote PC feature.

## LONG DESCRIPTION

Remote PC allows automatic publishing of a user's physical desktop within a XenDesktop site, so that it can be accessed remotely. The configuration of Remote PC within the Citrix Broker service specifies the rules and relationships that allow the machine to successfully register with a controller in the site, and be made available for the user to start remote sessions.

When Remote PC is configured, there are two steps required for publishing a machine as a Remote PC desktop.

First a VDA must be installed on the machine and it must be configured such that it registers with a controller in a site.

Second, a user must log on to the machine. When the Citrix Broker service detects an active console session for a user, it assigns the user to the machine and publishes it in a Remote PC desktop group.

Remote PC configuration consists of defining relationships between:

- o Machines and catalogs
- o Catalogs and desktop groups
- o Desktop groups and assignment policy rules
- o Assignment policy rules and users

The Citrix Broker service automates the assignment of Remote PC machines to users in two stages. The first stage automatically imports matching unconfigured machines into the site:

- o Suitable machines are automatically added to a Remote PC catalog.
- o The machines are temporarily configured to be in one of the Remote PC desktop groups associated with the catalog.

When a suitable user logs on to the console of the machine, the second stage of the automatic configuration is performed:

- o The machine is configured to be in the desktop group that is appropriate for the user.
- o The machine is assigned to the user that has logged on.

- o The machine desktop is made available to the user remotely, configured to appear as the machine hostname.

Catalogs and desktop groups can be marked as participating in RemotePC automation with the 'IsRemotePC' property, but this property can only be set to true if various other properties of the catalog or desktop group are appropriate. Catalogs must be single-sessioned and contain physical machines. Desktop Groups must be single session, configured to deliver private assigned machines, delivering desktop sessions only.

Catalogs and desktop groups can have the 'IsRemotePC' property cleared to remove them from the RemotePC automation, but only when all RemotePC associations relating to them through RemotePCAccounts and catalog/group relationships have first been removed.

## MACHINES AND CATALOGS

Mappings are defined between machines and Remote PC catalogs through the RemotePCAccount cmdlets.

The machine to catalog mappings support the automated addition of machines to catalogs.

## PROPERTIES

RemotePCAccounts expose sets of included and excluded machine name filters specified in DOMAIN\MACHINE format. A MachinesIncluded or MachinesExcluded entry can include asterisk wildcards to generalize matches.

Each RemotePCAccount can specify the Distinguished Name (DN) of an AD container in addition to the machine name filters, in the RemotePCAccount Organisational Unit (OU) property, and this limits the machines that the account objects act on to those that reside at or below that container in the AD domain hierarchy. An AllowSubfolderMatches setting on the RemotePCAccount indicates whether the computer must exist directly within the container to trigger a match, or whether it can be in a child below the defined container in the AD hierarchy.

Note that the AD container component is optional and a special value of 'any' can be supplied in the OU field to permit the RemotePCAccount to automatically match regardless of the AD machine object location in the AD domain hierarchy. A match is still subject to machine name filtering.

The last component of the RemotePCAccount is the CatalogUid. This indicates which catalog the Remote PC automation should move the machine into when a match is found.

## CONSTRAINTS

The IsRemotePC setting must be enabled on catalogs before they can be specified in a RemotePCAccount.

There can be any number of RemotePCAccounts configured in the site as long as each specifies a unique OU. There can be only one RemotePCAccount with the 'any' OU.

## AUTOMATION

When a machine matching the criteria set up in a RemotePCAccount instance registers with one of the brokers in the site, it is automatically added to the catalog defined by the RemotePCAccount. This can take up to 30 seconds to happen after the machine registers.

When the machine registration occurs, the machine details can match more than one RemotePCAccount instance, but the machine can only be placed into one catalog, so one RemotePCAccount instance is chosen from the list that best matches



the machine. This choice is made according to the length in nodes of the DN of the AD container specification associated with the RemotePCAccount, so more specific child OUs override specifications for their parent OUs if both are present. The RemotePCAccount for the 'any' OU is always last and used only if no other instances match.

A Windows eventlog message is generated when an automated catalog assignment is performed.

## NOTES

The AD distinguished name for the container is checked when the RemotePCAccount is created, but if the container is subsequently moved or deleted, the site does not automatically accommodate this, and the RemotePCAccount must be changed or removed manually.

## Related Cmdlets

- o [New-BrokerRemotePCAccount](#)
- o [Get-BrokerRemotePCAccount](#)
- o [Set-BrokerRemotePCAccount](#)
- o [Remove-BrokerRemotePCAccount](#)
- o [New-BrokerCatalog \[-IsRemotePC <Boolean>\]](#)
- o [Set-BrokerCatalog \[-IsRemotePC <Boolean>\]](#)

## CATALOGS AND DESKTOP GROUPS

A Remote PC catalog may be associated with one or more Remote PC desktop groups. The catalog to desktop group associations support automated publishing of machines to users.

## USAGE

An association is formed via:

```
Add-BrokerDesktopGroup -RemotePCCatalog <Catalog> [-Priority <Int32>]
```

An association is broken via:

```
Remove-BrokerDesktopGroup -RemotePCCatalog <Catalog>
```

To find associated desktop groups:

```
Get-BrokerDesktopGroup -RemotePCCatalogUid <Int32>
```

## AUTOMATION

When a machine in a Remote PC catalog is not already assigned to a user, Remote PC automation temporarily places the

machine in one of the desktop groups associated with the catalog. The temporary placing of the machine in a group will be adjusted as needed when the machine assignment to a user is first made.

The desktop group chosen as the temporary home for a machine is according to the priority value supplied when making the association between the group and the catalog. The group with the highest priority (lowest numerical priority value) is chosen.

A Windows eventlog message is generated when an automated machine assignment is performed.

## PRIORITY

Each desktop group to catalog association has a priority value that can be specified when the association is made or defaults to lower than the lowest existing association priority (highest numerical value) or zero if no other association exists yet. The lower the priority numerical value, the higher the priority level. The priority value for the association is used to choose the temporary desktop group to place a new RemotePC machine in when it first registers. It is also used to decide which group to finally place the machine in when the user that the machine is being assigned to is appropriate for more than one of the associated desktop groups, usually because the desktop groups are using AD security groups for the user associations with the desktop groups.

The priority values for each association between a desktop group and a catalog are automatically maintained as unique for each catalog, and priorities can be adjusted up or down accordingly by the system. The last association created without a specified priority is always arranged to have the least priority (the highest numerical priority value).

## NOTES

The temporary placement of the machines in a desktop group is not re-evaluated if settings change after the automatic placement.

## RELATED CMDLETS

- o [Add-BrokerDesktopGroup -RemotePCCatalog <Catalog>](#)
- o [Remove-BrokerDesktopGroup \[-RemotePCCatalog <Catalog>\]](#)
- o [Get-BrokerDesktopGroup \[-RemotePCCatalogUid <Int32>\]](#)
- o [New-BrokerCatalog \[-IsRemotePC <Boolean>\]](#)
- o [Set-BrokerCatalog \[-IsRemotePC <Boolean>\]](#)
- o [New-BrokerDesktopGroup \[-IsRemotePC <Boolean>\]](#)
- o [Set-BrokerDesktopGroup \[-IsRemotePC <Boolean>\]](#)

## DESKTOP GROUPS, ASSIGNMENT POLICY RULES, AND USERS

Assignment policy rules are used to define sets of users allowed to be assigned to machines by Remote PC automation, and to determine which desktop group the machine is placed in when the user assignment is made.

## AUTOMATION

A machine is automatically assigned to a user when the Citrix Broker service sees that the user has logged on to a RemotePC machine, and the user is amongst those configured in the desktop group assignment policy rule. If this is the first assignment of a user to a machine, all the assignment policy rules for all groups associated with the machine are checked,

and the machine can be moved to a different, more appropriate desktop group if needed.

A Windows eventlog message is generated when an automated assignment of a machine to a user is made.

## MULTIPLE ASSIGNMENTS

By default, multiple automatic assignments of users to the same machine can be established if multiple users log on to the RemotePC machine, but this can be disabled if desired using a registry setting (see CTX137805).

## NOTES

User to machine assignments can still be made and removed manually through the Powershell SDK for machines in Remote PC catalogs and desktop groups.

The automatic placement of a machine in an appropriate desktop group happens when the first automatic assignment of a user to the machine is made, and this placement will not be automatically updated if the configuration subsequently changes.

Only a single assignment policy rule is allowed for each RemotePC desktop group.

## RELATED CMDLETS

- o [New-BrokerUser](#)

- o [New-BrokerAssignmentPolicyRule](#)

- o [Set-BrokerAssignmentPolicyRule](#)

## EXAMPLE

The following example creates a simple configuration that allows any user and machine in the current AD domain to participate in RemotePC.

```
Create a Remote PC catalog
$catalog = New-BrokerCatalog -IsRemotePC $true
 -SessionSupport SingleSession
 -MachinesArePhysical $true
 -AllocationType Static
 -PersistUserChanges OnLocal
 -ProvisioningType Manual
 -Name RemotePCCatalog

Create a Remote PC desktop group
$dg = New-BrokerDesktopGroup -IsRemotePC $true
 -SessionSupport SingleSession
 -DeliveryType DesktopsOnly
 -DesktopKind Private
 -Name RemotePCDesktopGroup

Create an assignment policy rule for that desktop group allowing any
domain user to match.
```

```
New-BrokerAssignmentPolicyRule -DesktopGroupUid $dg.Uid
 -IncludedUsers 'domain users'
 -Name RemotePCAPR
```

```
Create a RemotePCAccount matching any unconfigured machine, causing the
machines to be added to the catalog by Remote PC automation.
```

```
New-BrokerRemotePCAccount -OU 'any'
 -CatalogUid $catalog.Uid
```

```
Associate the desktop group and catalog to permit domain users to be
automatically assigned to machines in that catalog
Add-BrokerDesktopGroup $dg -RemotePCCatalog $catalog
```

```
#Create an access policy rule, allowing access to the users to the
#Remote PC desktop group.
```

```
New-BrokerAccessPolicyRule -IncludedUsers 'domain users'
 -DesktopGroupUid $dg.Uid
 -IncludedUserFilterEnabled $true
 -Name RemotePCAccessPolicyRule
```

# Add-BrokerApplication

May 10, 2016

Adds applications to a desktop group.

## Syntax

```
Add-BrokerApplication [-InputObject] <Application[]> [-DesktopGroup <DesktopGroup>] [-Priority <Int32>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Add-BrokerApplication [-Name] <String> [-DesktopGroup <DesktopGroup>] [-Priority <Int32>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Add-BrokerApplication cmdlet is used to associate one or more applications with an existing desktop group.

There are two parameter sets for this cmdlet, allowing you to specify the application either by its BrowserName or by an array of object references. Uids can also be substituted for the object references.

See about\_Broker\_Desktops and about\_Broker\_Applications for more information.

## Related topics

[New-BrokerApplication](#)

[Add-BrokerApplication](#)

[Add-BrokerTag](#)

[Remove-BrokerApplication](#)

[Rename-BrokerApplication](#)

[Move-BrokerApplication](#)

[Set-BrokerApplication](#)

## Parameters

**-InputObject** <Application[]>

Specifies the application to associate. Its Uid can also be substituted for the object reference.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name**<String>

Specifies the name of the application to be associated with the desktop group.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-DesktopGroup**<DesktopGroup>

Specifies which desktop group this application should be associated with. Note that applications can only be associated with desktop groups of the AppsOnly or DesktopsAndApps delivery type.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Priority**<Int32>

Specifies the priority of the mapping between the application and desktop group where lower numbers imply higher priority with zero being highest.

If one association has a higher priority than the other, machines from that group will be selected for launching sessions until all machines are at maximum load, in maintenance mode, unregistered, or unavailable for any other reason. Only when all machines from the higher-priority group are unavailable will new connections be routed to the next lowest priority group.

If multiple associations have equal priority, load balancing does not occur among the desktop groups in these associations. Instead, the broker chooses one of these groups as the preferred group and machines from this group will be selected for launching sessions until all machines are at maximum load, in maintenance mode, unregistered, or unavailable for any other reason. Only when all machines from the preferred group are unavailable will new connections be routed to another one of these groups, which the broker chooses as next-most preferred.

Required?	false
Default Value	0
Accept Pipeline Input?	true (ByPropertyName)

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.Application, or as appropriate by property name You can pipe the application to be added to Add-BrokerApplication. You can also pipe some of the other parameters by name.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Add-BrokerApplication -BrowserName "Notepad" -DesktopGroup "Private DesktopGroup"
```

Adds the application with a BrowserName of "Notepad" to the desktop group called "Private DesktopGroup".

# Add-BrokerDesktopGroup

May 10, 2016  
Associate Remote PC desktop groups with the specified Remote PC catalog.

## Syntax

Add-BrokerDesktopGroup [-InputObject] <DesktopGroup[]> [-RemotePCCatalog <Catalog>] [-Priority <Int32>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Add-BrokerDesktopGroup [-Name] <String> [-RemotePCCatalog <Catalog>] [-Priority <Int32>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

This cmdlet forms relationships between Remote PC desktop groups and catalogs.

The Remote PC relationships are used by Remote PC automation to determine which desktop groups a machine in a particular Remote PC catalog can be published to. The assignment policy rules belonging to those desktop groups also determines the set of users that are allowed to be assigned to machines from the catalog.

## Related topics

[Remove-BrokerDesktopGroup](#)

[Add-BrokerCatalog](#)

[Remove-BrokerCatalog](#)

## Parameters

**-InputObject**<DesktopGroup[]>

Specifies one or more Remote PC desktop groups to add to a Remote PC catalog.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the Remote PC desktop groups to add to a Remote PC catalog based on their name properties.

Required?	true
Default Value	



Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-RemotePCCatalog**<Catalog>

The Remote PC catalog which the desktop groups are to be added to. Specified by name, Uid or instance.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Priority**<Int32>

Desktop group to catalog associations carry a priority number, where numerically lower values indicate a higher priority.

The priority relative to other associations determines which desktop group Remote PC automation will move a qualifying unconfigured machine into when it registers. Priority also determines which desktop group a machine will be published to when a user is assigned to the machine by Remote PC automation.

If a value is not supplied, then the desktop group association is automatically assigned a lower priority than any existing associations.

If a priority value is specified that conflicts with an existing association's priority value, then the new association is inserted with that value and existing associations are renumbered upwards to accommodate it.

Required?	false
Default Value	See description
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.DesktopGroup The set of Remote PC desktop groups to be added to the catalog can be piped into this cmdlet.

## Return Values

None

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Get-BrokerDesktopGroup -IsRemotePC $true | Add-BrokerDesktopGroup -RemotePCCatalog 42
```

Add all Remote PC desktop groups to Remote PC catalog 42.

### ----- **EXAMPLE 2** -----

```
C:\PS> Add-BrokerDesktopGroup -Name *MyGroup* -RemotePCCatalog RPCCat
```

Add desktop groups with names containing MyGroup to Remote PC catalog with name "RPCCat".

# Add-BrokerMachine

May 10, 2016  
Adds one or more machines to a desktop group.

## Syntax

Add-BrokerMachine [-InputObject <Machine[]> [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Add-BrokerMachine [-MachineName <String> [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The Add-BrokerMachine cmdlet adds specified machines to a desktop group. There are three forms:

- o Use the -InputObject parameter to add a single machine instance or array of instances to the group.
- o Use the -MachineName parameter to add a single, named machine to the group.
- o Use pipelining to pipe machines instances to the command.

The desktop group to which the machines are added can be specified by name, unique identifier (UID), or instance.

For a machine to be used in a site, the machine must be added to a desktop group. The machine and desktop group must be compatible in order for the process to succeed; for example a machine in a single-session catalog cannot be added to a multi-session desktop group.

For more information about machines, see about\_Broker\_Machines.

## Related topics

[Add-BrokerMachinesToDesktopGroup](#)

[Remove-BrokerMachine](#)

[Get-BrokerMachine](#)

## Parameters

**-InputObject**<Machine[]>

An array of machines to add to the group.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-MachineName**<String>

The name of the single machine to add (must match the MachineName property of the machine).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-DesktopGroup**<DesktopGroup>

The desktop group to which the machines are added, specified by name, Uid, or instance.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.Machine You can pipe in the machines you want to add.

## Return Values

None

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Add-BrokerMachine -InputObject $machine -DesktopGroup $desktopGroup
```

```
C:\PS> Add-BrokerMachine -InputObject $machine -DesktopGroup 2
```

```
C:\PS> Add-BrokerMachine $machine -DesktopGroup "MyDesktopGroup"
```

These examples all add a single machine instance to a desktop group, identifying the group by instance, UID, or name.

### ----- EXAMPLE 2 -----

```
C:\PS> Add-BrokerMachine -MachineName "MyDomain\MyMachine" -DesktopGroup 2
```

```
C:\PS> Add-BrokerMachine "MyDomain\MyMachine" -DesktopGroup "MyDesktopGroup"
```

```
C:\PS> Add-BrokerMachine "MyDomain\MyMachine" -DesktopGroup $desktopGroup
```

These examples add the machine called MyMachine to a desktop group.

### ----- EXAMPLE 3 -----

```
C:\PS> Get-BrokerMachine -Uid 3 | Add-BrokerMachine -DesktopGroup 2
```

```
C:\PS> Get-BrokerMachine -CatalogUid 4 | Add-BrokerMachine -DesktopGroup 2
```

These examples find specific machines and add them to a desktop group.

# Add-BrokerMachineConfiguration

May 10, 2016  
Adds a machine configuration to a desktop group.

## Syntax

Add-BrokerMachineConfiguration [-InputObject <MachineConfiguration[]>] [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Add-BrokerMachineConfiguration [-Name <String>] [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Associates a machine configuration with a desktop group. The settings in the machine configuration are then applied to the machines in the desktop group.

## Related topics

[New-BrokerMachineConfiguration](#)

[Set-BrokerMachineConfiguration](#)

[Rename-BrokerMachineConfiguration](#)

[Remove-BrokerMachineConfiguration](#)

## Parameters

**-InputObject**<MachineConfiguration[]>

Machine configuration to add to the desktop group.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Name of a machine configuration to add to the desktop group.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

### **-DesktopGroup**<DesktopGroup>

The desktop group to which the machine configurations are added, specified by name, Uid, or instance.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.MachineConfiguration The machine configuration to add to the desktop group.

### Return Values

None

### Examples

----- **EXAMPLE 1** -----

```
Add-BrokerMachineConfiguration -Name UPM\Conf1 -DesktopGroup 1
```

Adds the machine configuration named UPM\Conf1 to the desktop group with Uid 1.

----- **EXAMPLE 2** -----

```
$mc | Add-BrokerMachineConfiguration -DesktopGroup AdminDesktops
```

Adds the machine configuration \$mc to the desktop group named "AdminDesktops".



# Add-BrokerMachinesToDesktopGroup

May 10, 2016

Adds machines from a catalog to a desktop group.

## Syntax

```
Add-BrokerMachinesToDesktopGroup [-Catalog] <Catalog> [-DesktopGroup] <DesktopGroup> [-Count] <Int32> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Add-BrokerMachinesToDesktopGroup cmdlet adds a specified number of machines from a catalog to a desktop group.

The cmdlet adds as many machines as possible from the catalog to the desktop group, up to the specified number. The number of machines successfully added to the desktop group is returned.

The machines are added randomly from the catalog and are selected from those that are not already members of a desktop group, and not already assigned to a client, IP address, or user.

Both the catalog and desktop group can be referenced either by instance, name, or unique identifier (Uid). The allocation type of the catalog must be compatible with the type of desktop group. This means the session support (single/multi) and the allocation type (private/shared) of the catalog must match the session support and allocation type in the desktop group.

## Related topics

[Add-BrokerMachine](#)

[Remove-BrokerMachine](#)

## Parameters

**-Catalog**<Catalog>

The catalog from which the machines are taken.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-DesktopGroup**<DesktopGroup>

The desktop group to which the machines are added.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Count**<Int32>

The number of machines to add to the desktop group.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.Catalog, or the name or Uid of the catalog. You can pipe in the catalog from which the machines are taken. Alternatively, you can pipe the name or the Uid of the catalog.

### Return Values

System.Int32

The number of machines added to the desktop group.

### Examples

----- **EXAMPLE 1** -----

```
C:\PS> Add-BrokerMachinesToDesktopGroup -Catalog $catalog -DesktopGroup $desktopGroup -Count 1000
```

```
C:\PS> Add-BrokerMachinesToDesktopGroup -Catalog "MyCatalog" -DesktopGroup "MyDesktopGroup" -Count 1000
```

```
C:\PS> Add-BrokerMachinesToDesktopGroup -Catalog 23 -DesktopGroup 4 -Count 1000
```

All these examples request that a thousand machines from a catalog are added to a desktop group. The first example references both catalog and desktop group by instance. The second example references both catalog and desktop group by name. The third example references both catalog and desktop group by Uid.

----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerCatalog -ProvisioningType Manual | Add-BrokerMachinesToDesktopGroup -DesktopGroup $desktopGroup -Count 10
```

This example takes ten machines from each manually provisioned catalog and adds them to the specified desktop group.

# Add-BrokerScope

May 10, 2016

Add the specified catalog/desktop group to the given scope(s).

## Syntax

```
Add-BrokerScope [-InputObject] <Scope[]> [-Catalog <Catalog>] [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Add-BrokerScope cmdlet is used to associate a catalog or desktop group object with given scope(s).

To add a catalog/desktop group to a scope you need permission to change the scopes of the catalog/desktop group and permission to add objects to all of the scopes you have specified.

If the catalog/desktop group is already in any scope supplied, that scope will be silently ignored.

## Related topics

## Parameters

**-InputObject**<Scope[]>

Specifies the scope(s) to add the object to. Each can take the form of either the string form of the scope's GUID or its name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Catalog**<Catalog>

Specifies the catalog object to be added. This can take the form of an existing catalog object, a catalog Uid or name.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

**-DesktopGroup**<DesktopGroup>

Specifies the desktop group object to be added. This can take the form of an existing desktop group object, a desktop

group Uid or name.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.Scope You can pipe scopes to Add-BrokerScope.

#### Return Values

NONE

#### Examples

##### ----- EXAMPLE 1 -----

C:\PS> Add-BrokerScope -Scope Chalfont -DesktopGroup 27  
Adds the desktop group with Uid 27 to the Chalfont scope.

----- **EXAMPLE 2** -----

```
C:\PS> Add-BrokerScope BFC74867-C6EF-482C-996F-3E0D340E96AC -Catalog BangaloreMachines
```

Adds the BangaloreMachines catalog to the scope with the specified ScopeID.

# Add-BrokerTag

May 10, 2016

Associate a tag with another object.

## Syntax

```
Add-BrokerTag [-InputObject] <Tag[]> [-Desktop <Desktop>] [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Add-BrokerTag [-Name] <String> [-Desktop <Desktop>] [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Associates one or more tags with another object.

## Related topics

[Get-BrokerTag](#)

[New-BrokerTag](#)

[Remove-BrokerTag](#)

[Rename-BrokerTag](#)

## Parameters

**-InputObject**<Tag[]>

Specifies one or more tag objects.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies a tag by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-Desktop**<Desktop>

Associates the tag with a desktop.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-DesktopGroup**<DesktopGroup>

Associates the tag with a desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.



Accept Pipeline Input?	false
------------------------	-------

## Input Type

Citrix.Broker.Admin.SDK.Tag Tags may be specified through pipeline input.

## Return Values

None

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> $desktop = Get-BrokerDesktop -Uid 1
C:\PS> Add-BrokerTag -Name 'Tag1' -Desktop $desktop
Associates 'Tag1' with Desktop $desktop.
```

### ----- EXAMPLE 2 -----

```
C:\PS> $desktop = Get-BrokerDesktop -Uid 1
C:\PS> New-BrokerTag 'Tag2' | Add-BrokerTag -Desktop $desktop
Creates a new tag with name 'Tag2' and associates it with Desktop $desktop.
```

# Add-BrokerUser

May 10, 2016  
Creates an association between a user and another broker object

## Syntax

Add-BrokerUser [-InputObject <User[]>] [-Application <Application>] [-SessionLinger <SessionLinger>] [-SessionPreLaunch <SessionPreLaunch>] [-Machine <Machine>] [-PrivateDesktop <PrivateDesktop>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Add-BrokerUser [-Name] <String> [-Application <Application>] [-SessionLinger <SessionLinger>] [-SessionPreLaunch <SessionPreLaunch>] [-Machine <Machine>] [-PrivateDesktop <PrivateDesktop>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The Add-BrokerUser cmdlet adds broker user objects to another specified object, such as a broker private desktop. This depends on the target object type:

- o Machine - assign the broker machine to the specified user(s); when the machine is subsequently added to a desktop group, the desktop is also assigned to the same user(s).
- o PrivateDesktop - assign the desktop to the specified user(s).
- o Application - assign the application to the specified user(s).

## Related topics

- [Get-BrokerUser](#)
- [Remove-BrokerUser](#)

## Parameters

**-InputObject**<User[]>

The user objects to add.

Required?	true
Default Value	null
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

The name of the user or users to be added.

--	--

Required?	true
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

#### **-Application**<Application>

The application to which the user is to be associated.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-SessionLinger**<SessionLinger>

The session linger setting to which the user is to be assigned.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByValue)

#### **-SessionPreLaunch**<SessionPreLaunch>

The session pre-launch setting to which the user is to be assigned.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByValue)

#### **-Machine**<Machine>

The machine to which the user is to be assigned

Required?	false
-----------	-------

Default Value	null
Accept Pipeline Input?	true (ByValue)

### **-PrivateDesktop**<PrivateDesktop>

The desktop to which the user is to be assigned

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.User You can pipe the users to be added to Add-BrokerUser.

### Return Values

None

## Notes

Specify one of the -Machine or -PrivateDesktop or -Application parameters only.

## Examples

### ----- **EXAMPLE 1** -----

```
Add-BrokerUser "DOMAIN\UserName" -PrivateDesktop "DOMAIN\MachineName"
```

Assign the specified private desktop to the specified user.

### ----- **EXAMPLE 2** -----

```
Add-BrokerUser "DOMAIN\UserName" -Application "ApplicationName"
```

Assign the specified application to the specified user.

### ----- **EXAMPLE 3** -----

```
Add-BrokerUser "DOMAIN\UserName" -Application "FolderName"\ApplicationName"
```

Assign the specified application, in the specified folder, to the specified user.

# Disconnect-BrokerSession

May 10, 2016

Disconnect a session.

## Syntax

```
Disconnect-BrokerSession [-InputObject] <Session[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Disconnects the specified session.

If the session is active, no warning is issued to the user before that session is disconnected.

After disconnection, sessions enter a Disconnected state. In a Disconnected state, a session still exists but there is no remote connection to that session.

## Related topics

[Get-BrokerSession](#)

[Stop-BrokerSession](#)

## Parameters

**-InputObject**<Session[]>

Identifies the session(s) to disconnect. This can be expressed as either a session Uid or a session object.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.Session The sessions to disconnect can be piped into this cmdlet.

## Return Values

None

## Notes

This operation is non-blocking and returns before it completes. The operation, however, is unlikely to fail unless there are communication problems between the controller and the machine, or if bad arguments are passed to the cmdlet itself or if the machine cannot successfully execute the operation.

The transient nature of sessions means that the list of session objects or UIDs supplied to Disconnect-BrokerSession could consist of valid and invalid sessions. Invalid sessions are detected and disregarded and the disconnect session operation is invoked on only the valid sessions.

The system can fail to disconnect the session if the machine is not in an appropriate state or if there are problems in communicating with the machine. When a disconnect is requested the system detects if the operation was initiated successfully or not by the machine. As this operation is non-blocking the system doesn't detect or report whether the disconnect ultimately succeeded or failed after it was started.

Disconnect failures are reported through the broker SDK error handling mechanism (see about\_Broker\_ErrorHandling). In the event of errors the SdkErrorRecord error status is set to SessionOperationFailed and its error data dictionary is populated with the following entries:

- o OperationsAttemptedCount: The number of operations attempted.
- o OperationsFailedCount - The number of failed operations.
- o OperationsSucceededCount - The number of successfully executed operations.
- o UnresolvedSessionFailuresCount - The number of operations that failed due to invalid sessions being supplied.
- o OperationInvocationFailuresCount - The number of operations that failed because they could not be invoked on the machine.
- o DesktopExecutionFailuresCount - The number of operations that failed because they could not be successfully executed

by the machine.

The SdkErrorRecord message will also display the number of attempted, failed and successful operations in the following format:

"Session operation error - attempted:<OperationsAttemptedCount>, failed:<OperationsFailedCount>, succeeded:<OperationsSucceededCount>"

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerSession -UserName MyDomain\MyAccount | Disconnect-BrokerSession
Attempts to disconnect all of the sessions for the user MyDomain\MyAccount.
```

### ----- EXAMPLE 2 -----

```
C:\PS> $desktop = Get-BrokerDesktop -DNSName MyMachine.MyDomain.com
C:\PS> Disconnect-BrokerSession $desktop.SessionUid
Disconnects the session on MyMachine.
```

### ----- EXAMPLE 3 -----

```
C:\PS> trap [Citrix.Broker.Admin.SDK.SdkOperationException]
C:\PS> {
C:\PS> write $("Exception name = " + $_.Exception.GetType().FullName)
C:\PS> write $("SdkOperationException.Status = " + $_.Exception.Status)
C:\PS> write $("SdkOperationException.ErrorData=")
C:\PS> $_.Exception.ErrorData
C:\PS>
C:\PS> write $("SdkOperationException.InnerException = " + $_.Exception.InnerException)
C:\PS> $_.Exception.InnerException
C:\PS> continue
C:\PS> }
C:\PS>
C:\PS> Disconnect-BrokerSession -InputObject 10,11,12
Attempts to disconnect sessions 10, 11 and 12. Traps and displays the error information.
```



# Export-BrokerDesktopPolicy

May 10, 2016

Gets the site wide Citrix Group Policy settings.

## Syntax

```
Export-BrokerDesktopPolicy [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Export-BrokerDesktopPolicy returns an array of bytes containing the site-wide Citrix Group Policy settings. These policy settings are applied to every machine in the site.

## Related topics

[Import-BrokerDesktopPolicy](#)

[New-BrokerConfigurationSlot](#)

[New-BrokerMachineConfiguration](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

None

## Return Values

System.Byte[]

The configuration data as an opaque binary blob. This will be null if no site wide Citrix Group Policy settings are in place.

## Notes

Export-BrokerDesktopPolicy performs a specialized operation. Direct usage of it in scripts is discouraged, and could result in data corruption. It is recommended that this operation only be performed via the Citrix Studio.

## Examples

----- **EXAMPLE 1** -----

```
C:\PS> $policy = Export-BrokerDesktopPolicy
```

This command exports the site wide Citrix Group Policy settings.

# Get-BrokerAccessPolicyRule

May 10, 2016

Gets rules from the site's access policy.

## Syntax

```
Get-BrokerAccessPolicyRule [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerAccessPolicyRule [-Name] <String> [-AllowedConnections <AllowedConnection>] [-AllowedUsers
<AllowedUser>] [-Description <String>] [-DesktopGroupName <String>] [-DesktopGroupUid <Int32>] [-Enabled <Boolean>] [-
ExcludedClientIPFilterEnabled <Boolean>] [-ExcludedClientName <String>] [-ExcludedClientNameFilterEnabled <Boolean>] [-
ExcludedSmartAccessFilterEnabled <Boolean>] [-ExcludedSmartAccessTag <String>] [-ExcludedUser <User>] [-
ExcludedUserFilterEnabled <Boolean>] [-IncludedClientIPFilterEnabled <Boolean>] [-IncludedClientName <String>] [-
IncludedClientNameFilterEnabled <Boolean>] [-IncludedSmartAccessFilterEnabled <Boolean>] [-IncludedSmartAccessTag
<String>] [-IncludedUser <User>] [-IncludedUserFilterEnabled <Boolean>] [-Metadata <String>] [-ReturnTotalRecordCount] [-
MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Returns rules matching the specified search criteria from the site's access policy. If no search criteria are specified, all rules in the access policy are obtained.

An access policy rule defines a set of connection filters and access control rights relating to a desktop group. These allow fine-grained control of what access is granted to a desktop group based on details of, for example, a user's endpoint device, its address, and the user's identity.

----- BrokerAccessPolicyRule Object

A BrokerAccessPolicyRule object represents a single rule within the site's access policy. For a user to gain access to a desktop group via the rule their connection must match all its enabled include filters, and none of its enabled exclude filters. The object contains the following properties:

-- AllowedConnections (Citrix.Broker.Admin.SDK.AllowedConnection)

Controls whether connections must be local or via Access Gateway, and if so whether specified SmartAccess tags must be provided by Access Gateway with the connection. This property forms part of the included SmartAccess tags filter.

For a detailed description of this property see "help about\_Broker\_AccessPolicy".

-- AllowedProtocols (System.String[])

Protocols (for example HDX, RDP) available to the user for sessions delivered from the rule's desktop group. If the user gains access to a desktop group by multiple rules, the allowed protocol list is the combination of the protocol lists from all those rules.

If the protocol list is empty, access to the desktop group is implicitly denied.

-- AllowedUsers (Citrix.Broker.Admin.SDK.AllowedUser)

Controls the behavior of the included users filter. This can restrict access to a list of named users or groups, or allow access to any authenticated user. For a detailed description of this property see "help about\_Broker\_AccessPolicy".

-- AllowRestart (System.Boolean)

Indicates if the user can restart sessions delivered from the rule's desktop group. Session restart is handled as follows: For sessions on single-session power-managed machines, the machine is powered off, and a new session launch request made; for sessions on multi-session machines, a logoff request is issued to the session, and a new session launch request made; otherwise the property is ignored.

-- Description (System.String)

An optional description of the rule. The text is purely informational for the administrator, it is never visible to the end user.

-- DesktopGroupName (System.String)

The name of the desktop group to which the rule applies.

-- DesktopGroupUid (System.Int32)

The unique ID of the desktop group to which the rule applies.

-- Enabled (System.Boolean)

Indicates whether the rule is enabled. A disabled rule is ignored when evaluating the site's access policy.

-- ExcludedClientIPFilterEnabled (System.Boolean)

Indicates whether the excluded client IP filter is enabled. If the filter is disabled it is ignored when the rule is evaluated.

-- ExcludedClientIPs (Citrix.Broker.Admin.SDK.ChbIPAddressRange[])

IP addresses of user devices explicitly denied access to the rule's desktop group. Addresses can be specified as simple numeric addresses or as subnet masks (for example, 10.40.37.5 or 10.40.0.0/16). This property forms part of the excluded client IP address filter.

-- ExcludedClientNameFilterEnabled (System.Boolean)

Indicates whether the excluded client name filter is enabled. If the filter is disabled it is ignored when the rule is evaluated.

-- ExcludedClientNames (System.String[])

Names of user devices explicitly denied access to the rule's desktop group. This property forms part of the excluded client names filter.

-- ExcludedSmartAccessFilterEnabled (System.Boolean)

Indicates whether the excluded SmartAccess tags filter is enabled. If the filter is disabled it is ignored when the rule is evaluated.

-- ExcludedSmartAccessTags (System.String[])

SmartAccess tags which explicitly deny access to the rule's desktop group if any occur in those provided by Access Gateway with the user's connection. This property forms part of the excluded SmartAccess tags filter.

-- ExcludedUserFilterEnabled (System.Boolean)

Indicates whether the excluded users filter is enabled. If the filter is disabled it is ignored when the rule is evaluated.

-- ExcludedUsers (Citrix.Broker.Admin.SDK.ChbUser[])

Users and groups who are explicitly denied access to the rule's desktop group. This property forms part of the excluded users filter.

-- HdxSslEnabled (System.Boolean)

Indicates whether SSL encryption is enabled for sessions delivered from the rule's desktop group.

-- IncludedClientIPFilterEnabled (System.Boolean)

Indicates whether the included client IP filter is enabled. If the filter is disabled it is ignored when the rule is evaluated.

-- IncludedClientIPs (Citrix.Broker.Admin.SDK.ChbIPAddressRange[])

IP addresses of user devices allowed access to the rule's desktop group. Addresses can be specified as simple numeric addresses or as subnet masks (for example, 10.40.37.5 or 10.40.0.0/16). This property forms part of the included client IP address filter.

-- IncludedClientNameFilterEnabled (System.Boolean)

Indicates whether the included client names filter is enabled. If the filter is disabled it is ignored when the rule is evaluated.

-- IncludedClientNames (System.String[])

Names of user devices allowed access to the rule's desktop group. This property forms part of the included client names filter.

-- IncludedSmartAccessFilterEnabled (System.Boolean)

Indicates whether the included SmartAccess tags filter is enabled. If the filter is disabled it is ignored when the rule is evaluated.

-- IncludedSmartAccessTags (System.String[])

The SmartAccess tags which grant access to the rule's desktop group if any occur in those provided by Access Gateway with the user's connection. This property forms part of the excluded SmartAccess tags filter.

-- IncludedUserFilterEnabled (System.Boolean)

Indicates whether the included users filter is enabled. If the filter is disabled it is ignored when the rule is evaluated.

-- IncludedUsers (Citrix.Broker.Admin.SDK.ChbUser[])

Users and groups who are granted access to the rule's desktop group. This property forms part of the included users filter.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

A collection of arbitrary key/value pairs that can be associated with the rule. The administrator can use these values for any purpose; they are not used by the site itself in any way.

-- Name (System.String)

Administrative name of the rule. Each rule in the site's access policy must have a unique name.

-- Uid (System.Int32)

Unique ID of the rule itself.

## Related topics

[New-BrokerAccessPolicyRule](#)

[Set-BrokerAccessPolicyRule](#)

[Rename-BrokerAccessPolicyRule](#)

[Remove-BrokerAccessPolicyRule](#)

## Parameters

**-Uid**<Int32>

Gets only the rule with the specified unique ID.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Name**<String>

Gets only rules with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AllowedConnections**<AllowedConnection>

Gets only rules that have the specified value in the AllowedConnections property of their included SmartAccess tags filter.

Valid values are Filtered, NotViaAG, and ViaAG.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AllowedUsers**<AllowedUser>

Gets only rules that have the specified value in the AllowedUsers property of their included users filter.

Valid values are Filtered, AnyAuthenticated, Any, AnonymousOnly and FilteredOrAnonymous.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Description**<String>

Gets only rules with the specified description.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupName**<String>

Gets only rules applying to desktop groups with names matching the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupUid**<Int32>

Gets only rules that apply to the desktop group with the specified unique ID.

Required?	false

Default Value	
Accept Pipeline Input?	false

#### **-Enabled**<Boolean>

Gets only rules that are in the specified state, either enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedClientIPFilterEnabled**<Boolean>

Gets only rules that have their excluded client IP address filter enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedClientName**<String>

Gets only rules that have the specified client name in their excluded client names filter (whether the filter is enabled or not).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedClientNameFilterEnabled**<Boolean>

Gets only rules that have their excluded client name filter enabled (\$true) or disabled (\$false).

Required?	false
Default Value	



Accept Pipeline Input?	false
------------------------	-------

#### **-ExcludedSmartAccessFilterEnabled**<Boolean>

Gets only rules that have their excluded SmartAccess tags filter enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedSmartAccessTag**<String>

Gets only rules that have the specified SmartAccess tag in their excluded SmartAccess tags filter (whether the filter is enabled or not).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedUser**<User>

Gets only rules that have the specified user in their excluded users filter (whether the filter is enabled or not).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedUserFilterEnabled**<Boolean>

Gets only rules that have their excluded user filter enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

--	--

#### **-IncludedClientIPFilterEnabled**<Boolean>

Gets only rules that have their included client IP address filter enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedClientName**<String>

Gets only rules that have the specified user device name in their included client names filter (whether the filter is enabled or not).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedClientNameFilterEnabled**<Boolean>

Gets only rules that have their included client name filter enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedSmartAccessFilterEnabled**<Boolean>

Gets only rules that have their included SmartAccess tags filter enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-IncludedSmartAccessTag**<String>

Gets only rules that have the specified SmartAccess tag in their included SmartAccess tags filter (whether the filter is enabled or not).

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-IncludedUser**<User>

Gets only rules that have the specified user in their included users filter (whether the filter is enabled or not).

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-IncludedUserFilterEnabled**<Boolean>

Gets only rules that have their included user filter enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
-----------	-------

Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.AccessPolicyRule

Get-BrokerAccessPolicyRule returns all access policy rules that match the specified selection criteria.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerAccessPolicyRule
```

Returns all access policy rules. This offers a complete description of the current site's access policy.

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerAccessPolicyRule -Enabled $true -IncludedUser sales\tech-support
```

Returns all rules that are both enabled and explicitly include the SALES\tech-support group in their included users filter.

# Get-BrokerAdminFolder

May 10, 2016

Get the admin folders in this site.

## Syntax

```
Get-BrokerAdminFolder [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerAdminFolder [[-Name] <String>] [-DirectChildAdminFolders <Int32>] [-DirectChildApplications <Int32>] [-FolderName <String>] [-LastChangeId <Guid>] [-Metadata <String>] [-ParentAdminFolderUid <Int32>] [-TotalChildApplications <Int32>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Get-BrokerAdminFolder cmdlet gets admin folders in this site.

Without parameters, Get-BrokerAdminFolder gets all the admin folders that have been created. You can also use the parameters of Get-BrokerAdminFolder to filter the results to just the folders you're interested in. You can also identify folders by their UIDs or their FolderNames.

----- BrokerAdminFolder Object

A folder for use in the administration console for organising other objects. E.g. BrokerApplication objects

-- DirectChildAdminFolders (System.Int32)

The number of admin folders with this folder as a direct parent

-- DirectChildApplications (System.Int32)

The number of applications in this admin folder (does not include any applications in child folders)

-- FolderName (System.String)

The simple name of this folder within any parent folder

-- LastChangeId (System.Guid)

A random GUID assigned whenever there is a change anywhere in the hierarchy of admin folders below this node; each change updates this value on the changed folder and all parents all the way up to the root folder. Note that nodes below any change do not have their LastChangeId value updated

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

Holds any metadata associated with the admin folder

-- Name (System.String)

The full name of the folder including the full parent hierarchy separated by back-slash characters and including a trailing back-slash

-- ParentAdminFolderUid (System.Int32)

The UID of the parent admin folder; the root folder references itself (zero)

-- TotalChildApplications (System.Int32)

The number of applications in this admin folder (including any applications in child folders)

-- Uid (System.Int32)

The unique ID of the admin folder (the root folder has the value zero)

## Related topics

[New-BrokerAdminFolder](#)

[Remove-BrokerAdminFolder](#)

## Parameters

**-Uid**<Int32>

Gets only the admin folder with the specified unique identifier.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Name**<String>

Gets admin folders matching the specified name (if no trailing backslash is supplied, it is assumed).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DirectChildAdminFolders**<Int32>

Gets admin folders with the specified number of child folders.

Required?	false
Default Value	



Accept Pipeline Input?	false
------------------------	-------

#### **-DirectChildApplications<Int32>**

Gets admin folders with the specified number of applications (excluding those in sub-folders).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-FolderName<String>**

Gets only the admin folders matching the specified simple folder name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastChangeId<Guid>**

Gets only the admin folders with the specified value for LastChangeId.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Metadata<String>**

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ParentAdminFolderUid<Int32>**

Gets only admin folders with the specified parent admin folder UID value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-TotalChildApplications<Int32>**

Gets admin folders with the specified number of applications (including those in sub-folders).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount<SwitchParameter>**

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount<Int32>**

Specifies the maximum number of records to return.

Required?	false
Default Value	250

Accept Pipeline Input?	false
------------------------	-------

#### **-Skip<Int32>**

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy<String>**

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter<String>**

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

None Input cannot be piped to this cmdlet.

### Return Values

Citrix.Broker.Admin.SDK.AdminFolder

Returns admin folders.

### Examples

#### ----- **EXAMPLE 1** -----

C:\PS> Get-BrokerAdminFolder  
Return all administration folders.

#### ----- **EXAMPLE 2** -----

C:\PS> Get-BrokerAdminFolder -Uid 1  
Get the administration folder with Uid 1.

# Get-BrokerAppAssignmentPolicyRule

May 10, 2016

Gets application rules from the site's assignment policy.

## Syntax

```
Get-BrokerAppAssignmentPolicyRule [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerAppAssignmentPolicyRule [[-Name] <String>] [-Description <String>] [-DesktopGroupUid <Int32>] [-Enabled
<Boolean>] [-ExcludedUser <User>] [-ExcludedUserFilterEnabled <Boolean>] [-IncludedUser <User>] [-
IncludedUserFilterEnabled <Boolean>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy
<String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns application rules matching the specified search criteria from the site's assignment policy. If no search criteria are specified, all application rules in the assignment policy are obtained.

An application rule in the assignment policy defines the users who are entitled to a self-service persistent machine assignment from the rule's desktop group; once assigned the machine can run one or more applications published from the group.

----- BrokerAppAssignmentPolicyRule Object

The BrokerAppAssignmentPolicyRule object represents a single application rule within the site's assignment policy. It contains the following properties:

-- Description (System.String)

An optional description of the rule. The text is purely informational for the administrator, it is never visible to the end user.

-- DesktopGroupUid (System.Int32)

The unique ID of the desktop group to which the rule applies.

-- Enabled (System.Boolean)

Indicates whether the rule is enabled. A disabled rule is ignored when evaluating the site's assignment policy.

-- ExcludedUserFilterEnabled (System.Boolean)

Indicates whether the excluded users filter is enabled. If the filter is disabled then any user entries in the filter are ignored when assignment policy rules are evaluated.

-- ExcludedUsers (Citrix.Broker.Admin.SDK.ChbUser[])

The excluded users filter of the rule, that is, the users and groups who are explicitly denied an entitlement to a machine assignment from the rule's desktop group.

-- IncludedUserFilterEnabled (System.Boolean)

Indicates whether the included users filter is enabled. If the filter is disabled then any user who satisfies the requirements of

the access policy is implicitly entitled to the machine assignment described by the rule.

-- IncludedUsers (Citrix.Broker.Admin.SDK.ChbUser[])

The included users filter of the rule, that is, the users and groups who are entitled to a machine assignment from the rule's desktop group.

-- Name (System.String)

The administrative name of the rule. Each rule in the site's assignment policy must have a unique name (irrespective of whether they are desktop or application rules).

-- Uid (System.Int32)

The unique ID of the rule itself.

## Related topics

[New-BrokerAppAssignmentPolicyRule](#)

[Set-BrokerAppAssignmentPolicyRule](#)

[Rename-BrokerAppAssignmentPolicyRule](#)

[Remove-BrokerAppAssignmentPolicyRule](#)

## Parameters

**-Uid**<Int32>

Gets the application rule with the specified unique ID.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Name**<String>

Gets only application rules with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Description**<String>

Gets only application rules with the specified description.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopGroupUid**<Int32>

Gets only application rules that apply to the desktop group with the specified unique ID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Enabled**<Boolean>

Gets only application rules that are in the specified state, either enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ExcludedUser**<User>

Gets only application rules that have the specified user in their excluded users filter (whether the filter is enabled or not)

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ExcludedUserFilterEnabled**<Boolean>

Gets only application rules that have their excluded user filter enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedUser**<User>

Gets only application rules that have the specified user in their included users filter (whether the filter is enabled or not).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedUserFilterEnabled**<Boolean>

Gets only application rules that have their included user filter enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.



Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip<Int32>**

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy<String>**

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter<String>**

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.AppAssignmentPolicyRule

Get-BrokerAppAssignmentPolicyRule returns all application rules in the assignment policy that match the specified selection criteria.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerAppAssignmentPolicyRule
```

Returns all application rules from the assignment policy. This offers a complete description of the current site's assignment policy with respect to machine assignment entitlements for delivery of application sessions from private desktop groups.

##### ----- EXAMPLE 2 -----

```
C:\PS> $dg = Get-BrokerDesktopGroup 'Sales Support'
```

```
C:\PS> Get-BrokerAppAssignmentPolicyRule -DesktopGroupUid $dg.Uid
```

Returns the rule in the assignment policy that gives users entitlements to machine assignments in the Sales Support desktop group for delivery of application sessions.

# Get-BrokerAppEntitlementPolicyRule

May 10, 2016

Gets application rules from the site's entitlement policy.

## Syntax

```
Get-BrokerAppEntitlementPolicyRule [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerAppEntitlementPolicyRule [[-Name] <String>] [-Description <String>] [-DesktopGroupUid <Int32>] [-Enabled
<Boolean>] [-ExcludedUser <User>] [-ExcludedUserFilterEnabled <Boolean>] [-IncludedUser <User>] [-
IncludedUserFilterEnabled <Boolean>] [-SessionReconnection <SessionReconnection>] [-ReturnTotalRecordCount] [-
MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Returns application rules matching the specified search criteria from the site's entitlement policy. If no search criteria are specified, all application rules in the entitlement policy are obtained.

An application rule in the entitlement policy defines the users who are allowed per-session access to a machine to run one or more applications published from the rule's desktop group.

----- BrokerAppEntitlementPolicyRule Object

The BrokerAppEntitlementPolicyRule object represents a single application rule within the site's entitlement policy. It contains the following properties:

-- Description (System.String)

Optional description of the rule. The text is purely informational for the administrator, it is never visible to the end user.

-- DesktopGroupUid (System.Int32)

The unique ID of the desktop group to which the rule applies.

-- Enabled (System.Boolean)

Indicates whether the rule is enabled. A disabled rule is ignored when evaluating the site's entitlement policy.

-- ExcludedUserFilterEnabled (System.Boolean)

Indicates whether the excluded users filter of the rule is enabled. If the filter is disabled then any user entries in the filter are ignored when entitlement policy rules are evaluated.

-- ExcludedUsers (Citrix.Broker.Admin.SDK.ChbUser[])

The excluded users filter of the rule, that is, the users and groups who are explicitly denied entitlements to use published applications from the associated desktop group.

-- IncludedUserFilterEnabled (System.Boolean)

Indicates whether the included users filter of the rule is enabled. If the filter is disabled then any user who satisfies the

requirements of the access policy is implicitly granted an entitlement to an application session by the rule.

-- IncludedUsers (Citrix.Broker.Admin.SDK.ChbUser[])

The included users filter of the rule, that is, the users and groups who are granted an entitlement to an application session by the rule.

If a user appears explicitly in the excluded users filter of the rule or is a member of a group that appears in the excluded users filter, no entitlement is granted whether or not the user appears in the included users filter.

-- Name (System.String)

The administrative name of the rule. Each rule in the site's entitlement policy must have a unique name (irrespective of whether they are desktop or application rules).

-- SessionReconnection (Citrix.Broker.Admin.SDK.SessionReconnection)

Defines reconnection (roaming) behavior for sessions launched using this rule. Session reconnection control is an experimental and unsupported feature.

-- Uid (System.Int32)

The unique ID of the rule itself.

## Related topics

[New-BrokerAppEntitlementPolicyRule](#)

[Set-BrokerAppEntitlementPolicyRule](#)

[Rename-BrokerAppEntitlementPolicyRule](#)

[Remove-BrokerAppEntitlementPolicyRule](#)

## Parameters

**-Uid**<Int32>

Gets the application rule with the specified unique ID.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Name**<String>

Gets only application rules with the specified name.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-Description<String>**

Gets only application rules with the specified description.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupUid<Int32>**

Gets only the application rule that applies to the desktop group with the specified unique ID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Enabled<Boolean>**

Gets only application rules that are in the specified state, either enabled (\$true), or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedUser<User>**

Gets only application rules that have the specified user in their excluded users filter (whether the filter is enabled or not).

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ExcludedUserFilterEnabled**<Boolean>

Gets only application rules that have their excluded user filter enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedUser**<User>

Gets only application rules that have the specified user in their included users filter (whether the filter is enabled or not).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedUserFilterEnabled**<Boolean>

Gets only application rules that have their included user filter enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionReconnection**<SessionReconnection>

Gets only application rules with the specified session reconnection behavior.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
-----------	-------

Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.AppEntitlementPolicyRule



Get-BrokerAppEntitlementPolicyRule returns all application rules from the entitlement policy that match the specified selection criteria.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerAppEntitlementPolicyRule
```

Returns all application rules from the entitlement policy. This offers a complete description of the current site's entitlement policy with respect to application entitlements from shared desktop groups.

### ----- EXAMPLE 2 -----

```
C:\PS> $dg = Get-BrokerDesktopGroup 'Customer Support'
```

```
C:\PS> Get-BrokerAppEntitlementPolicyRule -DesktopGroupUid $dg.Uid
```

Returns the application rule in the entitlement policy that grants users an entitlement to application sessions in the Customer Support desktop group.

# Get-BrokerApplication

May 10, 2016

Get the applications published on this site.

## Syntax

```
Get-BrokerApplication [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerApplication [-Name] <String> [-AdminFolderName <String>] [-AdminFolderUid <Int32>] [-ApplicationName
<String>] [-ApplicationType <ApplicationType>] [-AssociatedDesktopGroupPriority <Int32>] [-AssociatedDesktopGroupUid
<Int32>] [-AssociatedDesktopGroupUUID <Guid>] [-AssociatedUserFullName <String>] [-AssociatedUserName <String>] [-
AssociatedUserUPN <String>] [-BrowserName <String>] [-ClientFolder <String>] [-CommandLineArguments <String>] [-
CommandLineExecutable <String>] [-CpuPriorityLevel <CpuPriorityLevel>] [-Description <String>] [-Enabled <Boolean>] [-
IconFromClient <Boolean>] [-IconUid <Int32>] [-MetadataKey <String>] [-Metadata <String>] [-PublishedName <String>] [-
SecureCmdLineArgumentsEnabled <Boolean>] [-ShortcutAddedToDesktop <Boolean>] [-ShortcutAddedToStartMenu
<Boolean>] [-StartMenuFolder <String>] [-UserFilterEnabled <Boolean>] [-UUID <Guid>] [-Visible <Boolean>] [-
WaitForPrinterCreation <Boolean>] [-WorkingDirectory <String>] [-DesktopUid <Int32>] [-SessionUid <Int64>] [-UserSID
<String>] [-DesktopGroupUid <Int32>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy
<String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Get-BrokerApplication cmdlet gets the published applications that are hosted on this site.

Without parameters, Get-BrokerApplication gets all the applications that have been published, regardless of whether they are visible to users or not. You can also use the parameters of Get-BrokerApplication to filter the results to just the applications you're interested in. You can also identify applications by their UIDs or their BrowserNames.

For more information about applications, see about\_Broker\_Applications.

----- BrokerApplication Object

The BrokerApplication object represents a published application in the site. It contains the following properties:

-- AdminFolderName (System.String)

The name of the admin folder the application is in (including trailing backslash), or the empty string if the application is at the root level

-- AdminFolderUid (System.Int32)

The Uid of the admin folder the application is in (if any)

-- ApplicationName (System.String)

The simple name of the application within its parent admin folder (if any)

-- ApplicationType (Citrix.Broker.Admin.SDK.ApplicationType)

The type of the application, whether HostedOnDesktop or InstalledOnClient.

-- AssociatedDesktopGroupPriorities (System.Int32[])

List of associated desktop group priorities. Associated desktop groups is the list of desktop groups on which the application is published. When launching an application an available machine from one of the associated groups is selected. Desktop groups are searched for available machines in order of their priority.

-- AssociatedDesktopGroupUids (System.Int32[])

List of associated desktop group uids. Associated desktop groups is the list of desktop groups on which the application is published. The list is sorted by priority, with the highest priority group first.

-- AssociatedDesktopGroupUUIDs (System.Guid[])

List of associated desktop group UUIDs. Associated desktop groups is the list of desktop groups on which the application is published. The list is sorted by priority, with the highest priority group first.

-- AssociatedUserFullNames (System.String[])

List of associated users (full names). Associated users is the list of users who are given access using the application/user mapping filter.

-- AssociatedUserNames (System.String[])

List of associated users (SAM names). Associated users is the list of users who are given access using the application/user mapping filter.

-- AssociatedUserUPNs (System.String[])

List of associated users (user principle names). Associated users is the list of users who are given access using the application/user mapping filter.

-- BrowserName (System.String)

Unique browser name used to identify this application to other components in the site. This value is not visible to the end users.

-- ClientFolder (System.String)

The folder that the application belongs to as the user sees it.

-- CommandLineArguments (System.String)

The command-line arguments to use when launching the executable.

-- CommandLineExecutable (System.String)

The name including the full path of the executable file to launch.

-- CpuPriorityLevel (Citrix.Broker.Admin.SDK.CpuPriorityLevel)

The CPU priority of the launched process. Valid values are: Low, BelowNormal, Normal, AboveNormal, and High.

-- Description (System.String)

Optional application description. This description is visible to the end users.

-- Enabled (System.Boolean)

Specifies whether or not this application can be launched.

-- IconFromClient (System.Boolean)

Specifies if the app icon should be retrieved from the application on the client. This is reserved for possible future use, and all applications of type HostedOnDesktop cannot set or change this value.

-- IconUid (System.Int32?)

The icon UID used for this application. If not specified a generic icon is used.

-- MetadataKeys (System.String[])

All key names of metadata items associated with this application.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

Metadata for this application.

-- Name (System.String)

Unique administrative name of application; this will include any parent admin folder hierarchy separated by backslash characters.

-- PublishedName (System.String)

Published name of application as seen by end user. If not specified value used defaults to the administrative name.

-- SecureCmdLineArgumentsEnabled (System.Boolean)

Specifies whether the command-line arguments should be secured.

-- ShortcutAddedToDesktop (System.Boolean)

Specifies whether a shortcut to the application should be placed on the user device.

-- ShortcutAddedToStartMenu (System.Boolean)

Specifies whether a shortcut to the application should be placed in the user's Start menu on their user device.

-- StartMenuFolder (System.String)

The name of the Start menu folder that holds the application shortcut.

-- Uid (System.Int32)

A unique identifier of an application.

-- UserFilterEnabled (System.Boolean)

Indicates if application-specific user filter is enabled.

-- UUID (System.Guid)

UUID of the application.

-- Visible (System.Boolean)

Specifies if the application is visible to users.

-- WaitForPrinterCreation (System.Boolean)

Specifies whether the VDA delays starting the app until printers are set up or not.

-- WorkingDirectory (System.String)

The working directory the executable is launched from.

## Related topics

[New-BrokerApplication](#)

[Add-BrokerApplication](#)

[Remove-BrokerApplication](#)

[Rename-BrokerApplication](#)

[Move-BrokerApplication](#)

[Set-BrokerApplication](#)

## Parameters

**-Uid**<Int32>

Gets only the application with the specified unique identifier.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Name**<String>

Gets only the applications matching the specified name (including any parent admin folder hierarchy).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminFolderName<String>**

Gets applications that are in admin folders matching the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminFolderUid<Int32>**

Gets applications that are in the specified admin folder.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ApplicationName<String>**

Gets applications that match the specified simple name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ApplicationType<ApplicationType>**

Gets applications that match the type specified: HostedOnDesktop or InstalledOnClient.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedDesktopGroupPriority<Int32>**

Gets applications with an associated desktop group identified by priority assigned to the pairing between an application and desktop group.

Associated desktop group is a desktop group on which the application is published.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedDesktopGroupUid<Int32>**

Gets applications with an associated desktop group identified by the desktop group UID.

Associated desktop group is a desktop group on which the application is published.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedDesktopGroupUUID<Guid>**

Gets applications with an associated desktop group identified by the desktop group UUID.

Associated desktop group is a desktop group on which the application is published.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserFullName<String>**

Gets applications with an associated user identified by their full name (usually 'first-name last-name').

If the 'UserFilterEnabled' property is true then access to the application is restricted to those users only, otherwise access is unrestricted (but always subject to other policy rules).

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserName<String>**

Gets applications with an associated user identified by their user name (in the form 'domain\user'). If the 'UserFilterEnabled' property is true then access to the application is restricted to those users only, otherwise access is unrestricted (but always subject to other policy rules).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserUPN<String>**

Gets applications with an associated user identified by their user principle name (in the form 'user@domain'). If the 'UserFilterEnabled' property is true then access to the application is restricted to those users only, otherwise access is unrestricted (but always subject to other policy rules).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-BrowserName<String>**

Gets only the applications that match the supplied name. The BrowserName is usually an internal name for the application and is unique in the site.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Client Folder<String>**

Gets only the applications that match the specified value for the folder the application belongs to as seen by the end-user.



This folder can be seen in the Citrix Online Plug-in, in Web Services, and also potentially in the user's start menu.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-CommandLineArguments<String>**

Gets only the applications that match the supplied arguments to the command-line executable.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-CommandLineExecutable<String>**

Gets only the applications that match the supplied command-line executable.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-CpuPriorityLevel<CpuPriorityLevel>**

Gets only the applications that have the specified value for the CPU priority level of the launched executable. Valid values are: Low, BelowNormal, Normal, AboveNormal, and High.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Description<String>**

Gets only the applications that match the supplied description.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Enabled**<Boolean>

Gets only the applications that have the specified value for whether the application is enabled. Disabled applications are still visible to users (that is controlled by the Visible setting) but cannot be launched.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IconFromClient**<Boolean>

Gets only the applications that have the specified value for whether the application icon should be retrieved from the user device.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IconUid**<Int32>

Gets only the applications that use the specified icon (identified by its Uid).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MetadataKey**<String>

Gets only applications whose associated metadata contains key names matching the specified value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Metadata<String>**

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PublishedName<String>**

Gets applications whose published name matches the supplied pattern.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecureCmdLineArgumentsEnabled<Boolean>**

Gets only the applications that have the specified value for whether the command-line arguments should be secured. This is reserved for possible future use, and all applications of type HostedOnDesktop can only have this value set to true.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ShortcutAddedToDesktop**<Boolean>

Gets only the applications that match depending on whether a shortcut for the application has been added to the user device or not.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ShortcutAddedToStart Menu**<Boolean>

Gets only the applications that match depending on whether a shortcut for the application has been added to Start Menu of the user device or not.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-StartMenuFolder**<String>

Gets only the applications that match the specified name for the start menu folder that holds the application shortcut. This is valid only for the Citrix Online Plug-in.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-UserFilterEnabled**<Boolean>

Gets only applications whose user filter is in the specified state.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UUID**<Guid>

Gets applications with the specified value of UUID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Visible**<Boolean>

Gets only the applications that have the specified value for whether it is visible to the users.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-WaitForPrinterCreation**<Boolean>

Gets only the applications that match depending on whether the VDA delays starting the application until printers are set up.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-WorkingDirectory**<String>

Gets only the applications that match the specified working directory.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopUid<Int32>**

Gets only the applications that have been associated (using a desktop group) to the specified desktop (identified by its Uid). Note that an application is not directly associated with a desktop, but only indirectly by which desktop group it has been published to.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionUid<Int64>**

Gets only the applications that are running in the specified session (identified by its Uid).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UserSID<String>**

Gets only applications with their accessibility restricted to include the specified user.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupUid<Int32>**

Gets only the applications that have been published to the specified desktop group (identified by its Uid).

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.

Accept Pipeline Input?	false
------------------------	-------

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.Application

Get-BrokerApplication returns an object for each application it gets.



## Notes

Get-BrokerApplication returns just the application object, and as such is not a complete picture. The returned objects do not tell you what File-Type Associations are configured for this application, etc.

Use the following cmdlets to gather data related to applications (shown with examples of syntax):

```
Get-BrokerConfiguredFTA -ApplicationUid $app.Uid
```

```
Get-BrokerTag -ApplicationUid $app.Uid
```

```
Get-BrokerDesktopGroup -ApplicationUid $app.Uid
```

```
Get-BrokerDesktop -PublishedApplication $app
```

```
Get-BrokerSession -ApplicationUid $app.Uid
```

```
Get-BrokerApplicationInstance -ApplicationUid $app.Uid
```

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerApplication Notepad
```

Returns the application with the Name of "Notepad".

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerApplication -PublishedName Note* -Enabled $true
```

Returns the applications that have a PublishedName starting with "Note" and that are enabled.

# Get-BrokerApplicationInstance

May 10, 2016

Gets the running applications on the desktops.

## Syntax

```
Get-BrokerApplicationInstance -Uid <Int64> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerApplicationInstance [-ApplicationName <String>] [-ApplicationUid <Int32>] [-ApplicationUUID <Guid>] [-Instances
<Int32>] [-MachineName <String>] [-MachineUid <Int32>] [-Metadata <String>] [-SessionKey <Guid>] [-SessionUid <Int64>] [-
UserName <String>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter
<String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Get-BrokerApplicationInstance gets the published applications that are running on desktops.

Only published applications that are launched from a Citrix client are returned. If a user launches an application from within a session (by double-clicking on an attachment from an email, for example) this will not show up in the list of running applications.

Also note that this is a list of launched published applications, not a list of processes running on the desktop. In some cases the original process associated with the published application may no longer be running, but if the session is still running the published application may be listed as running.

The number of instances for each published application running in a session is also returned. For example, if a user launches two Notepad applications from a Citrix client, and session-sharing occurs such that both Notepad applications run in the same session, then the Instances property indicates that 2 copies are running in the session.

See Get-BrokerApplication and Get-BrokerSession to get the details for the applications and sessions, respectively.

The Get-BrokerMachine cmdlet also returns a list of published applications that are running on a desktop. See the "ApplicationsInUse" attribute of the returned desktop objects.

----- BrokerApplicationInstance Object

The BrokerApplicationInstance object represents an instance of a published application in the site. It contains the following properties:

-- ApplicationName (System.String)

The administrative name of the application.

-- ApplicationUid (System.Int32)

The UID of the application.

-- ApplicationUUID (System.Guid)

The UUID of the application.

-- Instances (System.Int32)

The number of times this published application is running in the specified session.

-- MachineName (System.String)

Machine's SAM name (of the form domain\machine). If SAM name is unavailable, contains the machines's SID.

-- MachineUid (System.Int32)

UID of underlying machine.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

Metadata for this application instance.

-- SessionKey (System.Guid)

The key of the session.

-- SessionUid (System.Int64)

The UID of the session.

-- Uid (System.Int64)

The unique identifier for this application instance object itself, distinct from the Uids of either application or session objects.

-- UserName (System.String)

User name (SAMName).

## Related topics

[Get-BrokerApplication](#)

[Get-BrokerDesktop](#)

[Get-BrokerSession](#)

## Parameters

**-Uid**<Int64>

Gets only the application instances specified by the unique identifier. This is the unique identifier for the application instance object itself, and is distinct from the Uids of either application or session objects.

Required?	true
Default Value	
Accept Pipeline Input?	false

---

**-ApplicationName<String>**

Gets only application instances for the specified application name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ApplicationUid<Int32>**

Gets only application instances for the specified application Uid.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ApplicationUUID<Guid>**

Gets only the application instances for the specified application UUID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Instances<Int32>**

Gets only application instances that match the specified number of instances.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-MachineName<String>**

Gets only application instances running on the specified machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-MachineUid**<Int32>

Gets only application instances running on the machine with the specified UID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

**-SessionKey**<Guid>

Gets only application instances for the published applications running in the specified session.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-SessionUid<Int64>**

Gets only application instances for the published applications running in the specified session.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-UserName<String>**

Gets only application instances being run by the specified users.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount<SwitchParameter>**

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-MaxRecordCount<Int32>**

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

### **-Skip<Int32>**

Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy<String>**

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter<String>**

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	

Accept Pipeline Input?	false
------------------------	-------

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.ApplicationInstance

Get-BrokerApplicationInstance returns an object for each application instance it gets.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Get-BrokerApplicationInstance -Uid 3
```

Returns the application instance with a Uid of 3. Note that this is the unique identifier for application instances, which is distinct from the unique identifiers of either application or session objects.

##### ----- **EXAMPLE 2** -----

```
C:\PS> $app = Get-BrokerApplication Notepad
```

```
C:\PS> Get-BrokerApplicationInstance -ApplicationUid $app.Uid
```

Returns all the application instances for the Notepad application. Use this to see if there are any launched instances of Notepad running in your site and, if so, from which desktops.

##### ----- **EXAMPLE 3** -----

```
C:\PS> $sessions = Get-BrokerSession -MachineName "ACME\Worker1"
```

```
C:\PS> for ($i=0; $i -lt $sessions.Length; $i++) {
 Get-BrokerApplicationInstance -SessionUid $sessions[$i].SessionUid
}
```

Returns all the applications that are running on the "Worker1" machine in the "ACME" domain. Use this to see which published applications are running on a specific machine.

Note that the SessionUid, not the SessionId, is specified as a parameter to this cmdlet. The SessionId is a unique identifier that Remote Desktop Services uses to track the session, and is unique only on that machine. The SessionUid, on the other hand, is unique across the entire site.



The "ApplicationsInUse" attribute of the returned session object also provides a list of running launched applications, and in many cases might be more convenient to use. It returns a list of application BrowserNames.

# Get-BrokerAssignmentPolicyRule

May 10, 2016

Gets desktop rules from the site's assignment policy.

## Syntax

```
Get-BrokerAssignmentPolicyRule [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerAssignmentPolicyRule [[-Name] <String>] [-ColorDepth <ColorDepth>] [-Description <String>] [-DesktopGroupUid <Int32>] [-Enabled <Boolean>] [-ExcludedUser <User>] [-ExcludedUserFilterEnabled <Boolean>] [-IconUid <Int32>] [-IncludedUser <User>] [-IncludedUserFilterEnabled <Boolean>] [-MaxDesktops <Int32>] [-Metadata <String>] [-PublishedName <String>] [-SecureIcaRequired <Boolean>] [-UUID <Guid>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns desktop rules matching the specified search criteria from the site's assignment policy. If no search criteria are specified, all desktop rules in the assignment policy are obtained.

A desktop rule in the assignment policy defines the users who are entitled to self-service persistent machine assignments from the rule's desktop group. A rule defines how many machines a user is allowed from the group for delivery of full desktop sessions.

----- BrokerAssignmentPolicyRule Object

The BrokerAssignmentPolicyRule object represents a single desktop rule within the site's assignment policy. It contains the following properties:

-- ColorDepth (Citrix.Broker.Admin.SDK.ColorDepth?)

The color depth of desktop sessions launched by the user from machines assigned to them by the rule. If null, the equivalent setting from the rule's desktop group is used.

-- Description (System.String)

Optional description of the rule. The text may be visible to the end user, for example, as a tooltip associated with the desktop entitlement.

-- DesktopGroupUid (System.Int32)

The unique ID of the desktop group to which the rule applies.

-- Enabled (System.Boolean)

Indicates whether the rule is enabled. A disabled rule is ignored when evaluating the site's assignment policy.

-- ExcludedUserFilterEnabled (System.Boolean)

Indicates whether the excluded users filter is enabled. If the filter is disabled then any user entries in the filter are ignored when assignment policy rules are evaluated.

-- ExcludedUsers (Citrix.Broker.Admin.SDK.ChbUser[])

The excluded users filter of the rule, that is, the users and groups who are explicitly denied entitlements to machine assignments from the rule's desktop group.

-- IconUid (System.Int32?)

The unique ID of the icon used for the machine entitlement seen by the user or, after a machine is assigned by the rule, the icon for the desktop itself. If null, the equivalent setting from the rule's desktop group is used.

-- IncludedUserFilterEnabled (System.Boolean)

Indicates whether the included users filter is enabled. If the filter is disabled then any user who satisfies the requirements of the access policy is implicitly entitled to the machine assignments described by the rule.

For rules that relate to RemotePC desktop groups however, if the included user filter is disabled, the rule is effectively disabled.

-- IncludedUsers (Citrix.Broker.Admin.SDK.ChbUser[])

The included users filter of the rule, that is, the users and groups who are entitled to machine assignments from the rule's desktop group.

-- MaxDesktops (System.Int32)

The number of machines from the rule's desktop group to which a user is entitled. Where an entitlement is granted to a user group rather than an individual, the number of machines applies to each member of the user group independently.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

A collection of arbitrary key/value pairs that can be associated with the rule. The administrator can use these values for any purpose; they are not used by the site itself in any way.

-- Name (System.String)

The administrative name of the rule. Each rule in the site's assignment policy must have a unique name (irrespective of whether they are desktop or application rules).

-- PublishedName (System.String)

The published name of the desktop entitlement seen by the user or, after a machine is assigned by the rule, the published name of the desktop itself. If null, the equivalent setting from the rule's desktop group is used.

-- SecureIcaRequired (System.Boolean?)

Indicates whether the rule requires the SecureICA protocol for desktop sessions launched using a machine assigned by the rule. If null, the equivalent setting from the rule's desktop group is used.

-- Uid (System.Int32)

The unique ID of the rule itself.

-- UUID (System.Guid)

UUID of the rule.

## Related topics

[New-BrokerAssignmentPolicyRule](#)

[Set-BrokerAssignmentPolicyRule](#)

[Rename-BrokerAssignmentPolicyRule](#)

[Remove-BrokerAssignmentPolicyRule](#)

## Parameters

**-Uid**<Int32>

Gets the desktop rule with the specified unique ID.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Name**<String>

Gets only desktop rules with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ColorDepth**<ColorDepth>

Gets only desktop rules with the specified color depth.

Valid values are \$null, FourBit, EightBit, SixteenBit, and TwentyFourBit.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Description<String>**

Gets only desktop rules with the specified description.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopGroupUid<Int32>**

Gets only desktop rules that apply to the desktop group with the specified unique ID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Enabled<Boolean>**

Gets only rules that are in the specified state, either enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ExcludedUser<User>**

Gets only desktop rules that have the specified user in their excluded users filter (whether the filter is enabled or not).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ExcludedUserFilterEnabled<Boolean>**

Gets only desktop rules that have their excluded user filter enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IconUid**<Int32>

Gets only desktop rules using the icon with the specified unique ID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IncludedUser**<User>

Gets only desktop rules that have the specified user in their included users filter (whether the filter is enabled or not).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IncludedUserFilterEnabled**<Boolean>

Gets only desktop rules that have their included user filter enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-MaxDesktops**<Int32>

Gets only desktop rules granting the specified number of machine assignment entitlements.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Metadata<String>**

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PublishedName<String>**

Gets only desktop rules with the specified published name, that is, the desktop name that the end user sees.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecureIcaRequired<Boolean>**

Gets only desktop rules that require desktop sessions to machines assigned by the rule to use the SecureICA protocol (\$true) or not (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UUID<Guid>**

Gets rules with the specified value of UUID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

**-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false



### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See `about_Broker_Filtering` for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through `Select-Object`, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.

Accept Pipeline Input?	false
------------------------	-------

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

Citrix.Broker.Admin.SDK.AssignmentPolicyRule

Get-BrokerAssignmentPolicyRule returns all assignment policy rules that match the specified selection criteria.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerAssignmentPolicyRule
```

Returns all desktop rules from the assignment policy. This offers a complete description of the current site's assignment policy with respect to machine assignment entitlements for delivery of desktop sessions from private desktop groups.

### ----- EXAMPLE 2 -----

```
C:\PS> $dg = Get-BrokerDesktopGroup 'Sales Support'
```

```
C:\PS> Get-BrokerAssignmentPolicyRule -DesktopGroupUid $dg.Uid
```

Returns all rules in the assignment policy that give users entitlements to machine assignments in the Sales Support desktop group for delivery of full desktop sessions.

# Get-BrokerCatalog

May 10, 2016

Gets catalogs configured for this site.

## Syntax

```
Get-BrokerCatalog [-UId] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerCatalog [[-Name] <String>] [-AllocationType <AllocationType>] [-AssignedCount <Int32>] [-AvailableAssignedCount <Int32>] [-AvailableCount <Int32>] [-AvailableUnassignedCount <Int32>] [-Description <String>] [-HypervisorConnectionUid <Int32>] [-IsRemotePC <Boolean>] [-MachinesArePhysical <Boolean>] [-Metadata <String>] [-MinimumFunctionalLevel <FunctionalLevel>] [-PersistUserChanges <PersistUserChanges>] [-ProvisioningSchemeId <Guid>] [-ProvisioningType <ProvisioningType>] [-PvsAddress <String>] [-PvsDomain <String>] [-RemotePCDesktopGroupPriority <Int32>] [-RemotePCDesktopGroupUid <Int32>] [-RemotePCHypervisorConnectionUid <Int32>] [-Scopeld <Guid>] [-ScopeName <String>] [-SessionSupport <SessionSupport>] [-UnassignedCount <Int32>] [-UsedCount <Int32>] [-UUID <Guid>] [-MachineUid <Int32>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieves catalogs matching the specified criteria. If no parameters are specified this cmdlet enumerates all catalogs.

See [about\\_Broker\\_Filtering](#) for information about advanced filtering options.

----- BrokerCatalog Object

The catalog object returned represents a group of related physical or virtual machines that have been configured in the site.

See [about\\_Broker\\_Machines](#) for more information.

-- AllocationType (Citrix.Broker.Admin.SDK.AllocationType)

Denotes how the machines in the catalog are allocated to a user.

Possible values are:

o Static: Machines get assigned to a user either by the admin or on first use. This relationship is static and changes only if an admin explicitly changes the assignments.

o Random: Machines are allocated to users randomly from a pool of available machines.

-- AssignedCount (System.Int32)

The number of assigned machines (machines that have been assigned to a user/users or a client name/address).

-- AvailableAssignedCount (System.Int32)

The number of available machines (not in a desktop group), that are also assigned to users.

-- AvailableCount (System.Int32)

The number of available machines (those not in any desktop group).

-- AvailableUnassignedCount (System.Int32)

The number of available machines (those not in any desktop group) that are not assigned to users.

-- Description (System.String)

Description of the catalog.

-- HypervisorConnectionUid (System.Int32?)

The Uid of the hypervisor connection that is associated with the machines in the catalog. This property only applies to MCS provisioned catalogs. For other provisioning types machines can be from one or more different hypervisor connections.

-- IsRemotePC (System.Boolean)

Specifies whether or not the catalog is a RemotePC catalog. Remote PC catalogs automatically configure appropriate machines without the need for manual configuration. See about\_Broker\_RemotePC for more information.

-- MachinesArePhysical (System.Boolean)

Specifies whether or not the machines in the catalog can be power-managed by the broker.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

Holds any metadata associated with the catalog.

-- MinimumFunctionalLevel (Citrix.Broker.Admin.SDK.FunctionalLevel)

The expected minimal functional level of the machines in the catalog.

-- Name (System.String)

Name of the catalog.

-- PersistUserChanges (Citrix.Broker.Admin.SDK.PersistUserChanges)

Specifies how user changes are persisted on machines in the catalog. Possible values are:

- o OnLocal: User changes are stored on the machine's local storage.

- o Discard: User changes are discarded.

- o OnPvd: User changes are stored on the user's personal vDisk.

-- ProvisioningSchemeId (System.Guid?)

The GUID of the provisioning scheme (if any) associated with the catalog. This only applies if the provisioning type is MCS.

-- ProvisioningType (Citrix.Broker.Admin.SDK.ProvisioningType)

Specifies how the machines are provisioned in the catalog. Possible values are:

- o Manual: No provisioning.

- o PVS: Machine provisioned by PVS (machine may be physical, blade, VM...)

o MCS: Machine provisioned by MCS (machine must be a VM).

-- PvsAddress (System.String)

IP address of the PVS server to be used in a catalog with a PVS ProvisioningType.

-- PvsDomain (System.String)

The domain of the PVS server to be used in a catalog with a PVS ProvisioningType.

-- RemotePCDesktopGroupPriorities (System.Int32[])

Remote PC desktop groups' association priorities.

-- RemotePCDesktopGroupUids (System.Int32[])

UIDs of the Remote PC desktop groups associated with this catalog.

-- RemotePCHypervisorConnectionUid (System.Int32?)

UID of the hypervisor connection used for powering on RemotePC machines in this catalog (only for catalogs with IsRemotePC set to true).

-- Scopes (Citrix.Broker.Admin.SDK.ScopeReference[])

The list of the delegated admin scopes to which the catalog belongs.

-- SessionSupport (Citrix.Broker.Admin.SDK.SessionSupport)

Specifies the session support of the machines in the catalog. Valid values are:

SingleSession, MultiSession.

-- Uid (System.Int32)

Uid of the catalog.

-- UnassignedCount (System.Int32)

The number of unassigned machines (machines not assigned to users).

-- UsedCount (System.Int32)

The number of machines in the catalog that are in a desktop group.

-- UUID (System.Guid)

The global ID of the catalog.

Related topics

[New-BrokerCatalog](#)

[Remove-BrokerCatalog](#)

[Rename-BrokerCatalog](#)

## Set-BrokerCatalog

### Parameters

**-Uid**<Int32>

Get catalogs with the specified UID.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Name**<String>

Gets catalogs with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AllocationType**<AllocationType>

Gets catalogs that are of the specified allocation type. Values can be:

- o Static - Machines in a catalog of this type are permanently assigned to a user.
- o Permanent - equivalent to 'Static'.
- o Random - Machines in a catalog of this type are picked at random and temporarily assigned to a user.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AssignedCount**<Int32>

Gets catalogs containing a specified number of assigned machines (machines that have been assigned to users).

This property is typically used with advanced filtering; see about\_Broker\_Filtering.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AvailableAssignedCount**<Int32>

Gets catalogs containing a specified number of available machines (those not in any desktop group) that are also assigned to users.

This property is typically used with advanced filtering; see about\_Broker\_Filtering.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AvailableCount**<Int32>

Gets catalogs containing a specified number of available machines (those not in any desktop group).

This property is typically used with advanced filtering; see about\_Broker\_Filtering.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AvailableUnassignedCount**<Int32>

Gets catalogs containing a specified number of available machines (those not in any desktop group) that are not assigned to users.

This property is typically used with advanced filtering; see about\_Broker\_Filtering.

Required?	false
Default Value	
Accept Pipeline Input?	false

--	--

#### **-Description<String>**

Gets catalogs with the specified description.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionUid<Int32>**

Gets catalogs associated with the specified hypervisor connection.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IsRemotePC<Boolean>**

Gets catalogs with the specified IsRemotePC value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachinesArePhysical<Boolean>**

Specifies whether machines in the catalog can be power-managed by the Citrix Broker Service. Where the Citrix Broker Service cannot control the power state of the machine specify \$true, otherwise \$false. Can only be specified together with a provisioning type of Pvs or Manual, or if used with the legacy CatalogKind parameter only with Pvs or PvsPvd catalog kinds.

Required?	false
Default Value	



Accept Pipeline Input?	false
------------------------	-------

#### **-Metadata<String>**

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MinimumFunctionalLevel<FunctionalLevel>**

Gets catalogs with a specific MinimumFunctionalLevel.

Valid values are L5, L7, L7\_6

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PersistUserChanges<PersistUserChanges>**

Gets catalogs with the specified behavior when persisting changes made by the end user. Possible values are:

- o OnLocal - User changes are stored on the machine's local storage.
- o Discard - User changes are discarded.
- o OnPvd - User changes are stored on the user's personal vDisk.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ProvisioningSchemeld<Guid>**

Gets catalogs associated with the specified provisioning scheme.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ProvisioningType**<ProvisioningType>

Specifies the provisioning type for the catalog. Values can be:

- o Manual - No provisioning.
- o PVS - Machine provisioned by PVS (machine may be physical, blade, VM,...).
- o MCS - Machine provisioned by MCS (machine must be VM).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PvsAddress**<String>

Gets catalogs containing machines provided by the Provisioning Services server with the specified address.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PvsDomain**<String>

Gets catalogs containing machines provided by the Provisioning Services server in the specified domain.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-RemotePCDesktopGroupPriority<Int32>**

Gets Remote PC catalogs with a Remote PC desktop group association with the specified priority.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-RemotePCDesktopGroupUid<Int32>**

Gets Remote PC catalogs associated with the specified Remote PC desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-RemotePCHypervisorConnectionUid<Int32>**

Gets Remote PC catalogs associated with the specified Remote PC hypervisor connection.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ScopeId<Guid>**

Gets catalogs that are associated with the given scope identifier.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ScopeName**<String>

Gets catalogs that are associated with the given scope name.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-SessionSupport**<SessionSupport>

Gets catalogs that have the specified session capability. Values can be:

- o SingleSession - Single-session only machine.
- o MultiSession - Multi-session capable machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-UnassignedCount**<Int32>

Gets catalogs containing a specified number of unassigned machines (machines not assigned to users).

This property is typically used with advanced filtering; see about\_Broker\_Filtering.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-UsedCount**<Int32>

Gets catalogs containing a specified number of machines used in a desktop group.

This property is typically used with advanced filtering; see about\_Broker\_Filtering.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-UUID**<Guid>

Get catalogs with the specified global ID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineUid**<Int32>

Gets the catalog containing the machine referenced by the specified unique identifier (UID).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
-----------	-------

Default Value	250
Accept Pipeline Input?	false

### **-Skip<Int32>**

Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy<String>**

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter<String>**

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

--	--

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None

#### Return Values

Citrix.Broker.Admin.SDK.Catalog

Get-BrokerCatalog returns an object for each matching catalog.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Get-BrokerCatalog -AllocationType Random
```

```
C:\PS> Get-BrokerCatalog -Filter { AllocationType -eq 'Random'}
```

These commands return all catalogs containing machines that are randomly assigned to users. The second form uses advanced filtering (see about\_Broker\_Filtering).

##### ----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerCatalog -Filter { AvailableCount -gt 10}
```

This command returns catalogs with more than 10 unused machines that are available for assignment to users.

##### ----- **EXAMPLE 3** -----

```
C:\PS> Get-BrokerCatalog -MaxRecordCount 1 -ProvisioningType Manual -SortBy '-AvailableCount'
```

This command returns the unmanaged catalog with the highest number of available machines.

# Get-BrokerConfigurationSlot

May 10, 2016

Gets configuration slots configured for this site.

## Syntax

```
Get-BrokerConfigurationSlot [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerConfigurationSlot [[-Name] <String>] [-Metadata <String>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Get the list of configuration slots defined for this site. Each configuration slot determines a collection of related settings that can be specified in a machine configuration associated with this slot.

For example, a configuration slot may be defined to configure only "User Profile Manager" settings.

See [about\\_Broker\\_Filtering](#) for information about advanced filtering options.

----- BrokerConfigurationSlot Object

The configuration slot object returned represents a named collection of related settings.

-- Description (System.String)

Optional description of this configuration slot.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

A map of metadata associated with this configuration slot.

-- Name (System.String)

Unique name of this configuration slot.

-- SettingsGroup (System.String)

The encoded identity of the settings group that every setting in the associated machine configuration instances must belong to.

-- Uid (System.Int32)

Unique Uid of this configuration slot.

## Related topics

[New-BrokerConfigurationSlot](#)

[Remove-BrokerConfigurationSlot](#)

[Get-BrokerMachineConfiguration](#)



## Parameters

### **-Uid**<Int32>

Get only the configuration slot with the specified unique identifier.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Name**<String>

Get only the configuration slot with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	

Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.ConfigurationSlot

Get-BrokerConfigurationSlot returns an object for each matching slot.

#### Examples

##### ----- **EXAMPLE 1** -----

Get-ConfigurationSlot

Retrieves every configuration slot.

##### ----- **EXAMPLE 2** -----

Get-ConfigurationSlot -Name "AppV"

Retrieves the configuration slot named "AppV".

# Get-BrokerConfiguredFTA

May 10, 2016

Gets any file type associations configured for an application.

## Syntax

```
Get-BrokerConfiguredFTA -Uid <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerConfiguredFTA [-ApplicationUid <Int32>] [-ContentType <String>] [-ExtensionName <String>] [-HandlerDescription
<String>] [-HandlerName <String>] [-HandlerOpenArguments <String>] [-UUID <Guid>] [-ReturnTotalRecordCount] [-
MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Gets any file type associations that are configured for content redirection to a published application.

File type association associates a file extension (such as ".txt") with an application (such as Notepad). In a Citrix environment file type associations on a user device can be configured so that when an user clicks on a document it launches the appropriate published application. This is known as "content redirection".

Configured file type associations are different from imported file type associations. Configured file type associations are those that are actually associated with published applications for the purposes of content redirection. Imported file type associations are lists of known file type associations for a given desktop group. See Update-BrokerImportedFTA for more information about imported file type associations.

----- BrokerConfiguredFTA Object

The BrokerConfiguredFTA object represents a file type association configured for a published application. It contains the following properties:

-- ApplicationUid (System.Int32)

The Uid of the application configured for the file type association.

-- ContentType (System.String)

Content type of the file, such as "text/plain" or "application/vnd.ms-excel".

-- ExtensionName (System.String)

A single file extension, such as .txt, unique within the scope of a desktop group.

-- HandlerDescription (System.String)

File type description, such as "Test Document", "Microsoft Word Text Document", etc.

-- HandlerName (System.String)

File type handler name, e.g. "Word.Document.8" or TXTFILE.

-- HandlerOpenArguments (System.String)

The arguments used for the 'open' action on files of this type.

-- Uid (System.Int32)

Unique internal identifier of configured file type association.

-- UUID (System.Guid)

UUID of the configured file type association.

## Related topics

[New-BrokerConfiguredFTA](#)

[Remove-BrokerConfiguredFTA](#)

[Update-BrokerImportedFTA](#)

## Parameters

**-Uid**<Int32>

Gets only the configured file type association for the specified unique identifier.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-ApplicationUid**<Int32>

Gets only the configured file type associations for the specified application unique identifier.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ContentType**<String>

Gets only the configured file type associations for the specified content type (as seen in the Registry). For example, "text/plain" or "application/msword".

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-ExtensionName**<String>

Gets only the configured file type associations for the specified extension name. For example, ".txt" or ".doc".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HandlerDescription**<String>

Gets only the configured file type associations for the specified handler description. For example, "Text Document".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HandlerName**<String>

Gets only the configured file type associations for the specified handler name. For example, "TXTFILE" or "Word.Document.8".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HandlerOpenArguments**<String>

Gets only the configured file type associations for the specified open argument to the handler. For example, "%1".

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-UUID**<Guid>

Gets configured file type associations with the specified value of UUID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
-----------	-------



Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

None No input is accepted from the pipeline.

## Return Values

Citrix.Broker.Admin.SDK.ConfiguredFTA

This cmdlet returns one or more ConfiguredFTA objects.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerConfiguredFTA
```

Returns all configured file type associations.

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerConfiguredFTA -ExtensionName ".txt"
```

Returns only configured file type associations that have a ".txt" extension.

# Get-BrokerConnectionLog

May 10, 2016

Get entries from the site's session connection log.

## Syntax

```
Get-BrokerConnectionLog [-Uid] <Int64> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerConnectionLog [-MachineName] <String> [-BrokeringTime <DateTime>] [-BrokeringUserName <String>] [-BrokeringUserUPN <String>] [-ConnectionFailureReason <ConnectionFailureReason>] [-Disconnected <Boolean>] [-EndTime <DateTime>] [-EstablishmentTime <DateTime>] [-MachineDNSName <String>] [-MachineUid <Int32>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets connection log entries matching the specified criteria. If no parameters are specified all connection log entries are returned.

The session connection log contains entries describing each brokered connection, or reconnection, attempt to a session in the site.

Each log entry describes a single connection brokering attempt to a new or existing session within the site. A single session can have multiple entries in the connection log, for example where the end user brokers a connection to a new session, disconnects and later brokers a reconnection. Conversely, other sessions may have none (e.g. console sessions).

By default connection log entries are removed after 48 hours.

For information about advanced filtering options when using the `-Filter` parameter, see [about\\_Broker\\_Filtering](#); for information about machines, see [about\\_Broker\\_Machines](#).

### ----- BrokerConnectionLog Object

The `BrokerConnectionLog` object represents a single brokered connection attempt to a new or existing session on a machine in the site. It contains the following properties:

-- `BrokeringTime` (System.DateTime)

The time at which the connection attempt was made.

-- `BrokeringUserName` (System.String)

The name of the user making the connection (in DOMAIN\User format).

-- `BrokeringUserUPN` (System.String)

The name of the user making the connection (in user@upndomain.com format).

-- `ConnectionFailureReason` (Citrix.Broker.Admin.SDK.ConnectionFailureReason?)

The status of the connection attempt. A value of `None` indicates that the connection was successfully established, `$null`

that the attempt is still in progress, and other values indicate that the attempt failed for the specified reason.

-- Disconnected (System.Boolean?)

Indicates if the connection was ended by disconnection (True), logoff or establishment failure (False), or is still active (\$null).

-- EndTime (System.DateTime?)

The time at which the connection ended. If the connection ended by disconnection, the underlying machine session would still exist in a disconnected state.

-- EstablishmentTime (System.DateTime?)

The time at which the connection was successfully established. The value is \$null if the connection attempt failed or is still in progress.

-- MachineDNSName (System.String)

The name of the machine to which the connection was made (in machine@dnsdomain.com form).

-- MachineName (System.String)

The name of the machine to which the connection was made (in DOMAIN\Machine format).

-- MachineUid (System.Int32)

The UID of the machine to which the connection was made.

-- Uid (System.Int64)

The UID of the connection log entry itself.

Related topics

Parameters

**-Uid**<Int64>

Gets a specific connection log entry identified by its UID.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-MachineName**<String>

Gets connection log entries for the specified machines (in DOMAIN\Machine format).

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-BrokeringTime**<DateTime>

Gets connection log entries with a specified brokering time. For more flexibility when searching on brokering time use the -Filter parameter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-BrokeringUserName**<String>

Gets connection log entries for the specified users (in DOMAIN\User format).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-BrokeringUserUPN**<String>

Gets connection log entries for the specified users (in user@upndomain.com format).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ConnectionFailureReason**<ConnectionFailureReason>

Gets connection log entries which failed for the specified reason.

Required?	false

Default Value	
Accept Pipeline Input?	false

#### **-Disconnected**<Boolean>

Gets connection log entries with the specified disconnection status, that is, whether the connection was disconnected, or logged-off.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-EndTime**<DateTime>

Gets connection log entries with the specified end time. For more flexibility when searching on end time use the -Filter parameter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-EstablishmentTime**<DateTime>

Gets connection log entries with the specific establishment time. For more flexibility when searching on establishment time use the -Filter parameter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineDNSName**<String>

Gets connection log entries for the specified machines (in machine@dnsdomain.com format).

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-MachineUid**<Int32>

Gets connection log entries for a specific machine identified by its UID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

--	--

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through `Select-Object`, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host



name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

None You cannot pipe input into this cmdlet.

### Return Values

Citrix.Broker.Admin.SDK.ConnectionLog

An entry from the connection log.

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> $when = [DateTime]::Now - [TimeSpan]::FromMinutes(30)
```

```
C:\PS> Get-BrokerConnectionLog -Filter {BrokeringTime -gt $when} -SortBy '+MachineName,-EndTime'
```

Gets all connection log entries for sessions brokered in the past 30 minutes, ordered first by machine name (ascending), then by session end time (descending).

# Get-BrokerController

May 10, 2016

Gets Controllers running broker services in the site.

## Syntax

```
Get-BrokerController [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerController [-MachineName <String>] [-ControllerVersion <String>] [-DesktopsRegistered <Int32>] [-DNSName <String>] [-LastActivityTime <DateTime>] [-LastLicensingServerEvent <LicensingServerEvent>] [-LastLicensingServerEventTime <DateTime>] [-LastStartTime <DateTime>] [-LicensingGraceState <LicensingGraceState>] [-LicensingServerState <LicensingServerState>] [-Metadata <String>] [-OSType <String>] [-OSVersion <String>] [-SID <String>] [-State <ControllerState>] [-UUID <Guid>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets Controllers from the current site that match the specified search criteria.

Controllers are server machines running instances of the broker service. The broker service is responsible for the brokering of user sessions to desktops or applications, and for power management of the underlying machines. An operational site must contain at least one Controller.

If no search criteria are specified, all Controllers in the site are obtained.

----- BrokerController Object

The BrokerController object represents a single controller running an instance of the Broker service. It contains the following properties:

-- ActiveSiteServices (System.String[])

The Broker site services active on the controller.

-- AssociatedHypervisorConnectionUids (System.Int32[])

The UIDs of the hypervisor connections being managed by the Broker service on the controller.

-- ControllerVersion (System.String)

The version of the Broker service on the controller.

-- DesktopsRegistered (System.Int32)

The number of VDA machines registered with the Broker service on the controller.

-- DNSName (System.String)

The DNS name of the controller.

-- LastActivityTime (System.DateTime?)

The last reported activity time of the Broker service on the controller.

-- LastLicensingServerEvent (Citrix.Broker.Admin.SDK.LicensingServerEvent?)

Last significant licensing server event reported by the Broker service on the controller.

-- LastLicensingServerEventDetails (System.String[])

Additional details associated with the last significant licensing server event.

-- LastLicensingServerEventTime (System.DateTime?)

Time at which the last significant licensing server event was reported.

-- LastStartTime (System.DateTime?)

The last start-up time of the Broker service on the controller.

-- LicensingGracePeriodReasons (Citrix.Broker.Admin.SDK.LicensingGracePeriodReason[])

Current active or expired licensing grace periods in effect on the controller.

-- LicensingGracePeriodTimesRemaining (System.TimeSpan[])

Times remaining in currently active or expired licensing grace periods in effect on the controller. Expired grace periods are indicated by zero remaining time. The number and order of entries in this list matches that in the LicensingGracePeriodReasons list.

-- LicensingGraceState (Citrix.Broker.Admin.SDK.LicensingGraceState)

The licensing grace state currently in effect in the Broker service on the controller.

-- LicensingServerState (Citrix.Broker.Admin.SDK.LicensingServerState)

The licensing server state currently in effect in the Broker service on the controller.

-- MachineName (System.String)

The Windows name of the controller.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

The metadata for the controller.

-- OSType (System.String)

The Operating System type of the controller.

-- OSVersion (System.String)

The Operating System version of the controller.

-- SID (System.String)

The SID of the controller.

-- State (Citrix.Broker.Admin.SDK.ControllerState)

The state of the Broker service on the controller.

-- Uid (System.Int32)

The UID of the controller instance.

-- UUID (System.Guid)

A globally unique identifier of the controller instance.

## Related topics

[Get-BrokerDesktop](#)

## Parameters

**-Uid**<Int32>

Gets only Controller with the specified unique ID.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-MachineName**<String>

Gets only Controllers with the specified Windows name. ('domain\machine')

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ControllerVersion**<String>

Gets only Controllers running the specified version of the broker service.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopsRegistered<Int32>**

Gets only Controllers that have the specified number of desktops currently registered. This parameter is mainly of use with advanced filtering; see about\_Broker\_Filtering.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DNSName<String>**

Gets only Controllers with the specified DNS name ('machine.domain')

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastActivityTime<DateTime>**

Gets only Controllers last reported as active at the specified time. This parameter is mainly of use with advanced filtering; see about\_Broker\_Filtering.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastLicensingServerEvent<LicensingServerEvent>**

Gets only Controllers with the specified last license server event recorded.

Required?	false
Default Value	
Accept Pipeline Input?	false

--	--

#### **-LastLicensingServerEventTime**<DateTime>

Gets only Controllers with its last recorded licensing server event at the specified time. This parameter is mainly of use with advanced filtering; see about\_Broker\_Filtering.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastStartTime**<DateTime>

Gets only Controllers that last started-up at the specified time. This parameter is mainly of use with advanced filtering; see about\_Broker\_Filtering.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LicensingGraceState**<LicensingGraceState>

Gets only Controllers in the specified licensing grace state.

Valid values are: NotActive, InOutOfBoxGracePeriod, InSupplementalGracePeriod, InEmergencyGracePeriod and GracePeriodExpired.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LicensingServerState**<LicensingServerState>

Gets only Controllers in the specified licensing server state. Valid values are: ServerNotSpecified, NotConnected, OK, LicenseNotInstalled, LicenseExpired, Incompatible and Failed.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-Metadata<String>**

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OSType<String>**

Gets only Controllers running the specified Operating System type.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OSVersion<String>**

Gets only Controllers running the specified Operating System version.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SID<String>**

Gets only Controllers with the specified SID.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-State**<ControllerState>

Gets only Controllers currently in the specified state.

Valid values are: Failed, Off, On, and Active.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UUID**<Guid>

Gets only the Controller with the specified GUID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.



Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip<Int32>**

Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy<String>**

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter<String>**

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.Controller

Returns Controllers matching all specified selection criteria.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerController -State Active
```

Gets all Controllers in the site that are currently active (powered on and fully operational).

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerController -Filter 'LastStartTime -gt "-30:00"'
```

Gets all Controllers in the site that started-up in the last 30 minutes.

# Get-BrokerDBConnection

May 10, 2016

Gets the database connection string for the specified data store used by the Broker Service.

## Syntax

```
Get-BrokerDBConnection [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets the database connection string from the currently selected Broker Service instance.

If the returned string is blank, no valid connection string has been specified. In this case the service is running, but is idle and awaiting specification of a valid connection string.

The current service instance is the one on the local machine, or the one most recently specified using the -AdminAddress parameter of a Broker SDK cmdlet.

## Related topics

[Set-BrokerDBConnection](#)

[Get-BrokerServiceStatus](#)

[Test-BrokerDBConnection](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

System.String

The database connection string configured for the current Broker Service instance.

## Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

##### NoDBConnections

The database connection string for the Broker Service has not been specified.

##### PermissionDenied

You do not have permission to execute this command.

##### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

##### CommunicationError

There was a problem communicating with the remote service.

##### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

C:\PS> Get-BrokerDBConnection -AdminAddress controller1.mydomain.net

Gets the database connection string in use by the Broker Service instance running on controller "controller1.mydomain.net".

# Get-BrokerDBSchema

May 10, 2016

Gets SQL scripts to create or maintain the database schema for the Citrix Broker Service.

## Syntax

```
Get-BrokerDBSchema -DatabaseName <String> [-ServiceGroupName <String>] [-ScriptType <DatabaseScriptType>] [-SID <String>] [-LocalDatabase] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets SQL scripts that can be used to create a new Citrix Broker Service database schema, add a new Broker service to an existing site, remove a Broker service from a site, or create a database server logon for a Broker service.

If no Sid parameter is provided, the scripts obtained relate to the currently selected Broker service instance, otherwise the scripts relate to Broker service instance running on the machine identified by the Sid provided. When obtaining the Evict script, a Sid parameter must be supplied.

The current service instance is the one on the local machine, or the one most recently specified using the -AdminAddress parameter of a Broker SDK cmdlet.

The service instance used to obtain the scripts does not need to be a member of a site or to have had its database connection configured.

The database scripts support only Microsoft SQL Server, or SQL Server Express, and require Windows integrated authentication to be used. They can be run using SQL Server's SQLCMD utility, or by copying the script into an SQL Server Management Studio (SSMS) query window and executing the query. If using SSMS, the query must be executed in 'SQLCMD mode'.

The ScriptType parameter determines which script is obtained. If ScriptType is not specified, or is FullDatabase, the script contains:

- o Creation of service schema
- o Creation of database server logon
- o Creation of database user
- o Addition of database user to Broker service roles

If ScriptType is Instance, the returned script contains:

- o Creation of database server logon
- o Creation of database user
- o Addition of database user to Broker service roles

If ScriptType is Evict, the returned script contains:

- o Removal of Broker service instance from database

o Removal of database user

If ScriptType is Login, the returned script contains:

o Creation of database server logon only

If the service uses two data stores they can exist in the same database.

You do not need to configure a database before using this command.

## Related topics

[Set-BrokerDBConnection](#)

[Test-BrokerDBConnection](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database into which the new Broker service schema is to be placed, or in which it already exists. The database itself is not created by any of the script types; it must already exist before the scripts are run.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-ServiceGroupName**<String>

Specifies the name of the service group to be used when creating the database schema. The service group is a collection of all the Broker services that share the same database instance and are considered equivalent; that is, all the services within a service group can be used interchangeably.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ScriptType**<DatabaseScriptType>

Specifies the type of database script returned. Available script types are:

-- FullDatabase

Creates a database schema for the Citrix Broker Service in a database instance that does not already contain one. This is

used when creating a new site. DatabaseName and ServiceGroupName are required parameters for this script type.

#### -- Instance

Adds a Broker Service instance to a database and so to the associated site. Appropriate database server logons and users are created to allow the service instance access to the required service schemas.

#### -- Evict

Removes a Broker Service instance from the database and so from the site. All reference to the service instance is removed from the database. DatabaseName and Sid are required parameters for this script type.

#### -- Login

Adds a logon for the Broker Service instance to a database server. This is specifically for use when configuring SQL Server mirroring where the mirror server must have appropriate logons created for all service instances in the site.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -SID<String>

Specifies the SID of the controller on which the Broker Service instance to remove from the database is running (only valid for a script type of Evict).

Required?	false
Default Value	None
Accept Pipeline Input?	false

#### -LocalDatabase<SwitchParameter>

Specifies whether the database script is to be used in a database instance run on the same controller as other services in the service group. Including this parameter ensures the script creates only the required permissions for local services to access the database schema for Broker services. If this parameter is specified inappropriately, the service instance will not be able to connect to the database.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

System.String

A string containing the required SQL script for applying to a database.

#### Notes

The scripts returned support Microsoft SQL Server Express Edition, Microsoft SQL Server Standard Edition, and Microsoft SQL Server Enterprise Edition databases only, and are generated on the assumption that integrated authentication will be used.

If the ScriptType parameter is not included or set to 'FullDatabase', the full database script is returned, which will:

Create the database schema.

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist).

If the ScriptType parameter is set to 'Instance', the script will:

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist) and associate it with a user.

If the ScriptType parameter is set to 'Login', the script will:

Create the logon (providing the schema does not already exist) and associate it with a pre-existing user of the same name.

If the LocalDatabase parameter is included, the NetworkService account will be added to the list of accounts permitted to access the database. This is required only if the database is run on a controller.

If the command fails, the following errors can be returned.



## Error Codes

-----

### GetSchemasFailed

The database schema could not be found.

### ActiveDirectoryAccountResolutionFailed

The specified Active Directory account or Group could not be found.

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

### PermissionDenied

You do not have permission to execute this command.

### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

### CommunicationError

There was a problem communicating with the remote service.

### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS>Get-BrokerDBSchema -DatabaseName MySiteDB -ServiceGroupName MyServiceGroup > C:\BrokerSchema.sql
```

Gets a script to create the full database schema for the Citrix Broker Service and copies it to a file called "C:\BrokerSchema.sql"

This script can be used to create the service schema in a database with name "MySiteDB", which must already exist, and must not already contain a Broker service schema.

### ----- EXAMPLE 2 -----

```
C:\PS>Get-BrokerDBSchema -DatabaseName MySiteDB -ScriptType Login > C:\BrokerLogins.sql
```

Gets a script to create the appropriate database server logon for the Broker service. This can be used when configuring a mirror server for use.

# Get-BrokerDBVersionChangeScript

May 10, 2016

Gets an SQL service schema update script for the Citrix Broker Service.

## Syntax

```
Get-BrokerDBVersionChangeScript -DatabaseName <String> -TargetVersion <Version> [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Gets an SQL script that can be used to update the current Citrix Broker Service database schema. An update can be an upgrade or downgrade.

A script can be obtained to update the current service schema to any version that is reachable by applying available schema update packages that have been uploaded by the Citrix Broker Service.

Typically, this mechanism is used to update the current service schema to a newer version, however it can also be used to revert previously applied updates.

The SQL script obtained can be run using SQL Server's SQLCMD utility, or by copying the script into an SQL Server Management Studio (SSMS) query window and executing the query. If using SSMS, the query must be executed in 'SQLCMD mode'.

The schema update packages used to generate update scripts are stored in the database; because of this, any Citrix Broker Service in the site can be used to obtain a schema update script.

The fact that an update package is available in the database does not mean that the update has actually been applied to the service's schema. In addition, application of an update does not remove the associated update packages.

Take care when using the update scripts. Citrix recommends that where possible service schema upgrades are performed using Studio rather than manually via the SDK. The database should be backed-up before an update is attempted. The database script may also require exclusive use of the schema, in which case all Citrix Broker Services must be shutdown before applying the update.

Once an update has been applied to the service schema, any existing Citrix Broker Services that are incompatible with the updated schema will cease to operate. The service state, as reported by Get-BrokerServiceStatus, provides information about the service compatibility (e.g. DBNewerVersionThanService).

## Related topics

[Get-BrokerInstalledDBVersion](#)

[Get-BrokerServiceStatus](#)

[Get-BrokerDBSchema](#)

## Parameters

**-DatabaseName**<String>

The name of the database containing the Citrix Broker Service schema to be updated.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-TargetVersion**<Version>

The required target service schema version of the update. This is the service schema version obtained after the update script is applied.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

System.Management.Automation.PSObject

The Get-BrokerDBVersionChangeScript cmdlet returns a PSObject containing a script that can be used to update the Citrix Broker Service database schema. The object has the following properties:

-- Script

The raw text of the SQL script to apply the update.

-- CanUndo

If true, indicates that after the update has been applied, a script to revert from the updated schema to the schema state prior to the update can be obtained. Because Get-BrokerDBVersionChangeScript gets only update scripts relating to the current schema version, a script to revert an update can be obtained only after the update has been applied.

-- NeedExclusiveAccess

If true, indicates that the update requires exclusive access to the Citrix Broker Service's schema while the update is applied; all Citrix Broker Services must be shutdown during the update.

## Notes

The PSObject returned by this cmdlet contains the following properties:

-- Script The raw text of the SQL script to apply the update, or null in the case when no upgrade path to the specified target version exists.

-- NeedExclusiveAccess Indicates whether all services in the service group must be shut down during the update or not.

-- CanUndo Indicates whether the generated script allows the updated schema to be reverted to the state prior to the update.

Scripts to update the schema version are stored in the database so any service in the service group can obtain these scripts. Extreme caution should be exercised when using update scripts. Citrix recommends backing up the database before attempting to upgrade the schema. Database update scripts may require exclusive use of the schema and so may not be able to execute while any Broker services are running. However, this depends on the specific update being carried out.

After a schema update has been carried out, services that require the previous version of the schema may cease to operate. The ServiceState parameter reported by the Get-BrokerServiceStatus command provides information about service compatibility. For example, if the schema has been upgraded to a more recent version that a service cannot use, the service reports "DBNewerVersionThanService".

If the command fails, the following errors can be returned.

## Error Codes

-----

### NoOp

The operation was successful but had no effect.

### NoDBConnections

The database connection string for the Broker Service has not been specified.

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> $update = Get-BrokerDBVersionChangeScript -DatabaseName MyDb -TargetVersion 1.0.75.0
```

```
C:\PS> $update.Script > update_75.sql
```

Gets an SQL update script to update the current schema to version 1.0.75.0. The resulting update\_75.sql script is suitable for direct use with the SQL Server SQLCMD utility.

# Get-BrokerDelayedHostingPowerAction

May 10, 2016

Gets power actions that are executed after a delay.

## Syntax

```
Get-BrokerDelayedHostingPowerAction [-Uid] <Int64> [-Property <String[]>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Get-BrokerDelayedHostingPowerAction [[-MachineName] <String>] [-Action <PowerManagementAction>] [-ActionDueTime
<DateTime>] [-DNSName <String>] [-HostedMachineName <String>] [-HypervisorConnectionName <String>] [-
HypervisorConnectionUid <Int32>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-
Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Finds all delayed power actions that match the specified search criteria.

----- BrokerDelayedHostingPowerAction Object

The BrokerDelayedHostingPowerAction object represents an instance of a power action that is executed after a delay. It contains the following properties:

-- Action (Citrix.Broker.Admin.SDK.PowerManagementAction)

The power action to apply to the machine. Possible values are ShutDown and Suspend.

-- ActionDueTime (System.DateTime)

The UTC time at which the power action is due to be queued for execution.

-- DNSName (System.String)

The fully qualified DNS name of the machine that the power action applies to.

-- HostedMachineName (System.String)

The hypervisor's name for the machine that the power action applies to.

-- HypervisorConnectionUid (System.Int32)

The unique identifier of the hypervisor connection that is associated with the target machine.

-- MachineName (System.String)

The name of the machine that the power action applies to, in the form domain\machine.

-- Uid (System.Int64)

The unique identifier of the power action.

## Related topics

[New-BrokerDelayedHostingPowerAction](#)

[Remove-BrokerDelayedHostingPowerAction](#)

[Remove-BrokerHostingPowerAction](#)

## Parameters

**-Uid**<Int64>

Gets only the single action record whose ID matches the specified value.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-MachineName**<String>

Gets only the records for actions that are for machines whose name (of the form domain\machine) matches the specified string.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Action**<PowerManagementAction>

Gets only the records for actions with the specified action type.

Valid values are Shutdown and Suspend.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ActionDueTime**<DateTime>

Gets only the records for actions due to be queued for execution at the specified time. This is useful with advanced filtering; for more information, see [about\\_Broker\\_Filtering](#).



Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DNSName<String>**

Gets only the records for actions that are for machines whose DNS name matches the specified string.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HostedMachineName<String>**

Gets only the records for actions that are for machines whose Hosting Name (the machine name as understood by the hypervisor) matches the specified string.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionName<String>**

Gets only the records for actions for machines hosted through a hypervisor connection whose name matches the specified string.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionUid<Int32>**

Gets only the records for actions for machines hosted through a hypervisor connection whose ID matches the specified

value.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

**-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through `Select-Object`, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.

Accept Pipeline Input?	false
------------------------	-------

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

Citrix.Broker.Admin.SDK.DelayedHostingPowerAction

Get-BrokerDelayedHostingPowerAction returns all delayed power actions that match the specified selection criteria.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Get-BrokerDelayedHostingPowerAction
```

Fetches records for all known delayed power actions that have not yet been queued for execution.

# Get-BrokerDesktop

May 10, 2016

Gets desktops configured for this site.

## Syntax

```
Get-BrokerDesktop [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerDesktop [[-MachineName] <String>] [-AgentVersion <String>] [-ApplicationInUse <String>] [-AssignedClientName <String>] [-AssignedIPAddress <String>] [-AssociatedUserFullName <String>] [-AssociatedUserName <String>] [-AssociatedUserUPN <String>] [-AutonomouslyBrokered <Boolean>] [-CatalogName <String>] [-CatalogUid <Int32>] [-ClientAddress <String>] [-ClientName <String>] [-ClientVersion <String>] [-ColorDepth <ColorDepth>] [-ConnectedViaHostName <String>] [-ConnectedViaIP <String>] [-ControllerDNSName <String>] [-DeliveryType <DeliveryType>] [-Description <String>] [-DesktopCondition <String>] [-DesktopGroupName <String>] [-DesktopGroupUid <Int32>] [-DesktopKind <DesktopKind>] [-DeviceId <String>] [-DNSName <String>] [-FunctionalLevel <FunctionalLevel>] [-HardwareId <String>] [-HostedMachineId <String>] [-HostedMachineName <String>] [-HostingServerName <String>] [-HypervisorConnectionName <String>] [-HypervisorConnectionUid <Int32>] [-IconUid <Int32>] [-ImageOutOfDate <Boolean>] [-InMaintenanceMode <Boolean>] [-IPAddress <String>] [-IsAssigned <Boolean>] [-IsPhysical <Boolean>] [-LastConnectionFailure <ConnectionFailureReason>] [-LastConnectionTime <DateTime>] [-LastConnectionUser <String>] [-LastDeregistrationReason <DeregistrationReason>] [-LastDeregistrationTime <DateTime>] [-LastErrorReason <String>] [-LastErrorTime <DateTime>] [-LastHostingUpdateTime <DateTime>] [-LaunchedViaHostName <String>] [-LaunchedViaIP <String>] [-MachineInternalState <MachineInternalState>] [-MachineUid <Int32>] [-OSType <String>] [-OSVersion <String>] [-PersistUserChanges <PersistUserChanges>] [-PowerActionPending <Boolean>] [-PowerState <PowerState>] [-Protocol <String>] [-ProvisioningType <ProvisioningType>] [-PublishedApplication <String>] [-PublishedName <String>] [-PvdStage <PvdStage>] [-RegistrationState <RegistrationState>] [-SecureIcaActive <Boolean>] [-SecureIcaRequired <Boolean>] [-SessionHidden <Boolean>] [-SessionId <Int32>] [-SessionState <SessionState>] [-SessionStateChangeTime <DateTime>] [-SessionUid <Int64>] [-SessionUserName <String>] [-SessionUserSID <String>] [-SID <String>] [-SmartAccessTag <String>] [-StartTime <DateTime>] [-SummaryState <DesktopSummaryState>] [-Tag <String>] [-WillShutdownAfterUse <Boolean>] [-ApplicationUid <Int32>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet is now deprecated, please use Get-BrokerMachine.

Retrieves desktops matching the specified criteria. If no parameters are specified this cmdlet enumerates all desktops.

Get-BrokerDesktop returns objects that combine desktop configuration and state information.

For single-session desktops, session information is displayed if present. It is possible that there are more than one sessions present on single-session desktops if 'fast user switching' is enabled, this cmdlet will prefer to return information about brokered sessions (rather than, for example, unbrokered direct RDP sessions). If there is no session running, session related fields return \$null.

For multi-session desktops, no session information is ever displayed by this cmdlet, so session related fields always return \$null. Get-BrokerSession can be used to get information about sessions on both multi-session and single-session desktops.

To count desktops, rather than retrieve full details of each desktop, use Group-BrokerDesktop instead.

For information about advanced filtering options, see about\_Broker\_Filtering; for information about desktops, see about\_Broker\_Desktops.

### ----- BrokerDesktop Object

The desktop object returned represents a physical or virtual machine configured in the site that is able to run either a Microsoft Windows desktop environment, individual applications, or both.

-- AgentVersion (System.String)

Version of the Citrix Virtual Delivery Agent (VDA) installed on the desktop.

-- ApplicationsInUse (System.String[])

List of applications in use on the desktop (in the form of browser name).

-- AssignedClientName (System.String)

The name of the endpoint client device that the desktop has been assigned to.

-- AssignedIPAddress (System.String)

The IP address of the endpoint client device that the desktop has been assigned to.

-- AssociatedUserFullNames (System.String[])

Full names of the users that have been associated with the desktop (in the form "Firstname Lastname").

Associated users are the current user(s) for shared desktops and the assigned users for private desktops.

-- AssociatedUserNames (System.String[])

Username of the users that have been associated with the desktop (usually in the form "domain\user").

Associated users are the current user(s) for shared desktops and the assigned users for private desktops.

-- AssociatedUserUPNs (System.String[])

The user principal names of the users that have been associated with the desktop (in the form user@upndomain.com).

Associated users are the current user(s) for shared desktops and the assigned users for private desktops.

-- AutonomouslyBrokered (System.Boolean?)

Session property indicating if the current session is an HDX session established by direct connection without being brokered.

Session properties are always null for multi-session desktops.

-- CatalogName (System.String)

Name of the catalog the desktop is a member of.

-- CatalogUid (System.Int32)

UID of the catalog the desktop is a member of.

-- ClientAddress (System.String)

Session property indicating the IP address of the client connected to the desktop.

Session properties are always null for multi-session desktops.

-- ClientName (System.String)

Session property indicating the host name of the client connected to the desktop.

Session properties are always null for multi-session desktops.

-- ClientVersion (System.String)

Session property indicating the version of the Citrix Receiver running on the connected client.

Session properties are always null for multi-session desktops.

-- ColorDepth (Citrix.Broker.Admin.SDK.ColorDepth?)

The color depth setting configured on the desktop, possible values are:

\$null, FourBit, EightBit, SixteenBit, and TwentyFourBit.

-- ConnectedViaHostName (System.String)

Session property indicating the host name of the connection gateway, router or client.

Session properties are always null for multi-session desktops.

-- ConnectedViaIP (System.String)

Session property indicating the IP address of the connection gateway, router or client.

Session properties are always null for multi-session desktops.

-- ControllerDNSName (System.String)

The DNS host name of the controller that the desktop is registered to.

-- DeliveryType (Citrix.Broker.Admin.SDK.DeliveryType)

Denotes whether the desktop delivers desktops only, apps only or both.

-- Description (System.String)

Description of the desktop.

-- DesktopConditions (System.String[])

List of outstanding desktop conditions for the desktop.

-- DesktopGroupName (System.String)

Name of the desktop group the desktop has been assigned to.

-- DesktopGroupUid (System.Int32)

Uid of the desktop group the desktop has been assigned to.

-- DesktopKind (Citrix.Broker.Admin.SDK.DesktopKind)

Deprecated.

Denotes whether the desktop is private or shared. AllocationType should be used instead.

-- DeviceId (System.String)

Session property indicating a unique identifier for the client device that has most recently been associated with the current session.

Session properties are always null for multi-session desktops.

-- DNSName (System.String)

The DNS host name of the desktop.

-- FunctionalLevel (Citrix.Broker.Admin.SDK.FunctionalLevel?)

The functional level of the desktop, if known.

-- HardwareId (System.String)

Session property indicating a unique identifier for the client hardware that has been most recently associated with the current session.

Session properties are always null for multi-session desktops.

-- HostedMachineId (System.String)

Unique ID within the hosting unit of the target managed desktop.

-- HostedMachineName (System.String)

The friendly name of a hosted desktop as used by its hypervisor. This is not necessarily the DNS name of the desktop.

-- HostingServerName (System.String)

DNS name of the hypervisor that is hosting the desktop if managed.

-- HypervisorConnectionName (System.String)

The name of the hypervisor connection that the desktop's hosting server is accessed through, if managed.

-- HypervisorConnectionUid (System.Int32?)

The UID of the hypervisor connection that the desktop's hosting server is accessed through, if managed.

-- IconUid (System.Int32?)

The UID of the desktop's icon that is displayed in StoreFront.

-- ImageOutOfDate (System.Boolean?)

Denotes whether the VM image for a hosted desktop is out of date.

-- InMaintenanceMode (System.Boolean)

Denotes whether the desktop is in maintenance mode.

-- IPAddress (System.String)

The IP address of the desktop.

-- IsAssigned (System.Boolean)

Denotes whether a private desktop has been assigned to a user/users, or a client name/address. Users can be assigned explicitly or by assigning on first use of the desktop.

-- IsPhysical (System.Boolean)

This value is true if the desktop is physical (ie not power managed by the Citrix Broker Service), and false otherwise.

-- LastConnectionFailure (Citrix.Broker.Admin.SDK.ConnectionFailureReason)

The reason for the last failed connection between a client and the desktop.

-- LastConnectionTime (System.DateTime?)

Time of the last detected connection attempt that either failed or succeeded.

-- LastConnectionUser (System.String)

The SAM name (in the form DOMAIN\user) of the user that last attempted a connection with the desktop. If the SAM name is not available, the SID is used.

-- LastDeregistrationReason (Citrix.Broker.Admin.SDK.DeregistrationReason?)

The reason for the last deregistration of the desktop with the broker. Possible values are:

AgentShutdown, AgentSuspended, AgentRequested, IncompatibleVersion, AgentAddressResolutionFailed, AgentNotContactable, AgentWrongActiveDirectoryOU, EmptyRegistrationRequest, MissingRegistrationCapabilities, MissingAgentVersion, InconsistentRegistrationCapabilities, NotLicensedForFeature, UnsupportedCredentialSecurityVersion, InvalidRegistrationRequest, SingleMultiSessionMismatch, FunctionalLevelTooLowForCatalog, FunctionalLevelTooLowForDesktopGroup, PowerOff, DesktopRestart, DesktopRemoved, AgentRejectedSettingsUpdate, SendSettingsFailure, SessionAuditFailure, SessionPrepareFailure, ContactLost, SettingsCreationFailure, UnknownError and BrokerRegistrationLimitReached.

-- LastDeregistrationTime (System.DateTime?)

Time of the last deregistration of the desktop from the controller.

-- LastErrorReason (System.String)

The reason for the last error detected in the desktop.

-- LastErrorTime (System.DateTime?)

The time of the last detected error.

-- LastHostingUpdateTime (System.DateTime?)

Time of last update to any hosting data (such as power state) for this desktop reported by the hypervisor connection.

-- LaunchedViaHostName (System.String)

Session property that denotes the host name of the StoreFront server used to launch the current brokered session.

Session properties are always null for multi-session desktops.

-- LaunchedViaIP (System.String)

Session property that denotes the IP address of the StoreFront server used to launch the current brokered session.

Session properties are always null for multi-session desktops.

-- MachineInternalState (Citrix.Broker.Admin.SDK.MachineInternalState)



The internal state of the machine associated with the desktop; reported while the desktop is registered to a controller, plus some private Citrix Broker Service states while the machine is not registered.

-- MachineName (System.String)

DNS host name of the machine associated with the desktop.

-- MachineUid (System.Int32)

Uid of the associated machine.

-- OSType (System.String)

A string that can be used to identify the operating system that is running on the desktop.

-- OSVersion (System.String)

A string that can be used to identify the version of the operating system running on the desktop, if known

-- PersistUserChanges (Citrix.Broker.Admin.SDK.PersistUserChanges)

Describes whether/how the user changes are persisted. Possible values are:

o OnLocal - Persist the user changes on the local disk of the desktop.

o Discard - Discard user changes.

o OnPvd - Persist user changes on the Citrix Personal vDisk.

-- PowerActionPending (System.Boolean)

Property indicating whether there are any pending power actions for the desktop.

-- PowerState (Citrix.Broker.Admin.SDK.PowerState)

The current power state of the desktop. Possible values are: Unmanaged, Unknown, Unavailable, Off, On, Suspended, TurningOn, TurningOff, Suspending, resuming.

-- Protocol (System.String)

Session property that denotes the protocol that the current session is using, can be either HDX, RDP or Console. Console sessions on XenDesktop 5 VDAs appear with a blank protocol.

Session properties are always null for multi-session desktops.

-- ProvisioningType (Citrix.Broker.Admin.SDK.ProvisioningType)

Describes how the machine associated with the desktop was provisioned, possible values are:

o Manual: No automated provisioning.

o PVS: Machine provisioned by PVS (may be physical, blade, VM,...)

o MCS: Machine provisioned by MCS (machine must be VM)

-- PublishedApplications (System.String[])

List of applications published by the desktop (displayed as browser names).

-- PublishedName (System.String)

The name of the desktop that is displayed in StoreFront, if the desktop is published.

-- PvdStage (Citrix.Broker.Admin.SDK.PvdStage)

For a desktop supporting Personal vDisk technology (PvD), indicates the stage of the PvD image preparation.

-- RegistrationState (Citrix.Broker.Admin.SDK.RegistrationState)

Indicates the registration state of the desktop. Possible values are: Unregistered, Initializing, Registered, Agent Error.

-- SecureIcaActive (System.Boolean?)

Session property that indicates whether SecureICA is active on the current session.

Session properties are always null for multi-session desktops.

-- SecureIcaRequired (System.Boolean?)

Flag indicating whether SecureICA is required or not when starting a session on the desktop.

-- SessionHidden (System.Boolean?)

Session property that indicates if a session is hidden.

Session properties are always null for multi-session desktops.

-- SessionId (System.Int32?)

Deprecated. A unique identifier that Remote Desktop Services uses to track the session but it is only unique on that machine and only unique at any one particular time.

-- SessionState (Citrix.Broker.Admin.SDK.SessionState?)

Session property indicating the state of the current session.

Session properties are always null for multi-session desktops, possible values are: Other, PreparingSession, Connected, Active, Disconnected, Reconnecting, NonBrokeredSession and Unknown.

-- SessionStateChangeTime (System.DateTime?)

Session property indicating the time of the last state change of the current session.

Session properties are always null for multi-session desktops.

-- SessionUid (System.Int64?)

Session property indicating the UID of the current session.

Session properties are always null for multi-session desktops.

-- SessionUserName (System.String)

Session property indicates the name of the current sessions' user (in the form DOMAIN\user).

Session properties are always null for multi-session desktops.

-- SessionUserSID (System.String)

Session property indicates the SID of the current sessions' user.

Session properties are always null for multi-session desktops.

-- SID (System.String)

The SID of the desktop.

-- SmartAccessTags (System.String[])

Session property that indicates the Smart Access tags for the current session.

Session properties are always null on multi-session desktops.

-- StartTime (System.DateTime?)

Session property that indicates the start time of the current session.

Session properties are always null on multi-session desktops.

-- SummaryState (Citrix.Broker.Admin.SDK.DesktopSummaryState)

Indicates the overall state of the desktop. The overall state is a result of other more specific states such as session state, registration state and power state. Possible values: Off, Unregistered, Available, Disconnected, InUse, Preparing.

-- Tags (System.String[])

A list of tags for the desktop.

-- Uid (System.Int32)

UID of the desktop object.

-- WillShutdownAfterUse (System.Boolean)

Flag indicating whether this desktop is tainted and will be shut down after all sessions on the desktop have ended. This flag should only ever be true on power managed, single-session desktops.

Note: The desktop will not shut down if it is in maintenance mode, but will shut down after the desktop is taken out of maintenance mode.

## Related topics

[Group-BrokerMachine](#)

[Get-BrokerMachine](#)

## Parameters

**-Uid**<Int32>

Gets desktops with a specific UID.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-MachineName**<String>

Gets desktops with a specific machine name (in the form 'domain\machine').

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AgentVersion**<String>

Gets desktops with a specific Citrix Virtual Delivery Agent version.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ApplicationInUse**<String>

Gets desktops running a specified published application (identified by browser name).

String comparisons are case-insensitive.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-AssignedClientName<String>**

Gets desktops assigned to a specific client name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssignedIPAddress<String>**

Gets desktops assigned to a specific client IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserFullName<String>**

Gets desktops with an associated user identified by their full name (usually in the form 'first-name last-name').

Associated users are the current user for shared desktops, and the assigned users for private desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserName<String>**

Gets desktops with an associated user identified by their user name (in the form 'domain\user').

Associated users are the current user for shared desktops, and the assigned users for private desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserUPN<String>**

Gets desktops with an associated user identified by their User Principle Name (in the form 'user@domain').

Associated users are the current user for shared desktops, and the assigned users for private desktops.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AutonomouslyBrokered**<Boolean>

Gets desktops according to whether their current session is autonomously brokered or not. Autonomously brokered sessions are HDX sessions established by direct connection without being brokered.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-CatalogName**<String>

Gets desktops from the catalog with the specific name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-CatalogUid**<Int32>

Gets desktops from a catalog with a specific UID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ClientAddress**<String>

Gets desktops with a specific client IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ClientName**<String>

Gets desktops with a specific client name.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ClientVersion**<String>

Gets desktops with a specific client version.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ColorDepth**<ColorDepth>

Gets desktops configured with a specific color depth.

Valid values are FourBit, EightBit, SixteenBit, and TwentyFourBit.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ConnectedViaHostName**<String>

Gets desktops with a specific host name of the incoming connection. This is usually a proxy or Citrix Access Gateway server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ConnectedViaIP**<String>

Gets desktops with a specific IP address of the incoming connection.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ControllerDNSName**<String>

Gets desktops with a specific DNS name of the controller they are registered with.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DeliveryType**<DeliveryType>

Gets desktops of a particular delivery type.

Valid values are AppsOnly, DesktopsOnly, DesktopsAndApps

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Description**<String>

Gets desktops with a specific description.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopCondition**<String>

Gets desktop with an outstanding desktop condition condition.

Valid values are:

- o CPU: Indicates the machine has high CPU usage
- o ICALatency: Indicates the network latency is high
- o UPMLogonTime: Indicates that the profile load time was high

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupName**<String>

Gets desktops from a desktop group with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupUid**<Int32>

Gets desktops from a desktop group with the specified UID.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-DesktopKind**<DesktopKind>

Deprecated: Use AllocationType parameter.

Gets desktops of a particular kind.

Valid values are Private, Shared.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DeviceId**<String>

Gets desktops with a specific client device ID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DNSName**<String>

Gets desktops with a specific DNS name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-FunctionalLevel**<FunctionalLevel>

Gets desktops with a specific FunctionalLevel.

Valid values are L5, L7, L7\_6

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HardwareId**<String>

Gets desktops with a specific client hardware ID.

Required?	false
-----------	-------



Default Value	
Accept Pipeline Input?	false

#### **-HostedMachineId**<String>

Gets desktops with a specific machine ID known to the hypervisor.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HostedMachineName**<String>

Gets desktops with a specific machine name known to the hypervisor.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HostingServerName**<String>

Gets desktops with a specific name of the hosting hypervisor server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionName**<String>

Gets desktops with a specific name of the hosting hypervisor connection.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionUid**<Int32>

Gets desktops with a specific UID of the hosting hypervisor connection.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-IconUid<Int32>**

Gets desktops with a specific configured icon. Note that desktops with a null IconUid use the icon of the desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ImageOutOfDate<Boolean>**

Gets desktops by whether their disk image is out of date (for machines provisioned using MCS only).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-InMaintenanceMode<Boolean>**

Gets desktops with a specific InMaintenanceMode setting.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IPAddress<String>**

Gets desktops with a specific IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IsAssigned<Boolean>**

Gets desktops according to whether they are assigned or not. Desktops may be assigned to one or more users or groups, a client IP address or a client endpoint name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -IsPhysical<Boolean>

Specifies if machines in the catalog can be power managed by the Citrix Broker Service. Where the power state of the machine cannot be controlled, specify \$true, otherwise \$false. Can only be specified together with a provisioning type of Pvs or Manual, or if used with the deprecated CatalogKind parameter only with Pvs or PvsPvd catalog kinds.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -LastConnectionFailure<ConnectionFailureReason>

Gets desktops with a specific reason for the last recorded connection failure. This value is None if the last connection was successful or if there has been no attempt to connect to the desktop yet.

Valid values are None, SessionPreparation, RegistrationTimeout, ConnectionTimeout, Licensing, Ticketing, and Other.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -LastConnectionTime<DateTime>

Gets desktops that last connected at a specific time. This is the time that the broker detected that the connection attempt either succeeded or failed.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -LastConnectionUser<String>

Gets desktops where a specific user name last attempted a connection (in the form 'domain\user').

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -LastDeregistrationReason<DeregistrationReason>

Gets desktops whose broker last recorded a specific deregistration reason.

Valid values are \$null, AgentShutdown, AgentSuspended, AgentRequested, IncompatibleVersion, AgentAddressResolutionFailed, AgentNotContactable, AgentWrongActiveDirectoryOU, EmptyRegistrationRequest, MissingRegistrationCapabilities, MissingAgentVersion, InconsistentRegistrationCapabilities, NotLicensedForFeature, UnsupportedCredentialSecurityVersion, InvalidRegistrationRequest, SingleMultiSessionMismatch, FunctionalLevelTooLowForCatalog, FunctionalLevelTooLowForDesktopGroup, PowerOff, DesktopRestart, DesktopRemoved, AgentRejectedSettingsUpdate, SendSettingsFailure, SessionAuditFailure, SessionPrepareFailure, ContactLost, SettingsCreationFailure, UnknownError and BrokerRegistrationLimitReached.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-LastDeregistrationTime**<DateTime>

Gets desktops by the time that they were last deregistered.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastErrorReason**<String>

Gets desktops with the specified last error reason.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastErrorTime**<DateTime>

Gets desktops with the specified last error time.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastHostingUpdateTime**<DateTime>

Gets desktops with a specific time that the hosting information was last updated.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LaunchedViaHostName**<String>

Gets desktops with a specific host name of the StoreFront server from which the user launched the session.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-LaunchedViaIP**<String>

Gets desktops with a specific IP address of the StoreFront server from which the user launched the session.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineInternalState**<MachineInternalState>

Gets desktops with the specified internal machine state.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineUid**<Int32>

Gets desktops with a specific machine UID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OSType**<String>

Gets desktops by the type of operating system they are running.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OSVersion**<String>

Gets desktops by the version of the operating system they are running.

Required?	false
Default Value	
Accept Pipeline Input?	false

--	--

#### **-PersistUserChanges**<PersistUserChanges>

Gets desktops by the location where the user changes are persisted.

- o OnLocal - User changes are persisted locally.
- o Discard - User changes are discarded.
- o OnPvd - User changes are persisted on the Pvd.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PowerActionPending**<Boolean>

Gets desktops with a specific power action pending state.

Valid values are \$true or \$false.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PowerState**<PowerState>

Gets desktops with a specific power state.

Valid values are Unmanaged, Unknown, Unavailable, Off, On, Suspended, TurningOn, TurningOff, Suspending, and Resuming.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Protocol**<String>

Gets desktops with connections using a specific protocol, for example HDX, RDP, or Console.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ProvisioningType**<ProvisioningType>

Gets desktops that are in a catalog with a particular provisioning type. Values can be:

- o Manual - No provisioning.

o PVS - Machine provisioned by PVS (machine may be physical, blade, VM,...).

o MCS - Machine provisioned by MCS (machine must be VM).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PublishedApplication**<String>

Gets desktops with a specific application published to them.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PublishedName**<String>

Gets desktops with a specific published name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PvdStage**<PvdStage>

Gets desktops with a specific personal vDisk stage.

Valid values are None, Requested, Starting, Working and Failed.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-RegistrationState**<RegistrationState>

Gets desktops with a specific registration state.

Valid values are Unregistered, Initializing, Registered and AgentError.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecureIcaActive**<Boolean>

Gets desktops depending on whether the current session uses SecureICA or not.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecureIcaRequired**<Boolean>

Gets desktops configured with a particular SecureIcaRequired setting. Note that the desktop setting of \$null indicates that the desktop group value is used.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionHidden**<Boolean>

Gets desktops by whether their sessions are hidden or not. Hidden sessions are treated as though they do not exist when launching sessions; a hidden session cannot be reconnected to, but a new session may be launched using the same entitlement.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionId**<Int32>

Deprecated.

Gets desktops by session ID, a unique identifier that Remote Desktop Services uses to track the session but it is only unique on that machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionState**<SessionState>

Gets desktops with a specific session state.

Valid values are \$null, Other, PreparingSession, Connected, Active, Disconnected, Reconnecting, NonBrokeredSession, and Unknown.

Session properties are always null for multi-session desktops.

--	--



Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionStateChangeTime**<DateTime>

Gets desktops whose sessions last changed state at a specific time.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionUid**<Int64>

Gets single-session desktops with a specific session UID (\$null for no session).

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionUserName**<String>

Gets desktops with a specific user name for the current session (in the form 'domain\user').

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionUserSID**<String>

Gets desktops with a specific SID of the current session user.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SID**<String>

Gets desktops with a specific machine SID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-SmartAccessTag**<String>

Gets session desktops where the session has the specific SmartAccess tag.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-StartTime**<DateTime>

Gets desktops with a specific session start time.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-SummaryState**<DesktopSummaryState>

Gets desktops with a specific summary state.

Valid values are Off, Unregistered, Available, Disconnected, InUse and Preparing.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Tag**<String>

Gets desktops with a specific tag.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-WillShutDownAfterUse**<Boolean>

Gets desktops depending on whether they shut down after use or not.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ApplicationUid**<Int32>

Gets desktops with a specific published application (identified by its UID).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each

name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

**-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

None You cannot pipe input into this cmdlet.

Return Values

Citrix.Broker.Admin.SDK.Desktop

Get-BrokerDesktop returns an object for each matching desktop.

Notes

To compare dates or times, use -Filter and relative comparisons. For more information, see about\_Broker\_Filtering and the examples.

Examples

----- **EXAMPLE 1** -----

C:\PS> Get-BrokerDesktop -RegistrationState Unregistered

```
C:\PS> Get-BrokerDesktop -Filter { RegistrationState -ne 'Registered' }
```

Both commands retrieve desktops that are unregistered. The second command also includes desktops with a registration state of AgentError.

----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerDesktop -SessionUid $null | ft -a DNSName,SummaryState
```

Gets desktops without sessions, listing the DNS name and current state.

----- **EXAMPLE 3** -----

```
C:\PS> Get-BrokerDesktop -Filter { OSType -like "Windows XP*" -and ImageOutOfDate }
```

Finds all Windows XP desktops with an out-of-date image.

----- **EXAMPLE 4** -----

```
C:\PS> Get-BrokerDesktop -ApplicationInUse '*powerpoint'
```

Gets desktops running a published PowerPoint application. It matches any application browser name containing the word 'powerpoint'. String comparisons are case-insensitive.

----- **EXAMPLE 5** -----

```
C:\PS> Get-BrokerDesktop -DesktopCondition * DNSName,SessionUserName,DesktopConditions
```

Finds all desktops with an outstanding desktop condition, listing the affected desktop and user.

----- **EXAMPLE 6** -----

```
C:\PS> $d = (Get-Date).AddDays(-1)
```

```
C:\PS> Get-BrokerDesktop -Filter { StartTime -le $d } | ft MachineName,SessionUserName,StartTime,@{Label='Duration'; Expression=((Get-Date) - $_.StartTime)}
```

Finds users who have been logged on for more than a day, and outputs the machine name, start time, and duration the session has been logged on.

# Get-BrokerDesktopGroup

May 10, 2016

Gets broker desktop groups configured for this site.

## Syntax

```
Get-BrokerDesktopGroup [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerDesktopGroup [[-Name] <String>] [-AutomaticPowerOnForAssigned <Boolean>] [-AutomaticPowerOnForAssignedDuringPeak <Boolean>] [-ColorDepth <ColorDepth>] [-DeliveryType <DeliveryType>] [-Description <String>] [-DesktopKind <DesktopKind>] [-Enabled <Boolean>] [-InMaintenanceMode <Boolean>] [-IsRemotePC <Boolean>] [-Metadata <String>] [-MinimumFunctionalLevel <FunctionalLevel>] [-OffPeakBufferSizePercent <Int32>] [-OffPeakDisconnectAction <SessionChangeHostingAction>] [-OffPeakDisconnectTimeout <Int32>] [-OffPeakExtendedDisconnectAction <SessionChangeHostingAction>] [-OffPeakExtendedDisconnectTimeout <Int32>] [-OffPeakLogOffAction <SessionChangeHostingAction>] [-OffPeakLogOffTimeout <Int32>] [-PeakBufferSizePercent <Int32>] [-PeakDisconnectAction <SessionChangeHostingAction>] [-PeakDisconnectTimeout <Int32>] [-PeakExtendedDisconnectAction <SessionChangeHostingAction>] [-PeakExtendedDisconnectTimeout <Int32>] [-PeakLogOffAction <SessionChangeHostingAction>] [-PeakLogOffTimeout <Int32>] [-PublishedName <String>] [-ScopeId <Guid>] [-ScopeName <String>] [-SecureIcaRequired <Boolean>] [-SessionSupport <SessionSupport>] [-SettlementPeriodBeforeAutoShutdown <TimeSpan>] [-ShutdownDesktopsAfterUse <Boolean>] [-Tag <String>] [-TimeZone <String>] [-TotalApplications <Int32>] [-TurnOnAddedMachine <Boolean>] [-UUID <Guid>] [-ApplicationUid <Int32>] [-TagUid <Int32>] [-PowerTimeSchemeUid <Int32>] [-MachineConfigurationUid <Int32>] [-RemotePCCatalogUid <Int32>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieve desktop groups matching the specified criteria. If no parameters are specified this cmdlet enumerates all desktop groups.

Desktop groups represent groups of desktops that are managed together for brokering purposes.

----- BrokerDesktopGroup Object

A desktop group object represents a collection of machines that are fully configured in a site that is able to run either a Microsoft Windows desktop environment, individual applications, or both.

-- AutomaticPowerOnForAssigned (System.Boolean)

Specifies whether assigned desktops in the desktop group are automatically started at the start of peak time periods. Only relevant for groups whose DesktopKind is Private.

-- AutomaticPowerOnForAssignedDuringPeak (System.Boolean)

Specifies whether assigned desktops in the desktop are automatically started throughout peak time periods. Only relevant for groups whose DesktopKind is Private and which have AutomaticPowerOnForAssigned set to true.

-- ColorDepth (Citrix.Broker.Admin.SDK.ColorDepth)

Default color depth of sessions started with machines in the desktop group. Possible values are:

FourBit, EightBit, SixteenBit, TwentyFourBit.

-- ConfigurationSlotUids (System.Int32[])

Uids of any configuration slots which hold machine configurations associated with the desktop group. The order of slot UIDs in this list correspond with the order of items in the associated MachineConfigurationNames and MachineConfigurationUids list properties, and so the same slot UID can appear more than once.

-- DeliveryType (Citrix.Broker.Admin.SDK.DeliveryType)

The type of resources being published. Possible values are:

DesktopsOnly, AppsOnly, DesktopsAndApps.

-- Description (System.String)

Description of the desktop group.

-- DesktopKind (Citrix.Broker.Admin.SDK.DesktopKind)

The kind of the desktops being published, possible values are:

Private and Shared.

-- DesktopsAvailable (System.Int32)

The number of machines in the desktop group in state Available; this is the number of machines with no sessions present.

-- DesktopsDisconnected (System.Int32)

The number of disconnected sessions present on machines in the desktop group.

-- DesktopsInUse (System.Int32)

The number of machines in the desktop group in state InUse; this is the number of machines with at least one session present.

-- DesktopsNeverRegistered (System.Int32)

The number of machines in the desktop group that have never registered with the current site.

-- DesktopsPreparing (System.Int32)

The number of machines in the desktop group whose PvD disk image is being prepared.

-- DesktopsUnregistered (System.Int32)

The number of machines in the desktop group that are currently unregistered.

-- Enabled (System.Boolean)

Specifies whether the desktop group is enabled or not; disabled desktop groups do not appear to users.

-- IconUid (System.Int32)

The Uid of the icon to be used as a default for desktops in the desktop group. Individual desktop objects can override this default by setting the IconUid parameter on the desktop object.

-- InMaintenanceMode (System.Boolean)

Specifies whether the machines in the desktop group are in maintenance mode or not.

-- IsRemotePC (System.Boolean)

Specifies whether the desktop group is a Remote PC desktop group.

-- MachineConfigurationNames (System.String[])

The MachineConfiguration names associated with the desktop group.

-- MachineConfigurationUids (System.Int32[])

The MachineConfiguration uids associated with the desktop group.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

Metadata associated with the desktop group.

-- MinimumFunctionalLevel (Citrix.Broker.Admin.SDK.FunctionalLevel)

The minimum FunctionalLevel required for the machines in the desktop group to be able to register with the Citrix Broker Service.

-- Name (System.String)

Name of the desktop group.

-- OffPeakBufferSizePercent (System.Int32)

The percentage of machines that are kept available in an idle state outside peak hours.

-- OffPeakDisconnectAction (Citrix.Broker.Admin.SDK.SessionChangeHostingAction)

The action that is performed after a configurable period of a user session disconnecting outside peak hours. Possible values are Nothing, Suspend or Shutdown.

-- OffPeakDisconnectTimeout (System.Int32)

The number of minutes before the configured action is performed after a user session disconnects outside peak hours.

-- OffPeakExtendedDisconnectAction (Citrix.Broker.Admin.SDK.SessionChangeHostingAction)

The action performed after a second configurable period of a user session disconnecting outside peak hours. Possible values are Nothing, Suspend, or Shutdown.

-- OffPeakExtendedDisconnectTimeout (System.Int32)

The number of minutes before the second configured action is performed after a user session disconnects outside peak hours.



-- OffPeakLogOffAction (Citrix.Broker.Admin.SDK.SessionChangeHostingAction)

The action performed after a configurable period of a user session ending outside peak hours. Possible values are Nothing, Suspend, or Shutdown.

-- OffPeakLogOffTimeout (System.Int32)

The number of minutes before the configured action is performed after a user session ends outside peak hours.

-- PeakBufferSizePercent (System.Int32)

The percentage of machines in the desktop group that are kept available in an idle state in peak hours.

-- PeakDisconnectAction (Citrix.Broker.Admin.SDK.SessionChangeHostingAction)

The action performed after a configurable period of a user session disconnecting in peak hours. Possible values are Nothing, Suspend, or Shutdown.

-- PeakDisconnectTimeout (System.Int32)

The number of minutes before the configured action is performed after a user session disconnects in peak hours.

-- PeakExtendedDisconnectAction (Citrix.Broker.Admin.SDK.SessionChangeHostingAction)

The action performed after a second configurable period of a user session disconnecting in peak hours. Possible values are Nothing, Suspend, or Shutdown.

-- PeakExtendedDisconnectTimeout (System.Int32)

The number of minutes before the second configured action is performed after a user session disconnects in peak hours.

-- PeakLogOffAction (Citrix.Broker.Admin.SDK.SessionChangeHostingAction)

The action performed after a configurable period of a user session ending in peak hours. Possible values are Nothing, Suspend, or Shutdown.

-- PeakLogOffTimeout (System.Int32)

The number of minutes before the configured action is performed after a user session ends in peak hours.

-- ProtocolPriority (System.String[])

A list of protocol names in the order in which they are attempted for use during connection.

-- PublishedName (System.String)

The name of the desktop group as it is to appear to the user in StoreFront.

-- Scopes (Citrix.Broker.Admin.SDK.ScopeReference[])

The list of the delegated admin scopes to which the desktop group belongs.

-- SecureIcaRequired (System.Boolean)

Flag that specifies if the SecureICA encryption of the HDX protocol is required for sessions of desktops in the desktop

group.

-- Sessions (System.Int32)

The total number of user sessions currently running on all of the machines in the desktop group.

-- SessionSupport (Citrix.Broker.Admin.SDK.SessionSupport)

Specifies the session support (single/multi) of the machines in the desktop group. Machines with the incorrect session support for the desktop group will be unable to register with the Citrix Broker Service.

-- SettlementPeriodBeforeAutoShutdown (System.TimeSpan)

Time after a session ends during which automatic shutdown requests (for example, shutdown after use, idle pool management) are deferred. Any outstanding shutdown request takes effect after the settlement period expires. This is typically used to configure time to allow logoff scripts to complete.

-- ShutdownDesktopsAfterUse (System.Boolean)

Specifies if the desktops will shut down after they have been used and there are no sessions running on the machine. The machines will not shut down if they are placed into maintenance mode, even if this flag is set to \$true. The machines, however, will shutdown after the machine is taken out of maintenance mode if the flag is still set.

-- Tags (System.String[])

Tags associated with the desktop group.

-- TimeZone (System.String)

The timezone that desktops in the desktop group are in (for power policy purposes).

-- TotalApplications (System.Int32)

Total number of applications associated with the desktop group.

-- TotalDesktops (System.Int32)

Total number of machines in the desktop group.

-- TurnOnAddedMachine (System.Boolean)

Specifies whether the broker should attempt to turn on power-managed machines when they are added to the desktop group.

-- Uid (System.Int32)

Uid of the desktop group.

-- UUID (System.Guid)

UUID of the desktop group.

Related topics

[New-BrokerDesktopGroup](#)

[Set-BrokerDesktopGroup](#)

[Rename-BrokerDesktopGroup](#)

[Remove-BrokerDesktopGroup](#)

[Add-BrokerUser](#)

[Add-BrokerTag](#)

## Parameters

**-Uid**<Int32>

Gets desktop groups with the specified value of Uid.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Name**<String>

Gets desktop groups whose name matches the supplied pattern.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AutomaticPowerOnForAssigned**<Boolean>

Gets only desktop groups with the specified value of AutomaticPowerOnForAssigned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AutomaticPowerOnForAssignedDuringPeak**<Boolean>

Gets only desktop groups with the specified value of AutomaticPowerOnForAssignedDuringPeak.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ColorDepth**<ColorDepth>

Gets only desktop groups with the specified color depth.

Valid values are FourBit, EightBit, SixteenBit, and TwentyFourBit.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DeliveryType**<DeliveryType>

Gets desktop groups according to their delivery type.

Valid values are DesktopsOnly, AppsOnly and DesktopsAndApps.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Description**<String>

Gets desktop groups whose description matches the supplied pattern.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopKind**<DesktopKind>

Gets desktops of a particular kind.

Valid values are Private and Shared.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Enabled**<Boolean>

Gets desktop groups with the specified value of Enabled.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-InMaintenanceMode**<Boolean>

Gets desktop groups with the specified value of InMaintenanceMode.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IsRemotePC**<Boolean>

Gets desktop groups with the specified IsRemotePC value.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MinimumFunctionalLevel**<FunctionalLevel>

Gets desktop groups with a specific MinimumFunctionalLevel.

Valid values are L5, L7, L7\_6

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OffPeakBufferSizePercent**<Int32>

Gets desktop groups with the specified value of OffPeakBufferSizePercent.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OffPeakDisconnectAction**<SessionChangeHostingAction>

Gets desktop groups with the specified value of OffPeakDisconnectAction.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OffPeakDisconnectTimeout<Int32>**

Gets desktop groups with the specified value of OffPeakDisconnectTimeout.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OffPeakExtendedDisconnectAction<SessionChangeHostingAction>**

Gets desktop groups with the specified value of OffPeakExtendedDisconnectAction.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OffPeakExtendedDisconnectTimeout<Int32>**

Gets desktop groups with the specified value of OffPeakExtendedDisconnectTimeout.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OffPeakLogOffAction<SessionChangeHostingAction>**

Gets desktop groups with the specified value of OffPeakLogOffAction.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OffPeakLogOffTimeout<Int32>**

Gets desktop groups with the specified value of OffPeakLogOffTimeout.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PeakBufferSizePercent**<Int32>

Gets desktop groups with the specified value of PeakBufferSizePercent.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PeakDisconnectAction**<SessionChangeHostingAction>

Gets desktop groups with the specified value of PeakDisconnectAction.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PeakDisconnectTimeout**<Int32>

Gets desktop groups with the specified value of PeakDisconnectTimeout.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PeakExtendedDisconnectAction**<SessionChangeHostingAction>

Gets desktop groups with the specified value of PeakExtendedDisconnectAction.



Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PeakExtendedDisconnectTimeout<Int32>**

Gets desktop groups with the specified value of PeakExtendedDisconnectTimeout.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PeakLogOffAction<SessionChangeHostingAction>**

Gets desktop groups with the specified value of PeakLogOffAction.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PeakLogOffTimeout<Int32>**

Gets desktop groups with the specified value of PeakLogOffTimeout.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PublishedName<String>**

Gets desktop groups whose published name matches the supplied pattern.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-ScopeId**<Guid>

Gets desktop groups that are associated with the given scope identifier.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ScopeName**<String>

Gets desktop groups that are associated with the given scope name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecureIcaRequired**<Boolean>

Gets desktop groups with the specified value of SecureIcaRequired.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionSupport**<SessionSupport>

Gets desktop groups that have the specified session capability. Values can be:

- o SingleSession - Single-session only machine.
- o MultiSession - Multi-session capable machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SettlementPeriodBeforeAutoShutdown**<TimeSpan>

Gets desktop groups with the specified value of SettlementPeriodBeforeAutoShutdown.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ShutdownDesktopsAfterUse**<Boolean>

Gets desktop groups with the specified value of ShutdownDesktopsAfterUse.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Tag**<String>

Gets desktop groups tagged with the specified tag.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-TimeZone**<String>

Gets desktop groups with the specified value of TimeZone.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-TotalApplications<Int32>**

Gets desktop groups that are acting as delivery groups for the specified number of applications.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-TurnOnAddedMachine<Boolean>**

Gets desktop groups with the specified value of TurnOnAddedMachine value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UUID<Guid>**

Gets desktop groups with the specified value of UUID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ApplicationUid<Int32>**

Gets desktop groups that publish the specified application (identified by Uid)

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-TagUid<Int32>**

Gets desktop groups to which the specified tag (identified by its Uid) has been added to help identify it - see Add-BrokerTag for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PowerTimeSchemeUid<Int32>**

Gets desktop groups associated with the specified power time scheme (identified by its Uid).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineConfigurationUid<Int32>**

Gets desktop groups with the specified value of MachineConfiguration.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RemotePCCatalogUid<Int32>**

Gets Remote PC desktop groups associated with the specified catalog.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-Sort By**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
-----------	-------

Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through `Select-Object`, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.DesktopGroup

Get-BrokerDesktopGroup returns an object for each matching desktop group.

## Notes

To perform greater-than or less-than comparisons, use -Filter. For more information, see about\_Broker\_Filtering and the examples.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Get-BrokerDesktopGroup -PublishedName EMEA*
```

Finds all desktop groups with published names starting with "EMEA".

### ----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerDesktopGroup -InMaintenanceMode $true
```

Finds all desktop groups in maintenance mode.



# Get-BrokerDesktopUsage

May 10, 2016  
Get usage history of desktop groups.

## Syntax

Get-BrokerDesktopUsage [-DesktopGroupName <String>] [-DesktopGroupUid <Int32>] [-InUse <Int32>] [-Timestamp <DateTime>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Returns information, recorded by the broker on an hourly basis, about the number of desktops in use for each desktop group. Analyzing the historical usage records can give some guidance on usage patterns and help choosing idle pool settings.

Without parameters, Get-BrokerDesktopUsage returns the first 250 records. By using parameters, you can be more selective about the records that are returned.

To retrieve more than the default 250 records, use the -MaxRecordCount parameter. To select data for a specific desktop group, use either the -DesktopGroupName or -DesktopGroupUid parameters.

See the examples for this cmdlet and about\_Broker\_Filtering for details of how to perform advanced filtering.

----- BrokerDesktopUsage Object

Desktop usage object contains information to tell how many desktops in a desktop group are in use at a given time (identified by a timestamp).

-- DesktopGroupUid (System.Int32)

Uid of the desktop group that the usage data corresponds to.

-- InUse (System.Int32)

Specifies how many desktop are in use at the time the timestamp corresponds to.

-- Timestamp (System.DateTime)

Date and time the desktop usage information corresponds to.

## Related topics

## Parameters

**-DesktopGroupName**<String>

Gets usage records for the named desktop group or for multiple desktop groups if wildcards have been specified.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopGroupUid**<Int32>

Gets usage records for a specific desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-InUse**<Int32>

Gets usage records where the in-use count matches the specified value. This is useful when checking for zero or when used inside a -Filter expression.

Required?	false

Default Value	
Accept Pipeline Input?	false

#### **-Timestamp**<DateTime>

Gets usage records that occurred at the given time.

In general, Citrix recommends, using -Filter and relative comparisons. For a demonstration, see the examples.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See `about_Broker_Filtering` for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

**-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

None You cannot pipe objects to this cmdlet.

Return Values

Citrix.Broker.Admin.SDK.DesktopUsage

Get-BrokerDesktopUsage returns an object for each matching record.

Notes

Desktop usage information is automatically deleted after 7 days.

Examples

----- **EXAMPLE 1** -----

C:\PS> Get-BrokerDesktopUsage -DesktopGroupName TestGroup -MaxRecordCount 24 -SortBy 'Timestamp' | ft -a @{Name='Time';Expression='{0:t}' -f \$\_.Timestamp}},InUse  
Returns the last 24 hours of usage information for desktop group TestGroup, formatting it as two columns labeled Time and InUse.

----- **EXAMPLE 2** -----

C:\PS> \$d = Get-Date -Hour 0 -Minute 0 -Second 0  
C:\PS> Get-BrokerDesktopUsage -Filter { DesktopGroupName -eq 'TestGroup' -and Timestamp -ge \$d }  
Returns today's usage information for desktop group TestGroup.

----- **EXAMPLE 3** -----

C:\PS> \$dg = Get-BrokerDesktopGroup TestGroup  
C:\PS> Get-BrokerDesktopUsage -DesktopGroupUid \$dg.Uid | Select Timestamp,InUse,@{Name='Percent';Expression='{0:P0}' -f (\$\_.InUse / \$dg.TotalDesktops)}}  
Retrieves the usage history for desktop group TestGroup and adds a column showing the number of desktops in that group in use, as a percentage.

# Get-BrokerEntitlementPolicyRule

May 10, 2016

Gets desktop rules from the site's entitlement policy.

## Syntax

```
Get-BrokerEntitlementPolicyRule [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerEntitlementPolicyRule [[-Name] <String>] [-BrowserName <String>] [-ColorDepth <ColorDepth>] [-Description <String>] [-DesktopGroupUid <Int32>] [-Enabled <Boolean>] [-ExcludedUser <User>] [-ExcludedUserFilterEnabled <Boolean>] [-IconUid <Int32>] [-IncludedUser <User>] [-IncludedUserFilterEnabled <Boolean>] [-Metadata <String>] [-PublishedName <String>] [-SecureIcaRequired <Boolean>] [-SessionReconnection <SessionReconnection>] [-UUID <Guid>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns desktop rules matching the specified search criteria from the site's entitlement policy. If no search criteria are specified, all desktop rules in the entitlement policy are obtained.

A desktop rule in the entitlement policy defines the users who are allowed per-session access to a machine from the rule's associated desktop group to run a full desktop session.

----- BrokerEntitlementPolicyRule Object

The BrokerEntitlementPolicyRule object represents a single desktop rule within the site's entitlement policy. It contains the following properties:

-- BrowserName (System.String)

Site-wide unique name identifying this desktop entitlement to other components (for example StoreFront).

-- ColorDepth (Citrix.Broker.Admin.SDK.ColorDepth?)

The color depth of any desktop session launched by the user from the entitlement. If null, the equivalent setting from the rule's desktop group is used.

-- Description (System.String)

Optional description of the rule. The text may be visible to the end user, for example, as a tooltip associated with the desktop entitlement.

-- DesktopGroupUid (System.Int32)

The unique ID of the desktop group to which the rule applies.

-- Enabled (System.Boolean)

Indicates whether the rule is enabled. A disabled rule is ignored when evaluating the site's entitlement policy.

-- ExcludedUserFilterEnabled (System.Boolean)

Indicates whether the excluded users filter is enabled. If the filter is disabled then any user entries in the filter are ignored when entitlement policy rules are evaluated.

-- ExcludedUsers (Citrix.Broker.Admin.SDK.ChbUser[])

The excluded users filter of the rule, that is, the users and groups who are explicitly denied an entitlement to a desktop session from this rule.

-- IconUid (System.Int32?)

The unique ID of the icon used to display the desktop entitlement to the user. If null, the equivalent setting from the rule's desktop group is used.

-- IncludedUserFilterEnabled (System.Boolean)

Indicates whether the included users filter is enabled. If the filter is disabled then any user who satisfies the requirements of the access policy is implicitly granted an entitlement to a desktop session by the rule.

-- IncludedUsers (Citrix.Broker.Admin.SDK.ChbUser[])

The included users filter of the rule, that is, the users and groups who are granted an entitlement to a desktop session by the rule.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

A collection of arbitrary key/value pairs that can be associated with the rule. The administrator can use these values for any purpose; they are not used by the site itself in any way.

-- Name (System.String)

The administrative name of the rule. Each rule in the site's entitlement policy must have a unique name (irrespective of whether they are desktop or application rules).

-- PublishedName (System.String)

The name of the desktop session entitlement as seen by the user. If null, the equivalent setting from the rule's desktop group is used.

-- SecureIcaRequired (System.Boolean?)

Indicates whether the rule requires the SecureICA protocol for desktop sessions launched using the entitlement. If null, the equivalent setting from the rule's desktop group is used.

-- SessionReconnection (Citrix.Broker.Admin.SDK.SessionReconnection)

Defines reconnection (roaming) behavior for sessions launched using this rule. Session reconnection control is an experimental and unsupported feature.

-- Uid (System.Int32)

The unique ID of the rule itself.

-- UUID (System.Guid)

UUID of the rule.

## Related topics

[New-BrokerEntitlementPolicyRule](#)

[Set-BrokerEntitlementPolicyRule](#)

[Rename-BrokerEntitlementPolicyRule](#)

[Remove-BrokerEntitlementPolicyRule](#)

## Parameters

**-Uid**<Int32>

Gets the desktop rule with the specified unique ID.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Name**<String>

Gets only desktop rules with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-BrowserName**<String>

Gets only desktop rules with browser names matching the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ColorDepth**<ColorDepth>

Gets only desktop rules with the specified color depth.

Valid values are \$null, FourBit, EightBit, SixteenBit, and TwentyFourBit.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Description<String>**

Gets only desktop rules with the specified description.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopGroupUid<Int32>**

Gets only desktop rules that apply to the desktop group with the specified unique ID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Enabled<Boolean>**

Gets only desktop rules that are in the specified state, either enabled (\$true), or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ExcludedUser<User>**

Gets only desktop rules that have the specified user in their excluded users filter (whether the filter is enabled or not).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ExcludedUserFilterEnabled**<Boolean>

Gets only desktop rules that have their excluded user filter enabled (\$true) or disabled (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IconUid**<Int32>

Gets only desktop rules using the icon with the specified unique ID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IncludedUser**<User>

Gets only desktop rules that have the specified user in their included users filter (whether the filter is enabled or not).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IncludedUserFilterEnabled**<Boolean>

Gets only desktop rules that have their included user filter enabled (\$true) or disabled (\$false).



Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PublishedName**<String>

Gets only desktop rules with the specified published name, that is, the desktop session entitlement name that the end user sees.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecureIcaRequired**<Boolean>

Gets only desktop rules that require the desktop session to use the SecureICA protocol (\$true) or not (\$false).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionReconnection**<SessionReconnection>

Gets only desktop rules with the specified session reconnection behavior.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-UUID**<Guid>

Gets rules with the specified value of UUID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

**-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

**-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

**-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

Citrix.Broker.Admin.SDK.EntitlementPolicyRule

Get-BrokerEntitlementPolicyRule returns all desktop entitlement policy rules that match the specified selection criteria.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Get-BrokerEntitlementPolicyRule
```

Returns all desktop rules from the entitlement policy. This offers a complete description of the current site's entitlement policy with respect to desktops published from shared desktop groups.

### ----- **EXAMPLE 2** -----

```
C:\PS> $dg = Get-BrokerDesktopGroup 'Customer Support'
```

```
C:\PS> Get-BrokerEntitlementPolicyRule -DesktopGroupUid $dg.Uid
```

Returns all desktop rules in the entitlement policy that give users entitlements to desktop sessions in the Customer Support desktop group.

# Get-BrokerHostingPowerAction

May 10, 2016

Gets power actions queued for machines.

## Syntax

```
Get-BrokerHostingPowerAction [-Uid] <Int64> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerHostingPowerAction [[-MachineName] <String>] [-Action <PowerManagementAction>] [-ActionCompletionTime
<DateTime>] [-ActionStartTime <DateTime>] [-ActualPriority <Int32>] [-BasePriority <Int32>] [-DNSName <String>] [-
FailureReason <String>] [-HostedMachineName <String>] [-HypervisorConnectionName <String>] [-
HypervisorConnectionUid <Int32>] [-Metadata <String>] [-RequestTime <DateTime>] [-State <PowerActionState>] [-
ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>]
[-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Finds power actions matching the specified search criteria from the queue of all known power actions. These power actions can be waiting to be dealt with or can be part way through being processed by the relevant hypervisor, or they can be recently completed actions. Completed actions are removed from the queue after a configured period, the default being one hour.

If no search criteria are specified all actions in the queue are obtained.

A Hosting Power Action record defines the action that is to be performed or has been performed, the machine that the action is to be applied to, the priority of the action in relation to other actions in the queue, times for points in the life of the action, and any results if the action has completed.

For a detailed description of the queuing mechanism, see 'help about\_Broker\_PowerManagement'.

----- BrokerHostingPowerAction Object

The BrokerHostingPowerAction object represents an instance of a power action. It contains the following properties:

-- Action (Citrix.Broker.Admin.SDK.PowerManagementAction)

The power action to be performed. Possible values are: TurnOn, TurnOff, Shutdown, Reset, Restart, Suspend, Resume.

-- ActionCompletionTime (System.DateTime?)

The time when the power action was completed by the hypervisor connection.

-- ActionStartTime (System.DateTime?)

The time when the power action was started to be processed by the hypervisor.

-- ActualPriority (System.Int32)

The current priority of the operation after any priority boosting.

-- BasePriority (System.Int32)

The starting priority of the operation.

-- `DNSName` (System.String)

The fully qualified DNS name of the machine that the power action applies to.

-- `FailureReason` (System.String)

For failed power actions, an indication of the reason for the failure.

-- `HostedMachineName` (System.String)

The hypervisor's name for the machine that the power action applies to.

-- `HypervisorConnectionUid` (System.Int32)

The unique identifier of the hypervisor connection that is associated with the target machine.

-- `MachineName` (System.String)

The name of the machine that the power action applies to, in the form `domain\machine`.

-- `MetadataMap` (System.Collections.Generic.Dictionary<string, string>)

Metadata for this power action.

-- `RequestTime` (System.DateTime)

The timestamp of when the action was created and placed in the queue.

-- `State` (Citrix.Broker.Admin.SDK.PowerActionState)

The current state of this power action. Possible values are: Pending, Started, Completed, Failed, Canceled, Deleted, Lost.

-- `Uid` (System.Int64)

The unique identifier of the power action.

## Related topics

[New-BrokerHostingPowerAction](#)

[Set-BrokerHostingPowerAction](#)

[Remove-BrokerHostingPowerAction](#)

## Parameters

**-Uid**<Int64>

Gets only the single action record whose ID matches the specified value.

Required?	true
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-MachineName**<String>

Gets only the records for actions that are for machines whose name (of the form domain\machine) matches the specified string.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Action**<PowerManagementAction>

Gets only action records with the specified action type.

Valid values are TurnOn, TurnOff, ShutDown, Reset, Restart, Suspend and Resume.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ActionCompletionTime**<DateTime>

Gets only action records reported as having completed successfully at the specified time. This is useful with advanced filtering; for more information, see [about\\_Broker\\_Filtering](#).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ActionStartTime**<DateTime>

Gets only action records reported as starting to be processed by the relevant hypervisor at the specified time. This is useful with advanced filtering; for more information, see [about\\_Broker\\_Filtering](#).

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-ActualPriority<Int32>**

Gets only the records for actions whose current active priority matches the specified value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-BasePriority<Int32>**

Gets only the records for actions whose original priority matches the specified value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DNSName<String>**

Gets only the records for actions that are for machines whose DNS name matches the specified string.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-FailureReason<String>**

Gets only the records for actions that have failed and whose failure reason string matches the specified string.

Required?	false



Default Value	
Accept Pipeline Input?	false

#### **-HostedMachineName**<String>

Gets only the records for actions that are for machines whose Hosting Name (the machine name as understood by the hypervisor) matches the specified string.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionName**<String>

Gets only the records for actions for machines hosted via a hypervisor connection whose name matches the specified string.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionUid**<Int32>

Gets only the records for actions for machines hosted via a hypervisor connection whose ID matches the specified value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RequestTime**<DateTime>

Gets only the records for actions created and added to the queue at the specified time. This is useful with advanced filtering; for more information, see [about\\_Broker\\_Filtering](#).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-State**<PowerActionState>

Gets only the records for actions with the specified current state.

Valid values are Pending, Started, Completed, Failed, Canceled, Deleted and Lost.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

**-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

**-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

**-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### **Input Type**

None You cannot pipe input into this cmdlet.

### **Return Values**

Citrix.Broker.Admin.SDK.HostingPowerAction

Get-BrokerHostingPowerAction returns all power actions that match the specified selection criteria.

### **Examples**

#### **----- EXAMPLE 1 -----**

```
C:\PS> Get-BrokerHostingPowerAction
```

Fetches records for all known power actions either waiting to be processed, or currently being processed, or which have been processed in the last hour.

#### **----- EXAMPLE 2 -----**

```
C:\PS> Get-BrokerHostingPowerAction -State Pending -HypervisorConnectionName 'XenPool5'
```

Fetches records for all power actions that are waiting to be processed and where the action is for a virtual machine that is hosted by the hypervisor called 'XenPool5'.

# Get-BrokerHypervisorAlert

May 10, 2016

Gets hypervisor alerts recorded by the controller.

## Syntax

```
Get-BrokerHypervisorAlert -Uid <Int64> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerHypervisorAlert [-HostingServerName <String>] [-HypervisorConnectionUid <Int32>] [-Metadata <String>] [-Metric
<AlertMetric>] [-Severity <AlertSeverity>] [-Time <DateTime>] [-TriggerInterval <TimeSpan>] [-TriggerLevel <Double>] [-
TriggerPeriod <TimeSpan>] [-TriggerValue <Double>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>]
[-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Get-BrokerHypervisorAlert cmdlet gets alert objects reported by the hypervisors that the controller is monitoring.

Without parameters, Get-BrokerHypervisorAlert gets all of the alerts recorded. Use parameters to select which alerts are returned.

Once you have configured suitable alerts in your hypervisor, and configured the controller with your hypervisor details (see New-BrokerHypervisorConnection), the controller monitors each hypervisor for new alerts.

Four hypervisor alert metrics are recorded; these relate to the hypervisor host, not individual virtual machines:

- Cpu: Reports excessive CPU usage.
- Memory: Reports excessive memory usage.
- Network: Reports high network activity.
- Disk: Reports high disk activity.

Each alert also includes information about where and when the alert occurred, the severity of the alert (Red/Yellow), and the configuration of the triggered alert.

The following metrics are supported with these hypervisors:

- VMware ESX (Cpu, Memory, Network, Disk)
- Citrix XenServer (Cpu, Network)
- Microsoft Hyper-V (None)

----- BrokerHypervisorAlert Object

The BrokerHypervisorAlert represents a hypervisor alert object. It contains the following properties:

- HostingServerName (System.String)

The name of the hypervisor hosting this machine.

-- HypervisorConnectionUid (System.Int32)

The Uid of the hypervisor connection that reported this alert.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

Metadata for this hypervisor alert.

-- Metric (Citrix.Broker.Admin.SDK.AlertMetric)

The metric this alert relates to: Cpu, Memory, Network or Disk.

-- Severity (Citrix.Broker.Admin.SDK.AlertSeverity)

Severity of the alert (Red or Yellow). Red is more serious than Yellow.

-- Time (System.DateTime)

Time that the alert occurred.

-- TriggerInterval (System.TimeSpan?)

Number of ticks (100ns) before the alert can be raised again.

-- TriggerLevel (System.Double?)

Threshold level that the alert was configured to trigger at.

-- TriggerPeriod (System.TimeSpan?)

Duration the value was above the trigger level.

-- TriggerValue (System.Double?)

The value of the monitored metric that triggered the alert.

-- Uid (System.Int64)

The unique internal identifier of this alert.

## Related topics

[New-BrokerHypervisorConnection](#)

## Parameters

**-Uid<Int64>**

Gets the hypervisor alert with the specified UID.

Required?	true
Default Value	

Accept Pipeline Input?	false
------------------------	-------

**-HostingServerName<String>**

Gets alerts for the specified hosting hypervisor server.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-HypervisorConnectionUid<Int32>**

Gets alerts for the specified hypervisor connection.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Metadata<String>**

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Metric<AlertMetric>**

Gets alerts for a specified metric.

Valid values are: Cpu, Memory, Network and Disk.

Required?	false

Default Value	
Accept Pipeline Input?	false

### **-Severity**<AlertSeverity>

Gets alerts with the specified severity.

Valid values are: Red and Yellow.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Time**<DateTime>

Gets alerts that occurred at a specific time.

You can also use -Filter and relative comparisons; see the examples for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-TriggerInterval**<TimeSpan>

Gets alerts with a specific trigger interval. This is the interval before the alert is raised again.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-TriggerLevel**<Double>

Gets alerts with a specific trigger threshold level.

Required?	false
-----------	-------



Default Value	
Accept Pipeline Input?	false

#### **-TriggerPeriod**<TimeSpan>

Gets alerts with a specific trigger period. This is the duration the threshold level was exceeded for, prior to the alert triggering.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-TriggerValue**<Double>

Gets the value of the monitored metric that triggered the alert.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

--	--

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip<Int32>**

Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy<String>**

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter<String>**

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the

properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe objects to this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.HypervisorAlert

Get-BrokerHypervisorAlert returns an object for each matching alert.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerHypervisorAlert -HostingServerName TestHyp* -Severity Red
```

Returns all serious (Red) alerts for any hosting server with a name that starts with 'TestHyp'.

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerHypervisorAlert -Filter { Metric -eq 'Cpu' -and Time -ge '-1:0' }
```

Returns all CPU usage alerts that occurred in the last hour.

# Get-BrokerHypervisorConnection

May 10, 2016

Gets hypervisor connections matching the specified criteria.

## Syntax

```
Get-BrokerHypervisorConnection [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerHypervisorConnection [-Name] <String> [-HypHypervisorConnectionUid <Guid>] [-IsReady <Boolean>] [-MachineCount <Int32>] [-MaxAbsoluteActiveActions <Int32>] [-MaxAbsoluteNewActionsPerMinute <Int32>] [-MaxAbsolutePvdPowerActions <Int32>] [-MaxPercentageActiveActions <Int32>] [-MaxPvdPowerActionsPercentageOfDesktops <Int32>] [-Metadata <String>] [-PreferredController <String>] [-State <HypervisorConnectionState>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Get-BrokerHypervisorConnection cmdlet gets hypervisor connections matching the specified criteria. If no parameters are specified this cmdlet enumerates all hypervisor connections.

----- BrokerHypervisorConnection Object

The BrokerHypervisorConnection represents hypervisor connection object. It contains the following properties:

-- Capabilities (System.String[])

The set of capabilities as reported by the hypervisor.

-- HypHypervisorConnectionUid (System.Guid)

The Guid that identifies the hypervisor connection.

-- IsReady (System.Boolean)

Indicates that the connection is ready to be used in the configuration of managed machines.

-- MachineCount (System.Int32)

Count of machines associated with this hypervisor connection.

-- MaxAbsoluteActiveActions (System.Int32?)

Maximum number of active power actions allowed at any one time (defined in the metadata named 'Citrix\_Broker\_MaxAbsoluteActiveActions' on the hypervisor connection in the Citrix Hosting Service).

-- MaxAbsoluteNewActionsPerMinute (System.Int32?)

Maximum number of new actions that can be fired off to the hypervisor in any one minute (defined in the metadata named 'Citrix\_Broker\_MaxAbsoluteNewActionsPerMinute' on the hypervisor connection in the Citrix Hosting Service).

-- MaxAbsolutePvdPowerActions (System.Int32?)

Maximum number of active Pvd power actions allowed at any one time (defined in the metadata named 'Citrix\_Broker\_MaxAbsolutePvdPowerActions' on the hypervisor connection in the Citrix Hosting Service).

-- MaxPercentageActiveActions (System.Int32?)

Maximum percentage of machines on the connection that can have active power actions at any one time (defined in the metadata named 'Citrix\_Broker\_MaxPowerActionsPercentageOfDesktops' on the hypervisor connection in the Citrix Hosting Service).

-- MaxPvdPowerActionsPercentageOfDesktops (System.Int32?)

Maximum percentage of machines on the connection that can be in personal VDisk image preparation mode at any one time (defined in the metadata named 'Citrix\_Broker\_MaxPvdPowerActionsPercentageOfDesktops' on the hypervisor connection in the Citrix Hosting Service).

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

Collection of all the metadata associated to the hypervisor connection.

-- Name (System.String)

The display name of the hypervisor connection.

-- PreferredController (System.String)

The name of the controller which is preferred to be used, when available, to perform all communication to the hypervisor. The name is in DOMAIN\machine form. A preferred controller may have been automatically chosen when the hypervisor connection was created.

-- State (Citrix.Broker.Admin.SDK.HypervisorConnectionState)

The state of the hypervisor connection.

-- Uid (System.Int32)

Unique internal identifier of hypervisor connection.

Related topics

[New-BrokerHypervisorConnection](#)

[Remove-BrokerHypervisorConnection](#)

[Set-BrokerHypervisorConnection](#)

Parameters

**-Uid<Int32>**

Gets the hypervisor connection with the specified internal id.

Required?	true
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-Name<String>**

Gets hypervisor connections with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionUid<Guid>**

Gets hypervisor connections with the specified Guid.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IsReady<Boolean>**

Gets hypervisor connections with the specified value of the IsReady flag.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineCount<Int32>**

Gets hypervisor connections with the specified machine count.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-MaxAbsoluteActiveActions<Int32>**

Gets hypervisor connections with the specified MaxAbsoluteActiveActions value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxAbsoluteNewActionsPerMinute<Int32>**

Gets hypervisor connections with the specified MaxAbsoluteNewActionsPerMinute value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxAbsolutePvdPowerActions<Int32>**

Gets hypervisor connections with the specified MaxAbsolutePvdPowerActions value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxPercentageActiveActions<Int32>**

Gets hypervisor connections with the specified MaxPercentageActiveActions value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxPvdPowerActionsPercentageOfDesktops<Int32>**

Gets hypervisor connections with the specified MaxPvdPowerActionsPercentageOfDesktops value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Metadata<String>**

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PreferredController<String>**

Gets hypervisor connections with the specified preferred controller. Specify the SAM name of the controller.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-State<HypervisorConnectionState>**

Gets hypervisor connections with the specified connection state. Values can be can be:

- o Unavailable - The broker is unable to contact the hypervisor.
- o InMaintenanceMode - The hosting server is in maintenance mode.
- o On - The broker is in contact with the hypervisor.

Required?	false
-----------	-------



Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if

no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

**-Filter**<String>

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through `Select-Object`, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

None

## Return Values

Citrix.Broker.Admin.SDK.HypervisorConnection

Get-BrokerHypervisorConnection returns an object for each matching hypervisor connection.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS> $hvConn = Get-BrokerHypervisorConnection -Name "hvConnectionName"
```

Gets a hypervisor connection by name.

### ----- EXAMPLE 2 -----

```
c:\PS> $hvConn = Get-BrokerHypervisorConnection -PreferredController "domainName\controllerName"
```

Gets hypervisor connections by preferred controller.

### ----- EXAMPLE 3 -----

```
c:\PS> $machine = Get-BrokerMachine -Uid $machineUid
```

```
c:\PS> $hvConn = Get-BrokerHypervisorConnection -Uid $machine.HypervisorConnectionUid
```

Gets hypervisor connection used by a (power managed) machine.

# Get-BrokerIcon

May 10, 2016

Get stored icons.

## Syntax

```
Get-BrokerIcon -Uid <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerIcon [-Metadata <String>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>]
[-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Reads a specific icon by Uid, or enumerates icons by passing no Uid.

----- BrokerIcon Object

The BrokerIcon object represents a single instance of an icon. It contains the following properties:

-- EncodedIconData (System.String)

The Base64 encoded .ICO format of the icon.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

Metadata for this command

-- Uid (System.Int32)

The UID of the icon itself.

## Related topics

[New-BrokerIcon](#)

[Remove-BrokerIcon](#)

## Parameters

**-Uid**<Int32>

Gets only the icon specified by unique identifier.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

**-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0

Accept Pipeline Input?	false
------------------------	-------

### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through `Select-Object`, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

None Input cannot be piped to this cmdlet.

## Return Values

Citrix.Broker.Admin.SDK.Icon

Returns an Icon object for each enumerated icon.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerIcon
```

Enumerate all icons.

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerIcon -Uid 1
```

Get the icon with Uid 1.

# Get-BrokerImportedFTA

May 10, 2016

Gets the imported file type associations.

## Syntax

```
Get-BrokerImportedFTA [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerImportedFTA [[-ExtensionName] <String>] [-ContentType <String>] [-Description <String>] [-DesktopGroupUid
<Int32>] [-Edit <String>] [-EditArguments <String>] [-EditExecutableName <String>] [-HandlerName <String>] [-Open <String>]
[-OpenArguments <String>] [-OpenExecutableName <String>] [-PerceivedType <String>] [-Print <String>] [-PrintTo <String>] [-
ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>]
[-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the file type associations the system imports from worker machines.

File type association associates a file extension (such as ".txt") with an application (such as Notepad). In a Citrix environment file type associations on a user device can be configured so that when an user clicks on a document it launches the appropriate published application. This is known as "content redirection".

Imported file type associations are different from configured file type associations. Imported file type associations are lists of known file type associations for a given desktop group. Configured file type associations are those that are actually associated with published applications for the purposes of content redirection.

Initially the system is not aware of any extensions, and they must be imported by the Citrix administrator. See the Update-BrokerImportedFTA cmdlet for more information.

After file type extensions are imported, this cmdlet lets the administrator review which file type associations the system is aware of. ImportedFTA objects are also used when configuring content redirection. See the New-BrokerConfiguredFTA cmdlet for more information.

The imported file type associations are grouped according to the desktop group to which they belong, because the system expects all machines in the same desktop group to have the same file type associations. That may not be true, however, across desktop groups.

Note that the ImportedFTA object has several fields that are not currently used. Only those fields that are shared with the ConfiguredFTA object are actually used in some capacity.

----- BrokerImportedFTA Object

The BrokerImportedFTA object represents a file type association imported from worker machines. It contains the following properties:

-- ContentType (System.String)

Content type of the file, such as "text/plain" or "application/vnd.ms-excel".

-- Description (System.String)



File type description, such as "Test Document", "Microsoft Word Text Document", etc.

-- DesktopGroupUid (System.Int32)

The desktop group this file type belongs to.

-- Edit (System.String)

Edit command with full path to executable: "C:\Program Files (x86)\Microsoft Office\Office12\WINWORD.EXE" /n /dde

-- EditArguments (System.String)

The arguments used for the 'edit' action on files of this type. These are extracted from the full edit command, and may be empty.

-- EditExecutableName (System.String)

The executable name extracted from the Edit property, no path included. This is used for matching with published apps' executable when searching for the list of extensions an application is capable of handling.

-- ExtensionName (System.String)

A single file extension, such as .txt. Unique within the scope of a desktop group.

-- HandlerName (System.String)

File type handler name, e.g. "Word.Document.8" or TXT FILE.

-- Open (System.String)

Open command with full path to executable: "C:\Program Files (x86)\Microsoft Office\Office12\WINWORD.EXE" /n /dde

-- OpenArguments (System.String)

The arguments used for the 'open' action on files of this type. These are extracted from the full open command, and may be empty.

-- OpenExecutableName (System.String)

The executable name extracted from the Open property, no path included. This is used for matching with published apps' executable when searching for the list of extensions an application is capable of handling.

-- PerceivedType (System.String)

Perceived type, such as "text".

-- Print (System.String)

Print command: "C:\Program Files (x86)\Microsoft Office\Office12\WINWORD.EXE" /x /n /dde

-- PrintTo (System.String)

PrintTo command: "C:\Program Files (x86)\Microsoft Office\Office12\WINWORD.EXE" /n /dde

-- Uid (System.Int32)

Unique internal identifier of imported file type association.

## Related topics

[New-BrokerConfiguredFTA](#)

[Remove-BrokerImportedFTA](#)

[Update-BrokerImportedFTA](#)

## Parameters

**-Uid**<Int32>

Gets only the imported file type associations with the specified unique identifier.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-ExtensionName**<String>

Gets only the imported file type associations with the specified extension name. For example, ".txt" or ".png".

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ContentType**<String>

Gets only the imported file type associations with the specified content type (as listed in the Registry). For example, "application/msword".

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Description**<String>

Gets only the imported file type associations with the specified description (as listed in the Registry). For example, "Text Document" or "Microsoft Word text document".

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopGroupUid**<Int32>

Gets only the file type associations imported from a worker machine belonging to the specified desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Edit**<String>

Gets only the imported file type associations with the specified Edit command, that includes both the executable name and path, and any arguments to that executable.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-EditArguments**<String>

Gets only the imported file type associations with the specified arguments to the Edit command.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ExecutableName**<String>

Gets only the imported file type associations with the specified executable for the Edit command.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-HandlerName**<String>

Gets only the imported file type associations with the specified handler name (as listed in the Registry). For example, "TXT FILE" or "Word.Document.8".

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Open**<String>

Gets only the imported file type associations with the specified Open command, that includes both the executable name and path, and any arguments to that executable.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-OpenArguments**<String>

Gets only the imported file type associations with the specified arguments to the Open command.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-OpenExecutableName**<String>

Gets only the imported file type associations with the specified executable for the Open command.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PerceivedType**<String>

Gets only the imported file type associations with the specified perceived type (as listed in the Registry). For example, "document".

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Print**<String>

Gets only the imported file type associations with the specified Print command.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PrintTo**<String>

Gets only the imported file type associations with the specified PrintTo command.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

--	--

### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through `Select-Object`, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

None This cmdlet does not accept any input from the pipeline.

### Return Values

Citrix.Broker.Admin.SDK.ImportedFTA

One or more ImportedFTA objects are returned as output.

### Examples

----- **EXAMPLE 1** -----

```
C:\PS> Get-BrokerImportedFTA
```

Invoking this cmdlet with no arguments simply returns all of the imported file type association objects. By default, only the first 250 objects are returned. See the -MaxRecordCount and -Skip parameters for information about modifying this.

----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerImportedFTA -ExtensionName ".txt"
```

Retrieves all imported file type associations that have the extension ".txt". Note that because imported file type associations are per-desktop group, multiple instances may be returned.



# Get-BrokerInstalledDbVersion

May 10, 2016

Gets a list of all available database schema versions for the Broker Service.

## Syntax

```
Get-BrokerInstalledDbVersion [-Upgrade] [-Downgrade] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets the current version number of the Citrix Broker Service database schema when called with no parameters.

When called with the -Upgrade parameter, gets the service schema version numbers to which an upgrade could be performed.

When called with the -Downgrade parameter, gets the service schema version numbers to which a downgrade could be performed.

The SQL scripts to perform schema upgrades and downgrades can be obtained using the Get-BrokerDBVersionChangeScript cmdlet. Citrix recommends that where possible service schema upgrades are performed using Studio rather than manually via the SDK.

Only one of the -Upgrade or -Downgrade parameters may be supplied at once.

## Related topics

[Get-BrokerDBVersionChangeScript](#)

[Get-BrokerDBSchema](#)

## Parameters

**-Upgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be updated should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Downgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be reverted should be returned.

Required?	false

Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

None You cannot pipe input into this cmdlet.

### Return Values

System.Version

Get-BrokerInstalledDBVersion returns database schema version numbers as requested.

### Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

InvalidParameterCombination

Both the Upgrade and Downgrade flags were specified.

NoOp

The operation was successful but had no effect.

NoDBConnections

The database connection string for the Broker Service has not been specified.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS>Get-BrokerInstalledDBVersion
```

Gets the current Citrix Broker Service database schema version number.

#### ----- EXAMPLE 2 -----

```
C:\PS>Get-BrokerInstalledDBVersion -Upgrade
```

Get the versions of the Broker Service database schema for which upgrade scripts are supplied.

# Get-BrokerLease

May 10, 2016

Gets stored leases.

## Syntax

```
Get-BrokerLease [-Uid] <Int64> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerLease [[-Key] <String>] [-Expiration <DateTime>] [-LastModified <DateTime>] [-LeaseType <BrokerLeaseType>] [-OwnerSAMName <String>] [-OwnerSID <String>] [-OwnerUPN <String>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets leases filtered by specific Uid or Owner information.

----- BrokerLease Object

The BrokerLease object represents a single instance of a lease. It contains the following properties:

-- Expiration (System.DateTime)

The expiration time of the lease.

-- Key (System.String)

The SHA1 representation of the lease key.

-- LastModified (System.DateTime)

The modification time of the lease.

-- LeaseType (Citrix.Broker.Admin.SDK.BrokerLeaseType)

The type of lease.

-- OwnerSAMName (System.String)

The SAM name of the user associated with the lease.

-- OwnerSID (System.String)

The SID of the user associated with the lease.

-- OwnerUPN (System.String)

The UPN of the user associated with the lease.

-- Uid (System.Int64)

The UID of the lease itself.

-- Value (System.String)

The serialized lease data in XML.

## Related topics

[Remove-BrokerLease](#)

[Update-BrokerLocalLeaseCache](#)

## Parameters

**-Uid**<Int64>

Gets only the lease specified by unique identifier.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Key**<String>

Gets only the leases matching the specified lease key pattern.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Expiration**<DateTime>

Gets only the leases matching the specified expiration date and time.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LastModified**<DateTime>

Gets only the leases matching the specified modified date and time.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LeaseType**<BrokerLeaseType>

Gets only leases of a specific type. Possible values Enumeration, Launch.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OwnerSAMName**<String>

Gets only the leases associated with the specified Domain\User.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OwnerSID**<String>

Gets only the leases associated with the specified user SID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OwnerUPN**<String>

Gets only the leases associated with the specified user UPN.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if

no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through `Select-Object`, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type



None Input cannot be piped to this cmdlet.

## Return Values

Citrix.Broker.Admin.SDK.Lease

Returns an Lease object for each enumerated lease.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> $lease = Get-BrokerLease -Uid 1
```

Gets the lease with internal Uid 1.

### ----- **EXAMPLE 2** -----

```
C:\PS> $leases = Get-BrokerLease -OwnerSAMName Domain\User
```

Gets the leases associated with the specified user.

# Get-BrokerMachine

May 10, 2016

Gets machines belonging to this site.

## Syntax

```
Get-BrokerMachine [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerMachine [-MachineName] <String> [-AgentVersion <String>] [-AllocationType <AllocationType>] [-ApplicationInUse <String>] [-AssignedClientName <String>] [-AssignedIPAddress <String>] [-AssociatedUserFullName <String>] [-AssociatedUserName <String>] [-AssociatedUserSID <String>] [-AssociatedUserUPN <String>] [-BrowserName <String>] [-CatalogName <String>] [-CatalogUid <Int32>] [-CatalogUUID <Guid>] [-ColorDepth <ColorDepth>] [-ControllerDNSName <String>] [-DeliveryType <DeliveryType>] [-Description <String>] [-DesktopCondition <String>] [-DesktopGroupName <String>] [-DesktopGroupUid <Int32>] [-DesktopGroupUUID <Guid>] [-DesktopKind <DesktopKind>] [-DesktopUid <Int32>] [-DNSName <String>] [-FaultState <MachineFaultState>] [-FunctionalLevel <FunctionalLevel>] [-HostedMachineId <String>] [-HostedMachineName <String>] [-HostingServerName <String>] [-HypervisorConnectionName <String>] [-HypervisorConnectionUid <Int32>] [-HypHypervisorConnectionUid <Guid>] [-IconUid <Int32>] [-ImageOutOfDate <Boolean>] [-InMaintenanceMode <Boolean>] [-IPAddress <String>] [-IsAssigned <Boolean>] [-IsPhysical <Boolean>] [-LastConnectionFailure <ConnectionFailureReason>] [-LastConnectionTime <DateTime>] [-LastConnectionUser <String>] [-LastDeregistrationReason <DeregistrationReason>] [-LastDeregistrationTime <DateTime>] [-LastErrorReason <String>] [-LastErrorTime <DateTime>] [-LastHostingUpdateTime <DateTime>] [-LastPvdErrorReason <String>] [-LastPvdErrorTime <DateTime>] [-LoadIndex <Int32>] [-MachineInternalState <MachineInternalState>] [-Metadata <String>] [-OSType <String>] [-OSVersion <String>] [-PersistUserChanges <PersistUserChanges>] [-PowerActionPending <Boolean>] [-PowerState <PowerState>] [-ProvisioningType <ProvisioningType>] [-PublishedApplication <String>] [-PublishedName <String>] [-PvdEstimatedCompletionTime <DateTime>] [-PvdPercentDone <Int32>] [-PvdStage <PvdStage>] [-PvdUpdateStartTime <DateTime>] [-RegistrationState <RegistrationState>] [-ScheduledReboot <ScheduledReboot>] [-SecureIcaRequired <Boolean>] [-SessionAutonomouslyBrokered <Boolean>] [-SessionClientAddress <String>] [-SessionClientName <String>] [-SessionClientVersion <String>] [-SessionConnectedViaHostName <String>] [-SessionConnectedViaIP <String>] [-SessionCount <Int32>] [-SessionDeviceId <String>] [-SessionHardwareId <String>] [-SessionHidden <Boolean>] [-SessionKey <Guid>] [-SessionLaunchedViaHostName <String>] [-SessionLaunchedViaIP <String>] [-SessionProtocol <String>] [-SessionSecureIcaActive <Boolean>] [-SessionsEstablished <Int32>] [-SessionSmartAccessTag <String>] [-SessionsPending <Int32>] [-SessionStartTime <DateTime>] [-SessionState <SessionState>] [-SessionStateChangeTime <DateTime>] [-SessionSupport <SessionSupport>] [-SessionType <SessionType>] [-SessionUid <Int64>] [-SessionUserName <String>] [-SessionUserSID <String>] [-SID <String>] [-SummaryState <DesktopSummaryState>] [-SupportedPowerActions <String[]>] [-Tag <String>] [-UUID <Guid>] [-VMToolsState <VMToolsState>] [-WillShutdownAfterUse <Boolean>] [-WindowsConnectionSetting <WindowsConnectionSetting>] [-AssignedUserSID <String>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieves machines matching the specified criteria. If no parameters are specified, this cmdlet enumerates all machines.

Get-BrokerMachine returns objects that combine machine configuration and state information.

For single-session machines, session information is displayed if present. If "fast user switching" is enabled, more than one session may be present on single-session machines. Because this cmdlet returns information only for a single session, if two

sessions are present it will return information about the brokered session (rather than, for example, an unbrokered direct RDP session). If there is no session running, session-related fields return \$null.

For multi-session machines, no session information about single sessions is displayed by this cmdlet, and so are always \$null. Get-BrokerSession can be used to get information about sessions on both multi-session and single-session machines.

To count machines, rather than retrieve full details of each machine, use Group-BrokerMachine instead.

See about\_Broker\_Filtering for information about advanced filtering options, and about\_Broker\_Machines for background information about machines.

#### ----- BrokerMachine Object

The machine object returned represents a physical or virtual machine, which has been configured in the site.

-- AgentVersion (System.String)

Version of the Citrix Virtual Delivery Agent (VDA) installed on the machine.

-- AllocationType (Citrix.Broker.Admin.SDK.AllocationType)

Describes how the machine is allocated to the user, can be Permanent or Random.

-- ApplicationsInUse (System.String[])

List of applications in use on the machine (in the form of browser name).

-- AssignedClientName (System.String)

The name of the endpoint client device that the machine has been assigned to.

-- AssignedIPAddress (System.String)

The IP address of the endpoint client device that the machine has been assigned to.

-- AssociatedUserFullNames (System.String[])

Full names of the users that have been associated with the machine (usually in the form "Firstname Lastname").

Associated users are the current user(s) for shared machines and the assigned users for private machines.

-- AssociatedUserNames (System.String[])

Username of the users that have been associated with the machine (in the form "domain\user").

Associated users are the current user(s) for shared machines and the assigned users for private machines.

-- AssociatedUserSIDs (System.String[])

The SIDs of the users that have been associated with the machine.

Associated users are the current user(s) for shared machines and the assigned users for private machines.

-- AssociatedUserUPNs (System.String[])

The User Principal Names of the users associated with the machine (in the form user@domain).

Associated users are the current user(s) for shared machines and the assigned users for private machines.

-- BrowserName (System.String)

Site-wide unique name identifying associated desktop to other components (for example StoreFront). This is typically non-null only for machines backing assigned private desktops.

-- Capabilities (System.String[])

List of the capabilities that the machine supports. Valid capabilities are:

- o MultiSession: Indicates an RDS- (Terminal Services-) based machine, which supports multiple active sessions from different users.

- o CBP1\_5: Indicates the machine uses the CBP 1.5 protocol for communication.

-- CatalogName (System.String)

Name of the catalog the machine is a member of.

-- CatalogUid (System.Int32)

UID of the catalog the machine is a member of.

-- CatalogUUID (System.Guid)

UUID of the catalog the machine is a member of.

-- ColorDepth (Citrix.Broker.Admin.SDK.ColorDepth?)

The color depth setting configured on the machine, possible values are:

Snul, FourBit, EightBit, SixteenBit, and TwentyFourBit.

-- ControllerDNSName (System.String)

The DNS host name of the controller that the machine is registered to.

-- DeliveryType (Citrix.Broker.Admin.SDK.DeliveryType?)

Denotes whether the machine delivers desktops only, apps only or both.

-- Description (System.String)

Description of the machine.

-- DesktopConditions (System.String[])

List of outstanding desktop conditions for the machine.

-- DesktopGroupName (System.String)

Name of the desktop group the machine is a member of.

-- DesktopGroupUid (System.Int32?)

UID of the desktop group the machine is a member of.

-- DesktopGroupUUID (System.Guid?)

UUID of the desktop group the machine is a member of.

-- DesktopKind (Citrix.Broker.Admin.SDK.DesktopKind?)

Deprecated.

Denotes whether the machine is private or shared. Use AllocationType instead.

-- DesktopUid (System.Int32?)

The UID of the associated desktop object.

-- DNSName (System.String)

The DNS host name of the machine.

-- FaultState (Citrix.Broker.Admin.SDK.MachineFaultState)

Summary state of any current fault state of the machine. Can be one of the following:

o None - No fault; machine is healthy.

o FailedToStart - Last power-on operation for machine failed.

o StuckOnBoot - Machine does not seem to have booted following power on.

o Unregistered - Machine has failed to register within expected period, or its registration has been rejected.

o MaxCapacity - Machine is reporting itself at maximum capacity.

-- FunctionalLevel (Citrix.Broker.Admin.SDK.FunctionalLevel?)

Functional level of the machine, if known.

-- HostedMachineId (System.String)

Unique ID within the hosting unit of the target managed machine.

-- HostedMachineName (System.String)

The friendly name of a hosted machine as used by its hypervisor. This is not necessarily the DNS name of the machine.

-- HostingServerName (System.String)

DNS name of the hypervisor that is hosting the machine if managed.

-- HypervisorConnectionName (System.String)

The name of the hypervisor connection that the machine has been assigned to, if managed.

-- HypervisorConnectionUid (System.Int32?)

The UID of the hypervisor connection that the machine's hosting server is accessed through.

-- HypHypervisorConnectionUid (System.Guid?)

The UUID of the hypervisor connection that the machine's hosting server is accessed through

-- IconUid (System.Int32?)

The UID of the machine's icon that is displayed in StoreFront.

-- ImageOutOfDate (System.Boolean?)

Denotes if the VM image for a hosted machine is out of date.

-- InMaintenanceMode (System.Boolean)

Denotes if the machine is in maintenance mode.

-- IPAddress (System.String)

The IP address of the machine.

-- IsAssigned (System.Boolean)

Denotes whether a private desktop has been assigned to a user/users, or a client name/address. Users can be assigned explicitly or by assigning on first use of the machine.

-- IsPhysical (System.Boolean)

This value is true if the machine is physical (ie not power managed by the Citrix Broker service, and false otherwise.

-- LastConnectionFailure (Citrix.Broker.Admin.SDK.ConnectionFailureReason)

The reason for the last failed connection between a client and the machine.

-- LastConnectionTime (System.DateTime?)

Time of the last detected connection attempt that either failed or succeeded.

-- LastConnectionUser (System.String)

The SAM name (in the form DOMAIN\user) of the user that last attempted a connection with the machine. If the SAM name is not available, the SID is used.

-- LastDeregistrationReason (Citrix.Broker.Admin.SDK.DeregistrationReason?)

The reason for the last deregistration of the machine with the broker. Possible values are:

AgentShutdown, AgentSuspended, AgentRequested, IncompatibleVersion, AgentAddressResolutionFailed, AgentNotContactable, AgentWrongActiveDirectoryOU, EmptyRegistrationRequest, MissingRegistrationCapabilities, MissingAgentVersion, InconsistentRegistrationCapabilities, NotLicensedForFeature, UnsupportedCredentialSecurityVersion, InvalidRegistrationRequest, SingleMultiSessionMismatch, FunctionalLevelTooLowForCatalog, FunctionalLevelTooLowForDesktopGroup, PowerOff, DesktopRestart, DesktopRemoved, AgentRejectedSettingsUpdate, SendSettingsFailure, SessionAuditFailure, SessionPrepareFailure, ContactLost, SettingsCreationFailure, UnknownError and BrokerRegistrationLimitReached.

-- LastDeregistrationTime (System.DateTime?)

Time of the last deregistration of the machine from the controller.

-- LastErrorReason (System.String)

The reason for the last error detected in the machine.

-- LastErrorTime (System.DateTime?)

The time of the last detected error.

-- LastHostingUpdateTime (System.DateTime?)

Time of last update to any hosting data (such as power states) for this machine reported by the hypervisor connection.

-- LastPvdErrorReason (System.String)

The error text from the most recent failure of the Personal vDisk preparation process for this machine (if any).

-- LastPvdErrorTime (System.DateTime?)

The time of the most recent failure of the Personal vDisk preparation process for this machine (if any).

-- LoadIndex (System.Int32?)

Gives current effective load index for multi-session machines.

-- LoadIndexes (System.String[])

Gives the last reported individual load indexes that were used in the calculation of the LoadIndex value. Note that the LoadIndex value may have been subsequently adjusted due to session brokering operations. This value is only set for multi-session machines.

-- MachineInternalState (Citrix.Broker.Admin.SDK.MachineInternalState)

The internal state of the machine; reported while the machine is registered to a controller, plus some private Citrix Broker Service states while the machine is not registered.

-- MachineName (System.String)

DNS host name of the machine.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

Any metadata that is associated with the machine.

-- OSType (System.String)

A string that can be used to identify the operating system that is running on the machine.

-- OSVersion (System.String)

A string that can be used to identify the version of the operating system running on the machine, if known.

-- PersistUserChanges (Citrix.Broker.Admin.SDK.PersistUserChanges)

Describes if and how user changes are persisted. Possible values are:

- o OnLocal - Persist the user changes on the local disk of the machine.

- o Discard - Discard user changes.

- o OnPvd - Persist user changes on the Citrix Personal vDisk.

-- PowerActionPending (System.Boolean)

Indicates if there are any pending power actions for the machine.

-- PowerState (Citrix.Broker.Admin.SDK.PowerState)

The current power state of the machine. Possible values are: Unmanaged, Unknown, Unavailable, Off, On, Suspended, TurningOn, TurningOff, Suspending, resuming.

-- ProvisioningType (Citrix.Broker.Admin.SDK.ProvisioningType)

Describes how the machine was provisioned, possible values are:

- o Manual: No automated provisioning.

- o PVS: Machine provisioned by PVS (may be physical, blade, VM,...)

- o MCS: Machine provisioned by MCS (machine must be VM)

-- PublishedApplications (System.String[])

List of applications published by the machine (displayed as browser names).

-- PublishedName (System.String)

The name of the machine that is displayed in StoreFront, if the machine has been published.

-- PvdEstimatedCompletionTime (System.DateTime?)

If preparation of the Personal vDisk is currently in progress for this machine, this reports an estimation of the time at which the process will be complete.

-- PvdPercentDone (System.Int32?)

If preparation of the Personal vDisk is currently in progress for this machine, this reports how far the process has got as a percentage. This value will be zero if preparation is not in progress.

-- PvdStage (Citrix.Broker.Admin.SDK.PvdStage)

For a machine supporting Personal vDisk technology (PvD), indicates the stage of the PvD image preparation.

-- PvdUpdateStartTime (System.DateTime?)

If preparation of the Personal vDisk is currently in progress for this machine, this reports when the update process began.

-- RegistrationState (Citrix.Broker.Admin.SDK.RegistrationState)



Indicates the registration state of the machine. Possible values are: Unregistered, Initializing, Registered, AgentError.

-- ScheduledReboot (Citrix.Broker.Admin.SDK.ScheduledReboot)

Indicates the state of any scheduled reboot operation for a machine. Possible values:

o None: No reboot is scheduled.

o Pending: Machine is awaiting reboot but is available for use.

o Draining: Machine is awaiting reboot and is unavailable for new sessions; reconnections to existing connections are still allowed, however.

o InProgress: Machine is actively undergoing a scheduled reboot. o Natural: Natural reboot in progress. Machine is awaiting a restart.

-- SecureIcaRequired (System.Boolean?)

Flag indicating whether SecureICA is required or not when starting a session on the machine.

-- SessionAutonomouslyBrokered (System.Boolean?)

Session property indicating if the current session is an HDX session established by direct connection without being brokered.

Session properties are always null for multi-session machines.

-- SessionClientAddress (System.String)

Session property indicating the IP address of the client connected to the machine.

Session properties are always null for multi-session machines.

-- SessionClientName (System.String)

Session property indicating the host name of the client connected to the machine.

Session properties are always null for multi-session machines.

-- SessionClientVersion (System.String)

Session property indicating the version of the Citrix Receiver on the connected client.

Session properties are always null for multi-session machines.

-- SessionConnectedViaHostName (System.String)

Session property indicating the host name of the connection gateway, router or client.

Session properties are always null for multi-session machines.

-- SessionConnectedViaIP (System.String)

Session property indicating the IP address of the connection gateway, router or client.

Session properties are always null for multi-session machines.

-- SessionCount (System.Int32)

Count of number of sessions on the machine.

-- SessionDeviceId (System.String)

Session property indicating a unique identifier for the client device that has most recently been associated with the current session.

Session properties are always null for multi-session machines.

-- SessionHardwareId (System.String)

Session property indicating a unique identifier for the client hardware that has been most recently associated with the current session.

Session properties are always null for multi-session machines.

-- SessionHidden (System.Boolean?)

Session property that indicates if a session is hidden.

Session properties are always null for multi-session machines.

-- SessionKey (System.Guid?)

Session property indicating the key of the current session.

Session properties are always null for multi-session machines.

-- SessionLaunchedViaHostName (System.String)

Session property that denotes the host name of the StoreFront server used to launch the current brokered session.

Session properties are always null for multi-session machines.

-- SessionLaunchedViaIP (System.String)

Session property that denotes the IP address of the StoreFront server used to launch the current brokered session.

Session properties are always null for multi-session machines.

-- SessionProtocol (System.String)

Session property that denotes the protocol that the current session is using, can be either HDX, RDP or Console. Console sessions on XenDesktop 5 VDAs appear with a blank protocol.

Session properties are always null for multi-session machines.

-- SessionSecureIcaActive (System.Boolean?)

Session property that indicates whether SecureICA is active on the current session or not.

Session properties are always null for multi-session machines.

-- SessionsEstablished (System.Int32)

Number of established sessions on this machine. For multi-session machines this excludes established sessions which have not yet completed their logon processing.

-- SessionSmartAccessTags (System.String[])

Session property that indicates the Smart Access tags for the current session.

Session properties are always null on multi-session machines.

-- SessionsPending (System.Int32)

Number of pending (brokered but not yet established) sessions on this machine. For multi-session machines this also includes established sessions which have not yet completed their logon processing.

-- SessionStartTime (System.DateTime?)

Session property that indicates the start time of the current session.

Session properties are always null on multi-session machines.

-- SessionState (Citrix.Broker.Admin.SDK.SessionState?)

Session property indicating the state of the current session, possible values are:

Other, PreparingSession, Connected, Active, Disconnected, Reconnecting, NonBrokeredSession and Unknown. Session properties are always null for multi-session machines.

-- SessionStateChangeTime (System.DateTime?)

Session property indicating the time of the last state change of the current session.

Session properties are always null for multi-session machines.

-- SessionSupport (Citrix.Broker.Admin.SDK.SessionSupport)

Indicates the session support of the machine.

Possible values:

o SingleSession: Single-session only machine.

o MultiSession: Multi-session capable machine.

-- SessionType (Citrix.Broker.Admin.SDK.SessionType?)

Session property indicating the type of the current session.

Session properties are always null for multi-session machines.

-- SessionUid (System.Int64?)

Session property indicating the UID of the current session.

Session properties are always null for multi-session machines.

-- SessionUserName (System.String)

Session property indicates the name of the current session's user (in the form DOMAIN\user).

Session properties are always null for multi-session machines.

-- SessionUserSID (System.String)

Session property indicates the SID of the current session's user.

Session properties are always null for multi-session machines.

-- SID (System.String)

The SID of the machine.

-- SummaryState (Citrix.Broker.Admin.SDK.DesktopSummaryState)

Indicates the overall state of the desktop associated with the machine. The overall state is a result of other more specific states such as session state, registration state and power state. Possible values: Off, Unregistered, Available, Disconnected, InUse, Preparing.

-- SupportedPowerActions (System.String[])

A list of power actions supported by this machine.

-- Tags (System.String[])

A list of tags for the machine.

-- Uid (System.Int32)

UID of the machine object.

-- UUID (System.Guid)

UUID of the machine object.

-- VMToolsState (Citrix.Broker.Admin.SDK.VMToolsState)

State of the hypervisor tools present on the VM (if any).

Possible values are:

NotPresent, Unknown, NotStarted, Running.

-- WillShutdownAfterUse (System.Boolean)

Flag indicating if this machine is tainted and will be shut down after all sessions on the machine have ended. This flag is only ever true on power-managed, single-session machines.

Note: The machine will not shut down if it is in maintenance mode; it will shut down only after it is taken out of maintenance mode.

-- WindowsConnectionSetting (Citrix.Broker.Admin.SDK.WindowsConnectionSetting?)

The logon mode reported by Windows itself (multi-session machines only). For single-session machines the value is always

hardwired to LogonEnabled.

Possible values are:

LogonEnabled, Draining, DrainingUntilRestart and LogonDisabled.

## Related topics

[Group-BrokerMachine](#)

## Parameters

**-Uid**<Int32>

Gets a machine with a specific UID.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-MachineName**<String>

Gets machines with a specific machine name (in the form domain\machine).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AgentVersion**<String>

Gets machines with a specific Citrix Virtual Delivery Agent version.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AllocationType**<AllocationType>

Gets machines from catalogs with the specified allocation type.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ApplicationInUse<String>**

Gets machines running a specified published application (identified by browser name).

String comparisons are case-insensitive.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssignedClientName<String>**

Gets machines that have been assigned to the specific client name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssignedIPAddress<String>**

Gets machines that have been assigned to the specific IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserFullName<String>**

Gets machines with an associated user identified by their full name (usually 'first-name last-name').

Associated users are all current users of a desktop, plus the assigned users for private desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AssociatedUserName<String>**

Gets machines with an associated user identified by their user name (in the form 'domain\user').

Associated users are all current users of a desktop, plus the assigned users for private desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AssociatedUserSID<String>**

Gets machines with an associated user identified by their Windows SID.

Associated users are all current users of a desktop, plus the assigned users for private desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AssociatedUserUPN<String>**

Gets machines with an associated user identified by their User Principle Name (in the form 'user@domain').

Associated users are all current users of a desktop, plus the assigned users for private desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-BrowserName**<String>

Gets assigned machines backing desktop resources that have browser names matching the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-CatalogName**<String>

Gets machines from the catalog with the specific name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-CatalogUid**<Int32>

Gets machines from the catalog with the specific UID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-CatalogUUID**<Guid>

Gets machines from the catalog with the specific UUID.

Required?	false
Default Value	
Accept Pipeline Input?	false



#### **-ColorDepth**<ColorDepth>

Gets machines configured with a specific color depth.

Valid values are FourBit, EightBit, SixteenBit, and TwentyFourBit.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ControllerDNSName**<String>

Gets machines with a specific DNS name of the controller they are registered with.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DeliveryType**<DeliveryType>

Gets machines of a particular delivery type.

Valid values are AppsOnly, DesktopsOnly, DesktopsAndApps

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Description**<String>

Gets machines with a specific description.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopCondition<String>**

Gets machines with an outstanding desktop condition.

Valid values are:

- o CPU: Indicates the machine has high CPU usage
- o ICALatency: Indicates the network latency is high
- o UPMLogonTime: Indicates that the profile load time was high

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupName<String>**

Gets machines from a desktop group with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupUid<Int32>**

Gets machines from a desktop group with a specific UID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupUUID<Guid>**

Gets machines from a desktop group with a specific UUID.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-DesktopKind**<DesktopKind>

Deprecated: Use AllocationType parameter.

Gets machines of a particular kind.

Valid values are Private, Shared.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopUid**<Int32>

Gets the machine that corresponds to the desktop with the specific UID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DNSName**<String>

Gets machines with the specific DNS name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-FaultState**<MachineFaultState>

Gets machines currently in the specified fault state.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-FunctionalLevel**<FunctionalLevel>

Gets machines with a specific FunctionalLevel.

Valid values are L5, L7, L7\_6

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HostedMachineId**<String>

Gets machines with the specific machine ID known to the hypervisor.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HostedMachineName**<String>

Gets machines with the specific machine name known to the hypervisor.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HostingServerName**<String>

Gets machines by the name of the hosting hypervisor server.

--	--

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionName<String>**

Gets machines with the specific name of the hypervisor connection hosting them.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionUid<Int32>**

Gets machines with the specific UID of the hypervisor connection hosting them.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypHypervisorConnectionUid<Guid>**

Gets machines with the specific UUID of the hypervisor connection hosting them.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IconUid<Int32>**

Gets machines by configured icon. Note that machines with a null IconUid use the icon of the desktop group.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-ImageOutOfDate**<Boolean>

Gets machines depending on whether their disk image is out of date or not (for machines provisioned using MCS only).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-InMaintenanceMode**<Boolean>

Gets machines by whether they are in maintenance mode or not.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IPAddress**<String>

Gets machines with a specific IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IsAssigned**<Boolean>

Gets machines according to whether they are assigned or not. Machines may be assigned to one or more users or groups, a client IP address or a client endpoint name.

Required?	false

Default Value	
Accept Pipeline Input?	false

#### **-IsPhysical**<Boolean>

Gets machines according to whether they can be power managed by XenDesktop or not.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastConnectionFailure**<ConnectionFailureReason>

Gets machines with a specific reason for the last recorded connection failure. This value is None if the last connection was successful or if there has been no attempt to connect to the desktop yet.

Valid values are None, SessionPreparation, RegistrationTimeout, ConnectionTimeout, Licensing, Ticketing, and Other.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastConnectionTime**<DateTime>

Gets machines on which a user session connection occurred at a specific time. This is the time at which the broker detected that the connection attempt either succeeded or failed.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastConnectionUser**<String>

Gets machines where a specific user name last attempted a connection (in the form 'domain\user').

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-LastDeregistrationReason**<DeregistrationReason>

Gets machines whose broker last recorded a specific deregistration reason.

Valid values are \$null, AgentShutdown, AgentSuspended, AgentRequested, IncompatibleVersion, AgentAddressResolutionFailed, AgentNotContactable, AgentWrongActiveDirectoryOU, EmptyRegistrationRequest, MissingRegistrationCapabilities, MissingAgentVersion, InconsistentRegistrationCapabilities, NotLicensedForFeature, UnsupportedCredentialSecurityVersion, InvalidRegistrationRequest, SingleMultiSessionMismatch, FunctionalLevelTooLowForCatalog, FunctionalLevelTooLowForDesktopGroup, PowerOff, DesktopRestart, DesktopRemoved, AgentRejectedSettingsUpdate, SendSettingsFailure, SessionAuditFailure, SessionPrepareFailure, ContactLost, SettingsCreationFailure, UnknownError and BrokerRegistrationLimitReached.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastDeregistrationTime**<DateTime>

Gets machines by the time that they were last deregistered.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastErrorReason**<String>

Gets machines with the specified last error reason.

Required?	false
Default Value	
Accept Pipeline Input?	false



---

**-LastErrorTime<DateTime>**

Gets machines with the specified last error time.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LastHostingUpdateTime<DateTime>**

Gets machines with a specific time that the hosting information was last updated.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LastPvdErrorReason<String>**

Gets machines with the specified last Personal vDisk preparation error reason.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LastPvdErrorTime<DateTime>**

Gets machines with the specified last Personal vDisk preparation error time.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LoadIndex<Int32>**

Gets machines by their current load index.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-MachineInternalState**<MachineInternalState>

Gets machines with the specified internal state.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

**-OSType**<String>

Gets machines by the type of operating system they are running.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-OSVersion**<String>

Gets machines by the version of the operating system they are running.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-PersistUserChanges**<PersistUserChanges>

Gets machines by the location where the user changes are persisted.

- o OnLocal - User changes are persisted locally.
- o Discard - User changes are discarded.
- o OnPvd - User changes are persisted on the Pvd.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-PowerActionPending**<Boolean>

Gets machines depending on whether a power action is pending or not.

Valid values are \$true or \$false.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-PowerState**<PowerState>

Gets machines with a specific power state.

Valid values are Unmanaged, Unknown, Unavailable, Off, On, Suspended, TurningOn, TurningOff, Suspending, and Resuming.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-ProvisioningType**<ProvisioningType>

Gets machines that are in a catalog with a particular provisioning type. Values can be:

- o Manual - No provisioning.
- o PVS - Machine provisioned by PVS (machine may be physical, blade, VM,...).
- o MCS - Machine provisioned by MCS (machine must be VM).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PublishedApplication**<String>

Gets machines with a specific application published to them (identified by its browser name).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PublishedName**<String>

Gets desktops with a specific published name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PvdEstimatedCompletionTime**<DateTime>

If preparation of the Personal vDisk is currently in progress for this machine, this reports an estimation of the time at which the process will be complete.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PvdPercentDone<Int32>**

Gets machines a specific percentage through the Personal vDisk preparation process.

This property is typically used with advanced filtering; see about\_Broker\_Filtering.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PvdStage<PvdStage>**

Gets machines at a specific personal vDisk stage.

Valid values are None, Requested, Starting, Working and Failed.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PvdUpdateStartTime<DateTime>**

If preparation of the Personal vDisk is currently in progress for this machine, this reports when the update process began.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-RegistrationState**<RegistrationState>

Gets machines in a specific registration state.

Valid values are Unregistered, Initializing, Registered, and AgentError.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ScheduledReboot**<ScheduledReboot>

Gets machines according to their current status with respect to any scheduled reboots (for either scheduled desktop group reboots or image rollout purposes). Valid values are:

- o None - No reboot currently scheduled.
- o Pending - Reboot scheduled but machine still available for use.
- o Draining - Reboot scheduled. New logons are disabled, but reconnections to existing sessions are allowed.
- o InProgress - Machine is actively being rebooted.
- o Natural - Natural reboot in progress. Machine is awaiting a restart.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-SecureIcaRequired**<Boolean>

Gets machines configured with a particular SecureIcaRequired setting. Note that the machine setting of \$null indicates that the desktop group value is used.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionAutonomouslyBrokered**<Boolean>

Gets machines according to whether their current session is autonomously brokered or not. Autonomously brokered sessions are HDX sessions established by direct connection without being brokered.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionClientAddress**<String>

Gets machines with a specific client IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionClientName**<String>

Gets machines with a specific client name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionClientVersion**<String>

Gets machines with a specific client version.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionConnectedViaHostName<String>**

Gets machines with a specific incoming connection host name. This is usually a proxy or Citrix Access Gateway server.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionConnectedViaIP<String>**

Gets machines with a specific incoming connection IP address.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionCount<Int32>**

Gets machines according to the total number of both pending and established user sessions on the machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionDeviceId<String>**

Gets machines with a specific client device ID.

Required?	false
Default Value	
Accept Pipeline Input?	



Accept Pipeline Input?	false
------------------------	-------

#### **-SessionHardwareId**<String>

Gets machines with a specific client hardware ID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionHidden**<Boolean>

Gets machines depending on whether their sessions are hidden or not. Hidden sessions are treated as though they do not exist when launching sessions using XenDesktop; a hidden session cannot be reconnected to, but a new session may be launched using the same entitlement.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionKey**<Guid>

Gets machine running the session with a specified unique key.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionLaunchedViaHostName**<String>

Gets machines with a specific host name of the StoreFront server from which the user launched the session.

Session properties are always null for multi-session machines.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-SessionLaunchedViaIP<String>**

Gets machines with a specific IP address of the StoreFront server from which the user launched the session.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionProtocol<String>**

Gets machines with connections using a specific protocol, for example HDX, RDP, or Console.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionSecureIcaActive<Boolean>**

Gets machines depending on whether the current session uses SecureICA or not.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionsEstablished<Int32>**

Gets machines according to the number of established user sessions present on the machine.

--	--

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionSmartAccessTag<String>**

Gets machines where the session has the specific SmartAccess tag.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionsPending<Int32>**

Get machines according to the number of pending user sessions for the machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionStartTime<DateTime>**

Gets machines with a specific session start time.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionState<SessionState>**

Gets machines with a specific session state.

Valid values are \$null, Other, PreparingSession, Connected, Active, Disconnected, Reconnecting, NonBrokeredSession, and Unknown.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionStateChangeTime**<DateTime>

Gets machines whose sessions last changed state at a specific time.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionSupport**<SessionSupport>

Gets machines that have the specified session capability. Values can be:

- o SingleSession - Single-session only machine.
- o MultiSession - Multi-session capable machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionType**<SessionType>

Gets machines with a specific session state.

Session properties are always null for multi-session machines.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-SessionUid<Int64>**

Gets machines with a specific session UID (\$null for no session).

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionUserName<String>**

Gets machines with a specific user name for the current session (in the form 'domain\user').

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionUserSID<String>**

Gets machines with a specific SID of the current session user.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SID<String>**

Gets machines with a specific machine SID.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-SummaryState**<DesktopSummaryState>

Gets machines with a specific summary state.

Valid values are Off, Unregistered, Available, Disconnected, and InUse.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-SupportedPowerActions**<String[]>

A list of power actions supported by this machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Tag**<String>

Gets machines where the session has the given SmartAccess tag.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UUID**<Guid>

Gets machines with the specified value of UUID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-VMToolsState**<VMToolsState>

Gets machines with a specific VM tools state.

Valid values are NotPresent, Unknown, NotStarted, and Running.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-WillShutDownAfterUse**<Boolean>

Gets machines depending on whether they shut down after use or not.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-WindowsConnectionSetting**<WindowsConnectionSetting>

Gets machines according to their current Windows connection setting (logon mode). Valid values are:

- o LogonEnabled - All logons are enabled.
- o Draining - New logons are disabled, but reconnections to existing sessions are allowed.
- o DrainingUntilRestart - Same as Draining, but setting reverts to LogonEnabled when machine next restarts.
- o LogonDisabled - All logons and reconnections are disabled.

This is a Windows setting and is not controlled by XenDesktop. It applies only to multi-session machines; for single-session

machines its value is always LogonEnabled.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssignedUserSID<String>**

Gets machines with the specific SID of the user to whom the desktop is assigned.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount<SwitchParameter>**

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount<Int32>**

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip<Int32>**



Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

**-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

**-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

Citrix.Broker.Admin.SDK.Machine

Get-BrokerMachine returns an object for each matching desktop.

## Notes

It is generally better to compare dates and times using -Filter and relative comparisons. See about\_Broker\_Filtering and the examples in this topic for more information.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerMachine -PowerState Suspended
```

```
C:\PS> Get-BrokerMachine -Filter { PowerState -eq 'Suspended' }
```

These commands return all suspended machines. The second form uses advanced filtering (see about\_Broker\_Filtering).

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerMachine -DNSName '*.mydomain.mycompany.com'
```

This command returns all machines belonging to the DNS domain mydomain.mycompany.com.

### ----- EXAMPLE 3 -----

```
C:\PS> Get-BrokerMachine -Filter { RegistrationState -eq 'Registered' -and HypervisorConnectionUid -eq 5 }
```

This command returns all registered machines running on the specified hypervisor connection.

### ----- EXAMPLE 4 -----

```
C:\PS> Get-BrokerMachine -MachineName 'MyDomain\X*' | Remove-BrokerDesktopGroup -DesktopGroup 2
```

This command finds all of the machines in MyDomain with names beginning with X and removes them from the specified desktop group.

### ----- EXAMPLE 5 -----

```
C:\PS> Get-BrokerMachine -Filter { DesktopGroupUid -ne $null }
```

This command gets all desktops in a site. Use this instead of the deprecated Get-BrokerDesktop command.

# Get-BrokerMachineCommand

May 10, 2016

Get the list of commands queued for delivery to a desktop.

## Syntax

```
Get-BrokerMachineCommand -Uid <Int64> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerMachineCommand [-Category <String>] [-CommandName <String>] [-CompletionTime <DateTime>] [-MachineName <String>] [-MachineUid <Int32>] [-Metadata <String>] [-RequestTime <DateTime>] [-SendDeadline <TimeSpan>] [-SendDeadlineTime <DateTime>] [-SendTrigger <MachineCommandTrigger>] [-SessionUid <Int64>] [-State <MachineCommandState>] [-User <String>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Get the list of commands queued for delivery to a desktop. Commands are batched and can be configured to be delivered at various times during a desktop session's lifetime. Normally commands are sent within a few minutes of being queued, but it is also possible to queue a command for a user who is not currently logged on or a desktop that is currently switched off.

See `about_Broker_Filtering` for information about advanced filtering options.

----- BrokerMachineCommand Object

The command object returned represents a command handled by a specific service on a desktop as determined by the `Category` property.

-- `Category` (System.String)

Category of the command.

-- `CommandData` (System.Byte[])

Additional binary data included when the command is sent.

-- `CommandName` (System.String)

Name of the command.

-- `CompletionTime` (System.DateTime?)

Time at which the command was sent, expired or canceled.

-- `DesktopGroupNames` (System.String[])

List of desktop group names that the command was restricted to.

-- `MachineName` (System.String)

Name of the machine this command is targeted at.

-- `MachineUid` (System.Int32?)

Unique ID of the machine this command is targeted at.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

Metadata for this command.

-- RequestTime (System.DateTime)

Time at which this command was created.

-- SendDeadline (System.TimeSpan)

Duration after which this command expires if it has not been sent yet.

-- SendDeadlineTime (System.DateTime?)

Time at which this command expires if it has not been sent yet.

-- SendTrigger (Citrix.Broker.Admin.SDK.MachineCommandTrigger?)

Event that triggers the sending of the command. Valid values are NextContact, Broker, LogOn, Logoff, Disconnect and Reconnect.

-- SessionUid (System.Int64?)

Unique ID of the session this command is targeted at.

-- State (Citrix.Broker.Admin.SDK.MachineCommandState)

Indicates whether the command is pending, sent, expired or canceled.

-- Synchronous (System.Boolean)

Flag that indicates if this is a synchronous command.

-- Uid (System.Int64)

Unique identifier of this machine command.

-- User (System.String)

Name of the user this command is targeted at.

## Related topics

[New-BrokerMachineCommand](#)

[Remove-BrokerMachineCommand](#)

## Parameters

**-Uid**<Int64>

Get only the command with the specified unique identifier.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	false

#### **-Category**<String>

Get only commands targeted to the specified service category.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-CommandName**<String>

Get only commands with the specified command name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-CompletionTime**<DateTime>

Get only commands that entered the Sent, Failed, Canceled or Expired state at the specified time.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineName**<String>

Get only commands targeted to the specified machine.

Required?	false

Default Value	
Accept Pipeline Input?	false

#### **-MachineUid**<Int32>

Get only commands targeted to the specified machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RequestTime**<DateTime>

Get only commands that were requested at the specified time.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SendDeadline**<TimeSpan>

Get only commands that expire after the specified time span.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-SendDeadlineTime**<DateTime>

Get only commands that have the specified deadline time.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SendTrigger**<MachineCommandTrigger>

Get only commands that are due to be sent when the specified trigger occurs. Valid values are NextContact, Broker, LogOn, Logoff, Disconnect and Reconnect.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionUid**<Int64>

Get only commands targeted to the specified session.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-State**<MachineCommandState>

Get only commands in the specified state. Valid values are Pending, Sent, Failed, Canceled and Expired.

Required?	false



Default Value	
Accept Pipeline Input?	false

#### **-User**<String>

Get only commands targeted to the specified user.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
-----------	-------

Default Value	0
Accept Pipeline Input?	false

#### **-SortBy<String>**

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter<String>**

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through `Select-Object`, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

None No parameter is accepted from the input pipeline.

## Return Values

Citrix.Broker.Admin.SDK.MachineCommand

Returns Command objects matching all specified selection criteria.

## Examples

### ----- EXAMPLE 1 -----

Get-BrokerMachineCommand

Returns all pending, canceled, expired and sent commands.

### ----- EXAMPLE 2 -----

Get-BrokerMachineCommand -State Pending

Returns all queued commands.

# Get-BrokerMachineConfiguration

May 10, 2016

Gets machine configurations defined for this site.

## Syntax

```
Get-BrokerMachineConfiguration [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerMachineConfiguration [[-Name] <String>] [-ConfigurationSlotUid <Int32>] [-LeafName <String>] [-Metadata <String>] [-DesktopGroupUid <Int32>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieves machine configurations matching the specified criteria. If no parameters are specified this cmdlet enumerates all machine configurations.

Machine configurations contain binary arrays of settings data that are managed using SDK snap-ins. Each machine configuration is associated with a configuration slot and referenced by Name. The configuration slot restricts the settings that can be held by the machine configuration. For example, only configurations for Citrix User Profile Manager can be associated with the "User Profile Manager" slot.

See `about_Broker_Filtering` for information about advanced filtering options.

----- BrokerMachineConfiguration Object

The machine configuration object returned represents a named collection of related settings values that are applied to a desktop group.

-- ConfigurationSlotUid (System.Int32)

Uid of the associated configuration slot.

-- Description (System.String)

Optional description of the machine configuration.

-- DesktopGroupUids (System.Int32[])

List of desktop group Uids that this machine configuration has been added to.

-- LeafName (System.String)

Name of this machine configuration.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

Map of metadata associated with this machine configuration.

-- Name (System.String)

Unique "SlotName\MachineConfigurationName" for this machine configuration.

-- Policy (System.Byte[])

A binary array of encoded settings.

-- Uid (System.Int32)

Uid of this machine configuration.

## Related topics

[New-BrokerMachineConfiguration](#)

[Set-BrokerMachineConfiguration](#)

[Rename-BrokerMachineConfiguration](#)

[Remove-BrokerMachineConfiguration](#)

[Add-BrokerMachineConfiguration](#)

## Parameters

**-Uid**<Int32>

Get only the machine configuration with the specified unique identifier.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Name**<String>

Get only the machine configuration with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ConfigurationSlotUid**<Int32>

Get only the machine configurations associated with the specified configuration slot.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-LeafName**<String>

Get only the machine configurations that have the specified leaf name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupUid**<Int32>

Get only the machine configurations that have been assigned to the specified desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See `about_Broker_Filtering` for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through `Select-Object`, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

`Citrix.Broker.Admin.SDK.MachineConfiguration`

`Get-BrokerMachineConfiguration` returns an object for each matching machine configuration.

#### Examples

----- **EXAMPLE 1** -----

`Get-BrokerMachineConfiguration`



Retrieves a list of every defined machine configuration.

----- **EXAMPLE 2** -----

Get-BrokerMachineConfiguration -Name Receiver\Engineering

Retrieves the machine configuration named "Receiver\Engineering".

----- **EXAMPLE 3** -----

Get-BrokerMachineConfiguration -Name UPM\\*

Retrieves a list of every machine configuration associated with the configuration slot named "UPM".

----- **EXAMPLE 4** -----

Get-BrokerMachineConfiguration -LeafName "Dept\*"

Retrieves a list of every machine configuration with a LeafName that starts with "Dept", regardless of the associated configuration slot.

# Get-BrokerMachineStartMenuShortcutIcon

May 10, 2016

Retrieves a Start Menu Shortcut icon from the specified machine.

## Syntax

```
Get-BrokerMachineStartMenuShortcutIcon [-MachineName] <String> [-Path] <String> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieves the icon associated with a particular shortcut on a particular machine. This icon is usually used to help create a published application to access the shortcut.

## Related topics

[Get-BrokerMachine](#)

[New-BrokerIcon](#)

## Parameters

**-MachineName**<String>

Specify the name of the machine to use for icon retrieval for the specified shortcut path. The machine can be identified by DNS name, short name, SID, or name of the form domain\machine.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Path**<String>

The location of the shortcut in the specified machine whose icon is being fetched.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

---

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

System.String

Get-BrokerMachineStartMenuShortcutIcon generates a Base64 encoded string containing the icon for the specified shortcut. This can be used as input to New-BrokerIcon cmdlet.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> $shortcuts = Get-BrokerMachineStartMenuShortcuts -MachineName 'MyDomain\MyMachine'
C:\PS> $encodedIconData = Get-BrokerMachineStartMenuShortcutIcon -MachineName 'MyDomain\MyMachine' -Path $shortcuts[0].ShortcutPath
C:\PS> $brokerIcon = New-BrokerIcon -EncodedIconData $encodedIconData
C:\PS> Set-BrokerApplication 'Notepad' -IconUid $brokerIcon.Uid
```

This example retrieves all Start Menu Shortcuts from 'MyDomain\MyMachine', and then the icon for the first shortcut from the returned list. The icon is then associated with a published application called 'Notepad'.

# Get-BrokerMachineStartMenuShortcuts

May 10, 2016

Retrieves the Start Menu Shortcuts from the specified machine.

## Syntax

```
Get-BrokerMachineStartMenuShortcuts [-MachineName] <String> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieves the shortcuts defined for all the start menu items on a particular machine. The shortcuts obtained are from the 'All users' start menu; user-specific shortcuts are not found.

## Related topics

## Parameters

**-MachineName**<String>

Specify the name of the machine to use for shortcut retrieval. The machine can be identified by DNS name, short name, SID, or name of the form domain\machine.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

None or Citrix.Broker.Admin.SDK.StartMenuShortcut

Get-BrokerMachineStartMenuShortcuts generates an array of Citrix.Broker.Admin.SDK.StartMenuShortcut objects.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> $shortcuts = Get-BrokerMachineStartMenuShortcuts -MachineName 'MyDomain\MyMachine'
```

This example retrieves all Start Menu Shortcuts from 'MyDomain\MyMachine'.

# Get-BrokerPowerTimeScheme

May 10, 2016

Gets power management time schemes for desktop groups.

## Syntax

```
Get-BrokerPowerTimeScheme [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerPowerTimeScheme [[-Name] <String>] [-DesktopGroupUid <Int32>] [-Metadata <String>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Finds power time schemes matching the specified search criteria. Each desktop group in the site can have a number of power time schemes associated with it, and these time schemes control how the power states of machines in the group are managed.

If no search criteria are specified all power time schemes for all desktop groups are obtained.

Each power time scheme covers one or more days of the week, and defines which hours of those days are considered peak times and which are off-peak times. In addition, the time scheme defines a pool size value for each hour of the day for the days of the week covered by the time scheme. No one desktop group can be associated with two or more time schemes that cover the same day of the week.

For any day of the week not covered by any power time scheme, it is assumed that all hours are off-peak and no pool size management is required for any of the hours.

For more information about the power policy mechanism and pool size management, see 'help about\_Broker\_PowerManagement'.

----- BrokerPowerTimeScheme Object

The BrokerPowerTimeScheme object represents a power time scheme, defining peak/off-peak hours and idle pool sizes for desktop groups. It contains the following properties:

-- DaysOfWeek (Citrix.Broker.Admin.SDK.TimeSchemeDays)

The days of the week for which this scheme applies to (Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Weekdays, Weekend).

-- DesktopGroupUid (System.Int32)

The desktop group that this time scheme is for.

-- DisplayName (System.String)

The name of this time scheme, as displayed in the Studio console.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

Metadata for this power time scheme.

-- Name (System.String)

The unique name of this time scheme.

-- PeakHours (System.Boolean[])

A set of 24 boolean flag values, one for each hour of the day. The first value in the array relates to midnight to 00:59, the next one to 1 AM to 01:59 and so on, with the last array element relating to 11 PM to 11:59. If the flag is \$true it means that the associated hour of the day is considered a peak time; if it is \$false it means that it is considered off-peak.

-- PoolSize (System.Int32[])

A set of 24 integer values, one for each hour of the day. The first value in the array relates to midnight to 00:59, the next one to 1 AM to 01:59 and so on, with the last array element relating to 11 PM to 11:59. The value defines the number of machines (either as an absolute number or a percentage of the machines in the desktop group) that are to be maintained in a running state, whether they are in use or not. A value of -1 has special meaning: pool size management does not apply during such hours.

-- PoolUsingPercentage (System.Boolean?)

A boolean flag to indicate whether the integer values in the pool size array are to be treated as absolute values (if this value is \$false) or as percentages of the number of machines in the desktop group (if this value is \$true).

-- Uid (System.Int32)

Unique internal identifier of a time scheme.

## Related topics

[New-BrokerPowerTimeScheme](#)

[Set-BrokerPowerTimeScheme](#)

[Remove-BrokerPowerTimeScheme](#)

[Rename-BrokerPowerTimeScheme](#)

## Parameters

**-Uid**<Int32>

Gets only the power time scheme with the specified Uid.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Name**<String>

Gets only power time schemes with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-DesktopGroupUid**<Int32>

Gets only the power time schemes associated with the specified desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False



Accept Pipeline Input?	false
------------------------	-------

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.PowerTimeScheme

Get-BrokerPowerTimeScheme returns all power time schemes that match the specified selection criteria.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Get-BrokerPowerTimeScheme
```

Fetches all known power time schemes for all desktop groups in the site.

##### ----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerPowerTimeScheme -DesktopGroupUid (Get-BrokerDesktopGroup 'Sales Desktops').Uid
```

Fetches all the power time schemes for the desktop group called 'Sales Desktops'.

# Get-BrokerPrivateDesktop

May 10, 2016

Get private desktops configured for this site.

## Syntax

```
Get-BrokerPrivateDesktop [-UId] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerPrivateDesktop [[-MachineName] <String>] [-AgentVersion <String>] [-AssignedClientName <String>] [-AssignedIPAddress <String>] [-ColorDepth <ColorDepth>] [-ControllerDNSName <String>] [-Description <String>] [-DesktopGroupUId <Int32>] [-DNSName <String>] [-HostedMachineId <String>] [-HostedMachineName <String>] [-HostingServerName <String>] [-HypervisorConnectionUId <Int32>] [-IconUId <Int32>] [-InMaintenanceMode <Boolean>] [-IPAddress <String>] [-IsAssigned <Boolean>] [-LastDeregistrationReason <DeregistrationReason>] [-LastDeregistrationTime <DateTime>] [-LastHostingUpdateTime <DateTime>] [-OSType <String>] [-OSVersion <String>] [-PowerState <PowerState>] [-PublishedName <String>] [-RegistrationState <RegistrationState>] [-SecureIcaRequired <Boolean>] [-SID <String>] [-Tag <String>] [-WillShutdownAfterUse <Boolean>] [-AssignedUserSID <String>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet is deprecated, please use the Get-BrokerMachine cmdlet instead.

Retrieve private desktops matching the specified criteria. If no parameters are specified, this cmdlet enumerates all private desktops.

Get-BrokerPrivateDesktop returns configuration information only for private desktops (a DesktopKind of 'Private'). For more state information about desktops, or other types of desktop, use the Get-BrokerMachine cmdlet instead.

For information about advanced filtering options, see [about\\_Broker\\_Filtering](#); for more information about desktops, see [about\\_Broker\\_Desktops](#).

### ----- BrokerPrivateDesktop Object

Private desktops are machines that have been configured with a DesktopKind of 'Private'. They are allocated to either a user/users or a client name/address (but cannot be allocated to both).

-- AgentVersion (System.String)

Version of the Citrix Virtual Delivery Agent (VDA) installed on the desktop.

-- AssignedClientName (System.String)

Client name the desktop has been assigned to.

-- AssignedIPAddress (System.String)

IP Address the desktop has been assigned to.

-- ColorDepth (Citrix.Broker.Admin.SDK.ColorDepth?)

The color depth setting configured on the desktop, possible values are:

\$null, FourBit, EightBit, SixteenBit, and TwentyFourBit.

-- ControllerDNSName (System.String)

The DNS host name of the controller that the desktop is registered to.

-- Description (System.String)

Description of the private desktop.

-- DesktopGroupUid (System.Int32)

Uid of the desktop group the desktop has been assigned to.

-- DNSName (System.String)

The DNS host name of the desktop.

-- HostedMachineId (System.String)

Unique ID within the hosting unit of the target managed desktop.

-- HostedMachineName (System.String)

The friendly name of a hosted desktop as used by its hypervisor. This is not necessarily the DNS name of the desktop.

-- HostingServerName (System.String)

DNS name of the hypervisor that is hosting the desktop if managed.

-- HypervisorConnectionUid (System.Int32?)

The UID of the hypervisor connection that the desktop has been assigned to, if managed.

-- IconUid (System.Int32?)

The UID of the desktop's icon that is displayed in StoreFront. If this is \$null then the desktop will use the icon specified by the desktop group.

-- InMaintenanceMode (System.Boolean)

Denotes whether the desktop is in maintenance mode.

-- IPAddress (System.String)

The IP address of the desktop.

-- IsAssigned (System.Boolean)

Denotes whether a private desktop has been assigned to a user/users, or a client name/address. Users can be assigned explicitly or by assigning on first use of the desktop.

-- LastDeregistrationReason (Citrix.Broker.Admin.SDK.DeregistrationReason?)

The reason for the last deregistration of the desktop with the broker. Possible values are:

AgentShutdown, AgentSuspended, AgentRequested, IncompatibleVersion, AgentAddressResolutionFailed, AgentNotContactable, AgentWrongActiveDirectoryOU, EmptyRegistrationRequest, MissingRegistrationCapabilities, MissingAgentVersion, InconsistentRegistrationCapabilities, NotLicensedForFeature, UnsupportedCredentialSecurityVersion, InvalidRegistrationRequest, SingleMultiSessionMismatch, FunctionalLevelTooLowForCatalog, FunctionalLevelTooLowForDesktopGroup, PowerOff, DesktopRestart, DesktopRemoved, AgentRejectedSettingsUpdate, SendSettingsFailure, SessionAuditFailure, SessionPrepareFailure, ContactLost, SettingsCreationFailure, UnknownError and BrokerRegistrationLimitReached.

-- LastDeregistrationTime (System.DateTime?)

Time of the last deregistration of the desktop from the controller.

-- LastHostingUpdateTime (System.DateTime?)

Time of last update to any hosting data for this desktop reported by the hypervisor connection.

-- MachineName (System.String)

DNS host name of the machine associated with the desktop.

-- OSType (System.String)

A string that can be used to identify the operating system that is running on the desktop.

-- OSVersion (System.String)

A string that can be used to identify the version of the operating system running on the desktop, if known.

-- PowerState (Citrix.Broker.Admin.SDK.PowerState)

The current power state of the desktop. Possible values are: Unmanaged, Unknown, Unavailable, Off, On, Suspended, TurningOn, TurningOff, Suspending, Resuming.

-- PublishedName (System.String)

The name of the desktop that is displayed in StoreFront, if the desktop is published.

-- RegistrationState (Citrix.Broker.Admin.SDK.RegistrationState)

Indicates the registration state of the desktop. Possible values are: Unregistered, Initializing, Registered, AgentError.

-- SecureIcaRequired (System.Boolean?)

Flag indicating whether SecureICA is required or not when starting a session on the desktop.

-- SID (System.String)

Security identifier of the private desktop.

-- Uid (System.Int32)

Unique identifier of the private desktop.

-- WillShutdownAfterUse (System.Boolean)

Flag indicating whether this desktop is tainted and will be shutdown after all sessions on the desktop have ended. This flag should only ever be true on power managed, single-session desktops.

Note: The desktop will not shut down if it is in maintenance mode, but will shut down after the desktop is taken out of maintenance mode.

## Related topics

[Set-BrokerPrivateDesktop](#)

## Parameters

**-Uid**<Int32>

Gets desktops by Uid.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-MachineName**<String>

Gets desktops by machine name (in the form 'domain\machine').

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AgentVersion**<String>

Gets desktops with a specific Citrix Virtual Delivery Agent (VDA) version.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AssignedClientName**<String>

Gets desktops assigned to a specific client name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AssignedIPAddress<String>**

Gets desktops assigned to a specific IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ColorDepth<ColorDepth>**

Gets desktops configured with a specific color depth.

Valid values are FourBit, EightBit, SixteenBit, and TwentyFourBit.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ControllerDNSName<String>**

Gets desktops by the DNS name of the controller they are registered with.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Description<String>**

Gets desktops by description.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopGroupUid<Int32>**

Gets desktops from a desktop group with a specific Uid.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DNSName<String>**

Gets desktops by DNS name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-HostedMachineId<String>**

Gets desktops by the machine id known to the hypervisor.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-HostedMachineName<String>**

Gets desktops by the machine name known to the hypervisor.



Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HostingServerName<String>**

Gets desktops by the name of the hosting hypervisor server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionUid<Int32>**

Gets desktops by the uid of the hosting hypervisor connection.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IconUid<Int32>**

Gets desktops by configured icon. Note that desktops with a \$null IconUid use the icon of the desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-InMaintenanceMode<Boolean>**

Gets desktops by the InMaintenanceMode setting.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-IPAddress**<String>

Get desktops by their IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IsAssigned**<Boolean>

Gets desktops depending on whether they are assigned or not. Private desktops can be assigned to either a user/users or client names/addresses.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastDeregistrationReason**<DeregistrationReason>

Gets desktops whose broker last recorded a specific deregistration reason.

Valid values are \$null, AgentShutdown, AgentSuspended, AgentRequested, IncompatibleVersion, AgentAddressResolutionFailed, AgentNotContactable, AgentWrongActiveDirectoryOU, EmptyRegistrationRequest, MissingRegistrationCapabilities, MissingAgentVersion, InconsistentRegistrationCapabilities, NotLicensedForFeature, UnsupportedCredentialSecurityVersion, InvalidRegistrationRequest, SingleMultiSessionMismatch, FunctionalLevelTooLowForCatalog, FunctionalLevelTooLowForDesktopGroup, PowerOff, DesktopRestart, DesktopRemoved, AgentRejectedSettingsUpdate, SendSettingsFailure, SessionAuditFailure, SessionPrepareFailure, ContactLost, SettingsCreationFailure, UnknownError and BrokerRegistrationLimitReached.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastDeregistrationTime**<DateTime>

Gets desktops by the time that they were last deregistered.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastHostingUpdateTime**<DateTime>

Gets desktops by the time that the hosting information was last updated.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OSType**<String>

Gets desktops by the type of operating system they are running.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OSVersion**<String>

Gets desktops by the version of the operating system they are running.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PowerState**<PowerState>

Gets desktops by power state.

Valid values are Unmanaged, Unknown, Unavailable, Off, On, Suspended, TurningOn, TurningOff, Suspending, and Resuming.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PublishedName**<String>

Gets desktops by published name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RegistrationState**<RegistrationState>

Gets desktops by registration state.

Valid values are Registered, Unregistered, and AgentError.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecureIcaRequired**<Boolean>

Gets desktops configured with a particular SecureIcaRequired setting. Note that the desktop setting of \$null indicates that the desktop group value is used.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-SID<String>**

Gets desktops by machine SID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Tag<String>**

Gets desktops tagged with the given tag.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-WillShutDownAfterUse<Boolean>**

Gets desktops depending on whether they will be automatically shut down when the current session ends or not.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssignedUserSID<String>**

Gets desktops with the given assigned user (specified by SID).

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.

Accept Pipeline Input?	false
------------------------	-------

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.PrivateDesktop

Get-BrokerPrivateDesktop returns an object for each matching private desktop.

## Notes

To compare dates/times, use -Filter and relative comparisons. See about\_Broker\_Filtering for details.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> $list = 'Unmanaged','On','TurningOn','Resuming'
```

```
C:\PS> Get-BrokerPrivateDesktop -Filter { PowerState -in $list } | ft -a DNSName,PowerState
```

Get all private desktops that are turned on, or are turning on (assuming unmanaged desktops are powered on).

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerPrivateDesktop -Tag TestTag
```

Retrieve all private desktops tagged with the 'TestTag' tag.



# Get-BrokerRebootCycle

May 10, 2016

Gets one or more reboot cycles.

## Syntax

```
Get-BrokerRebootCycle -Uid <Int64> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerRebootCycle [-CatalogName <String>] [-CatalogUid <Int32>] [-DesktopGroupName <String>] [-DesktopGroupUid <Int32>] [-EndTime <DateTime>] [-MachinesCompleted <Int32>] [-MachinesFailed <Int32>] [-MachinesInProgress <Int32>] [-MachinesPending <Int32>] [-MachinesSkipped <Int32>] [-Metadata <String>] [-RebootDuration <Int32>] [-StartTime <DateTime>] [-State <RebootCycleState>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Get-BrokerRebootCycle cmdlet is used to enumerate reboot cycles that match all of the supplied criteria.

See about\_Broker\_Filtering for information about advanced filtering options.

----- BrokerRebootCycle Object

The reboot cycle object returned represents a single occurrence of the process of rebooting a portion (or all) of the machines in a desktop group.

-- CatalogName (System.String)

Name of the catalog whose machines are rebooted by this cycle if the cycle is associated with a catalog.

-- CatalogUid (System.Int32?)

Uid of the catalog whose machines are rebooted by this cycle if the cycle is associated with a catalog.

-- DesktopGroupName (System.String)

Name of the desktop group whose machines are rebooted by this cycle.

-- DesktopGroupUid (System.Int32)

Uid of the desktop group whose machines are rebooted by this cycle.

-- EndTime (System.DateTime?)

Time at which this cycle was completed, canceled or abandoned.

-- MachinesCompleted (System.Int32)

Number of machines successfully rebooted by this cycle.

-- MachinesFailed (System.Int32)

Number of machines issued with reboot requests where either the request failed or the operation did not complete within

the allowed time.

-- MachinesInProgress (System.Int32)

Number of machines issued with reboot requests but which have not yet completed the operation.

-- MachinesPending (System.Int32)

Number of outstanding machines to be rebooted during the cycle but on which processing has not yet started.

-- MachinesSkipped (System.Int32)

Number of machines scheduled for reboot during the cycle but which were not processed either because the cycle was canceled or abandoned or because the machine was unavailable for reboot processing throughout the cycle.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

Map of metadata associated with this cycle.

-- RebootDuration (System.Int32)

Approximate maximum number of minutes over which the reboot cycle runs.

-- StartTime (System.DateTime)

Time of day at which this reboot cycle was started.

-- State (Citrix.Broker.Admin.SDK.RebootCycleState)

The execution state of this cycle.

-- Uid (System.Int64)

Unique ID of this reboot cycle.

-- WarningDuration (System.Int32)

Number of minutes to display the warning message for.

-- WarningMessage (System.String)

Warning message to display to users in active sessions prior to rebooting the machine.

-- WarningTitle (System.String)

Title of the warning message dialog.

Related topics

[Start-BrokerRebootCycle](#)

[Stop-BrokerRebootCycle](#)

Parameters

**-Uid**<Int64>

Gets reboot cycles that have the specified Uid.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-CatalogName<String>**

Gets reboot cycles that relate to the named catalog.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-CatalogUid<Int32>**

Gets reboot cycles that relate to the catalog with a particular Uid.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopGroupName<String>**

Gets reboot cycles that relate to the named desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopGroupUid<Int32>**

Gets reboot cycles that relate to the desktop group with a particular Uid.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-EndTime<DateTime>**

Gets reboot cycles that have the specified time at which the reboot cycle was completed, canceled or abandoned.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachinesCompleted<Int32>**

Gets reboot cycles that have the specified count of machines successfully rebooted during the cycle.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachinesFailed<Int32>**

Gets reboot cycles that have the specified count of machines issued with reboot requests where either the request failed or the operation did not complete within the allowed time.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachinesInProgress<Int32>**

Gets reboot cycles that have the specified count of machines issued with reboot requests but which have not yet completed the operation.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachinesPending<Int32>**

Gets reboot cycles that have the specified count of outstanding machines to be rebooted during the cycle but on which processing has not yet started.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachinesSkipped<Int32>**

Gets reboot cycles that have the specified count of machines scheduled for reboot during the cycle but which were not processed either because the cycle was canceled or abandoned or because the machine was unavailable for reboot processing throughout the cycle.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Metadata<String>**

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

**-RebootDuration<Int32>**

Gets reboot cycles that have the specified approximate maximum duration in minutes over which the reboot cycle runs.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-StartTime<DateTime>**

Gets reboot cycles that have the specified time at which the reboot cycle started.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-State<RebootCycleState>**

Gets reboot cycles that have the specified overall state of the reboot cycle. Valid values are Initializing, Active, Completed, Canceled, and Abandoned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount<SwitchParameter>**

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

None Input cannot be piped to this cmdlet.

### Return Values

Citrix.Broker.Admin.SDK.RebootCycle

Returns matching reboot cycles.

### Examples

#### ----- EXAMPLE 1 -----

C:\PS> Get-BrokerRebootCycle  
Enumerate all reboot cycles.

#### ----- EXAMPLE 2 -----

C:\PS> Get-BrokerRebootCycle -State Completed  
Enumerates all reboot cycles that have successfully completed.

#### ----- EXAMPLE 3 -----



```
C:\PS> Get-BrokerRebootCycle -DesktopGroupName CallCenter
```

Enumerates all reboot cycles related to the desktop group named CallCenter.

# Get-BrokerRebootSchedule

May 10, 2016

Gets one or more reboot schedules.

## Syntax

```
Get-BrokerRebootSchedule [-DesktopGroupUid] <Int32> [-Property <String[]>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Get-BrokerRebootSchedule [[-DesktopGroupName] <String>] [-Active <Boolean>] [-Day <RebootScheduleDays>] [-Enabled
<Boolean>] [-Frequency <RebootScheduleFrequency>] [-RebootDuration <Int32>] [-StartTime <TimeSpan>] [-
ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>]
[-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Get-BrokerRebootSchedule cmdlet is used to enumerate desktop group reboot schedules that match all of the supplied criteria.

A reboot schedule can be configured to cause all of the machines in a desktop group to be rebooted at a particular time each day or each week, with the reboot of the individual machines spread out over the duration of the whole reboot cycle. A specific warning message can be configured to be displayed to users who are running sessions on the machines being rebooted. Note that each desktop group can only have a single reboot schedule configured.

See about\_Broker\_Filtering for information about advanced filtering options.

----- BrokerRebootSchedule Object

The reboot schedule object returned represents a regularly scheduled reboot of machines in a desktop group.

-- Active (System.Boolean)

True if there is an active reboot cycle for this schedule, false otherwise.

-- Day (Citrix.Broker.Admin.SDK.RebootScheduleDays)

When the frequency is weekly, day of the week on which the schedule reboot starts.

-- DesktopGroupName (System.String)

Name of the desktop group rebooted by this schedule.

-- DesktopGroupUid (System.Int32)

Uid of the desktop group rebooted by this schedule.

-- Enabled (System.Boolean)

True if this schedule is currently enabled, false otherwise.

-- Frequency (Citrix.Broker.Admin.SDK.RebootScheduleFrequency)

Whether the schedule runs daily or weekly.

-- RebootDuration (System.Int32)

Approximate maximum number of minutes over which the scheduled reboot cycle runs.

-- StartTime (System.TimeSpan)

Time of day at which the scheduled reboot cycle starts.

-- WarningDuration (System.Int32)

Number of minutes to display the warning message for.

-- WarningMessage (System.String)

Warning message to display to users in active sessions prior to rebooting the machine.

-- WarningTitle (System.String)

Title of the warning message dialog.

## Related topics

[Set-BrokerRebootSchedule](#)

[New-BrokerRebootSchedule](#)

[Remove-BrokerRebootSchedule](#)

[Get-BrokerRebootCycle](#)

## Parameters

**-DesktopGroupUid**<Int32>

Gets the reboot schedule for the desktop group having this Uid.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-DesktopGroupName**<String>

Gets the reboot schedule for the desktop group having this name.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-Active**<Boolean>

Gets desktop group reboot schedules according to whether they are currently active or not. A schedule is active if there is a reboot cycle currently running that was started as a result of the schedule.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Day**<RebootScheduleDays>

Gets the reboot schedules set to run on the specified day (one of Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Enabled**<Boolean>

Gets the reboot schedules with the specified Enabled value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Frequency**<RebootScheduleFrequency>

Gets the reboot schedules with the specified frequency (either Weekly or Daily).

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-RebootDuration**<Int32>

Gets the reboot schedules with the specified duration.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-StartTime**<TimeSpan>

Gets the reboot schedules with the specified start time (HH:MM).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250

Accept Pipeline Input?	false
------------------------	-------

### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None Input cannot be piped to this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.RebootSchedule

Returns matching reboot schedules.

#### Examples

##### ----- **EXAMPLE 1** -----

C:\PS> Get-BrokerRebootSchedule  
Enumerates all of the reboot schedules.

##### ----- **EXAMPLE 2** -----

C:\PS> Get-BrokerRebootSchedule -Enabled \$false -Frequency Daily  
Enumerates all disabled reboot schedules that are scheduled to run daily.

##### ----- **EXAMPLE 3** -----

C:\PS> Get-BrokerRebootSchedule -DesktopGroupUid 11  
Returns the unique reboot schedule for the desktop group having the Uid 11.

# Get-BrokerRemotePCAccount

May 10, 2016

Get RemotePCAccount entries configured for this site.

## Syntax

```
Get-BrokerRemotePCAccount -Uid <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerRemotePCAccount [-AllowSubfolderMatches <Boolean>] [-CatalogUid <Int32>] [-OU <String>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieves RemotePCAccounts matching the specified criteria. If no parameters are specified this cmdlet enumerates all RemotePCAccounts. Each RemotePCAccount object defines a set of machines either by machine name patterns or by where the machines are placed in Active Directory, and which RemotePC catalog the machines are to be associated with when they are discovered.

----- BrokerRemotePCAccount Object

RemotePCAccounts define a set of machines either by machine name patterns or by where the machines are placed in Active Directory, and which RemotePC catalog the machines are to be associated with when they are discovered.

-- AllowSubfolderMatches (System.Boolean)

Specifies whether machines subfolders of specified AD OUs are to be considered part of the RemotePCAccount.

-- CatalogUid (System.Int32)

The Uid of the RemotePC catalog to which machines in the RemotePCAccount automatically join during registration.

-- MachinesExcluded (System.String[])

A list of machines which are to be excluded from the RemotePCAccount. Wildcard matching is supported.

-- MachinesIncluded (System.String[])

A list of machines which are to be included in the RemotePCAccount. Wildcard matching is supported.

-- OU (System.String)

Machines within this specified AD OU are considered part of the RemotePCAccount, unless they are in they match the MachinesExcluded

-- Uid (System.Int32)

The Uid of the RemotePCAccount object.

## Related topics

[New-BrokerRemotePCAccount](#)



## Set-BrokerRemotePCAccount

## Remove-BrokerRemotePCAccount

### Parameters

**-Uid**<Int32>

Gets the RemotePCAccount with the specified unique ID.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-AllowSubfolderMatches**<Boolean>

Gets RemotePCAccounts with the specified value of AllowSubfolderMatches.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-CatalogUid**<Int32>

Gets RemotePCAccounts belonging to the specified Remote PC catalog.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-OU**<String>

Gets the RemotePCAccount with the specified OU.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-Sort By**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
-----------	-------

Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter<String>**

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through `Select-Object`, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.RemotePCAccount

Get-BrokerRemotePCAccount returns an object for each matching RemotePCAccount.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerRemotePCAccount
```

Find all RemotePCAccounts.

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerRemotePCAccount -CatalogUid 42
```

Find RemotePCAccounts belonging to Remote PC catalog 42.

# Get-BrokerResource

May 10, 2016

Gets resources that a user can broker connections to.

## Syntax

```
Get-BrokerResource [-User] <String> [-Groups <String[]>] [-ClientName <String>] [-ClientIP <String>] [-ViaAG <Boolean>] [-SmartAccessTags <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieve a list of resources that a user has access to, taking into account the site access policy, configuration of desktop groups, assignments, entitlements, and applications.

What a user has access to depends on a number of attributes:

- User's name or security identifier.
- Groups that the user is a member of (names or security identifiers).
- IP address of the client the user connects from.
- Name of the client that the user connects from.
- Whether the user is connecting via Citrix Access Gateway.
- SmartAccess tags when connecting via Citrix Access Gateway.

You must always specify the user's name or security identifier, but you will not always be able to predict what some of the other values will be. By omitting these values the corresponding access checks are ignored.

Consider for example, a site configuration that uses IP address ranges to allow access to private desktop A when connecting from the local network and private desktop B when connecting from home. Running this cmdlet without specifying a client IP address would return both A and B.

The output of this cmdlet depends on the available resources:

- Assigned private desktops are returned as PrivateDesktop objects.
- Shared desktops are returned as EntitlementPolicyRule objects.
- Assign-On-First-Use desktops that have not been assigned yet are returned as AssignmentPolicyRule objects.
- Application resources produce Application objects.

If more than one type of resource is available, the output pipeline contains a mixture of the above objects, in no particular order.

Only resources accessible based on the specified parameters, and visible to the administrator running this cmdlet are returned.

## Related topics

## Parameters

### **-User**<String>

Gets resources given the specified user name or security identifier.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Groups**<String[]>

Get resources accessible given a list of group names or security identifiers.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Client Name**<String>

Get resources given the specified client name.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Client IP**<String>

Get resources given the specified client IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ViaAG**<Boolean>

Gets resources given the specified ViaAG setting.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-SmartAccessTags**<String[]>

Get resources given the specified SmartAccess tags.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

None You cannot pipe input into this cmdlet.

### Return Values

Citrix.Broker.Admin.SDK.PrivateDesktop

Get-BrokerResource returns a PrivateDesktop for each accessible assigned private desktop.Citrix.Broker.Admin.SDK.EntitlementPolicyRule

Get-BrokerResource returns an EntitlementPolicyRule object for each accessible entitlement to a shared desktop.Citrix.Broker.Admin.SDK.AssignmentPolicyRule

Get-BrokerResource returns an AssignmentPolicyRule object for each accessible Assign-On-First-Use desktop.Citrix.Broker.Admin.SDK.Application

Get-BrokerResource returns an Application object for each accessible application.

## Examples

### ----- EXAMPLE 1 -----

Get-BrokerResource -User MYDOMAIN\User1 -Group MYDOMAIN\Accounts,MYDOMAIN\Managers  
List resources visible by User1 assuming membership of a couple of groups.

### ----- EXAMPLE 2 -----

```
[int[]]$groups = (Get-BrokerResource -User MYDOMAIN\User1 | %{ $_.DesktopGroupUid })
Get-BrokerDesktopGroup -Filter { Uid -in $groups } -Property Uid,Name
```

Get all of the desktop groups supporting the resources accessible by User1, outputting the uid and name of each desktop group.



# Get-BrokerScopedObject

May 10, 2016

Gets the details of the scoped objects for the Broker Service.

## Syntax

```
Get-BrokerScopedObject -ScopeId <Guid> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerScopedObject [-Description <String>] [-ObjectId <String>] [-ObjectName <String>] [-ObjectType
<ScopedObjectType>] [-ScopeName <String>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-
SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns a list of directly scoped objects including the names and identifiers of both the scope and object as well as the object description for display purposes.

There will be at least one result for every directly scoped object. When an object is associated with multiple scopes the output contains one result per scope duplicating the object details.

No records are returned for the All scope, though if an object is not in any scope a result with a null ScopeId and ScopeName will be returned.

----- BrokerScopedObject Object

A scoped, or scopeable object configured in the Broker.

-- Description (System.String)

Description of the object (possibly \$null if the object type does not have a description).

-- ObjectId (System.String)

Unique identifier of the object.

-- ObjectName (System.String)

Display name of the object.

-- ObjectType (Citrix.Broker.Admin.SDK.ScopedObjectType)

Type of the object this entry relates to.

-- ScopeId (System.Guid?)

Specifies the unique identifier of the scope.

-- ScopeName (System.String)

Specifies the display name of the scope.

## Related topics

## Parameters

### **-ScopeId**<Guid>

Gets scoped object entries for the given scope identifier.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Description**<String>

Gets scoped object entries for objects with the specified description.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ObjectId**<String>

Gets scoped object entries for objects with the specified object identifier.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ObjectName**<String>

Gets scoped object entries for objects with the specified object identifier.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ObjectType**<ScopedObjectType>

Gets scoped object entries for objects of the given type.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ScopeName**<String>

Gets scoped object entries with the given scope name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	

Accept Pipeline Input?	false
------------------------	-------

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Return Values

Citrix.Broker.Sdk.ScopedObject

The Get-BrokerScopedObject command returns an object containing the following properties:

ScopeId <Guid?>

Specifies the unique identifier of the scope.

ScopeName <String>

Specifies the display name of the scope.

ObjectType <ScopedObjectType>

Type of the object this entry relates to.

ObjectId <String>

Unique identifier of the object.

ObjectName <String>

Display name of the object

Description <String>

Description of the object (possibly \$null if the object type does not have a description).

Notes

If the command fails, the following errors can be returned.

Error Codes

-----

UnknownObject

One of the specified objects was not found.

PartialData

Only a subset of the available data was returned.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

CouldNotQueryDatabase

The query required to get the database was not defined.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Get-BrokerScopedObject -ObjectType Scheme

```
ScopeId : eff6f464-f1ee-4442-add3-99982e0cec01
ScopeName : Sales
ObjectType : Scheme
ObjectId : cd4174ee-9e4b-4e57-b126-9dbf757fe493
ObjectName : MyExampleScheme
Description : Test scheme
```

ScopeId : 304e0fa7-d390-47f0-a94f-7e956a324c41  
ScopeName : Finance  
ObjectType : Scheme  
ObjectId : cd4174ee-9e4b-4e57-b126-9dbf757fe493  
ObjectName : MyExampleScheme  
Description : Test scheme

ScopeId :  
ScopeName :  
ObjectType : Scheme  
ObjectId : 5062e46b-71bc-4ac9-901a-30fe6797e2f6  
ObjectName : AnotherScheme  
Description : Another scheme in no scopes

Gets all of the scoped objects with type Scheme. The example output shows a scheme object (MyExampleScheme) in two scopes Sales and Finance, and another scheme (AnotherScheme) that is not in any scope. The ScopeId and ScopeName values returned are null in the final record.

# Get-BrokerServiceAddedCapability

May 10, 2016

Gets any added capabilities for the Broker Service on the controller.

## Syntax

```
Get-BrokerServiceAddedCapability [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables updates to the Broker Service on the controller to be detected.

There is no requirement for a database connection to be configured in order for this command to be used.

## Related topics

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

System.String

String containing added capabilities.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

DatabaseError

An error occurred in the service while attempting a database operation.



#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS>Get-BrokerServiceAddedCapability
```

Get the added capabilities of the Broker Service.

# Get-BrokerServiceInstance

May 10, 2016

Gets the service instance entries for the Broker Service.

## Syntax

```
Get-BrokerServiceInstance [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns service interfaces published by instances of the Broker Service. Each instance of a service publishes multiple interfaces with distinct interface types, and each of these interfaces is represented as a ServiceInstance object. Service instances can be used to register the service with a central configuration service so that other services can use the functionality.

You do not need to configure a database connection to use this command.

## Related topics

[Get-BrokerServiceStatus](#)

[Reset-BrokerServiceGroupMembership](#)

## Parameters

**-AdminAddress** <String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Broker.Sdk.ServiceInstance

The Get-BrokerServiceInstance command returns an object containing the following properties.

ServiceGroupUid <Guid>

Specifies the unique identifier for the service group of which the service is a member.

ServiceGroupName <String>

Specifies the name of the service group of which the service is a member.

ServiceInstanceUID <Guid>

Specifies the unique identifier for registered service instances, which are service instances held by and obtained from a central configuration service. Unregistered service instances do not have unique identifiers.

ServiceType <String>

Specifies the service instance type. For this service, the service instance type is always Broker.

Address

Specifies the address of the service instance. The address can be used to access the service and, when registered in the central configuration service, can be used by other services to access the service.

Binding

Specifies the binding type that must be used to communicate with the service instance. In this release of XenDesktop, the binding type is always 'wcf\_HTTP\_kerb'. This indicates that the service provides a Windows Communication Foundation endpoint that uses HTTP binding with integrated authentication.

Version

Specifies the version of the service instance. The version number is used to ensure that the correct versions of the services are used for communications.

ServiceAccount <String>

Specifies the Active Directory account name for the machine on which the service instance is running. The account name is used to provide information about the permissions required for interservice communications.

ServiceAccountSid <String>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

InterfaceType <String>

Specifies the interface type. Each service can provide multiple service instances, each for a different purpose, and the interface defines the purpose. Available interfaces are:

SDK - for PowerShell operations

InterService - for operations between different services

Peer - for communications between services of the same type

Metadata <Citrix.Broker.Sdk.Metadata[]>

The collection of metadata associated with registered service instances, which are service instances held by and obtained from a central configuration service. Metadata is not stored for unregistered service instances.

Notes

If the command fails, the following errors can be returned.

Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- **EXAMPLE 1** -----

**c:\PS>Get-BrokerServiceInstance**

Get all instances of the Broker Service running on the specified machine. For remote services, use the AdminAddress parameter to define the service for which the interfaces are required. If the AdminAddress parameter has not been specified for the runspace, service instances running on the local machine are returned.

# Get-BrokerServiceStatus

May 10, 2016

Gets the current state of the Broker Service on the controller.

## Syntax

```
Get-BrokerServiceStatus [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables the status of the Broker Service on the controller to be determined. If the service has multiple data stores it will return the overall state as an aggregate of all the data store states. For example, if the site data store status is OK and the secondary data store status is DBUnconfigured then it will return DBUnconfigured. Before using this command, you don't have to configure the database connection to the Service.

## Related topics

[Set-BrokerDBConnection](#)

[Test-BrokerDBConnection](#)

[Get-BrokerDBConnection](#)

[Get-BrokerDBSchema](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Get-BrokerServiceStatus command returns an object containing the status of the Broker Service together with extra diagnostics information.

#### DBUnconfigured

The Broker Service does not have a database connection configured.

#### DBRejectedConnection

The database rejected the logon attempt from the Broker Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

#### InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the Broker Service schema has not been added to the database.

#### DBNotFound

The specified database could not be located with the configured connection string.

#### DBMissingOptionalFeature

The Broker is connected to a database that is valid, but it does not have the full functionality required for optimal performance. Upgrading the database is advisable.

#### DBMissingMandatoryFeature

The Broker is connected to a database that is valid, but it does not have the full functionality required so the Broker cannot function. Upgrading the database is required.

#### DBNewerVersionThanService

The version of the Broker Service currently in use is incompatible with the version of the Broker Service schema on the database. Upgrade the Broker Service to a more recent version.

#### DBOlderVersionThanService

The version of the Broker Service schema on the database is incompatible with the version of the Broker Service currently in use. Upgrade the database schema to a more recent version.

#### DBVersionChangeInProgress

A database schema upgrade is currently in progress.

#### OK

The Broker Service is running and is connected to a database containing a valid schema.

#### PendingFailure

Connectivity between the Broker Service and the database has been lost. This may be a transitory network error, but may indicate a loss of connectivity that requires administrator intervention.

#### Failed

Connectivity between the Broker and the database has been lost for an extended period of time, or has failed due to a configuration problem. The Broker service cannot operate while its connection to the database is unavailable.

## Unknown

The service status cannot be determined.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

### PermissionDenied

You do not have permission to execute this command.

### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

### CommunicationError

There was a problem communicating with the remote service.

### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

C:\PS>Get-BrokerServiceStatus

Get the current status of the Broker Service.

# Get-BrokerSession

May 10, 2016

Gets a list of sessions.

## Syntax

```
Get-BrokerSession [-Uid] <Int64> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerSession [[-SessionKey] <Guid>] [-AgentVersion <String>] [-ApplicationInUse <String>] [-AppState
<SessionAppState>] [-AppStateLastChangeTime <DateTime>] [-AutonomouslyBrokered <Boolean>] [-BrokeringDuration
<Int32>] [-BrokeringTime <DateTime>] [-BrokeringUserName <String>] [-BrokeringUserSID <String>] [-CatalogName
<String>] [-ClientAddress <String>] [-ClientName <String>] [-ClientPlatform <String>] [-ClientProductId <Int32>] [-ClientVersion
<String>] [-ConnectedViaHostName <String>] [-ConnectedViaIP <String>] [-ConnectionMode <ConnectionMode>] [-
ControllerDNSName <String>] [-DesktopGroupName <String>] [-DesktopGroupUid <Int32>] [-DesktopKind <DesktopKind>] [-
DesktopSID <String>] [-DesktopUid <Int32>] [-DeviceId <String>] [-DNSName <String>] [-EstablishmentDuration <Int32>] [-
EstablishmentTime <DateTime>] [-HardwareId <String>] [-Hidden <Boolean>] [-HostedMachineName <String>] [-
HostingServerName <String>] [-HypervisorConnectionName <String>] [-ImageOutOfDate <Boolean>] [-InMaintenanceMode
<Boolean>] [-IPAddress <String>] [-IsAnonymousUser <Boolean>] [-IsPhysical <Boolean>] [-LaunchedViaHostName
<String>] [-LaunchedViaIP <String>] [-LogoffInProgress <Boolean>] [-LogonInProgress <Boolean>] [-MachineName <String>]
[-MachineSummaryState <DesktopSummaryState>] [-MachineUid <Int32>] [-Metadata <String>] [-OSType <String>] [-
PersistUserChanges <PersistUserChanges>] [-PowerState <PowerState>] [-Protocol <String>] [-ProvisioningType
<ProvisioningType>] [-ReceiverIPAddress <String>] [-ReceiverName <String>] [-SecureIcaActive <Boolean>] [-SessionId
<Int32>] [-SessionState <SessionState>] [-SessionStateChangeTime <DateTime>] [-SessionSupport <SessionSupport>] [-
SessionType <SessionType>] [-StartTime <DateTime>] [-UntrustedUserName <String>] [-UserFullName <String>] [-
UserName <String>] [-UserSID <String>] [-UserUPN <String>] [-ApplicationUid <Int32>] [-SharedDesktopUid <Int32>] [-
ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>]
[-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieves sessions matching all the specified criteria. If no parameters are specified this cmdlet enumerates all sessions.

----- BrokerSession Object

The session object returned represents a session on a machine in the site. The session could be for a desktop or application

-- AgentVersion (System.String)

Version of the Citrix Virtual Delivery Agent (VDA) installed on the machine.

-- ApplicationsInUse (System.String[])

List of applications in use in the session. Applications are identified by their administrative name.

-- AppState (Citrix.Broker.Admin.SDK.SessionAppState)

The app state of the session. Valid values are PreLogon, PreLaunched, Active, Desktop, Lingering and NoApps.

-- AppStateLastChangeTime (System.DateTime?)



The time when the session entered the current app state.

-- AutonomouslyBrokered (System.Boolean)

Indicates whether this is an HDX session established by direct connection without being brokered.

-- BrokeringDuration (System.Int32?)

Time taken to broker the session (in milliseconds).

-- BrokeringTime (System.DateTime?)

Time at which the session was brokered.

-- BrokeringUserName (System.String)

The user name of the brokering user.

-- BrokeringUserSID (System.String)

The SID of the brokering user.

-- CatalogName (System.String)

The name of the catalog that the machine hosting the session is assigned to.

-- ClientAddress (System.String)

The IP address of the client connected to the session.

-- ClientName (System.String)

The host name of the client connected to the session.

-- ClientPlatform (System.String)

The name of client platform, as indicated by client product ID.

-- ClientProductId (System.Int32?)

The product ID of the client connected to the session.

-- ClientVersion (System.String)

The version of the Citrix Receiver running on the client connected to the session.

-- ConnectedViaHostName (System.String)

The host name of the incoming connection. This is usually a gateway, router or client.

-- ConnectedViaIP (System.String)

The IP address of the incoming connection This is usually a gateway, router or client.

-- ConnectionMode (Citrix.Broker.Admin.SDK.ConnectionMode?)

The way in which the most recent connection to the session was established. Valid modes are:

- o Brokered: established through the XenDesktop Broker (for example, using StoreFront).
- o Unbrokered: direct connection (for example, using the console or direct RDP/HDX).
- o LeasedConnection: established through the XenDesktop Broker using a connection lease.
- o VdaHighAvailabilityMode: direct connection while VDA in high-availability mode.
- o ThirdPartyBroker: established through a third-party Broker.
- o ThirdPartyBrokerWithLicensing: established and licensed through a third-party Broker.

-- ControllerDNSName (System.String)

The DNS host name of the controller that the session's hosting machine is registered with.

-- DesktopGroupName (System.String)

Name of the desktop group of the machine the session is on.

-- DesktopGroupUid (System.Int32)

UID of the desktop group of the machine the session is on.

-- DesktopKind (Citrix.Broker.Admin.SDK.DesktopKind)

Indicates if the session is shared or private.

-- DesktopSID (System.String)

The Windows SID of the machine the session is on.

-- DesktopUid (System.Int32)

For a desktop session, the unique identifier of the desktop.

-- DeviceId (System.String)

Unique identifier for the client device that has most recently been associated with the session.

-- DNSName (System.String)

The DNS host name of the machine hosting the session.

-- EstablishmentDuration (System.Int32?)

Duration that it took to establish the session.

-- EstablishmentTime (System.DateTime?)

Time at which the session was established.

-- HardwareId (System.String)

Unique identifier for the client hardware that has been most recently associated with the session.

-- Hidden (System.Boolean)

Flag to indicate if the session is currently hidden from the user and not to be reconnected to.

-- HostedMachineName (System.String)

The friendly name of a hosted machine running the session, as used by its hypervisor. This does not necessarily match either the DNS or AD name of the machine.

-- HostingServerName (System.String)

DNS name of the hypervisor that is hosting the machine hosting the session.

-- HypervisorConnectionName (System.String)

The name of the hypervisor connection that the machine hosting the session has been assigned to.

-- ImageOutOfDate (System.Boolean?)

Denotes whether the VM image for a hosted machine is out of date and due to be updated to a new master image when the machine next reboots.

-- InMaintenanceMode (System.Boolean)

Denotes whether the machine hosting the session is in maintenance mode.

-- IPAddress (System.String)

The IP address of the machine hosting the session.

-- IsAnonymousUser (System.Boolean)

Indicates whether the session was established anonymously (without user credentials), in this case a temporary local user account on the machine is used.

-- IsPhysical (System.Boolean)

This value is false if the machine hosting the session can be power managed, and true otherwise

-- LaunchedViaHostName (System.String)

The host name of the StoreFront server used to launch the session.

-- LaunchedViaIP (System.String)

The IP address of the StoreFront server used to launch the session.

-- LogoffInProgress (System.Boolean)

Indicates whether the session is in the process of being logged off.

-- LogonInProgress (System.Boolean)

Indicates whether the session is still executing user logon processing or not.

-- MachineName (System.String)

DNS host name of the machine hosting the session.

-- MachineSummaryState (Citrix.Broker.Admin.SDK.DesktopSummaryState)

The summary state of the machine (will be Unregistered, Disconnected, or InUse)

-- MachineUid (System.Int32)

UID of the machine hosting the session.

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

Map of metadata for this session.

-- OSType (System.String)

A string that can be used to identify the operating system that is running on the machine hosting the session.

-- PersistUserChanges (Citrix.Broker.Admin.SDK.PersistUserChanges)

Describes whether/how the user changes are persisted. Possible values are:

- o OnLocal - Persist the user changes locally.

- o Discard - Discard user changes.

- o OnPvd - Persist user changes on the Citrix Personal vDisk.

-- PowerState (Citrix.Broker.Admin.SDK.PowerState)

The current power state of the machine hosting the session. Possible values are: Unmanaged, Unknown, Unavailable, On, Suspended, TurningOn, TurningOff, Suspending and Resuming.

-- Protocol (System.String)

The protocol that the session is using, can be HDX, RDP, or Console. Console sessions on XenDesktop 5 VDAs appear with a blank protocol.

-- ProvisioningType (Citrix.Broker.Admin.SDK.ProvisioningType)

Describes how the machine hosting the session was provisioned, possible values are:

- o Manual: No provisioning.

- o PVS: Machine provisioned by PVS (may be physical, blade, VM,...)

- o MCS: Machine provisioned by MCS (machine must be VM)

-- ReceiverIPAddress (System.String)

The IP address of the client as supplied by Receiver (for example, StoreFront) when the session was launched, or reconnected.

-- ReceiverName (System.String)

The name of the client as supplied by Receiver (for example, StoreFront) when the session was launched, or reconnected.

-- SecureIcaActive (System.Boolean?)

Indicates whether SecureICA is active on the session.

-- SessionId (System.Int32)

Deprecated. A unique identifier that Remote Desktop Services uses to track the session but it is only unique on that machine.

-- SessionKey (System.Guid)

GUID that provides a unique identifier for this session.

-- SessionState (Citrix.Broker.Admin.SDK.SessionState)

The state of the session. Valid values are Connected, Active or Disconnected.

For a session on a machine with functional level below L7, the additional states PreparingSession, Reconnecting, NonBrokeredSession, Other, and Unknown can also occur.

-- SessionStateChangeTime (System.DateTime)

The time of the most recent state change for the session.

-- SessionSupport (Citrix.Broker.Admin.SDK.SessionSupport)

Indicates if the machine hosting the session supports multiple or single sessions.

-- SessionType (Citrix.Broker.Admin.SDK.SessionType)

Indicates if this is an Application or Desktop session.

-- SmartAccessTags (System.String[])

The Smart Access tags for this session.

-- StartTime (System.DateTime?)

Indicates when the session was started.

-- Uid (System.Int64)

Unique identifier of this session.

-- UntrustedUserName (System.String)

The name of the logged-on user reported directly from the machine (in the form DOMAIN\user). This may be useful where the user is logged in to a non-domain account, however the name cannot be verified and must therefore be considered untrusted.

-- UserFullName (System.String)

The full name of the user.

-- UserName (System.String)

The name of the user.

-- UserSID (System.String)

The user's Windows SID.

-- UserUPN (System.String)

The user's User Principal Name

## Related topics

[Disconnect-BrokerSession](#)

[Stop-BrokerSession](#)

[Get-BrokerDesktop](#)

## Parameters

**-Uid**<Int64>

Get session by its Uid.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-SessionKey**<Guid>

Gets session having the specified unique key.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AgentVersion**<String>

Gets sessions with a specific Virtual Desktop Agent version.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-ApplicationInUse<String>**

Gets sessions running specific applications (identified by their SDK Name property).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AppState<SessionAppState>**

Get sessions by their app state.

Valid values are PreLogon, PreLaunched, Active, Desktop, Linging and NoApps.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AppStateLastChangeTime<DateTime>**

Get sessions by their app state change time.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AutonomouslyBrokered<Boolean>**

Gets sessions according to whether they are autonomously brokered or not. Autonomously brokered sessions are HDX sessions established by direct connection without being brokered.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-BrokeringDuration**<Int32>

Gets session with a specific time taken to broker. In general, Citrix recommends using -Filter and relative comparisons.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-BrokeringTime**<DateTime>

Get sessions brokered at a specific time. In general, Citrix recommends using -Filter and relative comparisons.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-BrokeringUserName**<String>

Get sessions by brokering user.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-BrokeringUserSID**<String>

Get sessions by brokering user SID.

Required?	false
Default Value	



Accept Pipeline Input?	false
------------------------	-------

#### **-CatalogName**<String>

Gets sessions on machines from a specific catalog name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ClientAddress**<String>

Get sessions by client IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ClientName**<String>

Get sessions by client name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ClientPlatform**<String>

Get sessions by client platform.

Required?	false
Default Value	
Accept Pipeline Input?	false

--	--

#### **-ClientProductId<Int32>**

Get sessions by client product ID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ClientVersion<String>**

Get sessions by client version.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ConnectedViaHostName<String>**

Get sessions by host name of the incoming connection. This is usually a proxy or Citrix Access Gateway server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ConnectedViaIP<String>**

Get sessions by IP address of the incoming connection.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ConnectionMode**<ConnectionMode>

Gets sessions by the way in which the most recent connection to the session was established.

Valid modes are Brokered, Unbrokered, LeasedConnection, VdaHighAvailabilityMode, ThirdPartyBroker, and ThirdPartyBrokerWithLicensing.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ControllerDNSName**<String>

Gets sessions that are hosted on machines which are registered with a specific controller.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupName**<String>

Gets sessions from a desktop group with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupUid**<Int32>

Gets sessions from a desktop group with the specified UID.

Required?	false
Default Value	
Accept Pipeline Input?	false

---

**-DesktopKind**<DesktopKind>

Gets sessions on a desktop of a particular kind.

Valid values are Private and Shared.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopSID**<String>

Get sessions by desktop SID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopUid**<Int32>

Get sessions by desktop Uid.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DeviceId**<String>

Get sessions by client device id.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-DNSName<String>**

Gets sessions by their machine's DNS name.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-EstablishmentDuration<Int32>**

Gets sessions which took a specific time to establish. In general, Citrix recommends using -Filter and relative comparisons.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-EstablishmentTime<DateTime>**

Gets sessions which became established at a particular time. In general, Citrix recommends using -Filter and relative comparisons.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-HardwareId<String>**

Get sessions by client hardware id.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Hidden**<Boolean>

Get sessions by whether they are hidden or not. Hidden sessions are treated as though they do not exist when brokering sessions; a hidden session cannot be reconnected to, but a new session may be launched using the same entitlement.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HostedMachineName**<String>

Gets sessions by their machine's name as known to its hypervisor.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HostingServerName**<String>

Gets sessions hosted by a machine with a specific name of the hosting hypervisor server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionName**<String>

Gets sessions hosted by a machine with a specific name of the hosting hypervisor connection.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ImageOutOfDate<Boolean>**

Gets sessions hosted by a machine with a specific ImageOutOfDate setting.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-InMaintenanceMode<Boolean>**

Gets sessions hosted by a machine with a specific InMaintenanceMode setting.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IPAddress<String>**

Gets sessions hosted by a machine with a specific IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IsAnonymousUser<Boolean>**

Gets sessions depending on whether they were established anonymously (\$true) or not (\$false). An anonymous session is established without user credentials and a temporary local user account is used.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-IsPhysical**<Boolean>

Gets sessions hosted on machines where the flag indicating if the machine can be power managed by the Citrix Broker Service matches the requested value. Where the power state of the machine cannot be controlled, specify \$true, otherwise \$false.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-LaunchedViaHostName**<String>

Get sessions by the host name of the Web Interface server from which a user launches a session.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-LaunchedViaIP**<String>

Get sessions by the IP address of the Web Interface server from which a user launches a session.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-LogoffInProgress**<Boolean>

Gets sessions by whether they are in the process of being logged off or not.

Required?	false
Default Value	
Accept Pipeline Input?	false



#### **-LogonInProgress**<Boolean>

Gets sessions by whether they are still executing user logon processing or not.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineName**<String>

Gets sessions by their machine name (in the form DOMAIN\machine).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineSummaryState**<DesktopSummaryState>

Gets sessions on a machine with a specific summary state.

Valid values are Off, Unregistered, Available, Disconnected, Preparing, and InUse.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineUid**<Int32>

Gets sessions on a machine with the specified UID.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-OSType**<String>

Gets sessions with a specific type of operating system.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-PersistUserChanges**<PersistUserChanges>

Gets sessions where the user changes are persisted in a particular manner. Values can be:

- o OnLocal - User changes are persisted locally.
- o Discard - User changes are discarded.
- o OnPvd - User changes are persisted on the Pvd.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-PowerState**<PowerState>

Gets sessions on machines in the specified power state.

Valid values are Unmanaged, Unknown, Unavailable, On, Suspended, TurningOn, TurningOff, Suspending, and Resuming.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Protocol**<String>

Get sessions by connection protocol. Valid values are HDX, RDP, or Console.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ProvisioningType**<ProvisioningType>

Gets sessions hosted on machines provisioned in a particular manner. Values can be:

- o Manual - No automated provisioning.
- o PVS - Machine provisioned by PVS (machine may be physical, blade, VM,...).
- o MCS - Machine provisioned by MCS (machine must be VM).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReceiverIPAddress**<String>

Gets sessions with the specified client IP address supplied by Receiver (for example, StoreFront) when the session was launched, or reconnected.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ReceiverName**<String>

Gets sessions with the specified client name supplied by Receiver (for example, StoreFront) when the session was launched, or reconnected.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-SecureIcaActive**<Boolean>

Get sessions by their use of SecureICA.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-SessionId**<Int32>

Deprecated.

Gets sessions by session ID, a unique identifier that Remote Desktop Services uses to track the session but it is only unique on that machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-SessionState**<SessionState>

Get sessions by their state.

Valid values are Other, PreparingNewSession, Connected, Active, Disconnected, Reconnecting, NonBrokeredSession, and Unknown.

Required?	false

Default Value	
Accept Pipeline Input?	false

#### **-SessionStateChangeTime**<DateTime>

Get sessions by their last state change time. In general, Citrix recommends using -Filter and relative comparisons.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionSupport**<SessionSupport>

Gets sessions hosted on machines which support the required pattern of sessions. Values can be:

- o SingleSession - Single-session only machine.
- o MultiSession - Multi-session capable machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionType**<SessionType>

Get sessions by their type.

Valid values are Application and Desktop.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-StartTime**<DateTime>

Get sessions by their start time. In general, Citrix recommends using -Filter and relative comparisons.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UntrustedUserName**<String>

Gets sessions by the untrusted user name reported directly from the machine (in the form DOMAIN\user).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UserFullName**<String>

Gets sessions by user's full name (usually 'first-name last-name').

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UserName**<String>

Get sessions by user name (in the form DOMAIN\user).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UserSID**<String>

Get sessions by user's Windows SID.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-UserUPN**<String>

Gets sessions by user's User Principal Name (in the form user@domain).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ApplicationUid**<Int32>

Get sessions running the application with the specified Uid.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SharedDesktopUid**<Int32>

Get sessions by SharedDesktop Uid.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
-----------	-------

Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

--	--



Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

None You cannot pipe input into this cmdlet.

### Return Values

Citrix.Broker.Admin.SDK.Session

Returns sessions matching the specified criteria.

### Examples

----- **EXAMPLE 1** -----

C:\PS> Get-BrokerSession  
Enumerate all sessions.

----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerSession -UserName MyDomain\MyAccount -SessionState Disconnected
```

Get all disconnected sessions for a specific user.

# Get-BrokerSessionLinger

May 10, 2016

Gets one or more session lingering settings.

## Syntax

```
Get-BrokerSessionLinger [-DesktopGroupUid] <Int32> [-Property <String[]>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Get-BrokerSessionLinger [[-DesktopGroupName] <String>] [-AssociatedUserFullName <String>] [-AssociatedUserName
<String>] [-AssociatedUserUPN <String>] [-Enabled <Boolean>] [-MaxAverageLoadThreshold <Int32>] [-
MaxLoadPerMachineThreshold <Int32>] [-MaxTimeBeforeDisconnect <TimeSpan>] [-MaxTimeBeforeTerminate
<TimeSpan>] [-UserFilterEnabled <Boolean>] [-UserSID <String>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-
Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Get-BrokerSessionLinger cmdlet is used to enumerate desktop group session linger settings that match all of the supplied criteria.

Without parameters, Get-BrokerSessionLinger gets all the session linger settings that have been created. You can also use the parameters of Get-BrokerSessionLinger to filter the results to just the desktop group you're interested in.

Note that each desktop group can only have a single session linger setting. Session lingering only applies to application sessions.

See about\_Broker\_Filtering for information about advanced filtering options.

----- BrokerSessionLinger Object

The session linger object returned represents a session linger setting in a desktop group.

-- AssociatedUserFullNames (System.String[])

List of associated users (full names). Associated users is the list of users who are given access using the pre-launch/user mapping filter.

-- AssociatedUserNames (System.String[])

List of associated users (SAM names). Associated users is the list of users who are given access using the pre-launch/user mapping filter.

-- AssociatedUserUPNs (System.String[])

List of associated users (user principle names). Associated users is the list of users who are given access using the pre-launch/user mapping filter.

-- DesktopGroupName (System.String)

Name of the associated desktop group.

-- DesktopGroupUid (System.Int32)

Uid of the associated desktop group.

-- Enabled (System.Boolean)

Specifies whether or not session lingering is enabled for the desktop group.

-- MaxAverageLoadThreshold (System.Int32)

Specifies the average load threshold across the desktop group. After this threshold is hit lingering sessions will be terminated to reduce average load across the group. Sessions that have been lingering the longest will be chosen first.

-- MaxLoadPerMachineThreshold (System.Int32)

Specifies the maximum load threshold per machine in the desktop group. After this threshold is hit lingering sessions on loaded machines will be terminated to reduce load. Sessions that have been lingering the longest will be chosen first.

-- MaxTimeBeforeDisconnect (System.TimeSpan)

Specifies the maximum time by when a lingering session will be disconnected. The disconnect timer cannot be greater than the terminate timer. When the disconnect timer is same as the terminate timer, the session will be directly be terminated. The default value is 15 minutes. A value of 0 disables the disconnect timer.

-- MaxTimeBeforeTerminate (System.TimeSpan)

Specifies the maximum time by when a lingering session will be terminated. When the disconnect timer is same as the terminate timer, the session will be directly be terminated. The default value is 8 hours. A value of 0 disables the terminate timer.

-- UserFilterEnabled (System.Boolean)

Indicates if linger-specific user filter is enabled.

## Related topics

[New-BrokerSessionLinger](#)

[Set-BrokerSessionLinger](#)

[Remove-BrokerSessionLinger](#)

## Parameters

**-DesktopGroupUid**<Int32>

Gets session linger setting that is associated with the specified desktop group Uid.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupName<String>**

Gets session linger setting that is associated with the specified desktop group name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserFullName<String>**

Gets session linger settings with an associated user identified by their full name (usually 'first-name last-name'). If the 'UserFilterEnabled' property is true then access to the session linger is restricted to those users only, otherwise access is unrestricted (but always subject to other policy rules).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserName<String>**

Gets session linger settings with an associated user identified by their user name (in the form 'domain\user'). If the 'UserFilterEnabled' property is true then access to the session linger is restricted to those users only, otherwise access is unrestricted (but always subject to other policy rules).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserUPN<String>**

Gets session linger settings with an associated user identified by their user principle name (in the form 'user@domain'). If the 'UserFilterEnabled' property is true then access to the session linger is restricted to those users only, otherwise access is unrestricted (but always subject to other policy rules).

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-Enabled**<Boolean>

Gets only the session linger settings that have the specified value for whether the setting is enabled.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxAverageLoadThreshold**<Int32>

Gets only the session linger settings that have the specified average load threshold.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxLoadPerMachineThreshold**<Int32>

Gets only the session linger settings that have the specified maximum load threshold per machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxTimeBeforeDisconnect**<TimeSpan>

Gets only the session linger settings that have the specified idle disconnect time.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-MaxTimeBeforeTerminate**<TimeSpan>

Gets only the session linger settings that have the specified idle terminate time.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UserFilterEnabled**<Boolean>

Gets only session linger settings whose user filter is in the specified state.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UserSID**<String>

Gets only session linger settings with their accessibility restricted to include the specified user.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False

Accept Pipeline Input?	false
------------------------	-------

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
-----------	-------



Default Value	
Accept Pipeline Input?	false

### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

None You cannot pipe input into this cmdlet.

### Return Values

Citrix.Broker.Admin.SDK.SessionLinger

Get-BrokerSessionLinger returns an object for each session linger setting it gets.

### Examples

#### ----- **EXAMPLE 1** -----

```
C:\PS> Get-BrokerSessionLinger -DesktopGroupName "test"
```

Returns the session linger settings associated with the desktop group named 'test'.

# Get-BrokerSessionPreLaunch

May 10, 2016

Gets one or more session pre-launch settings.

## Syntax

```
Get-BrokerSessionPreLaunch [-DesktopGroupUid] <Int32> [-Property <String[]>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Get-BrokerSessionPreLaunch [[-DesktopGroupName] <String>] [-AssociatedUserFullName <String>] [-AssociatedUserName
<String>] [-AssociatedUserUPN <String>] [-Enabled <Boolean>] [-MaxAverageLoadThreshold <Int32>] [-
MaxLoadPerMachineThreshold <Int32>] [-MaxTimeBeforeDisconnect <TimeSpan>] [-MaxTimeBeforeTerminate
<TimeSpan>] [-UserFilterEnabled <Boolean>] [-UserSID <String>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-
Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Get-BrokerSessionPreLaunch cmdlet is used to enumerate desktop group session pre-launch settings that match all of the supplied criteria.

Without parameters, Get-BrokerSessionPreLaunch gets all the session pre-launch settings that have been created. You can also use the parameters of Get-BrokerSessionPreLaunch to filter the results to just the desktop group you're interested in.

Note that each desktop group can only have a single session pre-launch setting. Session pre-launch only applies to application sessions.

See about\_Broker\_Filtering for information about advanced filtering options.

----- BrokerSessionPreLaunch Object

The session pre-launch object returned represents a session pre-launch setting in a desktop group.

-- AssociatedUserFullNames (System.String[])

List of associated users (full names). Associated users is the list of users who are given access using the pre-launch/user mapping filter.

-- AssociatedUserNames (System.String[])

List of associated users (SAM names). Associated users is the list of users who are given access using the pre-launch/user mapping filter.

-- AssociatedUserUPNs (System.String[])

List of associated users (user principle names). Associated users is the list of users who are given access using the pre-launch/user mapping filter.

-- DesktopGroupName (System.String)

Name of the associated desktop group.

-- DesktopGroupUid (System.Int32)

Uid of the associated desktop group.

-- Enabled (System.Boolean)

Specifies whether or not session pre-launch is enabled for the desktop group.

-- MaxAverageLoadThreshold (System.Int32)

Specifies the average load threshold across the desktop group. After this threshold is hit pre-launched sessions will be terminated to reduce average load across the group. Sessions that have been pre-launched the longest will be chosen first.

-- MaxLoadPerMachineThreshold (System.Int32)

Specifies the maximum load threshold per machine in the desktop group. After this threshold is hit pre-launched sessions on loaded machines will be terminated to reduce load. Sessions that have been pre-launched the longest will be chosen first.

-- MaxTimeBeforeDisconnect (System.TimeSpan)

Specifies the maximum time by when a pre-launched session will be disconnected. The disconnect timer cannot be greater than the terminate timer. When the disconnect timer is same as the terminate timer, the session will be directly be terminated. The default value is 15 minutes. A value of 0 disables the disconnect timer.

-- MaxTimeBeforeTerminate (System.TimeSpan)

Specifies the maximum time by when a pre-launched session will be terminated. When the disconnect timer is same as the terminate timer, the session will be directly be terminated. The default value is 8 hours. A value of 0 disables the terminate timer.

-- UserFilterEnabled (System.Boolean)

Indicates if pre-launch-specific user filter is enabled.

## Related topics

[New-BrokerSessionPreLaunch](#)

[Set-BrokerSessionPreLaunch](#)

[Remove-BrokerSessionPreLaunch](#)

## Parameters

**-DesktopGroupUid**<Int32>

Gets session pre-launch setting that is associated with the specified desktop group Uid.

Required?	true
Default Value	
Accept Pipeline Input?	false

--	--

#### **-DesktopGroupName<String>**

Gets session pre-launch setting that is associated with the specified desktop group name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserFullName<String>**

Gets session pre-launch settings with an associated user identified by their full name (usually 'first-name last-name'). If the 'UserFilterEnabled' property is true then access to the session pre-launch is restricted to those users only, otherwise access is unrestricted (but always subject to other policy rules).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserName<String>**

Gets session pre-launch settings with an associated user identified by their user name (in the form 'domain\user'). If the 'UserFilterEnabled' property is true then access to the session pre-launch is restricted to those users only, otherwise access is unrestricted (but always subject to other policy rules).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserUPN<String>**

Gets session pre-launch settings with an associated user identified by their user principle name (in the form 'user@domain'). If the 'UserFilterEnabled' property is true then access to the session pre-launch is restricted to those users only, otherwise access is unrestricted (but always subject to other policy rules).

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-Enabled**<Boolean>

Gets only the session pre-launch settings that have the specified value for whether the setting is enabled.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxAverageLoadThreshold**<Int32>

Gets only the session pre-launch settings that have the specified average load threshold.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxLoadPerMachineThreshold**<Int32>

Gets only the session pre-launch settings that have the specified maximum load threshold per machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxTimeBeforeDisconnect**<TimeSpan>

Gets only the session pre-launch settings that have the specified idle disconnect time.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-MaxTimeBeforeTerminate**<TimeSpan>

Gets only the session pre-launch settings that have the specified idle terminate time.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UserFilterEnabled**<Boolean>

Gets only session pre-launch settings whose user filter is in the specified state.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UserSID**<String>

Gets only session pre-launch settings with their accessibility restricted to include the specified user.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False

Accept Pipeline Input?	false
------------------------	-------

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

None You cannot pipe input into this cmdlet.

### Return Values

Citrix.Broker.Admin.SDK.SessionPreLaunch

Get-BrokerSessionPreLaunch returns an object for each session pre-launch setting it gets.

### Examples

#### ----- **EXAMPLE 1** -----

```
C:\PS> Get-BrokerSessionPreLaunch -DesktopGroupName "test"
```

Returns the session pre-launch settings associated with the destkop group named 'test'.



# Get-BrokerSharedDesktop

May 10, 2016

Get shared desktops configured for this site.

## Syntax

```
Get-BrokerSharedDesktop [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerSharedDesktop [-MachineName] <String> [-AgentVersion <String>] [-ControllerDNSName <String>] [-DesktopGroupUid <Int32>] [-DNSName <String>] [-HostedMachineId <String>] [-HostedMachineName <String>] [-HostingServerName <String>] [-HypervisorConnectionUid <Int32>] [-InMaintenanceMode <Boolean>] [-IPAddress <String>] [-LastDeregistrationReason <DeregistrationReason>] [-LastDeregistrationTime <DateTime>] [-LastHostingUpdateTime <DateTime>] [-OSType <String>] [-OSVersion <String>] [-PowerState <PowerState>] [-RegistrationState <RegistrationState>] [-SID <String>] [-Tag <String>] [-WillShutdownAfterUse <Boolean>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet is deprecated, please use the Get-BrokerMachine cmdlet instead.

Retrieve shared desktops matching the specified criteria. If no parameters are specified, all shared desktops are enumerated.

Get-BrokerSharedDesktop returns configuration information only for shared desktops (a DesktopKind of 'Shared').

For information about advanced filtering options, see [about\\_Broker\\_Filtering](#); for information about desktops, see [about\\_Broker\\_Desktops](#).

### ----- BrokerSharedDesktop Object

Shared desktops are desktops that are assigned randomly to users upon connection from a pool of available machines.

-- AgentVersion (System.String)

Version of the Citrix Virtual Delivery Agent (VDA) installed on the desktop.

-- ControllerDNSName (System.String)

The DNS host name of the controller that the desktop is registered to.

-- DesktopGroupUid (System.Int32)

Uid of the desktop group the desktop has been assigned to.

-- DNSName (System.String)

The DNS host name of the desktop.

-- HostedMachineId (System.String)

Unique ID within the hosting unit of the target managed desktop.

-- HostedMachineName (System.String)

The friendly name of a hosted desktop as used by its hypervisor. This is not necessarily the DNS name of the desktop.

-- HostingServerName (System.String)

DNS name of the hypervisor that is hosting the desktop if managed.

-- HypervisorConnectionUid (System.Int32?)

The UID of the hypervisor connection that the desktop has been assigned to, if managed.

-- InMaintenanceMode (System.Boolean)

Denotes whether the desktop is in maintenance mode.

-- IPAddress (System.String)

The IP address of the desktop.

-- LastDeregistrationReason (Citrix.Broker.Admin.SDK.DeregistrationReason?)

The reason for the last deregistration of the desktop with the broker. Possible values are:

AgentShutdown, AgentSuspended, AgentRequested, IncompatibleVersion, AgentAddressResolutionFailed, AgentNotContactable, AgentWrongActiveDirectoryOU, EmptyRegistrationRequest, MissingRegistrationCapabilities, MissingAgentVersion, InconsistentRegistrationCapabilities, NotLicensedForFeature, UnsupportedCredentialSecurityVersion, InvalidRegistrationRequest, SingleMultiSessionMismatch, FunctionalLevelTooLowForCatalog, FunctionalLevelTooLowForDesktopGroup, PowerOff, DesktopRestart, DesktopRemoved, AgentRejectedSettingsUpdate, SendSettingsFailure, SessionAuditFailure, SessionPrepareFailure, ContactLost, SettingsCreationFailure, UnknownError and BrokerRegistrationLimitReached.

-- LastDeregistrationTime (System.DateTime?)

Time of the last deregistration of the desktop from the controller.

-- LastHostingUpdateTime (System.DateTime?)

Time of last update to any hosting data for this desktop reported by the hypervisor connection.

-- MachineName (System.String)

DNS host name of the machine associated with the desktop.

-- OSType (System.String)

A string that can be used to identify the operating system that is running on the desktop.

-- OSVersion (System.String)

A string that can be used to identify the version of the operating system running on the desktop, if known.

-- PowerState (Citrix.Broker.Admin.SDK.PowerState)

The current power state of the desktop. Possible values are: Unmanaged, Unknown, Unavailable, Off, On, Suspended,

TurningOn, TurningOff, Suspending, resuming.

-- RegistrationState (Citrix.Broker.Admin.SDK.RegistrationState)

Indicates the registration state of the desktop. Possible values are: Unregistered, Initializing, Registered, AgentError.

-- SID (System.String)

The security identifier of the shared desktop.

-- Uid (System.Int32)

The uid of the shared desktop.

-- WillShutdownAfterUse (System.Boolean)

Flag indicating whether this desktop is tainted and will be shutdown after all sessions on the desktop have ended. This flag should only ever be true on power managed, single-session desktops.

Note: The desktop will not shut down if it is in maintenance mode, but will shut down after the desktop is taken out of maintenance mode.

## Related topics

[Set-BrokerSharedDesktop](#)

## Parameters

**-Uid**<Int32>

Gets desktops by Uid.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-MachineName**<String>

Gets desktops by machine name (in the form 'domain\machine').

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AgentVersion**<String>

Gets desktops with a specific Virtual Desktop Agent version.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ControllerDNSName<String>**

Gets desktops by the DNS name of the controller they are registered with.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopGroupUid<Int32>**

Gets desktops from a desktop group with a specific Uid.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DNSName<String>**

Gets desktops by DNS name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-HostedMachineId<String>**

Gets desktops by the machine id known to the hypervisor.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HostedMachineName**<String>

Gets desktops by the machine name known to the hypervisor.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HostingServerName**<String>

Gets desktops by the name of the hosting hypervisor server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionUid**<Int32>

Gets desktops by the uid of the hosting hypervisor connection.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-InMaintenanceMode**<Boolean>

Gets desktops by the InMaintenanceMode setting.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-IPAddress<String>**

Gets desktops by IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastDeregistrationReason<DeregistrationReason>**

Gets desktops whose broker last recorded a specific deregistration reason.

Valid values are \$null, AgentShutdown, AgentSuspended, AgentRequested, IncompatibleVersion, AgentAddressResolutionFailed, AgentNotContactable, AgentWrongActiveDirectoryOU, EmptyRegistrationRequest, MissingRegistrationCapabilities, MissingAgentVersion, InconsistentRegistrationCapabilities, NotLicensedForFeature, UnsupportedCredentialSecurityVersion, InvalidRegistrationRequest, SingleMultiSessionMismatch, FunctionalLevelTooLowForCatalog, FunctionalLevelTooLowForDesktopGroup, PowerOff, DesktopRestart, DesktopRemoved, AgentRejectedSettingsUpdate, SendSettingsFailure, SessionAuditFailure, SessionPrepareFailure, ContactLost, SettingsCreationFailure, UnknownError and BrokerRegistrationLimitReached.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastDeregistrationTime<DateTime>**

Gets desktops by the time that they were last deregistered.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-LastHostingUpdateTime**<DateTime>

Gets desktops by the time that the hosting information was last updated.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-OSType**<String>

Gets desktops by the type of operating system they are running.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-OSVersion**<String>

Gets desktops by the version of the operating system they are running.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-PowerState**<PowerState>

Gets desktops by power state.

Valid values are Unmanaged, Unknown, Unavailable, Off, On, Suspended, TurningOn, TurningOff, Suspending, and Resuming.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-RegistrationState**<RegistrationState>

Gets desktops by registration state.

Valid values are Registered, Unregistered, and AgentError.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-SID**<String>

Gets desktops by machine SID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Tag**<String>

Get desktops tagged with the given tag.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-WillShutdownAfterUse**<Boolean>

Gets desktops depending on whether they shutdown after use or not.

Required?	false
Default Value	
Accept Pipeline Input?	false



### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.

Accept Pipeline Input?	false
------------------------	-------

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through `Select-Object`, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.SharedDesktop

Get-BrokerSharedDesktop returns an object for each matching shared desktop.

## Notes

To compare dates/times, use -Filter and relative comparisons. For more information, see [about\\_Broker\\_Filtering](#).

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerSharedDesktop -HostingServerName BigServer12*
```

Get all shared desktops hosted by the hypervisor server BigServer12.

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerSharedDesktop -OSType 'Windows XP*' | ft -a MachineName,OSType,OSVersion
```

List all shared desktops running Windows XP, including the machine name and OS details.

# Get-BrokerSite

May 10, 2016

Gets the current XenDesktop broker site.

## Syntax

```
Get-BrokerSite [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Get-BrokerSite cmdlet gets the current broker site.

The broker site is a top-level, logical representation of the XenDesktop site, from the perspective of the brokering services running within the site. It defines various site-wide default attributes used by the brokering services.

A XenDesktop installation has only a single broker site instance.

### ----- BrokerSite Object

The BrokerSite object represents logical representation of the XenDesktop site. It contains the following properties:

-- BaseOU (System.Guid?)

The objectGUID property identifying the base OU in Active Directory used for desktop registrations.

-- BrokerServiceGroupUid (System.Guid)

The Uid for the Broker Service Group.

-- ColorDepth (Citrix.Broker.Admin.SDK.ColorDepth)

The default color depth for new desktop groups.

-- ConfigurationServiceGroupUid (System.Guid?)

The Uid for the Configuration Service Group.

-- ConnectionLeasingEnabled (System.Boolean?)

The indicator for connection leasing active.

-- DesktopGroupIconUid (System.Int32)

The default desktop icon used for new desktop groups.

-- DnsResolutionEnabled (System.Boolean)

The setting to configure whether numeric IP address or the DNS name to be present in the ICA file.

-- LicensedSessionsActive (System.Int32?)

The count of active licensed session.

-- LicenseEdition (System.String)

The license edition for session brokering.

-- LicenseGraceSessionsRemaining (System.Int32?)

The count of License Grace Session Remaining

-- LicenseModel (Citrix.Broker.Admin.SDK.LicenseModel?)

The licensing model in use. Values can be 'Concurrent' or 'UserDevice'

-- LicenseServerName (System.String)

The DNS for License Server Name

-- LicenseServerPort (System.Int32)

The port for the License Server

-- LicensingBurnIn (System.String)

The date for the license to end in yyyy.MMdd format

-- LicensingBurnInDate (System.DateTime?)

The date for the license to end

-- LicensingGraceHoursLeft (System.Int32?)

The number of grace hours left after license expiry

-- LicensingGracePeriodActive (System.Boolean?)

The indicator for licensing grace period active

-- LicensingOutOfBoxGracePeriodActive (System.Boolean?)

The indicator for licensing out of the box grace period active

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

The metadata for this command.

-- Name (System.String)

The name of the site

-- SecureIcaRequired (System.Boolean)

The default SecureICA usage requirements for new desktop groups.

-- TrustRequestsSentToTheXmlServicePort (System.Boolean)

The XML Service trust settings.

## Related topics

[Set-BrokerSite](#)

[Get-BrokerIcon](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

Citrix.Broker.Admin.SDK.Site

Get-BrokerSite returns the single broker site instance.

## Examples

----- **EXAMPLE 1** -----

```
C:\PS> Get-BrokerSite
```

Gets the current broker site.

# Get-BrokerTag

May 10, 2016

Gets one or more tags.

## Syntax

```
Get-BrokerTag [-Uid] <Int32> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerTag [[-Name] <String>] [-Metadata <String>] [-UUID <Guid>] [-DesktopUid <Int32>] [-DesktopGroupUid <Int32>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets tags that match all of the supplied criteria.

----- BrokerTag Object

The BrokerTag object represents a single instance of a Tag associated to other objects. It contains the following properties:

-- MetadataMap (System.Collections.Generic.Dictionary<string, string>)

The metadata for this command.

-- Name (System.String)

The name of the Tag

-- Uid (System.Int32)

The Uid of the Tag

-- UUID (System.Guid)

The UUID associated to the Tag

## Related topics

[Add-BrokerTag](#)

[New-BrokerTag](#)

[Remove-BrokerTag](#)

[Rename-BrokerTag](#)

## Parameters

**-Uid**<Int32>

Gets the tag identified by Uid

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Name**<String>

Gets tags that match the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UUID**<Guid>

Gets tags associated with a given UUID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopUid**<Int32>

Gets tags associated with a Desktop.



Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupUid<Int32>**

Gets tags associated with a desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount<SwitchParameter>**

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount<Int32>**

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip<Int32>**

Skips the specified number of records before returning results. Also reduces the count returned by -

ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

**-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

**-Filter**<String>

Gets records that match a PowerShell style filter expression. See about\_Broker\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

None Input cannot be piped to this cmdlet.

## Return Values

Citrix.Broker.Admin.SDK.Tag

Returns matching tags.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Get-BrokerTag
Enumerate all tags
```

### ----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerTag -Uid 5
Get a single specific tag with a Uid of 5.
```

### ----- **EXAMPLE 3** -----

```
C:\PS> Get-BrokerTag -DesktopUid 1
Get tags associated with Desktop 1.
```

# Get-BrokerUnconfiguredMachine

May 10, 2016

Gets machines that have registered but are not yet configured in this site.

## Syntax

```
Get-BrokerUnconfiguredMachine -SID <String> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerUnconfiguredMachine [[-MachineName] <String>] [-AgentVersion <String>] [-ControllerDNSName <String>] [-DNSName <String>] [-FunctionalLevel <FunctionalLevel>] [-LastDeregistrationTime <DateTime>] [-OSType <String>] [-OSVersion <String>] [-SessionSupport <SessionSupport>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieve machines matching the specified criteria, where the machine has registered with a controller in the site but the machine has not yet been configured to be part of the site. If no parameters are specified, this cmdlet enumerates all such machines.

See about\_Broker\_Filtering for information about advanced filtering options, and about\_Broker\_Machines for background information about machines.

----- BrokerUnconfiguredMachine Object

An unconfigured machine is a machine that has registered with the site but is not configured in either a desktop group or a catalog.

-- AgentVersion (System.String)

Version of the Citrix Virtual Delivery Agent (VDA) installed on the unconfigured machine.

-- ControllerDNSName (System.String)

The DNS name of the controller that the unconfigured machine is registered with.

-- DNSName (System.String)

The DNS name of the unconfigured machine.

-- FunctionalLevel (Citrix.Broker.Admin.SDK.FunctionalLevel?)

The functional level of the unconfigured machine. This is determined by the version of the Citrix VDA software installed on the machine.

-- LastDeregistrationTime (System.DateTime?)

The time when the unconfigured machine last deregistered with the Citrix Broker Service.

-- MachineName (System.String)

The machine name of the unconfigured machine in the form domain\machine.

-- OSType (System.String)

The type of operating system installed on the unconfigured machine.

-- OSVersion (System.String)

The version of the operating system installed on the unconfigured machine.

-- SessionSupport (Citrix.Broker.Admin.SDK.SessionSupport?)

The session support of the unconfigured machine. Valid values are:

SingleSession, MultiSession

-- SID (System.String)

Security identifier of the unconfigured machine.

Related topics

[Get-BrokerMachine](#)

Parameters

**-SID**<String>

Gets machines by their machine SID.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-MachineName**<String>

Gets machines by their machine name (in the form domain\machine).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AgentVersion**<String>

Gets machines with a specific Virtual Desktop Agent version.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ControllerDNSName**<String>

Gets machines by the DNS name of the controller they are registered with.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DNSName**<String>

Gets machines by their DNS name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-FunctionalLevel**<FunctionalLevel>

Gets machines with a specific FunctionalLevel.

Valid values are L5, L7, L7\_6

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LastDeregistrationTime**<DateTime>

Gets machines by the time that they were last deregistered.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-OSType**<String>

Gets machines by the type of operating system they are running.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-OSVersion**<String>

Gets machines by the version of the operating system they are running.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-SessionSupport**<SessionSupport>

Gets machines that have the specified session capability. Values can be:

- o SingleSession - Single-session only machine.
- o MultiSession - Multi-session capable machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through `Select-Object`, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.UnconfiguredMachine

Get-BrokerUnconfiguredMachine returns an object for each matching machine

#### Notes

It is generally better to compare dates and times using `-Filter` and relative comparisons. See [about\\_Broker\\_Filtering](#) and the examples in this topic for more information.



## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerUnconfiguredMachine -Filter { ControllerDNSName -eq 'Controller1.domain.com' -and OSType -eq 'Windows XP Service Pack 3' }
```

This command returns all unconfigured XP SP3 machines which are registered with the controller called 'Controller1.domain.com'.

# Get-BrokerUser

May 10, 2016

Gets broker users configured for this site.

## Syntax

```
Get-BrokerUser -SID <String> [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Get-BrokerUser [[-Name] <String>] [-FullName <String>] [-UPN <String>] [-ApplicationUid <Int32>] [-SessionLingerDesktopGroupUid <Int32>] [-SessionPreLaunchDesktopGroupUid <Int32>] [-MachineUid <Int32>] [-PrivateDesktopUid <Int32>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-Property <String[]>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieve broker users matching the specified criteria. If no parameters are specified this cmdlet enumerates all broker users.

For information about advanced filtering options, see [about\\_Broker\\_Filtering](#).

----- BrokerUser Object

The BrokerUser object represents a single instance of an user. It contains the following properties:

-- FullName (System.String)

The full name of an user

-- Name (System.String)

The name of an user

-- SID (System.String)

The SID of an user

-- UPN (System.String)

The UPN of an user

## Related topics

[Add-BrokerUser](#)

[Remove-BrokerUser](#)

## Parameters

**-SID**<String>

Gets the broker user with the specified SID property value.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	false

#### **-Name<String>**

Gets the broker user with the specified Name property.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-FullName<String>**

Gets the broker user with the specified FullName property.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UPN<String>**

Gets the broker user with the specified UPN property value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ApplicationUid<Int32>**

Gets broker users associated with the application with the specified Uid.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-SessionLingerDesktopGroupUid<Int32>**

Gets broker users associated with the desktop group session linger settings with the specified Uid.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionPreLaunchDesktopGroupUid<Int32>**

Gets broker users associated with the desktop group session pre-launch settings with the specified Uid.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineUid<Int32>**

Gets broker users associated with the broker machine with the specified Uid.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PrivateDesktopUid<Int32>**

Gets broker users associated with the private desktop with the specified Uid.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
-----------	-------

Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through `Select-Object`, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.User

Get-BrokerUser returns an object for each matching broker user.

## Examples

### ----- **EXAMPLE 1** -----

```
Get-BrokerUser DOMAIN7*
```

Get all broker user objects with names matching the supplied pattern

### ----- **EXAMPLE 2** -----

```
$pdt = Get-BrokerPrivateDesktop DOMAIN\MACHINENAME*
```

```
Get-BrokerUser -PrivateDesktopUid $pdt.Uid
```

Get all broker user objects added to the specified private desktop

# Group-BrokerDesktop

May 10, 2016

Groups and counts desktops with the same value for a specified property.

## Syntax

```
Group-BrokerDesktop [-Uid <Int32>] -Property <String> [-AdminAddress <String>] [<CommonParameters>]
```

```
Group-BrokerDesktop -Property <String> [[-MachineName <String>] [-AgentVersion <String>] [-ApplicationInUse <String>] [-AssignedClientName <String>] [-AssignedIPAddress <String>] [-AssociatedUserFullName <String>] [-AssociatedUserName <String>] [-AssociatedUserUPN <String>] [-AutonomouslyBrokered <Boolean>] [-CatalogName <String>] [-CatalogUid <Int32>] [-ClientAddress <String>] [-ClientName <String>] [-ClientVersion <String>] [-ColorDepth <ColorDepth>] [-ConnectedViaHostName <String>] [-ConnectedViaIP <String>] [-ControllerDNSName <String>] [-DeliveryType <DeliveryType>] [-Description <String>] [-DesktopCondition <String>] [-DesktopGroupName <String>] [-DesktopGroupUid <Int32>] [-DesktopKind <DesktopKind>] [-DeviceId <String>] [-DNSName <String>] [-FunctionalLevel <FunctionalLevel>] [-HardwareId <String>] [-HostedMachineId <String>] [-HostedMachineName <String>] [-HostingServerName <String>] [-HypervisorConnectionName <String>] [-HypervisorConnectionUid <Int32>] [-IconUid <Int32>] [-ImageOutOfDate <Boolean>] [-InMaintenanceMode <Boolean>] [-IPAddress <String>] [-IsAssigned <Boolean>] [-IsPhysical <Boolean>] [-LastConnectionFailure <ConnectionFailureReason>] [-LastConnectionTime <DateTime>] [-LastConnectionUser <String>] [-LastDeregistrationReason <DeregistrationReason>] [-LastDeregistrationTime <DateTime>] [-LastErrorReason <String>] [-LastErrorTime <DateTime>] [-LastHostingUpdateTime <DateTime>] [-LaunchedViaHostName <String>] [-LaunchedViaIP <String>] [-MachineInternalState <MachineInternalState>] [-MachineUid <Int32>] [-OSType <String>] [-OSVersion <String>] [-PersistUserChanges <PersistUserChanges>] [-PowerActionPending <Boolean>] [-PowerState <PowerState>] [-Protocol <String>] [-ProvisioningType <ProvisioningType>] [-PublishedApplication <String>] [-PublishedName <String>] [-PvdStage <PvdStage>] [-RegistrationState <RegistrationState>] [-SecureIcaActive <Boolean>] [-SecureIcaRequired <Boolean>] [-SessionHidden <Boolean>] [-SessionId <Int32>] [-SessionState <SessionState>] [-SessionStateChangeTime <DateTime>] [-SessionUid <Int64>] [-SessionUserName <String>] [-SessionUserSID <String>] [-SID <String>] [-SmartAccessTag <String>] [-StartTime <DateTime>] [-SummaryState <DesktopSummaryState>] [-Tag <String>] [-WillShutdownAfterUse <Boolean>] [-ApplicationUid <Int32>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet is now deprecated, please use Group-BrokerMachine.

Filters desktops using the specified criteria, then groups and counts matching desktops with the same value for a particular property. The number of desktops in the group, and the property value for the group, is output. For example:

```
C:\PS> Group-BrokerDesktop -Property SummaryState
```

```
Count Name
```

```

```

```
43 Available
```

```
17 InUse
```

```
3 Disconnected
```

Filtering supports the same options as the Get-BrokerDesktop cmdlet, and allows filtering on both desktop and session properties.

Group-BrokerDesktop is similar to the standard PowerShell Group-Object, but is faster than piping the output of Get-BrokerDesktop into Group-Object when working with many desktops.

Note that all session information properties for multi-session desktops is always \$null, this means that it is not possible to group these desktops by session information using this command. Use Get-BrokerSession to get information on all current sessions.

Also note that the MaxRecordCount, ReturnTotalRecordCount, Skip, and SortBy parameters apply to GroupInfo records output rather than the filtered desktops.

## Related topics

[Get-BrokerDesktop](#)

Group-Object

Parameters

**-Uid**<Int32>



Gets desktops with a specific UID.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Property<String>**

Selects the property by which matching desktops are grouped.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-MachineName<String>**

Gets desktops with a specific machine name (in the form 'domain\machine').

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AgentVersion<String>**

Gets desktops with a specific Citrix Virtual Delivery Agent version.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ApplicationInUse<String>**

Gets desktops running a specified published application (identified by browser name).

String comparisons are case-insensitive.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AssignedClientName<String>**

Gets desktops assigned to a specific client name.

--	--

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssignedIPAddress<String>**

Gets desktops assigned to a specific IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserFullName<String>**

Gets desktops with an associated user identified by their full name (usually in the form 'first-name last-name').

Associated users are the current user for shared desktops, and the assigned users for private desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserName<String>**

Gets desktops with an associated user identified by their user name (in the form 'domain\user').

Associated users are the current user for shared desktops, and the assigned users for private desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserUPN<String>**

Gets desktops with an associated user identified by their User Principle Name (in the form 'user@domain').

Associated users are the current user for shared desktops, and the assigned users for private desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AutonomouslyBrokered<Boolean>**

Gets desktops according to whether their current session is autonomously brokered or not. Autonomously brokered sessions are HDX sessions established by

direct connection without being brokered.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-CatalogName<String>**

Gets desktops from the catalog with the specific name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-CatalogUid<Int32>**

Gets desktops from a catalog with a specific UID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ClientAddress<String>**

Gets desktops with a specific client IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ClientName<String>**

Gets desktops with a specific client name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ClientVersion<String>**

Gets desktops with a specific client version.

--	--

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ColorDepth**<ColorDepth>

Gets desktops configured with a specific color depth.

Valid values are FourBit, EightBit, SixteenBit, and TwentyFourBit.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ConnectedViaHostName**<String>

Gets desktops with a specific host name of the incoming connection. This is usually a proxy or Citrix Access Gateway server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ConnectedViaIP**<String>

Gets desktops with a specific IP address of the incoming connection.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ControllerDNSName**<String>

Gets desktops with a specific DNS name of the controller they are registered with.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DeliveryType**<DeliveryType>

Gets desktops of a particular delivery type.

Valid values are AppsOnly, DesktopsOnly, DesktopsAndApps

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-Description<String>**

Get desktops with a specific description.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopCondition<String>**

Gets desktop with an outstanding desktop condition condition.

Valid values are:

- o CPU: Indicates the machine has high CPU usage
- o ICALatency: Indicates the network latency is high
- o UPMLogonTime: Indicates that the profile load time was high

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupName<String>**

Gets desktops from a desktop group with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupUid<Int32>**

Gets desktops from a desktop group with the specified UID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopKind<DesktopKind>**

Deprecated: Use AllocationType parameter.

Gets desktops of a particular kind.

Valid values are Private, Shared.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DeviceId**<String>

Gets desktops with a specific client device ID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DNSName**<String>

Get desktops with a specific DNS name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-FunctionalLevel**<FunctionalLevel>

Gets desktops with a specific FunctionalLevel.

Valid values are L5, L7, L7\_6

Required?	false
Default Value	
Accept Pipeline Input?	false

**-HardwareId**<String>

Gets desktops with a specific client hardware ID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-HostedMachineId<String>**

Gets desktops with a specific machine ID known to the hypervisor.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-HostedMachineName<String>**

Gets desktops with a specific machine name known to the hypervisor.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-HostingServerName<String>**

Gets desktops with a specific name of the hosting hypervisor server.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-HypervisorConnectionName<String>**

Gets desktops with a specific name of the hosting hypervisor connection.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-HypervisorConnectionUid<Int32>**

Gets desktops with a specific UID of the hosting hypervisor connection.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IconUid<Int32>**

Gets desktops with a specific configured icon. Note that desktops with a null IconUid use the icon of the desktop group.

--	--

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ImageOutOfDate**<Boolean>

Gets desktops if they have an ImageOutOfDate flag.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-InMaintenanceMode**<Boolean>

Gets desktops with a specific InMaintenanceMode setting.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IPAddress**<String>

Gets desktops with a specific IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IsAssigned**<Boolean>

Gets desktops according to whether they are assigned or not. Desktops may be assigned to one or more users or groups, a client IP address or a client endpoint name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IsPhysical**<Boolean>

Specifies if machines in the catalog can be power managed by the Citrix Broker Service. Where the power state of the machine cannot be controlled, specify \$true, otherwise \$false. Can only be specified together with a provisioning type of Pvs or Manual, or if used with the deprecated CatalogKind parameter only with Pvs or PvsPvd catalog kinds.

Required?	false
-----------	-------



Default Value	
Accept Pipeline Input?	false

#### **-LastConnectionFailure**<ConnectionFailureReason>

Gets desktops with a specific reason for the last recorded connection failure. This value is None if the last connection was successful or if there has been no attempt to connect to the desktop yet.

Valid values are None, SessionPreparation, RegistrationTimeout, ConnectionTimeout, Licensing, Ticketing, and Other.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastConnectionTime**<DateTime>

Gets desktops that last connected at a specific time. This is the time that the broker detected that the connection attempt either succeeded or failed.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastConnectionUser**<String>

Gets desktops where a specific user name last attempted a connection (in the form 'domain\user').

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastDeregistrationReason**<DeregistrationReason>

Gets desktops whose broker last recorded a specific deregistration reason.

Valid values are \$null, AgentShutdown, AgentSuspended, AgentRequested, IncompatibleVersion, AgentAddressResolutionFailed, AgentNotContactable, AgentWrongActiveDirectoryOU, EmptyRegistrationRequest, MissingRegistrationCapabilities, MissingAgentVersion, InconsistentRegistrationCapabilities, NotLicensedForFeature, UnsupportedCredentialSecurityVersion, InvalidRegistrationRequest, SingleMultiSessionMismatch, FunctionalLevelTooLowForCatalog, FunctionalLevelTooLowForDesktopGroup, PowerOff, DesktopRestart, DesktopRemoved, AgentRejectedSettingsUpdate, SendSettingsFailure, SessionAuditFailure, SessionPrepareFailure, ContactLost, SettingsCreationFailure, UnknownError and BrokerRegistrationLimitReached.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LastDeregistrationTime**<DateTime>

Gets desktops that were last deregistered by a specific time.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LastErrorReason**<String>

Gets desktops with the specified last error reason.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LastErrorTime**<DateTime>

Gets desktops with the specified last error time.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LastHostingUpdateTime**<DateTime>

Gets desktops with a specific time that the hosting information was last updated.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LaunchedViaHostName**<String>

Gets desktops with a specific host name of the StoreFront server from which the user launched the session.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LaunchedViaIP**<String>

Gets desktops with a specific IP address of the StoreFront server from which the user launched the session.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineInternalState**<MachineInternalState>

Gets desktops with the specified internal machine state.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineUid**<Int32>

Gets desktops with a specific machine UID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OSType**<String>

Gets desktops by the type of operating system they are running.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OSVersion**<String>

Gets desktops by the version of the operating system they are running.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PersistUserChanges**<PersistUserChanges>

Gets desktops by the location where the user changes are persisted.

- o OnLocal - User changes are persisted locally.
- o Discard - User changes are discarded.
- o OnPvd - User changes are persisted on the Pvd.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PowerActionPending**<Boolean>

Gets desktops with a specific power action pending state.

Valid values are \$true or \$false.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PowerState**<PowerState>

Gets desktops with a specific power state.

Valid values are Unmanaged, Unknown, Unavailable, Off, On, Suspended, TurningOn, TurningOff, Suspending, and Resuming.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Protocol**<String>

Gets desktops with connections using a specific protocol, for example HDX, RDP, or Console.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ProvisioningType**<ProvisioningType>

Specifies the provisioning type for the catalog. Values can be:

- o Manual - No provisioning.
- o PVS - Machine provisioned by PVS (machine may be physical, blade, VM,...).
- o MCS - Machine provisioned by MCS (machine must be VM).

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-PublishedApplication**<String>

Gets desktops with a specific application published on them (identified by its browser name).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PublishedName**<String>

Gets desktops with a specific published name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PvdStage**<PvdStage>

Gets machines with a specific personal vDisk stage.

Valid values are None, Requested, Starting, Working and Failed.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RegistrationState**<RegistrationState>

Gets desktops with a specific registration state.

Valid values are Unregistered, Initializing, Registered and AgentError.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecureIcaActive**<Boolean>

Gets desktops depending on whether the current session uses SecureICA or not.

Session properties are always null for multi-session desktops.

--	--

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecurelcaRequired**<Boolean>

Gets desktops configured with a particular SecurelcaRequired setting. Note that the desktop setting of \$null indicates that the desktop group value is used.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionHidden**<Boolean>

Gets desktops by whether their sessions are hidden or not. Hidden sessions are treated as though they do not exist when launching sessions; a hidden session cannot be reconnected to, but a new session may be launched using the same entitlement.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionId**<Int32>

Deprecated.

Gets desktops by session ID, a unique identifier that Remote Desktop Services uses to track the session but it is only unique on that machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionState**<SessionState>

Gets desktops with a specific session state.

Valid values are \$null, Other, PreparingSession, Connected, Active, Disconnected, Reconnecting, NonBrokeredSession, and Unknown.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionStateChangeTime<DateTime>**

Gets desktops whose sessions last changed state at a specific time.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionUid<Int64>**

Gets desktops with a specific session UID (\$null for no session).

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionUserName<String>**

Gets desktops with a specific user name for the current session (in the form 'domain\user').

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionUserSID<String>**

Gets desktops with a specific SID of the current session user.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SID<String>**

Gets desktops with a specific machine SID.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-SmartAccessTag**<String>

Gets desktops where the session has the specific SmartAccess tag.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-StartTime**<DateTime>

Gets desktops with a specific session start time.

Session properties are always null for multi-session desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SummaryState**<DesktopSummaryState>

Gets desktops with a specific summary state.

Valid values are Off, Unregistered, Available, Disconnected, and InUse.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Tag**<String>

Gets desktops with a specific tag.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-WillShutdownAfterUse**<Boolean>

Gets desktops depending on whether they shut down after use or not.

Required?	false
-----------	-------



Default Value	
Accept Pipeline Input?	false

#### **-ApplicationUid**<Int32>

Gets desktops with a specific published application (identified by its UID).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.

Accept Pipeline Input?	false
------------------------	-------

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.GroupInfo

Each GroupInfo object represents one group, and contains the following properties:

-- Count: The count of desktops in this group.

-- Name: The value of the property the desktops were grouped by (as a string).

If you do not specify -SortBy, groups are sorted with the largest count first.

#### Notes

To compare dates or times, use -Filter and relative comparisons. For more information, see [about\\_Broker\\_Filtering](#) and the examples.

#### Examples

----- **EXAMPLE 1** -----

```
C:\PS> Group-BrokerDesktop -Property SummaryState -DesktopGroupName dg1
Group desktops from the dg1 group by summary state.
```

----- **EXAMPLE 2** -----

```
C:\PS> Group-BrokerDesktop -Property LastConnectionFailure -Filter { LastConnectionFailure -ne "None" -and LastConnectionTime -ge '-7' } -MaxRecordCount 1
For desktops where the last connection attempt failed, list the most common reason for failure, ignoring connections that failed over a week ago.
```

----- **EXAMPLE 3** -----

```
C:\PS> Group-BrokerDesktop -Property HostingServerName -DesktopCondition ICALatency -SortBy Name
List alphabetically the hypervisor servers hosting desktops that are currently experiencing high network latency.
```

# Group-BrokerMachine

May 10, 2016

Groups and counts machines with the same value for a specified property.

## Syntax

```
Group-BrokerMachine [-UId <Int32> -Property <String> [-AdminAddress <String>] [<CommonParameters>]
```

```
Group-BrokerMachine -Property <String> [[-MachineName <String>] [-AgentVersion <String>] [-AllocationType <AllocationType>] [-ApplicationInUse <String>] [-AssignedClientName <String>] [-AssignedIPAddress <String>] [-AssociatedUserFullName <String>] [-AssociatedUserName <String>] [-AssociatedUserSID <String>] [-AssociatedUserUPN <String>] [-BrowserName <String>] [-CatalogName <String>] [-CatalogUId <Int32>] [-CatalogUUID <Guid>] [-ColorDepth <ColorDepth>] [-ControllerDNSName <String>] [-DeliveryType <DeliveryType>] [-Description <String>] [-DesktopCondition <String>] [-DesktopGroupName <String>] [-DesktopGroupUId <Int32>] [-DesktopGroupUUID <Guid>] [-DesktopKind <DesktopKind>] [-DesktopUId <Int32>] [-DNSName <String>] [-FaultState <MachineFaultState>] [-FunctionalLevel <FunctionalLevel>] [-HostedMachineId <String>] [-HostedMachineName <String>] [-HostingServerName <String>] [-HypervisorConnectionName <String>] [-HypervisorConnectionUId <Int32>] [-HypHypervisorConnectionUId <Guid>] [-IconUId <Int32>] [-ImageOutOfDate <Boolean>] [-InMaintenanceMode <Boolean>] [-IPAddress <String>] [-IsAssigned <Boolean>] [-IsPhysical <Boolean>] [-LastConnectionFailure <ConnectionFailureReason>] [-LastConnectionTime <DateTime>] [-LastConnectionUser <String>] [-LastDeregistrationReason <DeregistrationReason>] [-LastDeregistrationTime <DateTime>] [-LastErrorReason <String>] [-LastErrorTime <DateTime>] [-LastHostingUpdateTime <DateTime>] [-LastPvdErrorReason <String>] [-LastPvdErrorTime <DateTime>] [-LoadIndex <Int32>] [-MachineInternalState <MachineInternalState>] [-Metadata <String>] [-OSType <String>] [-OSVersion <String>] [-PersistUserChanges <PersistUserChanges>] [-PowerActionPending <Boolean>] [-PowerState <PowerState>] [-ProvisioningType <ProvisioningType>] [-PublishedApplication <String>] [-PublishedName <String>] [-PvdEstimatedCompletionTime <DateTime>] [-PvdPercentDone <Int32>] [-PvdStage <PvdStage>] [-PvdUpdateStartTime <DateTime>] [-RegistrationState <RegistrationState>] [-ScheduledReboot <ScheduledReboot>] [-SecureIcaRequired <Boolean>] [-SessionAutonomouslyBrokered <Boolean>] [-SessionClientAddress <String>] [-SessionClientName <String>] [-SessionClientVersion <String>] [-SessionConnectedViaHostName <String>] [-SessionConnectedViaIP <String>] [-SessionCount <Int32>] [-SessionDeviceId <String>] [-SessionHardwareId <String>] [-SessionHidden <Boolean>] [-SessionKey <Guid>] [-SessionLaunchedViaHostName <String>] [-SessionLaunchedViaIP <String>] [-SessionProtocol <String>] [-SessionSecureIcaActive <Boolean>] [-SessionsEstablished <Int32>] [-SessionSmartAccessTag <String>] [-SessionsPending <Int32>] [-SessionStartTime <DateTime>] [-SessionState <SessionState>] [-SessionStateChangeTime <DateTime>] [-SessionSupport <SessionSupport>] [-SessionType <SessionType>] [-SessionUId <Int64>] [-SessionUserName <String>] [-SessionUserSID <String>] [-SID <String>] [-SummaryState <DesktopSummaryState>] [-SupportedPowerActions <String[]>] [-Tag <String>] [-UUID <Guid>] [-VMToolsState <VMToolsState>] [-WillShutdownAfterUse <Boolean>] [-WindowsConnectionSetting <WindowsConnectionSetting>] [-AssignedUserSID <String>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Filters machines using the specified criteria, then groups and counts matching machines with the same value for a particular property. The number of machines in the group, and the property value for the group, is output. For example:

```
C:\PS> Group-BrokerMachine -Property SummaryState
```

```
Count Name
```

```

```

```
43 Available
```

```
17 InUse
```

```
3 Disconnected
```

Filtering supports the same options as the Get-BrokerMachine cmdlet, and allows filtering on both machine and session properties.

Group-BrokerMachine is similar to the standard PowerShell Group-Object, but is faster than piping the output of Get-BrokerMachine into Group-Object when working with many machines.

Note that the MaxRecordCount, ReturnTotalRecordCount, Skip, and SortBy parameters apply to GroupInfo records output rather than the filtered machines.

## Related topics

[Get-BrokerMachine](#)

[Group-Object](#)

[Parameters](#)

**-UId**<Int32>

Gets a machine with a specific UID.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Property<String>**

Selects the property by which matching machines are grouped.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-MachineName<String>**

Gets machines with a specific machine name (in the form domain\machine).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AgentVersion<String>**

Gets machines with a specific Virtual Delivery Agent version.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AllocationType<AllocationType>**

Gets machines from catalogs with the specified allocation type.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ApplicationInUse<String>**

Gets machines running a specified published application. String comparisons are case-insensitive.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AssignedClientName<String>**

Gets machines that have been assigned to the specific client name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssignedIPAddress<String>**

Gets machines that have been assigned to the specific client IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserFullName<String>**

Gets machines with an associated user identified by their full name (usually 'first-name last-name').

Associated users are all current users of a desktop, plus the assigned users for private desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserName<String>**

Gets machines with an associated user identified by their user name (in the form 'domain\user').

Associated users are all current users of a desktop, plus the assigned users for private desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserSID<String>**

Gets machines with an associated user identified by their Windows SID.

Associated users are all current users of a desktop, plus the assigned users for private desktops.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AssociatedUserUPN<String>**

Gets machines with an associated user identified by their User Principle Name (in the form 'user@domain').

Associated users are all current users of a desktop, plus the assigned users for private desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-BrowserName<String>**

Gets assigned machines backing desktop resources that have browser names matching the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-CatalogName<String>**

Gets machines from the catalog with the specific name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-CatalogUid<Int32>**

Gets machines from the catalog with the specific UID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-CatalogUUID<Guid>**

Gets machines from the catalog with the specific UUID.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ColorDepth**<ColorDepth>

Gets machines configured with a specific color depth.

Valid values are FourBit, EightBit, SixteenBit, and TwentyFourBit.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ControllerDNSName**<String>

Gets machines by the DNS name of the controller they are registered with.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DeliveryType**<DeliveryType>

Gets machines of a particular delivery type.

Valid values are AppsOnly, DesktopsOnly, DesktopsAndApps

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Description**<String>

Get machines by description.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopCondition**<String>

Gets machines with an outstanding desktop condition.

Valid values are:

- o CPU: Indicates the machine has high CPU usage
- o ICALatency: Indicates the network latency is high

o UPMLogonTime: Indicates that the profile load time was high

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopGroupName<String>**

Gets machines from a desktop group with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopGroupUid<Int32>**

Gets machines from a desktop group with a specific UID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopGroupUUID<Guid>**

Gets machines from a desktop group with a specific UUID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopKind<DesktopKind>**

Deprecated: Use AllocationType parameter.

Gets machines of a particular kind.

Valid values are Private, Shared.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DesktopUid<Int32>**



Gets the machine that corresponds to the desktop with the specific UID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DNSName**<String>

Gets machines with the specific DNS name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-FaultState**<MachineFaultState>

Gets machines currently in the specified fault state.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-FunctionalLevel**<FunctionalLevel>

Gets machines with a specific FunctionalLevel.

Valid values are L5, L7, L7\_6

Required?	false
Default Value	
Accept Pipeline Input?	false

**-HostedMachineId**<String>

Gets machines with the specific machine ID known to the hypervisor.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-HostedMachineName**<String>

Gets machines with the specific machine name known to the hypervisor.

--	--

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HostingServerName<String>**

Gets machines by the name of the hosting hypervisor server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionName<String>**

Gets machines with the specific name of the hypervisor connection hosting them.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionUid<Int32>**

Gets machines with the specific UID of the hypervisor connection hosting them.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypHypervisorConnectionUid<Guid>**

Gets machines with the specific UUID of the hypervisor connection hosting them.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IconUid<Int32>**

Gets machines by configured icon. Note that machines with a null IconUid use the icon of the desktop group.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ImageOutOfDate<Boolean>**

Gets machines depending on whether their disk image is out of date or not (for machines provisioned using MCS only).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-InMaintenanceMode<Boolean>**

Gets machines by whether they are in maintenance mode or not.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IPAddress<String>**

Gets machines with a specific IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IsAssigned<Boolean>**

Gets machines according to whether they are assigned or not. Machines may be assigned to one or more users or groups, a client IP address or a client endpoint name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IsPhysical<Boolean>**

Gets machines according to whether they can be power managed by XenDesktop or not.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -LastConnectionFailure<ConnectionFailureReason>

Gets machines with a specific reason for the last recorded connection failure. This value is None if the last connection was successful or if there has been no attempt to connect to the machine yet.

Valid values are None, SessionPreparation, RegistrationTimeout, ConnectionTimeout, Licensing, Ticketing, and Other.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -LastConnectionTime<DateTime>

Gets machines to which a user session connection occurred at a specific time. This is the time that the broker detected that the connection attempt either succeeded or failed.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -LastConnectionUser<String>

Gets machines where a specific user name last attempted a connection (in the form 'domain\user').

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -LastDeregistrationReason<DeregistrationReason>

Gets machines whose broker last recorded a specific deregistration reason.

Valid values are Null, AgentShutdown, AgentSuspended, AgentRequested, IncompatibleVersion, AgentAddressResolutionFailed, AgentNotContactable, AgentWrongActiveDirectoryOU, EmptyRegistrationRequest, MissingRegistrationCapabilities, MissingAgentVersion, InconsistentRegistrationCapabilities, NotLicensedForFeature, UnsupportedCredentialSecurityVersion, InvalidRegistrationRequest, SingleMultiSessionMismatch, FunctionalLevelTooLowForCatalog, FunctionalLevelTooLowForDesktopGroup, PowerOff, DesktopRestart, DesktopRemoved, AgentRejectedSettingsUpdate, SendSettingsFailure, SessionAuditFailure, SessionPrepareFailure, ContactLost, SettingsCreationFailure, UnknownError and BrokerRegistrationLimitReached.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -LastDeregistrationTime<DateTime>

Gets machines by the time that they were last deregistered.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-LastErrorReason<String>**

Gets machines with the specified last error reason.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastErrorTime<DateTime>**

Gets machines with the specified last error time.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastHostingUpdateTime<DateTime>**

Gets machines with a specific time that the hosting information was last updated.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastPvdErrorReason<String>**

Gets machines with the specified last Personal vDisk preparation error reason.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LastPvdErrorTime<DateTime>**

Gets machines with the specified last Personal vDisk preparation error time.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoadIndex<Int32>**

Gets machines by their current load index.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineInternalState<MachineInternalState>**

Gets machines with the specified internal state.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Metadata<String>**

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OSType<String>**

Gets machines by the type of operating system they are running.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OSVersion<String>**

Gets machines by the version of the operating system they are running.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PersistUserChanges**<PersistUserChanges>

Gets machines according to the location where user changes are persisted. Values can be:

- o OnLocal - User changes are persisted locally.
- o Discard - User changes are discarded.
- o OnPvd - User changes are persisted on the Pvd.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PowerActionPending**<Boolean>

Gets machines depending on whether a power action is pending or not.

Valid values are \$true or \$false.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PowerState**<PowerState>

Gets machines with a specific power state.

Valid values are Unmanaged, Unknown, Unavailable, Off, On, Suspended, TurningOn, TurningOff, Suspending, and Resuming.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ProvisioningType**<ProvisioningType>

Specifies the provisioning type for the catalog. Values can be:

- o Manual - No provisioning.
- o PVS - Machine provisioned by PVS (machine may be physical, blade, VM,...).
- o MCS - Machine provisioned by MCS (machine must be VM).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PublishedApplication**<String>

Gets machines with a specific application published to them (identified by its browser name).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PublishedName<String>**

Gets desktops with a specific published name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PvdEstimatedCompletionTime<DateTime>**

If preparation of the Personal vDisk is currently in progress for this machine, this reports an estimation of the time at which the process will be complete.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PvdPercentDone<Int32>**

Gets machines a specific percentage through the Personal vDisk preparation process.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PvdStage<PvdStage>**

Gets machines at a specific personal vDisk stage.

Valid values are None, Requested, Starting, Working and Failed.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PvdUpdateStartTime<DateTime>**

If preparation of the Personal vDisk is currently in progress for this machine, this reports when the update process began.

--	--



Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RegistrationState**<RegistrationState>

Gets machines in a specific registration state.

Valid values are Unregistered, Initializing, Registered, and AgentError.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ScheduledReboot**<ScheduledReboot>

Gets machines according to their current status with respect to any scheduled reboots (for either scheduled desktop group reboots or image rollout purposes). Valid values are:

- o None - No reboot currently scheduled.
- o Pending - Reboot scheduled but machine still available for use.
- o Draining - Reboot scheduled. New logons are disabled, but reconnections to existing sessions are allowed.
- o InProgress - Machine is actively being rebooted.
- o Natural - Natural reboot in progress. Machine is awaiting a restart.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecureIcaRequired**<Boolean>

Gets machines configured with a particular SecureIcaRequired setting. Note that the machine setting of \$null indicates that the desktop group value is used.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionAutonomouslyBrokered**<Boolean>

Gets machines according to whether their current session is autonomously brokered or not. Autonomously brokered sessions are HDX sessions established by direct connection without being brokered.

Session properties are always null for multi-session machines.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-SessionClientAddress<String>**

Gets machines with a specific client IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionClientName<String>**

Gets machines with a specific client name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionClientVersion<String>**

Gets machines with a specific client version.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionConnectedViaHostName<String>**

Gets machines with a specific host name of the incoming connection. This is usually a proxy or Citrix Access Gateway server.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionConnectedViaIP<String>**

Gets machines with a specific IP address of the incoming connection.

Session properties are always null for multi-session machines.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-SessionCount**<Int32>

Gets machines according to the total number of both pending and established user sessions on the machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionDeviceId**<String>

Gets machines with a specific client device ID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionHardwareId**<String>

Gets machines with a specific client hardware ID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionHidden**<Boolean>

Gets machines by whether their sessions are hidden or not. Hidden sessions are treated as though they do not exist when launching sessions using XenDesktop; a hidden session cannot be reconnected to, but a new session may be launched using the same entitlement.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionKey**<Guid>

Gets machine running the session with the specified unique key.

Session properties are always null for multi-session machines.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-SessionLaunchedViaHostName<String>**

Gets machines with a specific host name of the Web Interface server from which the user launched the session.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionLaunchedViaIP<String>**

Gets machines with a specific IP address of the Web Interface server from which the user launched the session.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionProtocol<String>**

Gets machines with connections using a specific protocol, for example HDX, RDP, or Console.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionSecureIcaActive<Boolean>**

Gets machines depending on whether the current session uses SecureICA or not.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionsEstablished<Int32>**

Gets machines according to the number of established user sessions present on the machine.

Required?	false

Default Value	
Accept Pipeline Input?	false

#### **-SessionSmartAccessTag<String>**

Gets session machines where the session has the specific SmartAccess tag.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionsPending<Int32>**

Get machines according to the number of pending user sessions for the machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionStartTime<DateTime>**

Gets machines with a specific session start time.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionState<SessionState>**

Gets machines with a specific session state.

Valid values are \$null, Other, PreparingSession, Connected, Active, Disconnected, Reconnecting, NonBrokeredSession, and Unknown.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionStateChangeTime<DateTime>**

Gets machines whose sessions last changed state at a specific time.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionSupport**<SessionSupport>

Gets machines that have the specified session capability. Values can be:

- o SingleSession - Single-session only machine.
- o MultiSession - Multi-session capable machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionType**<SessionType>

Gets machines with a specific session state.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionUid**<Int64>

Gets single-session machines with a specific session UID (\$null for no session).

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionUserName**<String>

Gets machines with a specific user name for the current session (in the form 'domain\user').

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

--	--

#### **-SessionUserSID<String>**

Gets machines with a specific SID of the current session user.

Session properties are always null for multi-session machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SID<String>**

Gets machines with a specific machine SID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SummaryState<DesktopSummaryState>**

Gets machines with a specific summary state.

Valid values are Off, Unregistered, Available, Disconnected, and InUse.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SupportedPowerActions<String[]>**

A list of power actions supported by this machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Tag<String>**

Gets machines where the session has the given SmartAccess tag.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UUID**<Guid>

Gets machines with the specified value of UUID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-VMToolsState**<VMToolsState>

Gets machines with a specific VM tools state.

Valid values are NotPresent, Unknown, NotStarted, and Running.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-WillShutdownAfterUse**<Boolean>

Gets machines depending on whether they shut down after use or not.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-WindowsConnectionSetting**<WindowsConnectionSetting>

Gets machines according to their current Windows connection setting (logon mode). Valid values are:

- o LogonEnabled - All logons are enabled.
- o Draining - New logons are disabled, but reconnections to existing sessions are allowed.
- o DrainingUntilRestart - Same as Draining, but setting reverts to LogonEnabled when machine next restarts.
- o LogonDisabled - All logons and reconnections are disabled.

This is a Windows setting and is not controlled by XenDesktop. It applies only to multi-session machines; for single-session machines its value is always LogonEnabled.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AssignedUserSID**<String>

Gets machines with the specific SID of the user to whom the desktop is assigned.



Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by **-ReturnTotalRecordCount**.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See [about\\_Broker\\_Filtering](#) for details.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.GroupInfo

Each GroupInfo object represents one group, and contains the following properties:

-- Count: The count of machines in this group.

-- Name: The value of the property the machines were grouped by (as a string).

If you do not specify -SortBy, groups are sorted with the largest count first.

#### Notes

To compare dates or times, use -Filter and relative comparisons. For more information, see about\_Broker\_Filtering and the examples.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Group-BrokerMachine -Property SummaryState -DesktopGroupName dg1
Group machines from the dg1 group by summary state.
```

##### ----- EXAMPLE 2 -----

```
C:\PS> Group-BrokerMachine -Property LastConnectionFailure -Filter { LastConnectionFailure -ne "None" -and LastConnectionTime -ge '-7' } -MaxRecordCount 1
For machines where the last connection attempt failed, list the most common reason for failure, ignoring connections that failed over a week ago.
```

##### ----- EXAMPLE 3 -----

```
C:\PS> Group-BrokerMachine -Property HostingServerName -DesktopCondition ICALatency -SortBy Name
List alphabetically the hypervisor servers hosting machines that are currently experiencing high network latency.
```

# Group-BrokerSession

May 10, 2016

Groups and counts sessions with the same value for a specified property.

## Syntax

```
Group-BrokerSession [-Uid] <Int64> -Property <String> [-AdminAddress <String>] [<CommonParameters>]
```

```
Group-BrokerSession -Property <String> [-SessionKey <Guid>] [-AgentVersion <String>] [-ApplicationInUse <String>] [-AppState <SessionAppState>] [-AppStateLastChangeTime <DateTime>] [-AutonomouslyBrokered <Boolean>] [-BrokeringDuration <Int32>] [-BrokeringTime <DateTime>] [-BrokeringUserName <String>] [-BrokeringUserSID <String>] [-CatalogName <String>] [-ClientAddress <String>] [-ClientName <String>] [-ClientPlatform <String>] [-ClientProductId <Int32>] [-ClientVersion <String>] [-ConnectedViaHostName <String>] [-ConnectedViaIP <String>] [-ConnectionMode <ConnectionMode>] [-ControllerDNSName <String>] [-DesktopGroupName <String>] [-DesktopGroupUid <Int32>] [-DesktopKind <DesktopKind>] [-DesktopSID <String>] [-DesktopUid <Int32>] [-DeviceId <String>] [-DNSName <String>] [-EstablishmentDuration <Int32>] [-EstablishmentTime <DateTime>] [-HardwareId <String>] [-Hidden <Boolean>] [-HostedMachineName <String>] [-HostingServerName <String>] [-HypervisorConnectionName <String>] [-ImageOutOfDate <Boolean>] [-InMaintenanceMode <Boolean>] [-IPAddress <String>] [-IsAnonymousUser <Boolean>] [-IsPhysical <Boolean>] [-LaunchedViaHostName <String>] [-LaunchedViaIP <String>] [-LogoffInProgress <Boolean>] [-LogonInProgress <Boolean>] [-MachineName <String>] [-MachineSummaryState <DesktopSummaryState>] [-MachineUid <Int32>] [-Metadata <String>] [-OSType <String>] [-PersistUserChanges <PersistUserChanges>] [-PowerState <PowerState>] [-Protocol <String>] [-ProvisioningType <ProvisioningType>] [-ReceiverIPAddress <String>] [-ReceiverName <String>] [-SecureIcaActive <Boolean>] [-SessionId <Int32>] [-SessionState <SessionState>] [-SessionStateChangeTime <DateTime>] [-SessionSupport <SessionSupport>] [-SessionType <SessionType>] [-StartTime <DateTime>] [-UntrustedUserName <String>] [-UserFullName <String>] [-UserName <String>] [-UserSID <String>] [-UserUPN <String>] [-ApplicationUid <Int32>] [-SharedDesktopUid <Int32>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Filters sessions using the specified criteria, then groups and counts matching sessions with the same value for a particular property. The number of sessions in the group, and the property value for the group, is output. For example:

```
C:\PS> Group-BrokerSession -Property SessionState
```

```
Count Name
```

```

```

```
43 Active
```

```
17 NonBrokeredSession
```

```
3 Disconnected
```

Filtering supports the same options as the Get-BrokerSession cmdlet, and allows filtering on both machine and session properties.

Group-BrokerSession is similar to the standard PowerShell Group-Object, but is faster than piping the output of Get-

BrokerSession into Group-Object when working with many machines.

Note that the MaxRecordCount, ReturnTotalRecordCount, Skip, and SortBy parameters apply to GroupInfo records output rather than the filtered sessions.

## Related topics

[Get-BrokerSession](#)

Group-Object

Parameters

**-Uid**<Int64>

Get session by its Uid.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Property**<String>

Selects the property by which matching sessions are grouped.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-SessionKey**<Guid>

Gets session having the specified unique key.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AgentVersion**<String>

Gets sessions with a specific Virtual Desktop Agent version.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ApplicationInUse**<String>

Gets sessions running specific applications (identified by their SDK Name property).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AppState**<SessionAppState>

Get sessions by their app state.

Valid values are PreLogon, PreLaunched, Active, Desktop, Lingering and NoApps.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AppStateLastChangeTime**<DateTime>

Get sessions by their app state change time.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AutonomouslyBrokered**<Boolean>

Gets sessions according to whether they are autonomously brokered or not. Autonomously brokered sessions are HDX sessions established by direct connection without being brokered.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-BrokeringDuration**<Int32>

Gets session with a specific time taken to broker. In general, Citrix recommends using -Filter and relative comparisons.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-BrokeringTime**<DateTime>

Get sessions brokered at a specific time. In general, Citrix recommends using -Filter and relative comparisons.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-BrokeringUserName**<String>

Get sessions by brokering user.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-BrokeringUserSID**<String>

Get sessions by brokering user SID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-CatalogName**<String>

Gets sessions on machines from a specific catalog name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ClientAddress**<String>

Get sessions by client IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Client Name**<String>

Get sessions by client name.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ClientPlatform**<String>

Get sessions by client platform.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Client Product Id**<Int32>

Get sessions by client product ID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Client Version**<String>

Get sessions by client version.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ConnectedViaHostName**<String>

Get sessions by host name of the incoming connection. This is usually a proxy or Citrix Access Gateway server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ConnectedViaIP**<String>

Get sessions by IP address of the incoming connection.

Required?	false
-----------	-------



Default Value	
Accept Pipeline Input?	false

#### **-ConnectionMode**<ConnectionMode>

Gets sessions by the way in which the most recent connection to the session was established.

Valid modes are Brokered, Unbrokered, LeasedConnection, VdaHighAvailabilityMode, ThirdPartyBroker, and ThirdPartyBrokerWithLicensing.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ControllerDNSName**<String>

Gets sessions that are hosted on machines which are registered with a specific controller.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupName**<String>

Gets sessions from a desktop group with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupUid**<Int32>

Gets sessions from a desktop group with the specified UID.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-DesktopKind**<DesktopKind>

Gets sessions on a desktop of a particular kind.

Valid values are Private and Shared.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopSID**<String>

Get sessions by desktop SID.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopUid**<Int32>

Get sessions by desktop Uid.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DeviceId**<String>

Get sessions by client device id.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-DNSName<String>**

Gets sessions by their machine's DNS name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-EstablishmentDuration<Int32>**

Gets sessions which took a specific time to establish. In general, Citrix recommends using -Filter and relative comparisons.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-EstablishmentTime<DateTime>**

Gets sessions which became established at a particular time. In general, Citrix recommends using -Filter and relative comparisons.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HardwareId<String>**

Get sessions by client hardware id.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-Hidden**<Boolean>

Get sessions by whether they are hidden or not. Hidden sessions are treated as though they do not exist when brokering sessions; a hidden session cannot be reconnected to, but a new session may be launched using the same entitlement.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HostedMachineName**<String>

Gets sessions by their machine's name as known to its hypervisor.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HostingServerName**<String>

Gets sessions hosted by a machine with a specific name of the hosting hypervisor server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HypervisorConnectionName**<String>

Gets sessions hosted by a machine with a specific name of the hosting hypervisor connection.

Required?	false

Default Value	
Accept Pipeline Input?	false

#### **-ImageOutOfDate**<Boolean>

Gets sessions hosted by a machine with a specific ImageOutOfDate setting.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-InMaintenanceMode**<Boolean>

Gets sessions hosted by a machine with a specific InMaintenanceMode setting.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IPAddress**<String>

Gets sessions hosted by a machine with a specific IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IsAnonymousUser**<Boolean>

Gets sessions depending on whether they were established anonymously (\$true) or not (\$false). An anonymous session is established without user credentials and a temporary local user account is used.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-IsPhysical**<Boolean>

Gets sessions hosted on machines where the flag indicating if the machine can be power managed by the Citrix Broker Service matches the requested value. Where the power state of the machine cannot be controlled, specify \$true, otherwise \$false.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LaunchedViaHostName**<String>

Get sessions by the host name of the Web Interface server from which a user launches a session.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LaunchedViaIP**<String>

Get sessions by the IP address of the Web Interface server from which a user launches a session.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LogoffInProgress**<Boolean>

Gets sessions by whether they are in the process of being logged off or not.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-LogonInProgress**<Boolean>

Gets sessions by whether they are still executing user logon processing or not.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineName**<String>

Gets sessions by their machine name (in the form DOMAIN\machine).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineSummaryState**<DesktopSummaryState>

Gets sessions on a machine with a specific summary state.

Valid values are Off, Unregistered, Available, Disconnected, Preparing, and InUse.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachineUid**<Int32>

Gets sessions on a machine with the specified UID.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OSType**<String>

Gets sessions with a specific type of operating system.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PersistUserChanges**<PersistUserChanges>

Gets sessions where the user changes are persisted in a particular manner. Values can be:

- o OnLocal - User changes are persisted locally.
- o Discard - User changes are discarded.
- o OnPvd - User changes are persisted on the Pvd.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PowerState**<PowerState>



Gets sessions on machines in the specified power state.

Valid values are Unmanaged, Unknown, Unavailable, On, Suspended, TurningOn, TurningOff, Suspending, and Resuming.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Protocol<String>**

Get sessions by connection protocol. Valid values are HDX, RDP and Console.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ProvisioningType<ProvisioningType>**

Gets sessions hosted on machines provisioned in a particular manner. Values can be:

- o Manual - No automated provisioning.
- o PVS - Machine provisioned by PVS (machine may be physical, blade, VM,...).
- o MCS - Machine provisioned by MCS (machine must be VM).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReceiverIPAddress<String>**

Gets sessions with the specified client IP address supplied by Receiver (for example, StoreFront) when the session was launched, or reconnected.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ReceiverName**<String>

Gets sessions with the specified client name supplied by Receiver (for example, StoreFront) when the session was launched, or reconnected.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecureIcaActive**<Boolean>

Get sessions by their use of SecureICA.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionId**<Int32>

Deprecated.

Gets sessions by session ID, a unique identifier that Remote Desktop Services uses to track the session but it is only unique on that machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionState**<SessionState>

Get sessions by their state.

Valid values are Other, PreparingNewSession, Connected, Active, Disconnected, Reconnecting, NonBrokeredSession, and Unknown.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionStateChangeTime**<DateTime>

Get sessions by their last state change time. In general, Citrix recommends using -Filter and relative comparisons.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionSupport**<SessionSupport>

Gets sessions hosted on machines which support the required pattern of sessions. Values can be:

- o SingleSession - Single-session only machine.
- o MultiSession - Multi-session capable machine.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionType**<SessionType>

Get sessions by their type.

Valid values are Application and Desktop.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-StartTime<DateTime>**

Get sessions by their start time. In general, Citrix recommends using -Filter and relative comparisons.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UntrustedUserName<String>**

Gets sessions by the untrusted user name reported directly from the machine (in the form DOMAIN\user).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UserFullName<String>**

Gets sessions by user's full name (usually 'first-name last-name').

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UserName<String>**

Get sessions by user name (in the form DOMAIN\user).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UserSID<String>**

Get sessions by user's Windows SID.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-UserUPN<String>**

Gets sessions by user's User Principal Name (in the form user@domain).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ApplicationUid<Int32>**

Get sessions running the application with the specified Uid.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-SharedDesktopUid<Int32>**

Get sessions by SharedDesktop Uid.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount<SwitchParameter>**

When specified, this causes the cmdlet to output an error record containing the number of records available. This error

record is additional information and does not affect the objects written to the output pipeline. See about\_Broker\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell style filter expression. See `about_Broker_Filtering` for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

None You cannot pipe input into this cmdlet.

### Return Values

Citrix.Broker.Admin.SDK.GroupInfo

Each GroupInfo object represents one group, and contains the following properties:

-- Count: The count of sessions in this group.

-- Name: The value of the property the sessions were grouped by (as a string).

If you do not specify -SortBy, groups are sorted with the largest count first.

### Notes

To compare dates or times, use -Filter and relative comparisons. For more information, see `about_Broker_Filtering` and the examples.

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Group-BrokerSession -Property SessionState -DesktopGroupName dg1
```

Group sessions on machines from the dg1 group by session state.

----- **EXAMPLE 2** -----

```
C:\PS> Group-BrokerSession -Property ClientName -ClientName 'ThinClient*' -SortBy Name
```

List alphabetically the names of the clients connected to the site, but only show clients whose names starts with 'ThinClient'.



# Import-BrokerDesktopPolicy

May 10, 2016

Sets the site wide Citrix Group Policy settings for the site.

## Syntax

```
Import-BrokerDesktopPolicy [-Policy] <Byte[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Import-BrokerDesktopPolicy sets the site wide Citrix Group Policy settings. A successful call to this cmdlet will result in the supplied data being uploaded to every machine in the site prior to its next session launch.

## Related topics

[Export-BrokerDesktopPolicy](#)

[New-BrokerConfigurationSlot](#)

[New-BrokerMachineConfiguration](#)

## Parameters

**-Policy**<Byte[]>

The configuration data containing the Citrix Group Policy settings to apply to every machine in the site.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

System.Byte[] The configuration data as an opaque binary blob.

### Return Values

None

### Notes

Import-BrokerDesktopPolicy performs a specialized operation. Direct usage of it in scripts is discouraged, and could result in data corruption. It is recommended that this operation be performed via the Citrix Studio.

### Examples

#### ----- **EXAMPLE 1** -----

```
C:\PS> Import-BrokerDesktopPolicy $policyData
```

This command sets the Citrix Group Policy settings in the site. These policy settings are then applied to every machine prior to the next session launch.

# Move-BrokerAdminFolder

May 10, 2016

Moves a folder to another place in the hierarchy, optionally renaming it

## Syntax

Move-BrokerAdminFolder [-InputObject] <AdminFolder[]> [-Destination] <AdminFolder> [-NewName <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Move-BrokerAdminFolder [-Name] <String> [-Destination] <AdminFolder> [-NewName <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The Move-BrokerAdminFolder cmdlet moves a folder for organising objects for administration purposes (for example, Applications) to another position in the hierarchy.

The following special characters are not allowed in the new FolderName: \ / ; : # . \* ? = < > | [ ] ( ) " ' `

## Related topics

[Get-BrokerAdminFolder](#)

[New-BrokerAdminFolder](#)

## Parameters

**-InputObject**<AdminFolder[]>

The folder(s) to be moved

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

A pattern matching the names of folders to be moved

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Destination**<AdminFolder>

The destination folder the folder being moved should end up in

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-NewName**<String>

The name the new folder should have in the destination folder

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Depends on parameter Parameters can be piped by property name.

### Return Values

None or Citrix.Broker.Admin.SDK.AdminFolder

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.AdminFolder object.

### Examples

#### ----- **EXAMPLE 1** -----

```
Move-BrokerAdminFolder F1\XXX\ F2\
```

Moves the folder called XXX within the folder F1\ to a new home in F2\

#### ----- **EXAMPLE 2** -----

```
Move-BrokerAdminFolder F1\XXX\ F2\ -NewName YYY
```

Moves the folder called XXX within the folder F1\ to a new home in F2\ renaming it to YYY in the process

# Move-BrokerApplication

May 10, 2016

Move a published application from one admin folder to another

## Syntax

```
Move-BrokerApplication [-InputObject] <Application[]> [-Destination] <AdminFolder> [-NewName <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Move-BrokerApplication [-Name] <String> [-Destination] <AdminFolder> [-NewName <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Move-BrokerApplication cmdlet moves a published application from one place to another in the tree of admin folders, optionally renaming it in the process (if you only want to change the name of the application for administrative purposes and not its location in the tree, use the Rename-BrokerApplication cmdlet).

The location and name of a published application in this sense is only of interest to the administrator, changes do not affect the end-user experience.

## Related topics

[New-BrokerApplication](#)

[Add-BrokerApplication](#)

[Get-BrokerApplication](#)

[Remove-BrokerApplication](#)

[Rename-BrokerApplication](#)

[Set-BrokerApplication](#)

## Parameters

**-InputObject**<Application[]>

The application(s) to be moved

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

The application(s) to be moved

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Destination**<AdminFolder>

The destination location within the admin folder hierarchy

Required?	true
Default Value	
Accept Pipeline Input?	false

**-NewName**<String>

The new name of the application in its new destination

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.Application You can pipe applications to Move-BrokerApplication.

#### Return Values

None or Citrix.Broker.Admin.SDK.Application

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.Application object.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Move-BrokerApplication -Name 'App1' -Destination 'F1\'
Moves the application in the root folder called "App1" to the folder "F1\".
```

##### ----- EXAMPLE 2 -----

```
C:\PS> Move-BrokerApplication 'F1\App1' 'F2\' -NewName 'Application1'
Moves the application in folder "F1" called "App1" to the folder "F2", renaming it to "Application1" in the process.
```



# New-BrokerAccessPolicyRule

May 10, 2016  
Creates a new rule in the site's access policy.

## Syntax

New-BrokerAccessPolicyRule [-Name] <String> -DesktopGroupUid <Int32> [-AllowedConnections <AllowedConnection>] [-AllowedProtocols <String[]>] [-AllowedUsers <AllowedUser>] [-AllowRestart <Boolean>] [-Description <String>] [-Enabled <Boolean>] [-ExcludedClientIPFilterEnabled <Boolean>] [-ExcludedClientIPs <IPAddressRange[]>] [-ExcludedClientNameFilterEnabled <Boolean>] [-ExcludedClientNames <String[]>] [-ExcludedSmartAccessFilterEnabled <Boolean>] [-ExcludedSmartAccessTags <String[]>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-HdxSslEnabled <Boolean>] [-IncludedClientIPFilterEnabled <Boolean>] [-IncludedClientIPs <IPAddressRange[]>] [-IncludedClientNameFilterEnabled <Boolean>] [-IncludedClientNames <String[]>] [-IncludedSmartAccessFilterEnabled <Boolean>] [-IncludedSmartAccessTags <String[]>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

New-BrokerAccessPolicyRule [-Name] <String> -IncludedDesktopGroups <DesktopGroup[]> [-IncludedDesktopGroupFilterEnabled <Boolean>] [-AllowedConnections <AllowedConnection>] [-AllowedProtocols <String[]>] [-AllowedUsers <AllowedUser>] [-AllowRestart <Boolean>] [-Description <String>] [-Enabled <Boolean>] [-ExcludedClientIPFilterEnabled <Boolean>] [-ExcludedClientIPs <IPAddressRange[]>] [-ExcludedClientNameFilterEnabled <Boolean>] [-ExcludedClientNames <String[]>] [-ExcludedSmartAccessFilterEnabled <Boolean>] [-ExcludedSmartAccessTags <String[]>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-HdxSslEnabled <Boolean>] [-IncludedClientIPFilterEnabled <Boolean>] [-IncludedClientIPs <IPAddressRange[]>] [-IncludedClientNameFilterEnabled <Boolean>] [-IncludedClientNames <String[]>] [-IncludedSmartAccessFilterEnabled <Boolean>] [-IncludedSmartAccessTags <String[]>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The New-BrokerAccessPolicyRule cmdlet adds a new rule to the site's access policy.

An access policy rule defines a set of connection filters and access control rights relating to a desktop group. These allow fine-grained control of what access is granted to a desktop group based on details of, for example, a user's endpoint device, its address, and the user's identity.

Multiple rules in the access policy can apply to the same desktop group.

For a user to gain access to a desktop group via a rule their connection must match all its enabled include filters, and none of its enabled exclude filters. In addition, for a user to be able to launch a desktop or application resource session from the desktop group, they must have an entitlement to use the resource granted by the entitlement or assignment policies, or by direct machine assignment.

## Related topics

- [Get-BrokerAccessPolicyRule](#)
- [Set-BrokerAccessPolicyRule](#)
- [Rename-BrokerAccessPolicyRule](#)
- [Remove-BrokerAccessPolicyRule](#)

## Parameters

**-Name**<String>

Specifies the administrative name of the new rule. Each rule within the site's access policy must have a unique name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-DesktopGroupUid**<Int32>

Specifies the desktop group to which the new rule applies.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-IncludedDesktopGroups**<DesktopGroup[]>

This parameter is supported for backward compatibility only. If used only a single desktop group UID can be specified.

The IncludedDesktopGroups and IncludedDesktopGroupFilterEnabled parameters have been superseded by the DesktopGroupUid parameter.

Required?	true
Default Value	(empty list)
Accept Pipeline Input?	true (ByPropertyName)

**-AllowedConnections**<AllowedConnection>

Specifies whether connections must be local or via Access Gateway, and if so whether specified SmartAccess tags must be provided by Access Gateway with the connection. This property forms part of the included SmartAccess tags filter.

Valid values are Filtered, NotViaAG, and ViaAG.

For a detailed description of this property see "help about\_Broker\_AccessPolicy".

Required?	false
Default Value	Filtered
Accept Pipeline Input?	true (ByPropertyName)

**-AllowedProtocols**<String[]>

Specifies the protocols (for example HDX, RDP) available to the user for sessions delivered from the new rule's desktop group. If the user gains access to a desktop group by multiple rules, the allowed protocol list is the combination of the protocol lists from all those rules.

If the protocol list is empty, access to the desktop group is implicitly denied.

Required?	false
Default Value	HDX
Accept Pipeline Input?	true (ByPropertyName)

**-AllowedUsers**<AllowedUser>

Specifies the behavior of the included users filter of the new rule. This can restrict access to a list of named users or groups, allow access to any authenticated user, any user (whether authenticated or not), or only non-authenticated users. For a detailed description of this property see "help about\_Broker\_AccessPolicy".

Valid values are Filtered, AnyAuthenticated, Any, AnonymousOnly and FilteredOrAnonymous.

Required?	false
Default Value	Filtered
Accept Pipeline Input?	true (ByPropertyName)

**-AllowRestart**<Boolean>

Specifies if the user can restart sessions delivered from the new rule's desktop group. Session restart is handled as follows: For sessions on single-session power-managed machines, the machine is powered off, and a new session launch request made; for sessions on multi-session machines, a logoff request is issued to the session, and a new session launch request made; otherwise the property is ignored.

Required?	false
Default Value	false

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### -Description<String>

Specifies an optional description of the new rule. The text is purely informational for the administrator, it is never visible to the end user.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### -Enabled<Boolean>

Specifies whether the new rule is initially enabled. A disabled rule is ignored when evaluating the site's access policy.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

#### -ExcludedClientIPFilterEnabled<Boolean>

Specifies whether the excluded client IP address filter is initially enabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### -ExcludedClientIPs<IPAddressRange[]>

Specifies IP addresses of user devices explicitly denied access to the new rule's desktop group. Addresses can be specified as simple numeric addresses or as subnet masks (for example, 10.40.37.5 or 10.40.0.0/16). This property forms part of the excluded client IP address filter.

Required?	false
Default Value	(empty list)
Accept Pipeline Input?	true (ByPropertyName)

#### -ExcludedClientNameFilterEnabled<Boolean>

Specifies whether the excluded client names filter is initially enabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### -ExcludedClientNames<String[]>

Specifies names of user devices explicitly denied access to the new rule's desktop group. This property forms part of the excluded client names filter.

--	--

Required?	false
Default Value	(empty list)
Accept Pipeline Input?	true (ByPropertyName)

#### **-ExcludedSmartAccessFilterEnabled**<Boolean>

Specifies whether the excluded SmartAccess tags filter is initially enabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### **-ExcludedSmartAccessTags**<String[]>

Specifies SmartAccess tags which explicitly deny access to the new rule's desktop group if any occur in those provided by Access Gateway with the user's connection. This property forms part of the excluded SmartAccess tags filter.

Required?	false
Default Value	(empty list)
Accept Pipeline Input?	true (ByPropertyName)

#### **-ExcludedUserFilterEnabled**<Boolean>

Specifies whether the excluded users filter is initially enabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### **-ExcludedUsers**<User[]>

Specifies any users and groups who are explicitly denied access to the new rule's desktop group. This property forms part of the excluded users filter.

Required?	false
Default Value	(empty list)
Accept Pipeline Input?	true (ByPropertyName)

#### **-HdxSslEnabled**<Boolean>

Indicates whether SSL encryption is enabled for sessions delivered from the rule's desktop group.

Required?	false
Default Value	\$false
Accept Pipeline Input?	true (ByPropertyName)

#### **-IncludedClientIPFilterEnabled**<Boolean>

Specifies whether the included client IP address filter is initially enabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

**-IncludedClientIPs**<IPAddressRange[]>

Specifies IP addresses of user devices allowed access to the new rule's desktop group. Addresses can be specified as simple numeric addresses or as subnet masks (for example, 10.40.37.5 or 10.40.0.0/16). This property forms part of the included client IP address filter.

Required?	false
Default Value	(empty list)
Accept Pipeline Input?	true (ByPropertyName)

**-IncludedClientNameFilterEnabled**<Boolean>

Specifies whether the included client name filter is initially enabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

**-IncludedClientNames**<String[]>

Specifies names of user devices allowed access to the new rule's desktop group. This property forms part of the included client names filter.

Required?	false
Default Value	(empty list)
Accept Pipeline Input?	true (ByPropertyName)

**-IncludedSmartAccessFilterEnabled**<Boolean>

Specifies whether the included SmartAccess tags filter is initially enabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

**-IncludedSmartAccessTags**<String[]>

Specifies SmartAccess tags which grant access to the new rule's desktop group if any occur in those provided by Access Gateway with the user's connection. This property forms part of the excluded SmartAccess tags filter.

Required?	false
Default Value	(empty list)

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-IncludedUserFilterEnabled**<Boolean>

Specifies whether the included users filter is initially enabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### **-IncludedUsers**<User[]>

Specifies users and groups who are granted access to the new rule's desktop group. This property forms part of the included users filter.

Required?	false
Default Value	(empty list)
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### **-IncludedDesktopGroupFilterEnabled**<Boolean>

This parameter is supported for backward compatibility only. If used the supplied value must be \$true.

The IncludedDesktopGroups and IncludedDesktopGroupFilterEnabled parameters have been superseded by the DesktopGroupUid parameter.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

#### Input Type

None You cannot pipe input into this cmdlet.

## Return Values

Citrix.Broker.Admin.SDK.AccessPolicyRule

New-BrokerAccessPolicyRule returns the newly created access policy rule.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> $dg = Get-BrokerDesktopGroup 'Tech Support'
```

```
C:\PS> New-BrokerAccessPolicyRule 'UK Tech Support' -IncludedUserFilterEnabled $true -IncludedUsers support\uk-staff -DesktopGroupUid $dg.Uid -AllowedProtocols 'HDX'
```

Creates an access policy rule allowing access to the Tech Support desktop group for all users of the SUPPORT\uk-staff group. Connections to desktop or application resources in the group can only be made using the HDX protocol.

For users to gain access to resources in the group also requires that, depending on the desktop kind of the group, appropriate assignment or entitlement policy rules, or explicit machine assignments exist.

# New-BrokerAdminFolder

May 10, 2016

Creates a new admin folder.

## Syntax

```
New-BrokerAdminFolder [-FolderName] <String> [-ParentFolder <AdminFolder>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The New-BrokerAdminFolder cmdlet creates a new folder for organising objects for administration purposes (for example, Applications).

New-BrokerAdminFolder creates the folder object and optionally places it within an existing admin folder if required.

The following special characters are not allowed in the FolderName: \ / ; : # . \* ? = < > | [ ] ( ) " ' `

## Related topics

[Get-BrokerAdminFolder](#)

[Remove-BrokerAdminFolder](#)

## Parameters

**-FolderName**<String>

The simple name of the new folder within its parent (if any)

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ParentFolder**<AdminFolder>

The name or UID of the parent folder (if any)

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)



### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Depends on parameter Parameters can be piped by property name.

### Return Values

Citrix.Broker.Admin.SDK.AdminFolder

The new admin folder.

### Examples

#### ----- EXAMPLE 1 -----

New-BrokerAdminFolder F1

Creates an admin folder called F1 under the root folder (i.e. F1\)

#### ----- EXAMPLE 2 -----

New-BrokerAdminFolder F2 -AdminFolder F1\

Creates an admin folder called F2 under the folder F1\ (i.e. F1\F2\)

# New-BrokerAppAssignmentPolicyRule

May 10, 2016

Creates a new application rule in the site's assignment policy.

## Syntax

```
New-BrokerAppAssignmentPolicyRule [-Name] <String> -DesktopGroupUid <Int32> [-Description <String>] [-Enabled <Boolean>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The New-BrokerAppAssignmentPolicyRule cmdlet adds a new application rule to the site's assignment policy.

An application rule in the assignment policy defines the users who are entitled to a self-service persistent machine assignment from the rule's desktop group; once assigned the machine can run one or more applications published from the group.

The following constraints apply when creating an application assignment rule for a desktop group:

- o The group's desktop kind must be Private
- o The group's delivery type must be AppsOnly
- o Only a single application rule can apply to a given group
- o Application assignment rules cannot be applied to RemotePC groups.

When a user selects an application published from a private group, a currently unassigned machine is selected from the group and permanently assigned to the user. An application session is then launched to the machine. Subsequent launches are routed directly to the now assigned machine.

Once a machine has been assigned in this way, the original assignment rule plays no further part in access to the machine.

## Related topics

[Get-BrokerAppAssignmentPolicyRule](#)

[Set-BrokerAppAssignmentPolicyRule](#)

[Rename-BrokerAppAssignmentPolicyRule](#)

[Remove-BrokerAppAssignmentPolicyRule](#)

## Parameters

**-Name**<String>

Specifies the administrative name of the new application rule. Each rule in the site's assignment policy must have a unique name (irrespective of whether they are desktop or application rules).

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-DesktopGroupUid<Int32>**

Specifies the unique ID of the desktop group to which the new application rule applies.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Description<String>**

Specifies an optional description of the new application rule. The text is purely informational for the administrator, it is never visible to the end user.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

#### **-Enabled<Boolean>**

Specifies whether the new application rule is initially enabled. A disabled rule is ignored when evaluating the site's assignment policy.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

#### **-ExcludedUserFilterEnabled<Boolean>**

Specifies whether the excluded users filter is initially enabled. If the filter is disabled then any user entries in the filter are ignored when assignment policy rules are evaluated.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### **-ExcludedUsers<User[]>**

Specifies the excluded users filter of the new application rule, that is, the users and groups who are explicitly denied an entitlement to a machine assignment from the rule.

This can be used to exclude users or groups who would otherwise gain access by groups specified in the included users filter.

Required?	false
Default Value	(empty list)
Accept Pipeline Input?	true (ByPropertyName)

#### **-IncludedUserFilterEnabled<Boolean>**

Specifies whether the included users filter is initially enabled. If the filter is disabled then any user who satisfies the requirements of the access policy is implicitly granted an entitlement to a machine assignment by the new application rule.

Users who would be implicitly granted access when the filter is disabled can still be explicitly denied access using the excluded users filter.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

#### **-IncludedUsers<User[]>**

Specifies the included users filter of the new application rule, that is, the users and groups who are granted an entitlement to a machine assignment by the rule.

If a user appears explicitly in the excluded users filter of the rule or is a member of a group that appears in the excluded users filter, no entitlement is granted whether or not the user appears in the included users filter.

Required?	false

Default Value	(empty list)
Accept Pipeline Input?	true (ByPropertyName)

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

None You cannot pipe input into this cmdlet.

### Return Values

Citrix.Broker.Admin.SDK.AppAssignmentPolicyRule

New-BrokerAppAssignmentPolicyRule returns the newly created application rule in the assignment policy.

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> $dg = Get-BrokerDesktopGroup 'Sales Support'
```

```
C:\PS> New-BrokerAppAssignmentPolicyRule 'UK Office' -DesktopGroupUid $dg.Uid -IncludedUsers sales\uk-staff
```

Creates an application rule in the assignment policy that grants all members of the SALES\uk-staff group an entitlement to a single machine from the Sales Support desktop group. The machine can be used for running applications published from the group.

# New-BrokerAppEntitlementPolicyRule

May 10, 2016

Creates a new application rule in the site's entitlement policy.

## Syntax

```
New-BrokerAppEntitlementPolicyRule [-Name] <String> -DesktopGroupUid <Int32> [-Description <String>] [-Enabled <Boolean>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-SessionReconnection <SessionReconnection>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The New-BrokerAppEntitlementPolicyRule cmdlet adds a new application rule to the site's entitlement policy.

An application rule in the entitlement policy defines the users who are allowed per-session access to a machine to run one or more applications published from the rule's desktop group.

The following constraints apply when creating an application entitlement rule for a desktop group:

- o The group's desktop kind must be Shared
- o The group's delivery type must be AppsOnly or DesktopsAndApps
- o Only a single application rule can apply to a given group

When a user selects an application published from a shared group, a machine is selected from the group on which to run the application. No permanent association exists between the user and the selected machine; once the session ends the association also ends.

Even though only a single application entitlement and therefore session can be defined for a group, the user can still run multiple applications from the group because the applications run within the same session.

## Related topics

[Get-BrokerAppEntitlementPolicyRule](#)

[Set-BrokerAppEntitlementPolicyRule](#)

[Rename-BrokerAppEntitlementPolicyRule](#)

[Remove-BrokerAppEntitlementPolicyRule](#)

## Parameters

**-Name**<String>

Specifies the administrative name of the new application rule. Each rule in the site's entitlement policy must have a unique name (irrespective of whether they are desktop or application rules).

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-DesktopGroupUid<Int32>**

Specifies the unique ID of the desktop group to which the new application rule applies.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Description<String>**

Specifies an optional description of the new application rule. The text is purely informational for the administrator, it is never visible to the end user.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

#### **-Enabled<Boolean>**

Specifies whether the new application rule is initially enabled. A disabled rule is ignored when evaluating the site's entitlement policy.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

#### **-ExcludedUserFilterEnabled<Boolean>**

Specifies whether the excluded users filter is initially enabled. If the filter is disabled then any user entries in the filter are ignored when entitlement policy rules are evaluated.

--	--

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### **-ExcludedUsers<User[]>**

Specifies the excluded users filter of the application rule, that is, the users and groups who are explicitly denied entitlements to published applications from the desktop group.

This can be used to exclude users or groups who would otherwise gain access by groups specified in the included users filter.

Required?	false
Default Value	(empty list)
Accept Pipeline Input?	true (ByPropertyName)

#### **-IncludedUserFilterEnabled<Boolean>**

Specifies whether the included users filter is initially enabled. If the filter is disabled then any user who satisfies the requirements of the access policy is implicitly granted an entitlement to an application session by the new rule.

Users who would be implicitly granted access when the filter is disabled can still be explicitly denied access using the excluded users filter.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

#### **-IncludedUsers<User[]>**

Specifies the included users filter of the application rule, that is, the users and groups who are granted an entitlement to an application session by the new rule.

If a user appears explicitly in the excluded users filter of the rule or is a member of a group that appears in the excluded users filter, no entitlement is granted whether or not the user appears in the included users filter.

Required?	false
Default Value	(empty list)



Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-SessionReconnection**<SessionReconnection>

Defines reconnection (roaming) behavior for sessions launched using this rule. Session reconnection control is an experimental and unsupported feature.

Required?	false
Default Value	Always
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.AppEntitlementPolicyRule

New-BrokerAppEntitlementPolicyRule returns the newly created application rule in the entitlement policy.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> $dg = Get-BrokerDesktopGroup 'Customer Support'
```

```
C:\PS> New-BrokerAppEntitlementPolicyRule 'UK Office' -DesktopGroupUid $dg.Uid -IncludedUsers support\uk-staff
```

Creates an application rule in the entitlement policy that entitles all members of the SUPPORT\uk-staff group to a machine for running applications published from the Customer Support desktop group.

# New-BrokerApplication

May 10, 2016  
Creates a new published application.

## Syntax

New-BrokerApplication [-Name] <String> -CommandLineExecutable <String> -DesktopGroup <DesktopGroup> [-AdminFolder <AdminFolder>] [-ApplicationType <ApplicationType>] [-BrowserName <String>] [-ClientFolder <String>] [-CommandLineArguments <String>] [-CpuPriorityLevel <CpuPriorityLevel>] [-Description <String>] [-Enabled <Boolean>] [-IconFromClient <Boolean>] [-IconUid <Int32>] [-Priority <Int32>] [-PublishedName <String>] [-SecureCmdLineArgumentsEnabled <Boolean>] [-ShortcutAddedToDesktop <Boolean>] [-ShortcutAddedToStartMenu <Boolean>] [-StartMenuFolder <String>] [-UserFilterEnabled <Boolean>] [-UUID <Guid>] [-Visible <Boolean>] [-WaitForPrinterCreation <Boolean>] [-WorkingDirectory <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The New-BrokerApplication cmdlet creates a new published application in the site.

New-BrokerApplication creates the application object, and associates it with a desktop group. Application objects have three names that identify them (in addition to their Uid): the Name, BrowserName and the PublishedName. The BrowserName is unique across the entire site, and is primarily used internally. The Name is also unique and is what is seen by the administrator; it contains any prefix for an enclosing admin folder (if any). The PublishedName is not unique and is what is seen by the users.

You can create both HostedOnDesktop and InstalledOnClient applications but the ApplicationType cannot be changed later.

The following special characters are not allowed in the Name, BrowserName or the PublishedName properties: \ / ; : # . \* ? = < > | [ ] ( ) " ' "

In addition the ` character is not allowed in the Name property.

See about\_Broker\_Applications for more information.

## Related topics

- [Add-BrokerApplication](#)
- [Remove-BrokerApplication](#)
- [Get-BrokerApplication](#)
- [Remove-BrokerApplication](#)
- [Rename-BrokerApplication](#)
- [Move-BrokerApplication](#)
- [Set-BrokerApplication](#)

## Parameters

**-Name**<String>

Specifies the name of the application (must be unique within folder).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-CommandLineExecutable**<String>

Specifies the name of the executable file to launch. The full path need not be provided if it's already in the path. Environment variables can also be used.

--	--

Required?	true
Default Value	(required)
Accept Pipeline Input?	true (ByPropertyName)

#### **-DesktopGroup**<DesktopGroup>

Specifies which desktop group this application should be associated with. The association between application and desktop groups can be added or removed using the Add-BrokerApplication and Remove-BrokerApplication cmdlets.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-AdminFolder**<AdminFolder>

The folder in which the new application should reside (if any).

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-ApplicationType**<ApplicationType>

Specifies the type of the application: HostedOnDesktop or InstalledOnClient.

Required?	false
Default Value	(required)
Accept Pipeline Input?	true (ByPropertyName)

#### **-BrowserName**<String>

Specifies the internal name for this application. It must be unique in the site.

Required?	false
Default Value	(same as Name)
Accept Pipeline Input?	true (ByPropertyName)

#### **-ClientFolder**<String>

Specifies the folder that the application belongs to as the user sees it. This is the application folder that is seen in the Citrix Online Plug-in, in Web Services, and also in the end-user's Start menu. Subdirectories can be specified with '\' character. The following special characters are not allowed: / \* ? < > | " : . Note that this property cannot be set for applications of type InstalledOnClient.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

#### **-CommandLineArguments**<String>

Specifies the command-line arguments to use when launching the executable. Environment variables can be used.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

#### **-CpuPriorityLevel**<CpuPriorityLevel>

Specifies the CPU priority for the launched process. Valid values are: Low, BelowNormal, Normal, AboveNormal, and High. Note that this property cannot be set for applications of type InstalledOnClient.

Required?	false
Default Value	Normal
Accept Pipeline Input?	true (ByPropertyName)

#### **-Description**<String>

Specifies the description of the application. This is only seen by Citrix administrators and is not visible to users.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

#### **-Enabled**<Boolean>

Specifies whether or not this application can be launched.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

#### **-IconFromClient**<Boolean>

Specifies if the app icon should be retrieved from the application on the client. This is reserved for possible future use, and all applications of type HostedOnDesktop cannot set or change this value.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### **-IconUid<Int32>**

Specifies which icon to use for this application. This icon is visible both to the administrator (in the consoles) and to the user. If no icon is specified, then a generic built-in application icon is used.

Required?	false
Default Value	2
Accept Pipeline Input?	true (ByPropertyName)

#### **-Priority<Int32>**

Specifies the priority of the mapping between the application and desktop group. A value of zero has the highest priority, with increasing values indicating lower priorities.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PublishedName<String>**

The name seen by end users who have access to this application.

Required?	false
Default Value	The same value as that supplied for the name of the application.
Accept Pipeline Input?	true (ByPropertyName)

#### **-SecureCmdLineArgumentsEnabled<Boolean>**

Specifies whether the command-line arguments are secured or not. This is reserved for possible future use, and all applications of type HostedOnDesktop can only have this value set to true.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

#### **-ShortcutAddedToDesktop<Boolean>**

Specifies whether or not a shortcut to the application should be placed on the user device. This is valid only for the Citrix Online Plug-in.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

**-ShortcutAddedToStartMenu**<Boolean>

Specifies whether a shortcut to the application should be placed in the user's start menu on their user device.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

**-StartMenuFolder**<String>

Specifies the name of the start menu folder that holds the application shortcut (if any). This is valid only for the Citrix Online Plug-in. Subdirectories can be specified with '\' character. The following special characters are not allowed: / \* ? < > | " .:

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

**-UserFilterEnabled**<Boolean>

Specifies whether the application's user filter is enabled or disabled. Where the user filter is enabled, the application is visible only to users who appear in the filter (either explicitly or by virtue of group membership).

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

**-UUID**<Guid>

An optional GUID for this application.

Required?	false
Default Value	A new GUID is generated if none is supplied.
Accept Pipeline Input?	true (ByPropertyName)

**-Visible**<Boolean>

Specifies whether or not this application is visible to users. Note that it's possible for an application to be disabled and still visible.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

**-WaitForPrinterCreation**<Boolean>

Specifies whether or not the session waits for the printers to be created before allowing the user to interact with the session. Note that this property cannot be set for applications of type InstalledOnClient.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

**-WorkingDirectory**<String>

Specifies which working directory the executable is launched from. Environment variables can be used.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false



## Input Type

Depends on parameter Parameters can be piped by property name.

## Return Values

Citrix.Broker.Admin.SDK.Application

New-BrokerApplication returns an Application object.

## Notes

Usually only the Name is specified with the New-BrokerApplication cmdlet, and the system chooses a BrowserName and PublishedName for you. By default the BrowserName is the same as the Name, if it is unique in the site. If not, then "-x" is appended to the name, where "x" is a number. For instance, if there is already an application with a BrowserName of "Notepad" and a new application is created with a Name of "Notepad", then the new application gets a BrowserName of "Notepad-1". If another "Notepad" is published, it has a BrowserName of "Notepad-2".

That said, the BrowserName can optionally be specified as well.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> New-BrokerApplication -ApplicationType HostedOnDesktop -Name "Notepad" -CommandLineExecutable "notepad.exe" -DesktopGroup PrivateDG1
```

Creates and returns an object for a published application called "Notepad" that launches "notepad.exe".

### ----- EXAMPLE 2 -----

```
C:\PS> $dg = Get-BrokerDesktopGroup "SharedDG1"
C:\PS> $app = New-BrokerApplication -ApplicationType HostedOnDesktop -Name "Notepad" -CommandLineExecutable "notepad.exe" -DesktopGroup $dg
C:\PS> $group = Get-BrokerDesktopGroup -Name "Shared desktop group"
C:\PS> Add-BrokerApplication $app -DesktopGroup $group
C:\PS> $fta = Get-BrokerImportedFTA -ExtensionName ".txt"
C:\PS> New-BrokerConfiguredFTA -ImportedFTA $fta -ApplicationUid $app.Uid
```

This is a much more complete example. It creates an application object to publish Notepad and associates it first with the "SharedDG1" desktop group.

Next it adds an additional desktop group (one that can host applications), and publishes the application to that desktop group. It then gets the ImportedFTA object for the .txt file-type extension (this assumes file-type associations have already been imported), and then configures it so that ".txt" is associated with the published application.

Note: The appropriate access policy and app assignment/entitlement rules must also be configured to allow access to the application.

# New-BrokerAssignmentPolicyRule

May 10, 2016  
Creates a new desktop rule in the site's assignment policy.

## Syntax

New-BrokerAssignmentPolicyRule [-Name] <String> -DesktopGroupUid <Int32> [-ColorDepth <ColorDepth>] [-Description <String>] [-Enabled <Boolean>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-IconUid <Int32>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-MaxDesktops <Int32>] [-PublishedName <String>] [-SecureIcaRequired <Boolean>] [-UUID <Guid>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The New-BrokerAssignmentPolicyRule cmdlet adds a new desktop rule to the site's assignment policy.

A desktop rule in the assignment policy defines the users who are entitled to self-service persistent machine assignments from the rule's desktop group. A rule defines how many machines a user is allowed from the group for delivery of full desktop sessions.

The following constraints apply when creating a desktop assignment rule for a desktop group:

- o The group's desktop kind must be Private
- o The group's delivery type must be DesktopsOnly
- o Only one desktop assignment rule can be created for RemotePC groups.

When a user selects a machine assignment entitlement from a private group, a currently unassigned machine is selected from the group and permanently assigned to the user to create an assigned desktop. A desktop session is then launched to the machine. Subsequent launches are routed directly to the now assigned machine.

Once a machine has been assigned in this way, the original assignment rule plays no further part in access to the new desktop.

Multiple desktop rules in the assignment policy can apply to the same desktop group. Where a user is granted entitlements by more than one rule for the same group, they can have as many machine assignments from the group as the total of their entitlements.

## Related topics

- [Get-BrokerAssignmentPolicyRule](#)
- [Set-BrokerAssignmentPolicyRule](#)
- [Rename-BrokerAssignmentPolicyRule](#)
- [Remove-BrokerAssignmentPolicyRule](#)

## Parameters

**-Name**<String>

Specifies the administrative name of the new desktop rule. Each rule in the site's assignment policy must have a unique name (irrespective of whether they are desktop or application rules).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-DesktopGroupUid<Int32>**

Specifies the unique ID of the desktop group to which the new desktop rule applies.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-ColorDepth<ColorDepth>**

Specifies the color depth of any desktop sessions to machines assigned by the new rule.

Valid values are \$null, FourBit, EightBit, SixteenBit, and TwentyFourBit.

The default null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	null (dynamically inherited from the desktop group)
Accept Pipeline Input?	true (ByPropertyName)

#### **-Description<String>**

Specifies an optional description of the new desktop rule. The text may be visible to the end user, for example, as a tooltip associated with the desktop entitlement.

The default null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	null (dynamically inherited from the desktop group)
Accept Pipeline Input?	true (ByPropertyName)

#### **-Enabled<Boolean>**

Specifies whether the new desktop rule is initially enabled. A disabled rule is ignored when evaluating the site's assignment policy.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

#### **-ExcludedUserFilterEnabled<Boolean>**

Specifies whether the excluded users filter is initially enabled. If the filter is disabled then any user entries in the filter are ignored when

assignment policy rules are evaluated.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### **-ExcludedUsers<User[]>**

Specifies the excluded users filter of the new desktop rule, that is, the users and groups who are explicitly denied an entitlement to a machine assignment from the rule.

This can be used to exclude users or groups who would otherwise gain access by groups specified in the included users filter.

Required?	false
Default Value	(empty list)
Accept Pipeline Input?	true (ByPropertyName)

#### **-IconUid<Int32>**

Specifies the unique ID of the icon used to display the machine assignment entitlement to the user, and of the assigned desktop itself following the assignment.

The default null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	null (dynamically inherited from the desktop group)
Accept Pipeline Input?	true (ByPropertyName)

#### **-IncludedUserFilterEnabled<Boolean>**

Specifies whether the included users filter is initially enabled. If the filter is disabled then any user who satisfies the requirements of the access policy is implicitly granted an entitlement to a machine assignment by the new desktop rule.

Users who would be implicitly granted access when the filter is disabled can still be explicitly denied access using the excluded users filter.

For rules that relate to RemotePC desktop groups however, if the included user filter is disabled, the rule is effectively disabled.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

#### **-IncludedUsers<User[]>**

Specifies the included users filter of the new desktop rule, that is, the users and groups who are granted an entitlement to a machine assignment by the rule.

If a user appears explicitly in the excluded users filter of the rule or is a member of a group that appears in the excluded users filter, no entitlement is granted whether or not the user appears in the included users filter.

Required?	false
Default Value	(empty list)
Accept Pipeline Input?	true (ByPropertyName)

#### **-MaxDesktops**<Int32>

The number of machines from the rule's desktop group to which a user is entitled. Where an entitlement is granted to a user group rather than an individual, the number of machines applies to each member of the user group independently.

Required?	false
Default Value	1
Accept Pipeline Input?	true (ByPropertyName)

#### **-PublishedName**<String>

The name of the new machine assignment entitlement as seen by the user, and of the assigned desktop following its usage.

The default null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	null (dynamically inherited from the desktop group)
Accept Pipeline Input?	true (ByPropertyName)

#### **-SecureIcaRequired**<Boolean>

Specifies whether the new desktop rule requires the SecureICA protocol to be used for desktop sessions to machines assigned using the entitlement.

The default null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	null (dynamically inherited from the desktop group)
Accept Pipeline Input?	true (ByPropertyName)

#### **-UUID**<Guid>

An optional GUID for this rule.

Required?	false
Default Value	A new GUID is generated if none is supplied.
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.AssignmentPolicyRule

New-BrokerAssignmentPolicyRule returns the newly created desktop rule in the assignment policy rule.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> $dg = Get-BrokerDesktopGroup 'Sales Support'
C:\PS> New-BrokerAssignmentPolicyRule 'UK Office' -DesktopGroupUid $dg.Uid -IncludedUsers sales\uk-staff -PublishedName 'Sales Desktop'
```

Creates a desktop rule in the assignment policy that grants all members of the SALES\uk-staff group an entitlement to a single machine from the Sales Support desktop group. The entitlement name seen by users is Sales Desktop.

# New-BrokerCatalog

May 10, 2016  
Adds a new catalog to the site.

### Syntax

New-BrokerCatalog [-Name] <String> [-AllocationType] <AllocationType> [-CatalogKind] <CatalogKind> [-PvsForVM <String[]>] [-Description <String>] [-IsRemotePC <Boolean>] [-MachinesArePhysical <Boolean>] [-MinimumFunctionalLevel <FunctionalLevel>] [-PvsAddress <String>] [-PvsDomain <String>] [-RemotePCHypervisorConnectionUid <Int32>] [-Scope <String[]>] [-UUID <Guid>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

New-BrokerCatalog [-Name] <String> [-AllocationType] <AllocationType> [-ProvisioningType] <ProvisioningType> [-SessionSupport] <SessionSupport> [-PersistUserChanges] <PersistUserChanges> [-ProvisioningSchemeld <Guid>] [-Description <String>] [-IsRemotePC <Boolean>] [-MachinesArePhysical <Boolean>] [-MinimumFunctionalLevel <FunctionalLevel>] [-PvsAddress <String>] [-PvsDomain <String>] [-RemotePCHypervisorConnectionUid <Int32>] [-Scope <String[]>] [-UUID <Guid>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

### Detailed Description

New-BrokerCatalog adds a catalog through which machines can be provided to the site.

In order for a machine to register in a site, the machine must belong to a catalog with which it is compatible. The compatibility of a machine with a catalog is determined by two of the parameters of New-BrokerCatalog:

- o MinimalFunctionalLevel: The minimal functional level supported in the catalog. The functional level of the machine is determined by the capabilities of the Citrix VDA software on it.
- o SessionSupport: The session support (single/multi) of the catalog. The session support of the machine is determined by the variant of the Citrix VDA software installed (workstation/terminal services, respectively).

### Related topics

- [Get-BrokerCatalog](#)
- [Rename-BrokerCatalog](#)
- [Remove-BrokerCatalog](#)
- [Set-BrokerCatalog](#)

### Parameters

**-Name**<String>

Specifies a name for the catalog. Each catalog within a site must have a unique name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-AllocationType**<AllocationType>

Specifies how machines in the catalog are assigned to users. Values can be:

- o Static - Machines in a catalog of this type are permanently assigned to a user.
- o Permanent - equivalent to 'Static'.
- o Random - Machines in a catalog of this type are picked at random and temporarily assigned to a user.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-CatalogKind**<CatalogKind>

Deprecated: The type of machines the catalog will contain. Values can be: ThinCloned, SingleImage, PowerManaged, Unmanaged, Pvs, Pvd or PvsPvd.

Thin-Cloned, Single-Image and Personal vDisk Catalogs

-----

Thin-cloned and single-image catalog kinds are for machines created and managed with Provisioning Services for VMs. All machines in this type of catalog are managed, and so must be associated with a hypervisor connection.

A thin-cloned catalog is used for original golden VM images that are cloned when they are assigned to a VM, and users' changes to the VM image are retained after the VM is restarted.

A single-image catalog is used when multiple machines provisioned with Provisioning Services for VMs all share a single golden VM image when they run and, when restarted, they revert to the original VM image state.

A personal vDisk catalog is similar to a single-image catalog, but it also uses personal vDisk technology.

#### PowerManaged

This catalog kind is for managed machines that are manually provisioned by administrators. All machines in this type of catalog are managed, and so must be associated with a hypervisor connection.

#### Unmanaged

This catalog kind is for unmanaged machines, so there is no associated hypervisor connection.

#### PVS

This catalog kind is for managed machines that are provisioned using Provisioning Services. All machines in this type of catalog are managed, and so must be associated with a hypervisor connection. Only shared desktops are suitable for this catalog kind.

A Provisioning Services-personal vDisk (PvsPvd) catalog is similar to a Provisioning Services catalog, but it also uses personal vDisk technology.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### -ProvisioningType<ProvisioningType>

Specifies the ProvisioningType for the catalog. Values can be:

- o Manual - No provisioning.
- o PVS - Machine provisioned by PVS (machine may be physical, blade, VM,...).
- o MCS - Machine provisioned by MCS (machine must be VM).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### -SessionSupport<SessionSupport>

Specifies whether machines in the catalog are single or multi-session capable. Values can be:

- o SingleSession - Single-session only machine.
- o MultiSession - Multi-session capable machine.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### -PersistUserChanges<PersistUserChanges>

Specifies how user changes are persisted on machines in the catalog. Possible values are:

- o OnLocal: User changes are stored on the machine's local storage.
- o Discard: User changes are discarded.
- o OnPvd: User changes are stored on the user's personal vDisk.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### -PvsForVM<String[]>



Deprecated:

Identifies the provisioning scheme used by this catalog. To be specified in the format: ProvisioningSchemeGuid:ServiceGroupGuid. Applicable only to thin-cloned, single-image or personal vDisk catalogs.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Description**<String>

A description for the catalog.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-IsRemotePC**<Boolean>

Specifies whether this is to be a Remote PC catalog.

IsRemotePC can only be enabled when:

o SessionSupport is SingleSession

o MachinesArePhysical is true.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

**-MachinesArePhysical**<Boolean>

Specifies whether machines in the catalog can be power-managed by the Citrix Broker Service. Where the Citrix Broker Service cannot control the power state of themachine specify True, otherwise False. Can only be specified together with a provisioning type of Pvs or Manual, or if used with the legacy CatalogKind parameter only with Pvs or PvsPvd catalog kinds.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-MinimumFunctionalLevel**<FunctionalLevel>

The minimum FunctionalLevel required for machines to register in the site.

Valid values are L5, L7, L7\_6

Required?	false
Default Value	The FunctionalLevel of the current release (L7_6); by default no machines with less than the most current FunctionalLevel will be functional.
Accept Pipeline Input?	true (ByPropertyName)

**-PvsAddress**<String>

Specifies the URL of the Provisioning Services server. Only applicable to Provisioning Services or Provisioning Services-personal vDisk catalogs.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PvsDomain<String>**

Specifies the Active Directory domain of the Provisioning Services server. Only applicable to Provisioning Services or Provisioning Services-personal vDisk catalogs.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-RemotePCHypervisorConnectionUid<Int32>**

Specifies the hypervisor connection to use for powering on remote PCs in this catalog (only allowed when IsRemotePC is true).

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Scope<String[]>**

Specifies the name of the delegated administration scope to which the catalog belongs.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-UUID<Guid>**

An optional GUID for this catalog.

Required?	false
Default Value	A new GUID is generated if none is supplied.
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId<Guid>**

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### **-ProvisioningSchemeId<Guid>**

Specifies the identity of the MCS provisioning scheme the new catalog is associated with (can only be specified for new catalogs with a ProvisioningType of MCS).

Required?	false

Default Value	\$null
Accept Pipeline Input?	true (ByPropertyName)

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.Catalog

New-BrokerCatalog returns the created catalog.

#### Examples

##### ----- EXAMPLE 1 -----

C:\PS> New-BrokerCatalog -AllocationType Static -CatalogKind Unmanaged -Description "Catalog1 Description" -Name "Catalog1 Name"  
This command creates a catalog that can contain unmanaged physical or virtual machines that are permanently assigned to the user.

##### ----- EXAMPLE 2 -----

C:\PS> New-BrokerCatalog -AllocationType Random -CatalogKind PowerManaged -Description "catalog 2 Description" -Name "Catalog2 Name"  
This command creates a catalog that can contain power-managed machines that are randomly assigned to the user.

##### ----- EXAMPLE 3 -----

C:\PS> New-BrokerCatalog -AllocationType Random -CatalogKind PVS -Description "PVS Catalog Desc" -Name "PVS Catalog Name" -PvsAddress "pvsServer@pvsDomain.com" -PvsDomain "pvsDor"  
This command creates a catalog that can contain managed machines that are provisioned using Provisioning Services.

# New-BrokerConfigurationSlot

May 10, 2016

Creates a new configuration slot.

## Syntax

```
New-BrokerConfigurationSlot [-Name] <String> -SettingsGroup <String> [-Description <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Creates a new configuration slot. The SettingsGroup of the slot determines the particular collection of related settings that may be specified in a machine configuration associated with this slot.

For example, the configuration slot may be restricted to configuring Citrix User Profile Manager settings by specifying the SettingsGroup parameter as "G=UPM".

## Related topics

[Get-BrokerConfigurationSlot](#)

[Remove-BrokerConfigurationSlot](#)

[New-BrokerMachineConfiguration](#)

## Parameters

**-Name**<String>

Name of the new configuration slot. This must be alphanumeric and not contain white space.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-SettingsGroup**<String>

The settings group determines the particular collection of related settings that may be controlled by this slot. This must match the format of a Citrix Group Policy configuration group (e.g. "G=UPM"). Only settings that have this exact group may be specified in a machine configuration associated with this configuration slot.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-Description**<String>

Description of configuration slot.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.ConfigurationSlot

New-BrokerConfigurationSlot returns an object representing the newly created configuration slot

## Examples

### ----- EXAMPLE 1 -----

New-BrokerConfigurationSlot -Name "UPM" -SettingsGroup "G=UPM"

Create a new slot named "UPM" to configure settings specific to "User Profile Management"

# New-BrokerConfiguredFTA

May 10, 2016  
Creates a file type association with a published application.

## Syntax

New-BrokerConfiguredFTA -ImportedFTA <ImportedFTA> -ApplicationUid <Int32> [-UUID <Guid>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

New-BrokerConfiguredFTA -ExtensionName <String> -HandlerName <String> -ApplicationUid <Int32> [-ContentType <String>] [-HandlerDescription <String>] [-HandlerOpenArguments <String>] [-UUID <Guid>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Creates an association between a file type and a published application for the purposes of the content redirection.

File type association associates a file extension (such as ".txt") with an application (such as Notepad). In a Citrix environment file type associations on a user device can be configured so that when an user clicks on a document it launches the appropriate published application. This is known as "content redirection".

Configured file type associations are different from imported file type associations. Configured file type associations are those that are actually associated with published applications for the purposes of content redirection. Imported file type associations are lists of known file type associations for a given desktop group. See Update-BrokerImportedFTA for more information about imported file type associations.

This cmdlet has two parameter sets, which correspond to the cmdlet's two use cases.

The first use case leverages imported file type associations to configure file types for published applications. Information about the file type association is read from the imported object. See the Update-BrokerImportedFTA cmdlet for more information about importing file type associations from a worker machine.

The second use case is more complex and allows you to create your own file type association without having to import it first. This also lets you create custom file type associations that may not already exist on the worker machines. This use case is more error-prone, however, because the individual attributes of the file type association must be correctly specified by you.

## Related topics

- [Get-BrokerImportedFTA](#)
- [Get-BrokerConfiguredFTA](#)
- [Remove-BrokerConfiguredFTA](#)

## Parameters

**-ApplicationUid**<Int32>

Specifies the application with which the file type should be associated.

Required?	true
Default Value	(required)
Accept Pipeline Input?	true (ByPropertyName)

**-ImportedFTA**<ImportedFTA>

Specifies the ImportedFTA object to use for creating the ConfiguredFTA object. All values needed to create a ConfiguredFTA object are read from the ImportedFTA object.

Required?	true
Default Value	(required)
Accept Pipeline Input?	true (ByPropertyName)

**-ExtensionName**<String>

Specifies the extension name for the file type association. For example, ".txt" or ".doc".

Required?	true
Default Value	(required)
Accept Pipeline Input?	true (ByPropertyName)

**-HandlerName**<String>

Specifies the name of the handler for the file type association (as seen in the Registry). For example, "TXTFILE" or "Word.Document.8".

Required?	true
-----------	------

Default Value	(required)
Accept Pipeline Input?	true (ByPropertyName)

#### **-UUID<Guid>**

An optional GUID for this ConfiguredFTA.

Required?	false
Default Value	A new GUID is generated if none is supplied.
Accept Pipeline Input?	false

#### **-LoggingId<Guid>**

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### **-ContentType<String>**

Specifies the content type of the file type (as listed in the Registry). For example, content type would be "text/plain" or "application/msword".

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

#### **-HandlerDescription<String>**

Specifies the description of the handler for the file type association.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

#### **-HandlerOpenArguments<String>**

Specifies the arguments for the open command that the handler should use. For example, "%1".

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)



## Input Type

Variable, based on property name. This cmdlet does accept input from the pipeline but only by property name.

## Return Values

Citrix.Broker.Admin.SDK.ConfiguredFTA

This cmdlet returns a single ConfiguredFTA object.

## Examples

### EXAMPLE 1

```
C:\PS> $app = Get-BrokerApplication "Notepad"
```

```
C:\PS> $fta = Get-BrokerImportedFTA -ExtensionName ".txt"
```

```
C:\PS> New-BrokerConfiguredFTA -ImportedFTA $fta -ApplicationUid $app.Uid
```

Gets the Uid for the application, gets the ImportedFTA object for the file extension, and finally associates ".txt" with the published "Notepad" application.

Note that the Get-BrokerImportedFTA cmdlet may return more than one ImportedFTA objects for a specific extension name. See the help for that cmdlet for more details.

### EXAMPLE 2

```
C:\PS> $app = Get-BrokerApplication "Notepad"
```

```
C:\PS> New-BrokerConfiguredFTA -ApplicationUid $app.Uid -ExtensionName ".txt" -HandlerName "txtfile" -ContentType "text/plain" -HandlerDescription "Text Document" -HandlerOpenArguments "%1"
```

This example is identical to the first, but shows the the second use case of the cmdlet, specifying each attribute manually.

# New-BrokerDelayedHostingPowerAction

May 10, 2016

Causes a power action to be queued after a delay.

## Syntax

```
New-BrokerDelayedHostingPowerAction [-MachineName] <String> -Action <PowerManagementAction> -Delay <TimeSpan> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Causes a power action to be queued after the specified period of time.

Only ShutDown or Suspend actions can be requested to be delayed in this manner.

For a detailed description of the queuing mechanism, see 'help about\_Broker\_PowerManagement'.

## Related topics

[Get-BrokerDelayedHostingPowerAction](#)

[New-BrokerDelayedHostingPowerAction](#)

## Parameters

**-MachineName**<String>

Specifies the machine that the action is to be performed on.

The machine can be identified by DNS name, short name, SID, or name of the form domain\machine.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Action**<PowerManagementAction>

Specifies the power state change action that is to be performed on the specified machine after the specified delay.

Valid values are Shutdown and Suspend.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

--	--

#### **-Delay**<TimeSpan>

Specifies a timespan delay before the action is queued.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.DelayedHostingPowerAction

New-BrokerDelayedHostingPowerAction returns the created delayed power action.

## Examples

### ----- **EXAMPLE 1** -----

C:\PS> New-BrokerDelayedHostingPowerAction -Action Shutdown -MachineName 'XD\_VDA1' -Delay '00:02:00'  
Causes the machine called XD\_VDA1 to be shut down after a delay of two minutes.

# New-BrokerDesktopGroup

May 10, 2016

Create a new desktop group for managing the brokering of groups of desktops.

## Syntax

```
New-BrokerDesktopGroup [-Name] <String> -DesktopKind <DesktopKind> [-AutomaticPowerOnForAssigned <Boolean>] [-AutomaticPowerOnForAssignedDuringPeak <Boolean>] [-ColorDepth <ColorDepth>] [-DeliveryType <DeliveryType>] [-Description <String>] [-Enabled <Boolean>] [-IconUid <Int32>] [-InMaintenanceMode <Boolean>] [-IsRemotePC <Boolean>] [-MinimumFunctionalLevel <FunctionalLevel>] [-OffPeakBufferSizePercent <Int32>] [-OffPeakDisconnectAction <SessionChangeHostingAction>] [-OffPeakDisconnectTimeout <Int32>] [-OffPeakExtendedDisconnectAction <SessionChangeHostingAction>] [-OffPeakExtendedDisconnectTimeout <Int32>] [-OffPeakLogOffAction <SessionChangeHostingAction>] [-OffPeakLogOffTimeout <Int32>] [-PeakBufferSizePercent <Int32>] [-PeakDisconnectAction <SessionChangeHostingAction>] [-PeakDisconnectTimeout <Int32>] [-PeakExtendedDisconnectAction <SessionChangeHostingAction>] [-PeakExtendedDisconnectTimeout <Int32>] [-PeakLogOffAction <SessionChangeHostingAction>] [-PeakLogOffTimeout <Int32>] [-ProtocolPriority <String[]>] [-PublishedName <String>] [-Scope <String[]>] [-SecureIcaRequired <Boolean>] [-SessionSupport <SessionSupport>] [-SettlementPeriodBeforeAutoShutdown <TimeSpan>] [-ShutdownDesktopsAfterUse <Boolean>] [-TimeZone <String>] [-TurnOnAddedMachine <Boolean>] [-UUID <Guid>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The New-BrokerDesktopGroup cmdlet creates a new broker desktop group that can then be used to manage the brokering settings of all desktops within that desktop group. Once the desktop group has been created, you can create desktops in it by adding the appropriate broker machines to it using the Add-BrokerMachine or Add-BrokerMachinesToDesktopGroup cmdlets.

Desktop groups hold settings that apply to all desktops they contain.

For any automatic power management settings of a desktop group to take effect, the group's TimeZone property must be specified. Automatic power management operations include pool management (power time schemes), reboot schedules, session disconnect and logoff actions, and powering on assigned machines etc.

## Related topics

[Get-BrokerDesktopGroup](#)

[Set-BrokerDesktopGroup](#)

[Rename-BrokerDesktopGroup](#)

[Remove-BrokerDesktopGroup](#)

[Add-BrokerMachine](#)

[Add-BrokerMachinesToDesktopGroup](#)

[Get-BrokerSite](#)

## Parameters

#### **-Name<String>**

The name of the new broker desktop group.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-DesktopKind<DesktopKind>**

The kind of desktops this group will hold. Valid values are Private and Shared.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-AutomaticPowerOnForAssigned<Boolean>**

Specifies whether assigned desktops in the desktop group should be automatically started at the start of peak time periods. Only relevant for groups whose DesktopKind is Private.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

#### **-AutomaticPowerOnForAssignedDuringPeak<Boolean>**

Specifies whether assigned desktops in the desktop group should be automatically started throughout peak time periods. Only relevant for groups whose DesktopKind is Private and which have AutomaticPowerOnForAssigned set to true.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### **-ColorDepth**<ColorDepth>

Specifies the color depth that the ICA session should use for desktops in this group. Valid values are FourBit, EightBit, SixteenBit, and TwentyFourBit.

Required?	false
Default Value	TwentyFourBit
Accept Pipeline Input?	true (ByPropertyName)

#### **-DeliveryType**<DeliveryType>

Specifies whether desktops, applications, or both, can be delivered from machines contained within the new desktop group. Desktop groups with a DesktopKind of Private cannot be used to deliver both desktops and applications. Defaults to DesktopsOnly if not specified.

Valid values are DesktopsOnly, AppsOnly, and DesktopsAndApps.

Required?	false
Default Value	DesktopsOnly
Accept Pipeline Input?	true (ByPropertyName)

#### **-Description**<String>

A description for this desktop group useful for administrators of the site.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

#### **-Enabled**<Boolean>

Whether the desktop group should be in the enabled state; disabled desktop groups do not appear to users.

Required?	false
Default Value	true

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-IconUid**<Int32>

The UID of the broker icon to be displayed to users for their desktop(s) in this desktop group.

Required?	false
Default Value	The Uid of the default desktop icon in this site - use the Get-BrokerSite cmdlet to find this value.
Accept Pipeline Input?	true (ByPropertyName)

#### **-InMaintenanceMode**<Boolean>

Whether the desktop should be created in maintenance mode; a desktop group in maintenance mode will not allow users to connect or reconnect to their desktops.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### **-IsRemotePC**<Boolean>

Specifies whether this is to be a Remote PC desktop group.

IsRemotePC can only be enabled when:

- o SessionSupport is SingleSession
- o DeliveryType is DesktopsOnly
- o DesktopKind is Private

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### **-MinimumFunctionalLevel**<FunctionalLevel>



The minimum FunctionalLevel required for machines to work successfully in the desktop group.

Valid values are L5, L7, L7\_6

Required?	false
Default Value	The FunctionalLevel of the current release (L7_6); by default no machines with less than the most current FunctionalLevel will be functional.
Accept Pipeline Input?	true (ByPropertyName)

#### **-OffPeakBufferSizePercent**<Int32>

The percentage of machines in the desktop group that should be kept available in an idle state outside peak hours.

Required?	false
Default Value	0
Accept Pipeline Input?	true (ByPropertyName)

#### **-OffPeakDisconnectAction**<SessionChangeHostingAction>

The action to be performed after a configurable period of a user session disconnecting outside peak hours. Possible values are Nothing, Suspend, or Shutdown

Required?	false
Default Value	Nothing
Accept Pipeline Input?	true (ByPropertyName)

#### **-OffPeakDisconnectTimeout**<Int32>

The number of minutes before the configured action should be performed after a user session disconnects outside peak hours.

Required?	false
Default Value	0
Accept Pipeline Input?	true (ByPropertyName)

#### **-OffPeakExtendedDisconnectAction**<SessionChangeHostingAction>

The action to be performed after a second configurable period of a user session disconnecting outside peak hours. Possible values are Nothing, Suspend, or Shutdown.

Required?	false
Default Value	Nothing
Accept Pipeline Input?	true (ByPropertyName)

#### **-OffPeakExtendedDisconnectTimeout**<Int32>

The number of minutes before the second configured action should be performed after a user session disconnects outside peak hours.

Required?	false
Default Value	0
Accept Pipeline Input?	true (ByPropertyName)

#### **-OffPeakLogOffAction**<SessionChangeHostingAction>

The action to be performed after a configurable period of a user session ending outside peak hours. Possible values are Nothing, Suspend, or Shutdown.

Required?	false
Default Value	Nothing
Accept Pipeline Input?	true (ByPropertyName)

#### **-OffPeakLogOffTimeout**<Int32>

The number of minutes before the configured action should be performed after a user session ends outside peak hours.

Required?	false
Default Value	0
Accept Pipeline Input?	

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-PeakBufferSizePercent**<Int32>

The percentage of machines in the desktop group that should be kept available in an idle state in peak hours.

Required?	false
Default Value	0
Accept Pipeline Input?	true (ByPropertyName)

#### **-PeakDisconnectAction**<SessionChangeHostingAction>

The action to be performed after a configurable period of a user session disconnecting in peak hours. Possible values are Nothing, Suspend, or Shutdown.

Required?	false
Default Value	Nothing
Accept Pipeline Input?	true (ByPropertyName)

#### **-PeakDisconnectTimeout**<Int32>

The number of minutes before the configured action should be performed after a user session disconnects in peak hours.

Required?	false
Default Value	0
Accept Pipeline Input?	true (ByPropertyName)

#### **-PeakExtendedDisconnectAction**<SessionChangeHostingAction>

The action to be performed after a second configurable period of a user session disconnecting in peak hours. Possible values are Nothing, Suspend, or Shutdown.

Required?	false
Default Value	Nothing

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-PeakExtendedDisconnectTimeout<Int32>**

The number of minutes before the second configured action should be performed after a user session disconnects in peak hours.

Required?	false
Default Value	0
Accept Pipeline Input?	true (ByPropertyName)

#### **-PeakLogOffAction<SessionChangeHostingAction>**

The action to be performed after a configurable period of a user session ending in peak hours. Possible values are Nothing, Suspend, or Shutdown.

Required?	false
Default Value	Nothing
Accept Pipeline Input?	true (ByPropertyName)

#### **-PeakLogOffTimeout<Int32>**

The number of minutes before the configured action should be performed after a user session ends in peak hours.

Required?	false
Default Value	0
Accept Pipeline Input?	true (ByPropertyName)

#### **-ProtocolPriority<String[]>**

A list of protocol names in the order in which they should be attempted for use during connection.

Required?	false
Default Value	null

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-PublishedName**<String>

The name that will be displayed to users for their desktop(s) in this desktop group.

Required?	false
Default Value	The same value as that supplied for the name of the desktop group.
Accept Pipeline Input?	true (ByPropertyName)

#### **-Scope**<String[]>

Specifies the name of the delegated administration scope to which the desktop group should belong.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-SecureIcaRequired**<Boolean>

Whether HDX connections to desktops in the new desktop group require the use of a secure protocol.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### **-SessionSupport**<SessionSupport>

Specifies whether machines in the desktop group are single or multi-session capable. Values can be:

- o SingleSession - Single-session only machine.
- o MultiSession - Multi-session capable machine.

Required?	false
Default Value	

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-SettlementPeriodBeforeAutoShutdown**<TimeSpan>

Time after a session ends during which automatic shutdown requests (for example, shutdown after use, idle pool management) are deferred. Any outstanding shutdown request takes effect after the settlement period expires. This is typically used to configure time to allow logoff scripts to complete.

Required?	false
Default Value	0
Accept Pipeline Input?	true (ByPropertyName)

#### **-ShutdownDesktopsAfterUse**<Boolean>

Whether desktops in this desktop group should be automatically shut down when each user session completes (only relevant to power-managed desktops).

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### **-TimeZone**<String>

The time zone in which this desktop group's machines reside.

The time zone must be specified for any of the group's automatic power management settings to take effect. Automatic power management operations include pool management (power time schemes), reboot schedules, session disconnect and logoff actions, and powering on assigned machines etc.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

#### **-TurnOnAddedMachine**<Boolean>

This flag specifies whether the Broker Service should attempt to power on machines when they are added to the desktop

group.

Required?	false
Default Value	\$false for single session machines and \$true for multi-session machines.
Accept Pipeline Input?	true (ByPropertyName)

**-UUID**<Guid>

An optional GUID for this desktop group.

Required?	false
Default Value	A new GUID is generated if none is supplied.
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

Citrix.Broker.Admin.SDK.DesktopGroup

The newly created desktop group.

## Notes

Once a new desktop group is created, you can create desktops in it by adding the appropriate broker machines to it using the Add-BrokerMachine or Add-BrokerMachinesToDesktopGroup cmdlets.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> New-BrokerDesktopGroup "Assigned Desktops" -PublishedName "MyDesktop" -DesktopKind Private
```

Create a desktop group to manage the brokering of private desktops, which will appear to users with the name "MyDesktop".



# New-BrokerEntitlementPolicyRule

May 10, 2016  
Creates a new desktop rule in the site's entitlement policy.

## Syntax

New-BrokerEntitlementPolicyRule [-Name] <String> -DesktopGroupUid <Int32> [-ColorDepth <ColorDepth>] [-Description <String>] [-Enabled <Boolean>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-IconUid <Int32>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-PublishedName <String>] [-SecureIcaRequired <Boolean>] [-SessionReconnection <SessionReconnection>] [-UUID <Guid>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The New-BrokerEntitlementPolicyRule cmdlet adds a new desktop rule to the site's entitlement policy.

A desktop rule in the entitlement policy defines the users who are allowed per-session access to a machine from the rule's associated desktop group to run a full desktop session.

The following constraints apply when creating a desktop entitlement rule for a desktop group:

- o The group's desktop kind must be Shared
- o The group's delivery type must be DesktopsOnly or DesktopsAndApps

When a user selects a desktop entitlement published from a shared group, a machine is selected from the group on which to run the desktop session. No permanent association exists between the user and the selected machine; once the session ends the association also ends.

Multiple desktop rules in the entitlement policy can apply to the same desktop group. Where a user is granted an entitlement by more than one rule for the same group, they can use as many desktop sessions at the same time as they have entitlements.

## Related topics

- [Get-BrokerEntitlementPolicyRule](#)
- [Set-BrokerEntitlementPolicyRule](#)
- [Rename-BrokerEntitlementPolicyRule](#)
- [Remove-BrokerEntitlementPolicyRule](#)

## Parameters

**-Name**<String>

Specifies the administrative name of the new desktop rule. Each rule in the site's entitlement policy must have a unique name (irrespective of whether they are desktop or application rules).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-DesktopGroupUid**<Int32>

Specifies the unique ID of the desktop group to which the new desktop rule applies.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-ColorDepth**<ColorDepth>

Specifies the color depth of any desktop sessions launched by a user from this entitlement.

Valid values are \$null, FourBit, EightBit, SixteenBit, and TwentyFourBit.

The default null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	null (dynamically inherited from the desktop group)
Accept Pipeline Input?	true (ByPropertyName)

#### **-Description**<String>

Specifies an optional description of the new desktop rule. The text may be visible to the end user, for example, as a tooltip associated with the desktop entitlement.

The default null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	null (dynamically inherited from the desktop group)
Accept Pipeline Input?	true (ByPropertyName)

#### **-Enabled**<Boolean>

Specifies whether the new desktop rule is initially enabled. A disabled rule is ignored when evaluating the site's entitlement policy.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

#### **-ExcludedUserFilterEnabled**<Boolean>

Specifies whether the excluded users filter is initially enabled. If the filter is disabled then any user entries in the filter are ignored when entitlement policy rules are evaluated.

Required?	false
Default Value	false

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-ExcludedUsers<User[]>**

Specifies the excluded users filter of the desktop rule, that is, the users and groups who are explicitly denied an entitlement to a desktop session from the new rule.

This can be used to exclude users or groups who would otherwise gain access by groups specified in the included users filter.

Required?	false
Default Value	(empty list)
Accept Pipeline Input?	true (ByPropertyName)

#### **-IconUid<Int32>**

Specifies the unique ID of the icon used to display the desktop session entitlement to the user.

The default null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	null (dynamically inherited from the desktop group)
Accept Pipeline Input?	true (ByPropertyName)

#### **-IncludedUserFilterEnabled<Boolean>**

Specifies whether the included users filter is initially enabled. If the filter is disabled then any user who satisfies the requirements of the access policy is implicitly granted an entitlement to a desktop session by the new rule.

Users who would be implicitly granted access when the filter is disabled can still be explicitly denied access using the excluded users filter.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

#### **-IncludedUsers<User[]>**

Specifies the included users filter of the rule, that is, the users and groups who are granted an entitlement to a desktop session by the new rule.

If a user appears explicitly in the excluded users filter of the rule or is a member of a group that appears in the excluded users filter, no entitlement is granted whether or not the user appears in the included users filter.

Required?	false
Default Value	(empty list)
Accept Pipeline Input?	true (ByPropertyName)

#### **-PublishedName<String>**

The name of the new desktop session entitlement as seen by the user.

The default null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	null (dynamically inherited from the desktop group)
Accept Pipeline Input?	true (ByPropertyName)

#### **-SecureIcaRequired<Boolean>**

Specifies whether the new desktop rule requires the SecureICA protocol for desktop sessions launched using the entitlement.

The default null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	null (dynamically inherited from the desktop group)
Accept Pipeline Input?	true (ByPropertyName)

#### **-SessionReconnection<SessionReconnection>**

Defines reconnection (roaming) behavior for sessions launched using this rule. Session reconnection control is an experimental and unsupported feature.

Required?	false
Default Value	Always
Accept Pipeline Input?	true (ByPropertyName)

#### **-UUID<Guid>**

An optional GUID for this rule.

Required?	false
Default Value	A new GUID is generated if none is supplied.
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId<Guid>**

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation

and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.EntitlementPolicyRule

New-BrokerEntitlementPolicyRule returns the newly created desktop rule in the entitlement policy.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> $dg = Get-BrokerDesktopGroup 'Customer Support'
```

```
C:\PS> New-BrokerEntitlementPolicyRule 'UK Office' -DesktopGroupUid $dg.Uid -IncludedUsers support\uk-staff -PublishedName 'Support Desktop'
```

Creates an desktop rule in the entitlement policy that entitles all members of the SUPPORT\uk-staff group to a desktop session from the Customer Support desktop group. The desktop entitlement name seen by users is Support Desktop.

# New-BrokerHostingPowerAction

May 10, 2016

Creates a new action in the power action queue.

## Syntax

```
New-BrokerHostingPowerAction [-MachineName] <String> -Action <PowerManagementAction> [-ActualPriority <Int32>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The New-BrokerHostingPowerAction cmdlet adds a new power action record into the queue of power actions to be performed. The power actions in the queue are processed on a priority basis and sent to the relevant hypervisor to change the power state of a virtual machine.

A power action record defines the action to be performed, the machine on which the action is to be performed, and an initial priority value for the action. Multiple actions may be created that relate to the same machine.

For a detailed description of the queuing mechanism, see 'help about\_Broker\_PowerManagement'.

## Related topics

[Get-BrokerHostingPowerAction](#)

[Set-BrokerHostingPowerAction](#)

[Remove-BrokerHostingPowerAction](#)

## Parameters

**-MachineName**<String>

Specifies the machine that the action is to be performed on.

The machine can be identified by DNS name or short name or SID or 'machine\domain' form name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Action**<PowerManagementAction>

Specifies the power state change action that is to be performed on the specified machine.

Valid values are: TurnOn, TurnOff, ShutDown, Reset, Restart, Suspend and Resume.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-ActualPriority**<Int32>

Specifies an initial priority value for the action in the queue.

This priority is the current action priority; the 'base' priority for actions created via this cmdlet is always 30. Numerically lower priority values indicate more important actions that are processed in preference to actions with numerically higher priority settings.

Required?	false
Default Value	30
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

Citrix.Broker.Admin.SDK.HostingPowerAction

New-BrokerHostingPowerAction returns the newly created power action record.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> New-BrokerHostingPowerAction -Action Shutdown -MachineName 'XD_VDA1'
```

Causes the machine called 'XD\_VDA1' to be shut down.



# New-BrokerHypervisorConnection

May 10, 2016  
Creates a new hypervisor connection.

## Syntax

New-BrokerHypervisorConnection [-HypHypervisorConnectionUid <Guid>] [-PreferredController <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The New-BrokerHypervisorConnection cmdlet creates a new hypervisor connection.

## Related topics

- [Get-BrokerHypervisorConnection](#)
- [Remove-BrokerHypervisorConnection](#)
- [Set-BrokerHypervisorConnection](#)

## Parameters

**-HypHypervisorConnectionUid**<Guid>

The Guid that identifies the hypervisor connection, as defined in DUM.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-PreferredController**<String>

The preferred controller machine for the hypervisor connection. Can be specified as (first match is used):

- o Full SAM name.
- o Full DNS name.
- o SID value.
- o NetBIOS name (SAM without domain).
- o Partial DNS name (DNS name without some or all domain information).

Where not specified, the system selects preferred controller machine based on loading.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None

#### Return Values

Citrix.Broker.Admin.SDK.HypervisorConnection

New-BrokerHypervisorConnection returns an opaque object containing information about the hypervisor connection.

#### Examples

##### ----- EXAMPLE 1 -----

C:\PS> New-BrokerHypervisorConnection -PreferredController "domainName\controllerName" -HypHypervisorConnectionUid "d16f4e56-b85e-4ba6-b745-0e978ae4f192"  
This command creates a new hypervisor connection with a preferred controller.

##### ----- EXAMPLE 2 -----

C:\PS> New-BrokerHypervisorConnection -HypHypervisorConnectionUid "d16f4e56-b85e-4ba6-b745-0e978ae4f192"  
This command creates a new hypervisor connection, and leaves it to the system to select a preferred controller.

# New-BrokerIcon

May 10, 2016

Creates a new icon.

## Syntax

```
New-BrokerIcon [-EncodedIconData] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Accepts Base64 encoded .ICO format icon data, stores it in the database and returns an Icon object containing the Uid assigned to it.

New-BrokerIcon can be used with the Get-CtxIcon cmdlet from Citrix.Common.Commands, to obtain the Base64 icon. See Examples for a demonstration.

## Related topics

[Get-CtxIcon](#)

[Get-BrokerIcon](#)

[Remove-BrokerIcon](#)

## Parameters

**-EncodedIconData**<String>

Specifies the Base64 encoded .ICO format icon data.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

None Input cannot be piped to this cmdlet.

## Return Values

Citrix.Broker.Admin.SDK.Icon

Returns an Icon object.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Add-PSSnapin Citrix.Common.Commands
C:\PS> $ctxIcon = Get-CtxIcon -FileName C:\Windows\System32\notepad.exe -index 0
C:\PS> $brokerIcon = New-BrokerIcon -EncodedIconData $ctxIcon.EncodedIconData
C:\PS> $desktopGroup = Get-BrokerDesktopGroup -Name 'MyDesktopGroup'
C:\PS> Set-BrokerDesktopGroup $desktopGroup -IconUid $brokerIcon.Uid
```

Extracts the first icon resource from notepad.exe, and sets this as the icon for a desktop group.

# New-BrokerMachine

May 10, 2016  
Adds a machine that can be used to run desktops and applications.

## Syntax

New-BrokerMachine [-MachineName] <String> -CatalogUid <Int32> [-AssignedClientName <String>] [-AssignedIPAddress <String>] [-HostedMachineId <String>] [-HypervisorConnectionUid <Int32>] [-InMaintenanceMode <Boolean>] [-UUID <Guid>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

By adding a machine to a catalog, New-BrokerMachine adds a machine to the site, and is the first step in making the machine available to run users' desktops and applications. The machine may be physical or virtual.

For physical machines, you must specify the machine's SID and the catalog to which it will belong. For virtual machines which are not provisioned by MCS, you must also provide the hypervisor connection responsible for running the machine and the hosted machine ID by which the hypervisor recognizes the machine.

The machine must support the expected capabilities of the catalog: the catalog specifies a SessionType and a MinimalFunctionalLevel. The session support of the machine is determined by the type of Citrix VDA software installed (server or workstation) and the functional level depends on the version of the Citrix VDA software installed. The New-BrokerMachine command will complete successfully if these are not correct but the machine will be unable to register.

For more information about machines, see about\_Broker\_Machines.

## Related topics

[Add-BrokerMachine](#)

## Parameters

**-MachineName**<String>

Specify the name of the machine to create (in the form 'domain\machine'). A SID can also be specified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-CatalogUid**<Int32>

The catalog to which this machine will belong.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-AssignedClientName**<String>

The client name to which this machine will be assigned. Machines can be assigned to multiple users, a single client IP address, or a single client name, but only to one of these categories at a time.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-AssignedIPAddress**<String>

The client IP address to which this machine will be assigned. Machines can be assigned to multiple users, a single client IP address, or a single client name, but only to one of these categories at a time.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-HostedMachineId**<String>

The unique ID by which the hypervisor recognizes the machine. Omit this for physical machines or MCS-provisioned VMs.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

**-HypervisorConnectionUid**<Int32>

The hypervisor connection that runs the machine. Omit this for physical machines or MCS-provisioned VMs.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

**-InMaintenanceMode**<Boolean>

Specifies whether the machine is initially in maintenance mode. A machine in maintenance mode is not available for new sessions, and for managed machines all automatic power management is disabled.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

**-UUID**<Guid>

An optional GUID for this machine.

Required?	false
Default Value	A new GUID is generated if none is supplied.
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.Machine

New-BrokerMachine returns the created machine.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> New-BrokerMachine -CatalogUid 2 -MachineName 'domain\machine'
```

This adds the physical machine with the specified SAM name to this site and places it in the specified catalog.

##### ----- EXAMPLE 2 -----

```
C:\PS> New-BrokerMachine -CatalogUid 2 -MachineName 'S-1-5-12-1234567890-1234567890-1234567890-1234'
```

This adds the physical machine with the specified SID to this site and places it in the specified catalog.

##### ----- EXAMPLE 3 -----

```
C:\PS> New-BrokerMachine -CatalogUid 2 -MachineName 'domain\machine' -HostedMachineId 'F8143B4F-7371-4efa-868A-54787EF9F64E' -HypervisorConnectionUid 5
```

This adds the virtual machine, running on the specified hypervisor, to this site and places it in the catalog.

##### ----- EXAMPLE 4 -----

```
C:\PS> $m = New-BrokerMachine -CatalogUid 2 -MachineName 'domain\machine'
```

```
C:\PS> Add-BrokerMachine -InputObject $m -DesktopGroup 3
```

This adds the specified physical machine to the site and uses Add-BrokerMachine to add it to a desktop group.

# New-BrokerMachineCommand

May 10, 2016  
Creates a new command to deliver to a desktop.

## Syntax

```
New-BrokerMachineCommand -User <String> -Category <String> -CommandName <String> [-DesktopGroups <DesktopGroup[]>] [-SendTrigger <MachineCommandTrigger>] [-SendDeadline <TimeSpan>] [-CommandData <Byte[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

New-BrokerMachineCommand -SessionUid <Int64> -Category <String> -CommandName <String> [-SendTrigger <MachineCommandTrigger>] [-SendDeadline <TimeSpan>] [-CommandData <Byte[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

New-BrokerMachineCommand -MachineUid <Int32> -Category <String> -CommandName <String> [-SendTrigger <MachineCommandTrigger>] [-SendDeadline <TimeSpan>] [-CommandData <Byte[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

New-BrokerMachineCommand -Synchronous -MachineUid <Int32> -Category <String> -CommandName <String> [-CommandData <Byte[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Create a new command queued for delivery to a desktop. Commands are sent to a specific handler installed on the desktop using the Category parameter. Each handler has its own list of commands identified by the CommandName parameter. Optional command data can be provided using the CommandData parameter in a format specified by the handler.

Commands are targeted at a specific user, session or machine. Commands targeted at a user can be further be restricted to one or more desktop groups.

The SendTrigger is used to restrict the command to a specific event related to the target. For example, when the target machine registers or when the target user reconnects to a session. The command will be sent to the machine when the SendTrigger occurs for the target.

If the Synchronous switch is provided, the target must be a machine and no SendTrigger can be specified. The command is sent immediately to the machine if it is currently registered and fails if the machine is not registered.

Note that the combined length of the Category and CommandName is limited to 64 characters. The Category and CommandName must both be entirely alphanumeric and not include any white space.

## Related topics

- [Get-BrokerMachineCommand](#)
- [Remove-BrokerMachineCommand](#)

## Parameters

**-Category**<String>  
The service on the desktop to send the command to.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)



--	--

#### **-CommandName**<String>

The name of the command to send (as defined by the service).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-User**<String>

User whose desktop or session should receive the command.

Required?	true
Default Value	Any user.
Accept Pipeline Input?	true (ByPropertyName)

#### **-SessionUid**<Int64>

Currently logged on user session that should receive the command.

Required?	true
Default Value	Any session.
Accept Pipeline Input?	true (ByPropertyName)

#### **-MachineUid**<Int32>

Specific machine that should receive the command.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Synchronous**<SwitchParameter>

Send the command immediately and block while waiting for the reply.

Required?	true
-----------	------

Default Value	false
Accept Pipeline Input?	false

#### **-CommandData**<Byte[]>

Optional additional data to include with the command.

Required?	false
Default Value	None
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### **-DesktopGroups**<DesktopGroup[]>

Further restrict the command targeted at a user to machines in these desktop groups.

Required?	false
Default Value	No restriction by desktop group.
Accept Pipeline Input?	true (ByPropertyName)

### **-SendTrigger**<MachineCommandTrigger>

Queue command for delivery until this particular event occurs. Valid values are NextContact, Broker, LogOn, Logoff, Disconnect and Reconnect.

Required?	false
Default Value	Default value is 'NextContact' so the command is sent during the next communication with the desktop.
Accept Pipeline Input?	true (ByPropertyName)

### **-SendDeadline**<TimeSpan>

Automatically cancel the command if it not delivered before the specified time span passes.

Required?	false
Default Value	Command expires after 24 hours.
Accept Pipeline Input?	true (ByPropertyName)

### Input Type

None No parameter is accepted from the input pipeline.

### Return Values

Citrix.Broker.Admin.SDK.MachineCommand

New command that was added to the command queue.Citrix.Broker.Admin.SDK.MachineSynchronousCommandResponse

When the Synchronous option is used, the command is immediately sent to the specified machine and processed. The SDK object returned describes the command and the result of this command processing.

### Notes

Commands are subject to delegated administration restrictions based on the desktop group, category and command name.

### Examples

#### ----- **EXAMPLE 1** -----

New-BrokerMachineCommand -Category "UPM" -CommandName "ResetProfile" -DesktopGroups 1 -UserId 23 -SendTrigger Authentication  
Instruct the User Profile Manager service to execute the "ResetProfile" command when user 23 logs on to any machine in desktop group 1

#### ----- **EXAMPLE 2** -----

New-BrokerMachineCommand -Synchronous -Category "MonitorService" -CommandName "EnableLogging" -MachineUid 13  
Instruct the monitor service to immediately execute the "EnableLogging" command on the machine having Uid 13.

# New-BrokerMachineConfiguration

May 10, 2016  
Creates a new machine configuration associated with an existing configuration slot.

## Syntax

New-BrokerMachineConfiguration -ConfigurationSlotUid <Int32> -LeafName <String> -Policy <Byte[]> [-Description <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [-CommonParameters<String>]

## Detailed Description

Creates a new machine configuration containing settings that match the SettingsGroup of the associated configuration slot. This machine configuration can then be applied to a desktop group to have the settings applied to machines in that group.

The SettingsGroup of the configuration slot restricts the permitted settings. Use the SDK snap-in that matches the SettingsGroup to create the encoded settings data.

## Related topics

[Get-BrokerMachineConfiguration](#)

[Set-BrokerMachineConfiguration](#)

[Rename-BrokerMachineConfiguration](#)

[Remove-BrokerMachineConfiguration](#)

[Add-BrokerMachineConfiguration](#)

## Parameters

**-ConfigurationSlotUid**<Int32>

Unique identifier of the configuration slot to associate with this machine configuration.

Required?	true
Default Value	None
Accept Pipeline Input?	true (ByPropertyName)

**-LeafName**<String>

Name of the new machine configuration. This must be unique amongst the machine configurations associated with the same configuration slot.

Required?	true
Default Value	None
Accept Pipeline Input?	true (ByPropertyName)

**-Policy**<Byte[]>

A binary array of encoded settings (policy) data created with the SDK snap-in that matches the SettingsGroup of the configuration slot.

Required?	true
Default Value	None
Accept Pipeline Input?	true (ByPropertyName)

**-Description**<String>

Description of the new machine configuration.

--	--

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.MachineConfiguration

New-BrokerMachineConfiguration returns the newly created configuration

#### Notes

Delegated Administration can be used to restrict the configuration slots that an administrator can use and hence which components of the system that can be configured.

#### Examples

##### ----- EXAMPLE 1 -----

New-BrokerMachineConfiguration -LeafName "Finance Department" -Description "Finance Dept. User Profile Management policy" -Policy \$policy -ConfigurationSlotUid \$csUid  
Creates a new configuration named "%SlotName%\Finance Department" where %SlotName% is the name of the configuration slot having the Uid \$csUid. The encoded settings in the \$policy variable must match the SettingsGroup of the configuration slot having the Uid \$csUid.

# New-BrokerPowerTimeScheme

May 10, 2016  
Creates a new power time scheme for a desktop group.

## Syntax

New-BrokerPowerTimeScheme [-Name] <String> -DaysOfWeek <TimeSchemeDays> -DesktopGroupUid <Int32> [-DisplayName <String>] [-PeakHours <Boolean[]>] [-PoolSize <Int32[]>] [-PoolUsingPercentage <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The New-BrokerPowerTimeScheme cmdlet adds a new power time scheme to be associated with a desktop group. The power time scheme must relate to days of the week that are not already covered by an existing power time scheme.

Each power time scheme is associated with a particular desktop group, and covers one or more days of the week, defining which hours of those days are considered peak times and which are off-peak times. In addition, the time scheme defines a pool size value for each hour of the day for the days of the week covered by the time scheme. No one desktop group can be associated with two or more time schemes that cover the same day of the week.

See 'help about\_Broker\_PowerManagement' for a detailed description of the power policy mechanism and pool size management.

## Related topics

[Get-BrokerPowerTimeScheme](#)

[Set-BrokerPowerTimeScheme](#)

[Remove-BrokerPowerTimeScheme](#)

[Rename-BrokerPowerTimeScheme](#)

## Parameters

**-Name**<String>

Specifies the administrative name of the new power time scheme. Each scheme must have a name which is unique within the site.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-DaysOfWeek**<TimeSchemeDays>

Specifies the pattern of days of the week that the power time scheme covers.

Valid values are (singly or a list of) Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday, Weekdays and Weekend.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-DesktopGroupUid**<Int32>

Specifies the desktop group that the power time scheme applies to.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-DisplayName<String>**

Specifies the name of the new power time scheme as displayed in the DesktopStudio console. Each scheme associated with a desktop group must have a display name which is unique within its desktop group, although the same display name can be used on power schemes for different desktop groups.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-PeakHours<Boolean[]>**

A set of 24 boolean flag values, one for each hour of the day. The first value in the array relates to midnight to 00:59, the next one to 1 AM to 01:59 and so on, with the last array element relating to 11 PM to 11:59. If the flag is \$true it means that the associated hour of the day is considered a peak time; if \$false it means that it is considered off-peak.

Required?	false
Default Value	24 \$false values, meaning all hours are off-peak
Accept Pipeline Input?	true (ByPropertyName)

**-PoolSize<Int32[]>**

A set of 24 integer values, one for each hour of the day. The first value in the array relates to midnight to 00:59, the next one to 1 AM to 01:59 and so on, with the last array element relating to 11 PM to 11:59. The value defines the number of machines (either as an absolute number or a percentage of the machines in the desktop group) that are to be maintained in a running state, whether they are in use or not. A value of -1 has special meaning: pool size management does not apply during such hours.

Required?	false
Default Value	24 values of '-1', meaning no pool size management is to be performed
Accept Pipeline Input?	true (ByPropertyName)

**-PoolUsingPercentage<Boolean>**

A boolean flag to indicate whether the integer values in the pool size array are to be treated as absolute values (if this value is \$false) or as percentages of the number of machines in the desktop group (if this value is \$true).

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId<Guid>**

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

None You cannot pipe input into this cmdlet.

Return Values

Citrix.Broker.Admin.SDK.PowerTimeScheme

New-BrokerPowerTimeScheme returns the newly created power time scheme.

Examples

----- **EXAMPLE 1** -----

C:\PS> New-BrokerPowerTimeScheme -Name 'First Half Week' -DaysOfWeek Weekend,Monday,Tuesday -DesktopGroupUid 3 -PeakHours (0..23 | %{ \$\_ -gt 8 -and \$\_ -lt 18 } )  
Creates a new scheme attached to the desktop group whose UID value is 3. This new scheme covers the weekend and Monday and Tuesday, and defines 'peak' hours as 9am to 17:59, with all other times being 'off-peak'. No pool size values are supplied, so all size values for all the hours default to -1.



# New-BrokerRebootSchedule

May 10, 2016  
Creates a new reboot schedule for a desktop group.

### Syntax

New-BrokerRebootSchedule [-DesktopGroupName <String> -RebootDuration <Int32> [-Day <RebootScheduleDays>] [-Enabled <Boolean>] [-Frequency <RebootScheduleFrequency>] [-StartTime <TimeSpan>] [-WarningDuration <Int32>] [-WarningMessage <String>] [-WarningTitle <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

New-BrokerRebootSchedule -DesktopGroupUid <Int32> -RebootDuration <Int32> [-Day <RebootScheduleDays>] [-Enabled <Boolean>] [-Frequency <RebootScheduleFrequency>] [-StartTime <TimeSpan>] [-WarningDuration <Int32>] [-WarningMessage <String>] [-WarningTitle <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

### Detailed Description

The New-BrokerRebootSchedule cmdlet is used to define a reboot schedule for a desktop group.

### Related topics

- [Get-BrokerRebootSchedule](#)
- [Set-BrokerRebootSchedule](#)
- [Remove-BrokerRebootSchedule](#)
- [Start-BrokerRebootCycle](#)

### Parameters

**-DesktopGroupName**<String>

The name of the desktop group that this reboot schedule is applied to.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-RebootDuration**<Int32>

Approximate maximum number of minutes over which the scheduled reboot cycle runs.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-DesktopGroupUid**<Int32>

The Uid of the desktop group that this reboot schedule is applied to.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Day**<RebootScheduleDays>

For weekly schedules, the day of the week on which the scheduled reboot-cycle starts (one of Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday).

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Enabled**<Boolean>

Boolean that indicates if the new reboot schedule is enabled.

--	--

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### -Frequency<RebootScheduleFrequency>

Frequency with which this schedule runs (either Weekly or Daily).

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### -StartTime<TimeSpan>

Time of day at which the scheduled reboot cycle starts (HH:MM).

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### -WarningDuration<Int32>

Time prior to the initiation of a machine reboot at which warning message is displayed in all user sessions on that machine. If the warning duration is zero then no message is displayed.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### -WarningMessage<String>

Warning message displayed in user sessions on a machine scheduled for reboot. If the message is blank then no message is displayed.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### -WarningTitle<String>

The window title used when showing the warning message in user sessions on a machine scheduled for reboot.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### -LoggingId<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

None Input cannot be piped to this cmdlet.

Return Values

Citrix.Broker.Admin.SDK.RebootSchedule

Examples

----- **EXAMPLE 1** -----

C:\PS> New-BrokerRebootSchedule -DesktopGroupName BankTellers -Frequency Daily -StartTime "02:00" -Enabled \$true -Duration 120  
Schedules the machines in the desktop group named 'BankTellers' to be rebooted every night between 2 AM and 4 AM.

----- **EXAMPLE 2** -----

C:\PS> New-BrokerRebootSchedule -DesktopGroupUid 17 -Frequency Weekly -Day Saturday -StartTime "01:00" -Enabled \$true -Duration 240 -WarningTitle "WARNING: Reboot pending" -WarningMes  
Schedules the machines in the desktop group having Uid 17 to be rebooted every Saturday night between 1 AM and 5 AM. Ten minutes prior to rebooting, each machine will display a message box with the title "WARNING: Reboot pending" and message "Save your work" in every user session.

# New-BrokerRemotePCAccount

May 10, 2016

Create a new RemotePCAccount.

## Syntax

```
New-BrokerRemotePCAccount -CatalogUid <Int32> -OU <String> [-AllowSubfolderMatches <Boolean>] [-MachinesExcluded <String[]>] [-MachinesIncluded <String[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Create a new RemotePCAccount. A RemotePCAccount defines machine filters to support Remote PC automation adding unconfigured machines to catalogs.

## Related topics

[Get-BrokerRemotePCAccount](#)

[Set-BrokerRemotePCAccount](#)

[Remove-BrokerRemotePCAccount](#)

## Parameters

**-CatalogUid**<Int32>

Specifies the catalog which Remote PC automation adds an unconfigured machine to if it matches this RemotePCAccount.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-OU**<String>

Specifies the DN of an AD container, or has the special value 'any'.

When an AD container is specified a machine may only match with the RemotePCAccount when the AD computer object is located relative to the OU.

When 'any' is specified the location of the AD computer object is ignored for purposes of matching this RemotePCAccount. The machine must still meet the MachinesIncluded and MachinesExcluded filters for a match to occur.

In the event that a machine matches with multiple RemotePCAccounts then the RemotePCAccount OU with the longest canonical name takes precedence. The special 'any' OU is treated as lowest priority.

Note that the OU value of every RemotePCAccount must be unique, and this includes only one 'any' entry being permitted.

Required?	true
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

**-AllowSubfolderMatches**<Boolean>

When true a machine matches this RemotePCAccount if the AD computer object exists within the container specified by the OU property, or within a child container of the OU.

When false the AD computer object only matches if it exists directly in the AD container specified by the OU property.

This property is not meaningful when OU has the special value 'any'.

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### **-MachinesExcluded**<String[]>

MachinesExcluded specifies a set of strings that can include asterisk wildcards. If a machine name matches any entries in MachinesExcluded then it cannot match with this RemotePCAccount regardless of whether there is a MachinesIncluded match.

Matches are performed against the domain name joined with the machine name by a backslash (DOMAIN\MACHINE), e.g.:

DOMAIN1\M\*

DOMAIN\*\M\*

\*\M\*

Required?	false
Default Value	@()
Accept Pipeline Input?	true (ByPropertyName)

#### **-MachinesIncluded**<String[]>

MachinesIncluded specifies a set of strings that can include asterisk wildcards. A machine may only match with this RemotePCAccount if it matches a MachinesIncluded entry and does not match any MachinesExcluded entries.

Matches are performed against the domain name joined with the machine name by a backslash (DOMAIN\MACHINE), e.g.:

DOMAIN1\M\*

DOMAIN\*\M\*

\*\M\*

Required?	false
Default Value	@('*')
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and

Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

**Input Type**

None You cannot pipe input into this cmdlet.

**Return Values**

Citrix.Broker.Admin.SDK.RemotePCAccount

The newly created RemotePCAccount.

**Examples**

----- **EXAMPLE 1** -----

```
C:\PS> New-BrokerRemotePCAccount -OU 'ou=MyOU,dc=MyDomain,dc=com' -CatalogUid 42
```

Create a RemotePCAccount that adds unconfigured machines with computer objects in MyOU, into catalog 42.

----- **EXAMPLE 2** -----

```
C:\PS> New-BrokerRemotePCAccount -OU 'any' -CatalogUid 42 -MachinesIncluded @('DOMAIN1*') -MachinesExcluded @('DOMAIN1*JOHNDOE*')
```

Create a RemotePCAccount matching unconfigured machines from DOMAIN1, except those with hostnames containing JOHNDOE, and add them to catalog 42.

----- **EXAMPLE 3** -----

```
C:\PS> New-BrokerRemotePCAccount -OU 'any' -CatalogUid 42
```

Create a RemotePCAccount that matches any unconfigured machine, causing automation to add matching machines to catalog 42.

# New-BrokerSessionLinger

May 10, 2016

Creates a new session linger setting for a desktop group.

## Syntax

**New-BrokerSessionLinger** [-DesktopGroupName] <String> [-Enabled <Boolean>] [-MaxAverageLoadThreshold <Int32>] [-MaxLoadPerMachineThreshold <Int32>] [-MaxTimeBeforeDisconnect <TimeSpan>] [-MaxTimeBeforeTerminate <TimeSpan>] [-UserFilterEnabled <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

**New-BrokerSessionLinger** -DesktopGroupUid <Int32> [-Enabled <Boolean>] [-MaxAverageLoadThreshold <Int32>] [-MaxLoadPerMachineThreshold <Int32>] [-MaxTimeBeforeDisconnect <TimeSpan>] [-MaxTimeBeforeTerminate <TimeSpan>] [-UserFilterEnabled <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The New-BrokerSessionLinger cmdlet is used to define a session linger setting for a desktop group.

Note that each desktop group can only have a single session linger setting. Session lingering only applies to application sessions.

## Related topics

[Get-BrokerSessionLinger](#)

[Set-BrokerSessionLinger](#)

[Remove-BrokerSessionLinger](#)

## Parameters

**-DesktopGroupName**<String>

The name of the desktop group that this linger setting is applied to.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-DesktopGroupUid**<Int32>

The Uid of the desktop group that this linger setting is applied to.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Enabled**<Boolean>

Boolean that indicates if the new session linger is enabled.

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

#### **-MaxAverageLoadThreshold<Int32>**

Specifies the average load threshold across the desktop group. When the threshold hits, lingering sessions across the group be terminated to reduce load. Sessions that have been lingering the longest will be chosen first.

Required?	false
Default Value	0
Accept Pipeline Input?	true (ByPropertyName)

#### **-MaxLoadPerMachineThreshold<Int32>**

Specifies the maximum load threshold per machine in the desktop group. When the threshold hits, lingering sessions on each loaded machine will be terminated to reduce load. Sessions that have been lingering the longest will be chosen first.

Required?	false
Default Value	0
Accept Pipeline Input?	true (ByPropertyName)

#### **-MaxTimeBeforeDisconnect<TimeSpan>**

Specifies the time by which a lingering session will be disconnected. The disconnect timer is optional, but when specified the terminate timer needs to be also set. The disconnect time cannot be greater than the terminate time. When the disconnect and terminate times are the same, the terminate timer takes precedence. The disconnect timer needs to be paired with a session termination condition like the terminate timer or one of load threshold settings.

Required?	false
Default Value	15 minutes
Accept Pipeline Input?	true (ByPropertyName)

#### **-MaxTimeBeforeTerminate<TimeSpan>**

Specifies the time by which a lingering session will be terminated. The terminate timer is not optional when timers are configured. When the disconnect and terminate times are the same, the terminate timer takes precedence.

Required?	false
-----------	-------



Default Value	8 hours
Accept Pipeline Input?	true (ByPropertyName)

#### **-UserFilterEnabled**<Boolean>

Specifies whether the session linger's user filter is enabled or disabled. Where the user filter is enabled, lingering is enabled only to users who appear in the filter (either explicitly or by virtue of group membership).

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Depends on parameter Parameters can be piped by property name.

#### Return Values

Citrix.Broker.Admin.SDK.SessionLinger

New-BrokerSessionLinger returns a session linger object.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> New-BrokerSessionLinger -DesktopGroupName test -Enabled $true -MaxTimeBeforeDisconnect 0:30 -MaxTimeBeforeTerminate 1:00
```

Creates a new session linger setting with a disconnect timer of 30 minutes and terminate timer of 1 hour.

# New-BrokerSessionPreLaunch

May 10, 2016  
Creates a new session pre-launch setting for a desktop group.

## Syntax

New-BrokerSessionPreLaunch [-DesktopGroupName] <String> [-Enabled <Boolean>] [-MaxAverageLoadThreshold <Int32>] [-MaxLoadPerMachineThreshold <Int32>] [-MaxTimeBeforeDisconnect <TimeSpan>] [-MaxTimeBeforeTerminate <TimeSpan>] [-UserFilterEnabled <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

New-BrokerSessionPreLaunch -DesktopGroupUid <Int32> [-Enabled <Boolean>] [-MaxAverageLoadThreshold <Int32>] [-MaxLoadPerMachineThreshold <Int32>] [-MaxTimeBeforeDisconnect <TimeSpan>] [-MaxTimeBeforeTerminate <TimeSpan>] [-UserFilterEnabled <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The New-BrokerSessionPreLaunch cmdlet is used to define a session pre-launch setting for a desktop group.

Note that each desktop group can only have a single session pre-launch setting. Session pre-launch only applies to application sessions.

## Related topics

- [Get-BrokerSessionPreLaunch](#)
- [Set-BrokerSessionPreLaunch](#)
- [Remove-BrokerSessionPreLaunch](#)

## Parameters

**-DesktopGroupName**<String>

The name of the desktop group that this pre-launch setting is applied to.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-DesktopGroupUid**<Int32>

The Uid of the desktop group that this pre-launch setting is applied to.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Enabled**<Boolean>

Boolean that indicates if the new session pre-launch is enabled.

--	--

Required?	false
Default Value	true
Accept Pipeline Input?	true (ByPropertyName)

#### **-MaxAverageLoadThreshold**<Int32>

Specifies the average load threshold across the desktop group. When the threshold hits, pre-launched sessions across the group be terminated to reduce load. Sessions that have been pre-launched the longest will be chosen first.

Required?	false
Default Value	0
Accept Pipeline Input?	true (ByPropertyName)

#### **-MaxLoadPerMachineThreshold**<Int32>

Specifies the maximum load threshold per machine in the desktop group. When the threshold hits, pre-launched sessions on each loaded machine will be terminated to reduce load. Sessions that have been pre-launched the longest will be chosen first.

Required?	false
Default Value	0
Accept Pipeline Input?	true (ByPropertyName)

#### **-MaxTimeBeforeDisconnect**<TimeSpan>

Specifies the time by which a pre-launched session will be disconnected. The disconnect timer is optional, but when specified the terminate timer needs to be also set. The disconnect time cannot be greater than the terminate time. When the disconnect and terminate times are the same, the terminate timer takes precedence. The disconnect timer needs to be paired with a session termination condition like the terminate timer or one of load threshold settings.

Required?	false
Default Value	15 minutes
Accept Pipeline Input?	true (ByPropertyName)

#### **-MaxTimeBeforeTerminate**<TimeSpan>

Specifies the time by which a pre-launched session will be terminated. The terminate timer is not optional when timers are configured. When the disconnect and terminate times are the same, the terminate timer takes precedence.

Required?	false
Default Value	8 hours

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-UserFilterEnabled**<Boolean>

Specifies whether the session pre-launch's user filter is enabled or disabled. Where the user filter is enabled, pre-launch is enabled only to users who appear in the filter (either explicitly or by virtue of group membership).

Required?	false
Default Value	false
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Depends on parameter Parameters can be piped by property name.

#### Return Values

Citrix.Broker.Admin.SDK.SessionPreLaunch

New-BrokerSessionPreLaunch returns a session pre-launch object.

#### Examples

----- **EXAMPLE 1** -----

C:\PS> New-BrokerSessionPreLaunch -DesktopGroupName test -Enabled \$true -MaxTimeBeforeDisconnect 0:30 -MaxTimeBeforeTerminate 1:00  
Creates a new session pre-launch setting with a disconnect timer of 30 minutes and terminate timer of 1 hour.



# New-BrokerTag

May 10, 2016

Creates a new tag.

## Syntax

New-BrokerTag [-Name] <String> [-UUID <Guid>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Creates a tag that can be associated with other objects using Add-BrokerTag.

## Related topics

[Add-BrokerTag](#)

[Get-BrokerTag](#)

[Remove-BrokerTag](#)

[Rename-BrokerTag](#)

## Parameters

**-Name**<String>

Specifies a name for the new tag.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-UUID**<Guid>

Specifies a UUID for the new tag. When not specified, a UUID is automatically assigned.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None Input cannot be piped to this cmdlet.

#### Return Values

Citrix.Broker.Admin.SDK.Tag

Outputs the generated tag.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> New-BrokerTag -Name 'Tag1'
```

Creates a new tag with name 'Tag1'.

##### ----- EXAMPLE 2 -----

```
C:\PS> New-BrokerTag 'Tag2' | Add-BrokerTag -Desktop $desktop
```

Creates a new tag with name 'Tag2' and associates it with Desktop \$desktop.



# New-BrokerUser

May 10, 2016

Creates a new broker user object

## Syntax

New-BrokerUser [-SID] <SecurityIdentifier> [-AdminAddress <String>] [<CommonParameters>]

New-BrokerUser [-Name] <String> [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The New-BrokerUser cmdlet creates a new broker object to represent a user identity (or the identity of a group of users). The object is created local to the PowerShell environment in which the cmdlet is run; no new user object is created in the broker configuration, unless the object is added to another broker object, such as a machine or a desktop. For details, see Add-BrokerUser.

The identity of the user or group must be specified using either the Name or SID parameter

## Related topics

[Add-BrokerUser](#)

[Get-BrokerUser](#)

[Remove-BrokerUser](#)

## Parameters

**-SID**<SecurityIdentifier>

The SID of the user or group

Required?	true
Default Value	null
Accept Pipeline Input?	false

**-Name**<String>

The name of the user or group

Required?	true
Default Value	null

Accept Pipeline Input?	false
------------------------	-------

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

None You cannot pipe input into this cmdlet.

### Return Values

Citrix.Broker.Admin.SDK.User

The broker user object

### Notes

Typically, broker user objects are created implicitly using the Add-BrokerUser cmdlet with a user name or SID.

### Examples

#### ----- **EXAMPLE 1** -----

```
$user = New-BrokerUser DOMAIN\UserName
```

Create a broker user object for the specified user.

#### ----- **EXAMPLE 2** -----

```
$user = New-BrokerUser -SID S-1-5-23-1763203430-193137401-908696819-3450
```

Create a broker user object for the specified user.

# Remove-BrokerAccessPolicyRule

May 10, 2016

Deletes a rule from the site's access policy.

## Syntax

```
Remove-BrokerAccessPolicyRule [-InputObject] <AccessPolicyRule[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-BrokerAccessPolicyRule [-Name] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerAccessPolicyRule cmdlet deletes a rule from the site's access policy.

An access policy rule defines a set of connection filters and access control rights relating to a desktop group. These allow fine-grained control of what access is granted to a desktop group based on details of, for example, a user's endpoint device, its address, and the user's identity.

Deleting a rule does not affect existing user sessions, but it may result in users being unable to launch new sessions, or reconnect to disconnected sessions if access to the desktop group delivering those sessions was granted by the deleted rule.

## Related topics

[New-BrokerAccessPolicyRule](#)

[Get-BrokerAccessPolicyRule](#)

[Set-BrokerAccessPolicyRule](#)

[Rename-BrokerAccessPolicyRule](#)

## Parameters

**-InputObject** <AccessPolicyRule[]>

The access policy rule to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name** <String>

The name of the access policy rule to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.AccessPolicyRule The access policy rule to be deleted.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

C:\PS> Remove-BrokerAccessPolicyRule 'Temp Staff'

Deletes the access policy rule called Temp Staff. Existing sessions are not affected, but if access was granted by the deleted rule users may be unable to reconnect to sessions if they are subsequently disconnected.

----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerAccessPolicyRule -IncludedUsers sales\johndoe | Remove-BrokerAccessPolicyRule
```

Deletes all access policy rules explicitly granting user SALES\johndoe access to any desktop group in the site. Any existing desktop sessions for the user are not affected. The user may still be able to access site resources by access policy rules that grant access through group membership or non-user-based connection filters.

# Remove-BrokerAccessPolicyRuleMetadata

May 10, 2016

Deletes AccessPolicyRule Metadata from the AccessPolicyRule objects

## Syntax

```
Remove-BrokerAccessPolicyRuleMetadata [-InputObject] <AccessPolicyRule[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerAccessPolicyRuleMetadata cmdlet deletes Metadata from the AccessPolicyRule objects.

## Related topics

## Parameters

**-InputObject**<AccessPolicyRule[]>

Specifies the AccessPolicyRule object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerAccessPolicyRule You can pipe the AccessPolicyRule to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerAccessPolicyRuleMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the AccessPolicyRule whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerAccessPolicyRule | Remove-BrokerAccessPolicyRuleMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the AccessPolicyRule in the site

# Remove-BrokerAdminFolder

May 10, 2016

Removes an admin folder.

## Syntax

```
Remove-BrokerAdminFolder [-InputObject] <AdminFolder[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-BrokerAdminFolder [-Name] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerAdminFolder cmdlet removes an existing admin folder.

Remove-BrokerAdminFolder will not remove a folder if it contains any other objects (e.g. sub-folders or applications).

## Related topics

[Get-BrokerAdminFolder](#)

[New-BrokerAdminFolder](#)

## Parameters

**-InputObject**<AdminFolder[]>

Identifies the folder to remove

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

The name pattern of folder(s) to remove

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>



Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.AdminFolder The admin folder objects can be specified as input.

#### Return Values

None

This cmdlet does not return any output.

#### Examples

##### ----- **EXAMPLE 1** -----

Remove-BrokerAdminFolder F1\F2\

Removes the folder called F2 within the folder F1\

# Remove-BrokerAdminFolderMetadata

May 10, 2016

Deletes AdminFolder Metadata from the AdminFolder objects

## Syntax

```
Remove-BrokerAdminFolderMetadata [-InputObject] <AdminFolder[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerAdminFolderMetadata cmdlet deletes Metadata from the AdminFolder objects.

## Related topics

## Parameters

**-InputObject**<AdminFolder[]>

Specifies the AdminFolder object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerAdminFolder You can pipe the AdminFolder to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerAdminFolderMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the AdminFolder whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerAdminFolder | Remove-BrokerAdminFolderMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the AdminFolder in the site

# Remove-BrokerAppAssignmentPolicyRule

May 10, 2016

Deletes an application rule from the site's assignment policy.

## Syntax

```
Remove-BrokerAppAssignmentPolicyRule [-InputObject] <AppAssignmentPolicyRule[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-BrokerAppAssignmentPolicyRule [-Name] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerAppAssignmentPolicyRule cmdlet deletes an application rule from the site's assignment policy.

An application rule in the assignment policy defines the users who are entitled to a self-service persistent machine assignment from the rule's desktop group; once assigned the machine can run one or more applications published from the group.

Deleting an application rule does not remove machine assignments that have already been made by the rule, nor does it affect active sessions to those machines in any way.

## Related topics

[New-BrokerAppAssignmentPolicyRule](#)

[Get-BrokerAppAssignmentPolicyRule](#)

[Set-BrokerAppAssignmentPolicyRule](#)

[Rename-BrokerAppAssignmentPolicyRule](#)

## Parameters

**-InputObject** <AppAssignmentPolicyRule[]>

The application rule to be deleted from the assignment policy.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name** <String>

The name of the application rule to be deleted from the assignment policy.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.AppAssignmentPolicyRule The application rule to be deleted from the assignment policy.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerAppAssignmentPolicyRule 'Temp Staff'
```

Deletes the application rule called Temp Staff from the assignment policy. Access to machines already assigned by this rule is not affected in any way.

----- **EXAMPLE 2** -----

```
C:\PS> $dg = Get-BrokerDesktopGroup 'Sales Support'
```

```
C:\PS> Get-BrokerAppAssignmentPolicyRule -DesktopGroupUid $dg.Uid | Remove-BrokerAppAssignmentPolicyRule
```

Deletes the application rule for the Sales Support desktop group from the site's assignment policy. This prevents any further machine assignments being made from this group, but it does not affect existing assignments made by the rule.

# Remove-BrokerAppEntitlementPolicyRule

May 10, 2016

Deletes an application rule from the site's entitlement policy.

## Syntax

```
Remove-BrokerAppEntitlementPolicyRule [-InputObject] <AppEntitlementPolicyRule[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-BrokerAppEntitlementPolicyRule [-Name] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerAppEntitlementPolicyRule cmdlet deletes an application rule from the site's entitlement policy.

An application rule in the entitlement policy defines the users who are allowed per-session access to a machine to run one or more applications published from the rule's desktop group.

Deleting a rule does not affect existing sessions launched using the rule, but users cannot reconnect to those sessions if they are subsequently disconnected.

## Related topics

[New-BrokerAppEntitlementPolicyRule](#)

[Get-BrokerAppEntitlementPolicyRule](#)

[Set-BrokerAppEntitlementPolicyRule](#)

[Rename-BrokerAppEntitlementPolicyRule](#)

## Parameters

**-InputObject**<AppEntitlementPolicyRule[]>

The application rule to be deleted from the entitlement policy.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

The name of the application rule to be deleted from the entitlement policy.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.AppEntitlementPolicyRule The application rule to be deleted from the entitlement policy.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerAppEntitlementPolicyRule 'Temp Workers'
```

Deletes the application rule called Temp Workers from the entitlement policy rule. Existing application sessions launched using that rule are not affected, but users cannot reconnect to those sessions if they are subsequently disconnected.

#### ----- EXAMPLE 2 -----



```
C:\PS> $dg = Get-BrokerDesktopGroup 'Customer Support'
```

```
C:\PS> Get-BrokerAppEntitlementPolicyRule -DesktopGroupUid $dg.Uid | Remove-BrokerAppEntitlementPolicyRule
```

Deletes the application rule from the entitlement policy rule applied to the Customer Support desktop group. This effectively removes all access to the applications published from this group. Existing application sessions are not affected, but users cannot reconnect to those sessions if they are subsequently disconnected.

# Remove-BrokerApplication

May 10, 2016

Deletes one or more applications, or an association of an application.

## Syntax

```
Remove-BrokerApplication [-InputObject] <Application[]> [-Force] [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-BrokerApplication [-Name] <String> [-Force] [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerApplication cmdlet deletes one or more applications, or you can use it to delete just the association of an application to a desktop group.

Its usage dictates the behavior of the cmdlet. If only the application is specified as a parameter, then the cmdlet deletes the application. It also deletes any associations this application has with other objects, such as with access policy rules or desktop groups. More specifically, when an application is deleted the following happens:

- o The association to any desktop groups is removed.
- o The association to any tags is removed.
- o Any configured file-type association objects for this application are deleted.
- o The association to any user accounts is removed.
- o The association to any access session conditions is removed.
- o The access policy rule object for this application, if one existed, is deleted.
- o Finally, the application object itself is deleted.

Note that if the application is in use by a user then the application cannot be deleted.

If more than just the application is supplied as a parameter to the cmdlet (for instance, if a DesktopGroup object is also specified) then the application is not deleted. Instead, only the association from the application to that desktop group is removed.

## Related topics

[New-BrokerApplication](#)

[Add-BrokerApplication](#)

[Get-BrokerApplication](#)

[Rename-BrokerApplication](#)

[Move-BrokerApplication](#)

## Set-BrokerApplication

### Parameters

**-InputObject**<Application[]>

Specifies the applications to delete. The Uid can also be substituted for the application objects.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the application to remove.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Force**<SwitchParameter>

Remove application even if it's in use. Removing an application that is currently in use, can potentially leave an application session containing no applications. If all the applications that are currently active in a disconnected application session are removed, the user will be unable to reconnect to the session. Forcing removal of an in-use application does not impact the actual session itself.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-DesktopGroup**<DesktopGroup>

Specifies the desktop group that this application should no longer be associated with. The Uid or Name can also be substituted for the desktop group object.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.Application The application objects can be specified as input.

#### Return Values

None

This cmdlet does not return any output.

#### Examples

----- **EXAMPLE 1** -----

C:\PS> Remove-BrokerApplication "Notepad"

This command deletes the application that has a BrowserName of "Notepad".

----- **EXAMPLE 2** -----

```
C:\PS> $app = Get-BrokerApplication -BrowserName "Notepad"
```

```
C:\PS> $group = Get-BrokerDesktopGroup -Name "Private DesktopGroup"
```

```
C:\PS> Remove-BrokerApplication -InputObject $app -DesktopGroup $group
```

This command removes the association of the desktop group that has a name of "Private DesktopGroup" from the application that has a BrowserName of "Notepad". It does not otherwise modify the application.

# Remove-BrokerApplicationInstanceMetadata

May 10, 2016

Deletes ApplicationInstance Metadata from the ApplicationInstance objects

## Syntax

```
Remove-BrokerApplicationInstanceMetadata [-InputObject] <ApplicationInstance[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerApplicationInstanceMetadata cmdlet deletes Metadata from the ApplicationInstance objects.

## Related topics

## Parameters

**-InputObject**<ApplicationInstance[]>

Specifies the ApplicationInstance object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerApplicationInstance You can pipe the ApplicationInstance to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerApplicationInstanceMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the ApplicationInstance whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerApplicationInstance | Remove-BrokerApplicationInstanceMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the ApplicationInstance in the site

# Remove-BrokerApplicationMetadata

May 10, 2016

Deletes Application Metadata from the Application objects

## Syntax

```
Remove-BrokerApplicationMetadata [-InputObject] <Application[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerApplicationMetadata cmdlet deletes Metadata from the Application objects.

## Related topics

## Parameters

**-InputObject**<Application[]>

Specifies the Application object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------



Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerApplication You can pipe the Application to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerApplicationMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the Application whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerApplication | Remove-BrokerApplicationMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the Application in the site

# Remove-BrokerAssignmentPolicyRule

May 10, 2016

Deletes a desktop rule from the site's assignment policy.

## Syntax

```
Remove-BrokerAssignmentPolicyRule [-InputObject] <AssignmentPolicyRule[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-BrokerAssignmentPolicyRule [-Name] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerAssignmentPolicyRule cmdlet deletes a desktop rule from the site's assignment policy.

A desktop rule in the assignment policy defines the users who are entitled to self-service persistent machine assignments from the rule's desktop group. A rule defines how many machines a user is allowed from the group for delivery of full desktop sessions.

Deleting a desktop rule does not remove machine assignments that have already been made by the rule, nor does it affect active sessions to those machines in any way.

## Related topics

[New-BrokerAssignmentPolicyRule](#)

[Get-BrokerAssignmentPolicyRule](#)

[Set-BrokerAssignmentPolicyRule](#)

[Rename-BrokerAssignmentPolicyRule](#)

## Parameters

**-InputObject** <AssignmentPolicyRule[]>

The desktop rule to be deleted from the assignment policy.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name** <String>

The name of the desktop rule to be deleted from the assignment policy.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.AssignmentPolicyRule The desktop rule to be deleted from the assignment policy.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerAssignmentPolicyRule 'Temp Staff'
```

Deletes the desktop rule called Temp Staff from the assignment policy. Access to machines already assigned by this rule is not affected in any way.

----- **EXAMPLE 2** -----

```
C:\PS> $dg = Get-BrokerDesktopGroup 'Sales Support'
```

```
C:\PS> Get-BrokerAssignmentPolicyRule -DesktopGroupUid $dg.Uid | Remove-BrokerAssignmentPolicyRule
```

Deletes all desktop rules for the Sales Support desktop group from the site's assignment policy. This prevents any further machine assignments being made from this group, but it does not affect existing assignments made by these rules.

# Remove-BrokerAssignmentPolicyRuleMetadata

May 10, 2016

Deletes AssignmentPolicyRule Metadata from the AssignmentPolicyRule objects

## Syntax

```
Remove-BrokerAssignmentPolicyRuleMetadata [-InputObject] <AssignmentPolicyRule[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerAssignmentPolicyRuleMetadata cmdlet deletes Metadata from the AssignmentPolicyRule objects.

## Related topics

## Parameters

**-InputObject**<AssignmentPolicyRule[]>

Specifies the AssignmentPolicyRule object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerAssignmentPolicyRule You can pipe the AssignmentPolicyRule to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerAssignmentPolicyRuleMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the AssignmentPolicyRule whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerAssignmentPolicyRule | Remove-BrokerAssignmentPolicyRuleMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the AssignmentPolicyRule in the site

# Remove-BrokerCatalog

May 10, 2016

Removes catalogs from the site.

## Syntax

```
Remove-BrokerCatalog [-InputObject] <Catalog[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-BrokerCatalog [-Name] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Remove catalogs from the site.

In order to remove a catalog from a site, the catalog must not contain machines. To remove a machine from a catalog use the Remove-BrokerMachine cmdlet. Note: in order to remove a machine from a catalog, it must not belong to a desktop group.

## Related topics

[New-BrokerCatalog](#)

[Get-BrokerCatalog](#)

[Rename-BrokerCatalog](#)

[Set-BrokerCatalog](#)

[New-BrokerDesktopGroup](#)

[Remove-BrokerDesktopGroup](#)

## Parameters

**-InputObject**<Catalog[]>

Specifies the catalog objects to delete.

Required?	true
Default Value	null
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the catalog to delete.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.Catalog You can pipe the catalogs to be deleted to Remove-BrokerCatalog.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerCatalog -Name "MyCatalog"
```

```
C:\PS> Remove-BrokerCatalog -InputObject (Get-BrokerCatalog -Name "MyCatalog")
```

These commands delete the catalog with the name "MyCatalog".

##### ----- EXAMPLE 2 -----



```
C:\PS> Remove-BrokerCatalog -Name 'test'
```

This command deletes all catalogs with names beginning with "test".

----- **EXAMPLE 3** -----

```
C:\PS> Get-BrokerCatalog -RemotePCDesktopGroupUid 42 | Remove-BrokerCatalog -RemotePCDesktopGroup 42
```

Remove all the Remote PC catalogs that are associated with desktop group 42. Note that this only breaks the Remote PC relationships and does not delete the desktop groups.

# Remove-BrokerCatalogMetadata

May 10, 2016

Deletes Catalog Metadata from the Catalog objects

## Syntax

```
Remove-BrokerCatalogMetadata [-InputObject] <Catalog[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Remove-BrokerCatalogMetadata cmdlet deletes Metadata from the Catalog objects.

## Related topics

## Parameters

**-InputObject**<Catalog[]>

Specifies the Catalog object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerCatalog You can pipe the Catalog to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerCatalogMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the Catalog whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerCatalog | Remove-BrokerCatalogMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the Catalog in the site

# Remove-BrokerConfigurationSlot

May 10, 2016

Removes a configuration slot.

## Syntax

```
Remove-BrokerConfigurationSlot [-InputObject] <ConfigurationSlot[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-BrokerConfigurationSlot [-Name] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Removes a configuration slot from the site. All machine configurations associated with this slot are also removed.

## Related topics

[New-BrokerConfigurationSlot](#)

[Get-BrokerConfigurationSlot](#)

## Parameters

**-InputObject**<ConfigurationSlot[]>

Configuration slot to remove.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Name of configuration slot to remove.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop

Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Configuration slot to remove Configuration slots may be specified through pipeline input.

#### Return Values

None

#### Examples

##### ----- **EXAMPLE 1** -----

Remove-BrokerConfigurationSlot -Name "User Profile Manager"  
Remove the configuration slot named "User Profile Manager".

# Remove-BrokerConfigurationSlotMetadata

May 10, 2016

Deletes ConfigurationSlot Metadata from the ConfigurationSlot objects

## Syntax

```
Remove-BrokerConfigurationSlotMetadata [-InputObject] <ConfigurationSlot[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerConfigurationSlotMetadata cmdlet deletes Metadata from the ConfigurationSlot objects.

## Related topics

## Parameters

**-InputObject**<ConfigurationSlot[]>

Specifies the ConfigurationSlot object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerConfigurationSlot You can pipe the ConfigurationSlot to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerConfigurationSlotMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the ConfigurationSlot whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerConfigurationSlot | Remove-BrokerConfigurationSlotMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the ConfigurationSlot in the site

# Remove-BrokerConfiguredFTA

May 10, 2016

Deletes one or more configured file type associations.

## Syntax

```
Remove-BrokerConfiguredFTA [-InputObject] <ConfiguredFTA[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Deletes one or more file type associations configured for a published application. At least one configured file type association object must be specified.

## Related topics

[Get-BrokerConfiguredFTA](#)

[New-BrokerConfiguredFTA](#)

## Parameters

**-InputObject**<ConfiguredFTA[]>

Specifies the ConfiguredFTA objects to delete.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>



Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.ConfiguredFTA[] One or more ConfiguredFTA objects can be supplied as input.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> $ftas = Get-BrokerConfiguredFTA -ExtensionName ".txt"
```

```
C:\PS> Remove-BrokerConfiguredFTA $ftas
```

Deletes all configured file type associations with an extension name of ".txt".

# Remove-BrokerControllerMetadata

May 10, 2016

Deletes Controller Metadata from the Controller objects

## Syntax

```
Remove-BrokerControllerMetadata [-InputObject] <Controller[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerControllerMetadata cmdlet deletes Metadata from the Controller objects.

## Related topics

## Parameters

**-InputObject**<Controller[]>

Specifies the Controller object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerController You can pipe the Controller to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerControllerMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the Controller whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerController | Remove-BrokerControllerMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the Controller in the site

# Remove-BrokerDelayedHostingPowerAction

May 10, 2016

Cancels one or more delayed power actions.

## Syntax

```
Remove-BrokerDelayedHostingPowerAction [-InputObject] <DelayedHostingPowerAction[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-BrokerDelayedHostingPowerAction [-MachineName] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Removes one or more delayed power actions that have not yet been queued for execution.

## Related topics

[Get-BrokerDelayedHostingPowerAction](#)

[New-BrokerDelayedHostingPowerAction](#)

## Parameters

**-InputObject**<DelayedHostingPowerAction[]>

The power action to be cancelled.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-MachineName**<String>

Cancels only actions for machines whose name (of the form domain\machine) matches the specified string.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.DelayedHostingPowerAction The power action to be cancelled.

#### Return Values

None

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Remove-BrokerHostingPowerAction -MachineName 'XD_VDA1'
```

Cancels any pending delayed power actions for the machine called XD\_VDA1.

# Remove-BrokerDesktopGroup

May 10, 2016

Remove desktop groups from the system or remove them from a Remote PC catalog.

## Syntax

```
Remove-BrokerDesktopGroup [-InputObject] <DesktopGroup[]> [-Force] [-RemotePCCatalog <Catalog>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-BrokerDesktopGroup [-Name] <String> [-Force] [-RemotePCCatalog <Catalog>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet has 2 functions:

- o Remove desktop groups from the system.
- o Break Remote PC associations between desktop groups and a catalog.

The Remote PC relationships are used by Remote PC automation to determine which desktop groups a machine in a particular Remote PC catalog can be published to. The assignment policy rules belonging to those desktop groups also determines the set of users that are allowed to be assigned to machines from the catalog.

## Related topics

[Get-BrokerDesktopGroup](#)

[New-BrokerDesktopGroup](#)

[Set-BrokerDesktopGroup](#)

[Add-BrokerDesktopGroup](#)

[Rename-BrokerDesktopGroup](#)

[New-BrokerCatalog](#)

[Remove-BrokerCatalog](#)

## Parameters

**-InputObject** <DesktopGroup[]>

Specifies the desktop groups to remove.

Required?	true
Default Value	null
Accept Pipeline Input?	true (ByValue)

--	--

#### **-Name**<String>

Specifies the name of the desktop group to remove.

Required?	true
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

#### **-Force**<SwitchParameter>

Remove desktop groups even if there are active sessions.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### **-RemotePCCatalog**<Catalog>

When this parameter is specified, Remote PC desktop groups are removed from the specified Remote PC catalog.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.DesktopGroup You can pipe desktop groups to Remove-BrokerDesktopGroup.

### Return Values

None

### Notes

If a desktop group contains desktops when it is removed, these desktops are also removed (but the underlying broker machine remains).

A desktop group that still has active sessions cannot be removed unless the -Force switch is used.

### Examples

#### ----- **EXAMPLE 1** -----

```
C:\PS> Remove-BrokerDesktopGroup EMEA*
Remove all desktop groups with names starting with "EMEA".
```

#### ----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerDesktopGroup -Enabled $false | Remove-BrokerDesktopGroup -Force
Remove all desktops that are currently disabled even if there are active sessions.
```

#### ----- **EXAMPLE 3** -----

```
C:\PS> Get-BrokerDesktopGroup -RemotePCCatalogUid 42 | Remove-BrokerDesktopGroup -RemotePCCatalog 42
Remove all the Remote PC desktop groups that are associated with catalog 42. Note that this only breaks the Remote PC relationships and does not delete the desktop groups.
```



# Remove-BrokerDesktopGroupMetadata

May 10, 2016

Deletes DesktopGroup Metadata from the DesktopGroup objects

## Syntax

```
Remove-BrokerDesktopGroupMetadata [-InputObject] <DesktopGroup[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerDesktopGroupMetadata cmdlet deletes Metadata from the DesktopGroup objects.

## Related topics

## Parameters

**-InputObject**<DesktopGroup[]>

Specifies the DesktopGroup object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerDesktopGroup You can pipe the DesktopGroup to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerDesktopGroupMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the DesktopGroup whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerDesktopGroup | Remove-BrokerDesktopGroupMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the DesktopGroup in the site

# Remove-BrokerEntitlementPolicyRule

May 10, 2016

Deletes a desktop rule from the site's entitlement policy.

## Syntax

```
Remove-BrokerEntitlementPolicyRule [-InputObject] <EntitlementPolicyRule[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-BrokerEntitlementPolicyRule [-Name] <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Remove-BrokerEntitlementPolicyRule cmdlet deletes a desktop rule from the site's entitlement policy.

A desktop rule in the entitlement policy defines the users who are allowed per-session access to a machine from the rule's associated desktop group to run a full desktop session.

Deleting a rule does not affect existing sessions launched using the rule, but users cannot reconnect to those sessions if they are subsequently disconnected.

## Related topics

[New-BrokerEntitlementPolicyRule](#)

[Get-BrokerEntitlementPolicyRule](#)

[Set-BrokerEntitlementPolicyRule](#)

[Rename-BrokerEntitlementPolicyRule](#)

## Parameters

**-InputObject**<EntitlementPolicyRule[]>

The desktop rule to be deleted from the entitlement policy.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

The name of the desktop rule to be deleted from the entitlement policy.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.EntitlementPolicyRule The desktop rule to be deleted from the entitlement policy.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerEntitlementPolicyRule 'Temp Workers'
```

Deletes the desktop rule called Temp Workers from the entitlement policy. Existing desktop sessions launched using the rule are not affected, but users cannot reconnect to sessions if they are subsequently disconnected.

##### ----- EXAMPLE 2 -----

```
C:\PS> $dg = Get-BrokerDesktopGroup 'Customer Support'
```

```
C:\PS> Get-BrokerEntitlementPolicyRule -DesktopGroupUid $dg.Uid | Remove-BrokerEntitlementPolicyRule
```

Deletes all desktop rules from the entitlement policy applying to the Customer Support desktop group. This effectively removes all access to the desktops published from this group. Existing desktop sessions are not affected, but users cannot reconnect to sessions if they are subsequently disconnected.

# Remove-BrokerEntitlementPolicyRuleMetadata

May 10, 2016

Deletes EntitlementPolicyRule Metadata from the EntitlementPolicyRule objects

## Syntax

```
Remove-BrokerEntitlementPolicyRuleMetadata [-InputObject] <EntitlementPolicyRule[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerEntitlementPolicyRuleMetadata cmdlet deletes Metadata from the EntitlementPolicyRule objects.

## Related topics

## Parameters

**-InputObject**<EntitlementPolicyRule[]>

Specifies the EntitlementPolicyRule object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerEntitlementPolicyRule You can pipe the EntitlementPolicyRule to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerEntitlementPolicyRuleMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the EntitlementPolicyRule whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerEntitlementPolicyRule | Remove-BrokerEntitlementPolicyRuleMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the EntitlementPolicyRule in the site

# Remove-BrokerHostingPowerAction

May 10, 2016

Cancel one or more pending power actions.

## Syntax

```
Remove-BrokerHostingPowerAction [-InputObject] <HostingPowerAction[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-BrokerHostingPowerAction [-MachineName] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Causes one or more of the pending power actions in the queue to be marked as cancelled. The affected power actions are not sent to the hypervisor for processing, and take no further part in the queuing activity.

Power actions cannot be cancelled once they have started to be processed by the hypervisor.

## Related topics

[Get-BrokerHostingPowerAction](#)

[New-BrokerHostingPowerAction](#)

[Set-BrokerHostingPowerAction](#)

## Parameters

**-InputObject**<HostingPowerAction[]>

The power action to be cancelled.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-MachineName**<String>

Cancels only actions for machines whose name (of the form domain\machine) matches the specified string.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically



create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.HostingPowerAction The power action to be cancelled.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerHostingPowerAction -MachineName 'XD_VDA1'
```

Cancels any pending power actions for the machine called 'XD\_VDA1'.

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerHostingPowerAction -Filter { State -eq "Pending" -and RequestTime -gt "-00:05" } | Remove-BrokerHostingPowerAction
```

Cancels any pending power actions requested in the last five minutes.

# Remove-BrokerHostingPowerActionMetadata

May 10, 2016

Deletes HostingPowerAction Metadata from the HostingPowerAction objects

## Syntax

```
Remove-BrokerHostingPowerActionMetadata [-InputObject] <HostingPowerAction[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerHostingPowerActionMetadata cmdlet deletes Metadata from the HostingPowerAction objects.

## Related topics

## Parameters

**-InputObject**<HostingPowerAction[]>

Specifies the HostingPowerAction object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerHostingPowerAction You can pipe the HostingPowerAction to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerHostingPowerActionMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the HostingPowerAction whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerHostingPowerAction | Remove-BrokerHostingPowerActionMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the HostingPowerAction in the site

# Remove-BrokerHypervisorAlertMetadata

May 10, 2016

Deletes HypervisorAlert Metadata from the HypervisorAlert objects

## Syntax

```
Remove-BrokerHypervisorAlertMetadata [-InputObject] <HypervisorAlert[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerHypervisorAlertMetadata cmdlet deletes Metadata from the HypervisorAlert objects.

## Related topics

## Parameters

**-InputObject**<HypervisorAlert[]>

Specifies the HypervisorAlert object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerHypervisorAlert You can pipe the HypervisorAlert to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerHypervisorAlertMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the HypervisorAlert whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerHypervisorAlert | Remove-BrokerHypervisorAlertMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the HypervisorAlert in the site

# Remove-BrokerHypervisorConnection

May 10, 2016

Removes a hypervisor connection from the system.

## Syntax

```
Remove-BrokerHypervisorConnection [-InputObject] <HypervisorConnection[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-BrokerHypervisorConnection [-Name] <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Remove-BrokerHypervisorConnection removes a hypervisor connection from the system. A hypervisor connection cannot be removed if it's being used by a machine.

## Related topics

[Get-BrokerHypervisorConnection](#)

[Set-BrokerHypervisorConnection](#)

[New-BrokerHypervisorConnection](#)

## Parameters

**-InputObject** <HypervisorConnection[]>

Specifies the hypervisor connection object to remove.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name** <String>

Specifies the name of the hypervisor connection object to remove.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.HypervisorConnection You can pipe the hypervisor connection to be removed to Remove-BrokerHypervisorConnection.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

```
c:\PS> Remove-BrokerHypervisorConnection -Name "Xen Server Connection"
```

This command removes a hypervisor connection by name.

#### ----- EXAMPLE 2 -----

```
c:\PS> $hvConn = Get-BrokerHypervisorConnection -PreferredController "controllerName" -Name "Xen Server Connection"
```

```
c:\PS> Remove-BrokerHypervisorConnection -InputObject $hvConn
```

Gets a hypervisor connection by preferred controller and removes it.

# Remove-BrokerHypervisorConnectionMetadata

May 10, 2016

Deletes HypervisorConnection Metadata from the HypervisorConnection objects

## Syntax

```
Remove-BrokerHypervisorConnectionMetadata [-InputObject] <HypervisorConnection[]> -Name <String> [-LoggingId <Guid>]
[-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerHypervisorConnectionMetadata cmdlet deletes Metadata from the HypervisorConnection objects.

## Related topics

## Parameters

**-InputObject**<HypervisorConnection[]>

Specifies the HypervisorConnection object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------



Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerHypervisorConnection You can pipe the HypervisorConnection to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerHypervisorConnectionMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the HypervisorConnection whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerHypervisorConnection | Remove-BrokerHypervisorConnectionMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the HypervisorConnection in the site

# Remove-BrokerIcon

May 10, 2016

Remove an icon.

## Syntax

```
Remove-BrokerIcon [-InputObject] <Icon[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Removes an icon from the database.

## Related topics

[Get-BrokerIcon](#)

[New-BrokerIcon](#)

[Set-BrokerIconMetadata](#)

## Parameters

**-InputObject**<Icon[]>

Specifies the icon to remove.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.Icon The icon to be removed can be piped into the cmdlet.

### Return Values

None

### Notes

Note that if the icon is currently in use, for example, as a desktop icon, it cannot be removed until the association is cleared.

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerIcon 3
Removes the icon with Uid 3.
```

# Remove-BrokerIconMetadata

May 10, 2016

Deletes Icon Metadata from the Icon objects

## Syntax

```
Remove-BrokerIconMetadata [-InputObject] <Icon[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Remove-BrokerIconMetadata cmdlet deletes Metadata from the Icon objects.

## Related topics

## Parameters

**-InputObject**<Icon[]>

Specifies the Icon object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerIcon You can pipe the Icon to delete the metadata.

#### Return Values

None

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Remove-BrokerIconMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the Icon whose instance is pointed by \$obj-Uid

##### ----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerIcon | Remove-BrokerIconMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the Icon in the site

# Remove-BrokerImportedFTA

May 10, 2016

Deletes one or more imported file type associations.

## Syntax

```
Remove-BrokerImportedFTA -DesktopGroupUids <Int32[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Deletes all of the imported file type associations belonging to one or more desktop groups. At least one desktop group must be specified.

Imported file type associations are grouped together based on the desktop group of the machine from which they were imported. All file types for a desktop group are deleted. There is no mechanism for deleting a subset imported file type associations for a specific desktop group.

Imported file type associations are different from configured file type associations. Imported file type associations are lists of known file type associations for a given desktop group. Configured file type associations are those that are actually associated with published applications for the purposes of content redirection.

## Related topics

[Get-BrokerImportedFTA](#)

[Update-BrokerImportedFTA](#)

## Parameters

**-DesktopGroupUids**<Int32[]>

Deletes the imported file type associations belonging to specified desktop groups.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Int32[] An array of Uids for the desktop groups can be supplied as input. The desktop groups must be of the Private or Shared desktop kind.

#### Return Values

None

#### Notes

If an imported file type association is used to create a new configured file type association and the imported file type association is subsequently deleted, the configured file type association is not affected.

#### Examples

----- **EXAMPLE 1** -----

```
C:\PS> $dg = Get-BrokerDesktopGroup -Name "Sales VMs"
```

```
C:\PS> Remove-BrokerImportedFTA -DesktopGroupUids $dg.Uid
```

Deletes all imported file type associations belonging to the "Sales VMs" desktop group.

# Remove-BrokerLease

May 10, 2016

Remove the specified lease in the Database.

## Syntax

Remove-BrokerLease [-InputObject] <Lease[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Remove-BrokerLease [-Key] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Marks the specified lease for deletion. Note that the lease is eventually fully deleted when enough time has been allowed for the deletion to propagate to all controller machines in the site, but is immediately removed from lease search results.

## Related topics

[Update-BrokerLocalLeaseCache](#)

Remove-BrokerCache

## Parameters

**-InputObject**<Lease[]>

Specifies the lease to remove.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Key**<String>

Specifies the lease key of the lease to remove. A pattern can be specified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop



Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.Lease The lease to be removed can be piped into the cmdlet.

#### Return Values

None

This cmdlet does not return any output.

#### Notes

The lease is marked for deletion after this cmdlet is run. Note that the lease is eventually fully deleted when enough time has been allowed for the deletion to propagate to all controller machines in the site, but is immediately removed from lease search results.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> $lease = Get-BrokerLease -Uid 1
```

```
C:\PS> Remove-BrokerLease $lease
```

Marks the specified lease for deletion.

# Remove-BrokerLeaseMetadata

May 10, 2016

Deletes Lease Metadata from the Lease objects

## Syntax

```
Remove-BrokerLeaseMetadata [-InputObject] <Lease[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Remove-BrokerLeaseMetadata cmdlet deletes Metadata from the Lease objects.

## Related topics

## Parameters

**-InputObject**<Lease[]>

Specifies the Lease object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerLease You can pipe the Lease to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerLeaseMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the Lease whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerLease | Remove-BrokerLeaseMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the Lease in the site

# Remove-BrokerMachine

May 10, 2016

Removes one or more machines from its desktop group or catalog.

## Syntax

```
Remove-BrokerMachine [-InputObject] <Machine[]> [-Force] [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-BrokerMachine [-MachineName] <String> [-Force] [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerMachine cmdlet removes one or more machines from their desktop group or catalog. There are three forms:

- o Use the -InputObject parameter to remove a single machine instance or array of instances from their desktop group or catalog.
- o Use the -MachineName parameter to remove the single named machine from its group or catalog.
- o Use pipelining to pipe machine instances to the command.

To remove machines from their desktop group use the -DesktopGroup parameter; the specified group must be the one that contains the machines. If more than one machine is being removed from its group they must all be members of the same group.

If the -DesktopGroup parameter is not used then the machines are removed from their catalog. Removing a machine from its catalog deletes the record of the machine from the Citrix Broker Service.

Machines cannot be removed from their catalog while they are members of a desktop group.

## Related topics

[Add-BrokerMachine](#)

[Get-BrokerMachine](#)

## Parameters

**-InputObject**<Machine[]>

An array of machines to be removed from their desktop group or catalog.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-MachineName**<String>

The name of the single machine to remove (must match the MachineName property of the machine).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Force**<SwitchParameter>

Forces removal of machine from a desktop group even if it is still in use (that is, there are user sessions running on the machine). Forcing removal of a machine does not disconnect or logoff the user sessions.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroup**<DesktopGroup>

The desktop group from which the machines are to be removed, specified by name, UID, or instance.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

Citrix.Broker.Admin.SDK.Machine You can pipe in the machines to be removed.

Return Values

None

Examples

----- **EXAMPLE 1** -----

```
C:\PS> Remove-BrokerMachine -InputObject $machine -DesktopGroup $desktopGroup
C:\PS> Remove-BrokerMachine -InputObject $machine -DesktopGroup 2
C:\PS> Remove-BrokerMachine $machine -DesktopGroup "MyDesktopGroup"
```

These all remove a single machine from a desktop group, identifying the group by instance, UID, or name.

----- **EXAMPLE 2** -----

```
C:\PS> Remove-BrokerMachine -MachineName "DOMAIN\MyMachine" -DesktopGroup 2
C:\PS> Remove-BrokerMachine DOMAIN\MyMachine -DesktopGroup "MyDesktopGroup"
C:\PS> Remove-BrokerMachine DOMAIN\MyMachine -DesktopGroup $desktopGroup
```

These remove the machine called "DOMAIN\MyMachine" from its desktop group.

----- **EXAMPLE 3** -----

```
C:\PS> Remove-BrokerMachine -MachineName DOMAIN\MyMachine
C:\PS> Remove-BrokerMachine "DOMAIN\MyMachine"
C:\PS> Remove-BrokerMachine $machine
```

These all remove a machine from its catalog.

----- **EXAMPLE 4** -----

```
C:\PS> Get-BrokerMachine -Uid 3 | Remove-BrokerMachine -DesktopGroup $dg
C:\PS> Get-BrokerMachine -CatalogUid 4 | Remove-BrokerMachine
```

These find specific machines and remove them from their desktop group or catalog.

# Remove-BrokerMachineCommand

May 10, 2016

Cancel a pending command queued for delivery to a desktop.

## Syntax

```
Remove-BrokerMachineCommand [-InputObject] <MachineCommand[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Sets the state of a pending command queued for delivery to a desktop to Canceled. The command is not removed from the system.

## Related topics

[Get-BrokerMachineCommand](#)

[New-BrokerMachineCommand](#)

## Parameters

**-InputObject**<MachineCommand[]>

Commands to cancel.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.MachineCommand Commands to cancel.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

Get-BrokerMachineCommand | Remove-BrokerMachineCommand  
Cancel all pending commands.

#### ----- EXAMPLE 2 -----

Get-BrokerMachineCommand -Category "UPM" | Remove-BrokerMachineCommand  
Cancel all pending commands that have the category "UPM".



# Remove-BrokerMachineCommandMetadata

May 10, 2016

Deletes MachineCommand Metadata from the MachineCommand objects

## Syntax

```
Remove-BrokerMachineCommandMetadata [-InputObject] <MachineCommand[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerMachineCommandMetadata cmdlet deletes Metadata from the MachineCommand objects.

## Related topics

## Parameters

**-InputObject**<MachineCommand[]>

Specifies the MachineCommand object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerMachineCommand You can pipe the MachineCommand to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerMachineCommandMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the MachineCommand whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerMachineCommand | Remove-BrokerMachineCommandMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the MachineCommand in the site

# Remove-BrokerMachineConfiguration

May 10, 2016

Deletes a machine configuration from the site or removes the association from a desktop group.

## Syntax

```
Remove-BrokerMachineConfiguration [-InputObject] <MachineConfiguration[]> [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-BrokerMachineConfiguration [-Name] <String> [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-MachineConfiguration cmdlet deletes machine configurations from the site. A machine configuration cannot be removed while it is applied to a desktop group.

When a desktop group is provided, the Remove-MachineConfiguration cmdlet removes the association between machine configuration and the desktop group. In this case, the machine configuration is not removed from the site.

## Related topics

[New-BrokerMachineConfiguration](#)

[Get-BrokerMachineConfiguration](#)

[Set-BrokerMachineConfiguration](#)

[Rename-BrokerMachineConfiguration](#)

[Add-BrokerMachineConfiguration](#)

## Parameters

**-InputObject**<MachineConfiguration[]>

Machine configuration to remove.

Required?	true
Default Value	None
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Name of machine configuration for which the remove operation applies.

Required?	true
-----------	------

Default Value	None
Accept Pipeline Input?	true (ByPropertyName)

#### **-DesktopGroup**<DesktopGroup>

The desktop group from which this machine configuration is to be removed.

Required?	false
Default Value	None
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.MachineConfiguration Machine configuration to remove

## Return Values

None

## Notes

A machine configuration can only be removed if it is not currently applied to a desktop group.

## Examples

### ----- **EXAMPLE 1** -----

`Remove-BrokerMachineConfiguration -Name UPM\Finance`

Removes the machine configuration named "UPM\Finance".

### ----- **EXAMPLE 2** -----

`Remove-BrokerMachineConfiguration -Name Receiver\HumanResources -DesktopGroup SharedWorkers`

Removes the association of the machine configuration named "Receiver\HumanResources" from the "SharedWorkers" desktop group.

# Remove-BrokerMachineConfigurationMetadata

May 10, 2016

Deletes MachineConfiguration Metadata from the MachineConfiguration objects

## Syntax

```
Remove-BrokerMachineConfigurationMetadata [-InputObject] <MachineConfiguration[]> -Name <String> [-LoggingId <Guid>]
[-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerMachineConfigurationMetadata cmdlet deletes Metadata from the MachineConfiguration objects.

## Related topics

## Parameters

**-InputObject**<MachineConfiguration[]>

Specifies the MachineConfiguration object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerMachineConfiguration You can pipe the MachineConfiguration to delete the metadata.

#### Return Values

None

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Remove-BrokerMachineConfigurationMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the MachineConfiguration whose instance is pointed by \$obj-Uid

##### ----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerMachineConfiguration | Remove-BrokerMachineConfigurationMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the MachineConfiguration in the site

# Remove-BrokerMachineMetadata

May 10, 2016

Deletes Machine Metadata from the Machine objects

## Syntax

```
Remove-BrokerMachineMetadata [-InputObject] <Machine[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Remove-BrokerMachineMetadata cmdlet deletes Metadata from the Machine objects.

## Related topics

## Parameters

**-InputObject**<Machine[]>

Specifies the Machine object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------



Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerMachine You can pipe the Machine to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerMachineMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the Machine whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerMachine | Remove-BrokerMachineMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the Machine in the site

# Remove-BrokerPowerTimeScheme

May 10, 2016  
Deletes an existing power time scheme.

## Syntax

```
Remove-BrokerPowerTimeScheme [-InputObject <PowerTimeScheme[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
Remove-BrokerPowerTimeScheme [-Name <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerPowerTimeScheme cmdlet deletes a power time scheme from the system, and leaves the days that the time scheme used to cover for the associated desktop group as defaulting to all hours off-peak and all hours with pool size of -1.

Each power time scheme is associated with a particular desktop group, and covers one or more days of the week, defining which hours of those days are considered peak times and which are off-peak times. In addition, the time scheme defines a pool size value for each hour of the day for the days of the week covered by the time scheme. No one desktop group can be associated with two or more time schemes that cover the same day of the week.

For more information about the power policy mechanism and pool size management, see 'help about\_Broker\_PowerManagement'.

## Related topics

- [Get-BrokerPowerTimeScheme](#)
- [Set-BrokerPowerTimeScheme](#)
- [New-BrokerPowerTimeScheme](#)
- [Rename-BrokerPowerTimeScheme](#)

## Parameters

**-InputObject**<PowerTimeScheme[]>  
The power time scheme to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>  
The name of the power time scheme to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>  
Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High

Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

Citrix.Broker.Admin.SDK.PowerTimeScheme The power time scheme to be deleted.

Return Values

None

Examples

----- **EXAMPLE 1** -----

C:\PS> Remove-BrokerPowerTimeScheme -Name 'Development Weekdays'  
Deletes the power time scheme named 'Development Weekdays'.

----- **EXAMPLE 2** -----

C:\PS> Get-BrokerPowerTimeScheme -DesktopGroupUid (Get-BrokerDesktopGroup -name 'Finance desk1').Uid | Remove-BrokerPowerTimeScheme  
Deletes all power time schemes for the desktop group named 'Finance desk1'.

# Remove-BrokerPowerTimeSchemeMetadata

May 10, 2016

Deletes PowerTimeScheme Metadata from the PowerTimeScheme objects

## Syntax

```
Remove-BrokerPowerTimeSchemeMetadata [-InputObject] <PowerTimeScheme[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerPowerTimeSchemeMetadata cmdlet deletes Metadata from the PowerTimeScheme objects.

## Related topics

## Parameters

**-InputObject**<PowerTimeScheme[]>

Specifies the PowerTimeScheme object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerPowerTimeScheme You can pipe the PowerTimeScheme to delete the metadata.

#### Return Values

None

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Remove-BrokerPowerTimeSchemeMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the PowerTimeScheme whose instance is pointed by \$obj-Uid

##### ----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerPowerTimeScheme | Remove-BrokerPowerTimeSchemeMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the PowerTimeScheme in the site

# Remove-BrokerRebootCycleMetadata

May 10, 2016

Deletes RebootCycle Metadata from the RebootCycle objects

## Syntax

```
Remove-BrokerRebootCycleMetadata [-InputObject] <RebootCycle[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerRebootCycleMetadata cmdlet deletes Metadata from the RebootCycle objects.

## Related topics

## Parameters

**-InputObject**<RebootCycle[]>

Specifies the RebootCycle object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerRebootCycle You can pipe the RebootCycle to delete the metadata.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerRebootCycleMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the RebootCycle whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerRebootCycle | Remove-BrokerRebootCycleMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the RebootCycle in the site

# Remove-BrokerRebootSchedule

May 10, 2016

Removes the reboot schedule.

## Syntax

```
Remove-BrokerRebootSchedule [-InputObject] <RebootSchedule[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-BrokerRebootSchedule [-DesktopGroupName] <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Remove-BrokerRebootSchedule cmdlet is used to delete an existing reboot schedule.

## Related topics

[Get-BrokerRebootSchedule](#)

[Set-BrokerRebootSchedule](#)

[New-BrokerRebootSchedule](#)

[Stop-BrokerRebootCycle](#)

## Parameters

**-InputObject**<RebootSchedule[]>

The reboot schedule to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-DesktopGroupName**<String>

The name of the desktop group whose reboot schedule is to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)



### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.RebootSchedule Reboot schedules may be specified through pipeline input.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerRebootSchedule | Remove-BrokerRebootSchedule
```

Deletes every reboot schedule.

#### ----- EXAMPLE 2 -----

```
C:\PS> Remove-BrokerRebootSchedule 12
```

Deletes the reboot schedule for the desktop group having Uid 12.

#### ----- EXAMPLE 3 -----

```
C:\PS> Remove-BrokerRebootSchedule -DesktopGroupName Accounting
```

Deletes the reboot schedule for the desktop group named Accounting.



# Remove-BrokerRemotePCAccount

May 10, 2016

Delete RemotePCAccounts from the system.

## Syntax

```
Remove-BrokerRemotePCAccount [-InputObject] <RemotePCAccount[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Delete RemotePCAccounts from the site.

## Related topics

[Get-BrokerRemotePCAccount](#)

[New-BrokerRemotePCAccount](#)

[Set-BrokerRemotePCAccount](#)

## Parameters

**-InputObject**<RemotePCAccount[]>

Specifies the RemotePCAccounts to delete.

Required?	true
Default Value	null
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.RemotePCAccount You can pipe the RemotePCAccounts to be deleted into this cmdlet.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerRemotePCAccount 42
Delete RemotePCAccount 42.
```

#### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerRemotePCAccount -OU 'any' | Remove-BrokerRemotePCAccount
Delete the 'any' OU RemotePCAccount.
```

#### ----- EXAMPLE 3 -----

```
C:\PS> Get-BrokerRemotePCAccount | Remove-BrokerRemotePCAccount
Delete all RemotePCAccounts.
```

# Remove-BrokerScope

May 10, 2016

Remove the specified catalog/desktop group from the given scope(s).

## Syntax

```
Remove-BrokerScope [-InputObject] <Scope[]> [-Catalog <Catalog>] [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerScope cmdlet is used to remove a catalog/desktop group object from the given scope(s).

To remove a catalog/desktop group from a scope you need permission to change the scopes of the catalog/desktop group.

If the catalog/desktop group is not in a specified scope, that scope will be silently ignored.

## Related topics

### Parameters

**-InputObject**<Scope[]>

Specifies the scopes to remove the object from.

Required?	true
Default Value	null
Accept Pipeline Input?	true (ByValue)

**-Catalog**<Catalog>

Specifies the catalog object to be removed.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

**-DesktopGroup**<DesktopGroup>

Specifies the desktop group object to be removed.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.Scope You can pipe scopes to Remove-BrokerScope.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerScope -Scope Chalfont,Cambourne -DesktopGroup "Win7 Desktops"
```

Removes the "Win7 Desktops" desktop group from both the Chalfont and Cambourne scopes.

##### ----- EXAMPLE 2 -----

```
C:\PS> 'Chalfont','Cambourne' | Remove-BrokerScope -DesktopGroup 'Win7 Desktops'
```

Removes the "Win7 Desktops" desktop group from both the Chalfont and Cambourne scopes.

# Remove-BrokerSessionLinger

May 10, 2016

Removes a session linger setting.

## Syntax

```
Remove-BrokerSessionLinger [-InputObject] <SessionLinger[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-BrokerSessionLinger [-DesktopGroupName] <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Remove-BrokerSessionLinger cmdlet is used to delete an existing session linger setting.

## Related topics

[New-BrokerSessionLinger](#)

[Get-BrokerSessionLinger](#)

[Set-BrokerSessionLinger](#)

## Parameters

**-InputObject**<SessionLinger[]>

The session linger setting to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-DesktopGroupName**<String>

The name of the desktop group whose session linger setting is to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.SessionLinger Session linger settings may be specified through pipeline input.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerSessionLinger | Remove-BrokerSessionLinger
```

Deletes every session linger setting for all desktop groups.



# Remove-BrokerSessionMetadata

May 10, 2016

Deletes Session Metadata from the Session objects

## Syntax

```
Remove-BrokerSessionMetadata [-InputObject] <Session[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Remove-BrokerSessionMetadata cmdlet deletes Metadata from the Session objects.

## Related topics

## Parameters

**-InputObject**<Session[]>

Specifies the Session object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.BrokerSession You can pipe the Session to delete the metadata.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerSessionMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the Session whose instance is pointed by \$obj-Uid

#### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerSession | Remove-BrokerSessionMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the Session in the site

# Remove-BrokerSessionPreLaunch

May 10, 2016

Removes a session pre-launch setting.

## Syntax

```
Remove-BrokerSessionPreLaunch [-InputObject] <SessionPreLaunch[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-BrokerSessionPreLaunch [-DesktopGroupName] <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Remove-BrokerSessionPreLaunch cmdlet is used to delete an existing session pre-launch setting.

## Related topics

[New-BrokerSessionPreLaunch](#)

[Get-BrokerSessionPreLaunch](#)

[Set-BrokerSessionPreLaunch](#)

## Parameters

**-InputObject**<SessionPreLaunch[]>

The session pre-launch setting to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-DesktopGroupName**<String>

The name of the desktop group whose session pre-launch setting is to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.SessionPreLaunch Session pre-launch settings may be specified through pipeline input.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerSessionPreLaunch | Remove-BrokerSessionPreLaunch
```

Deletes every session pre-launch setting for all desktop groups.

# Remove-BrokerSiteMetadata

May 10, 2016

Deletes Site Metadata from the Site objects

## Syntax

```
Remove-BrokerSiteMetadata -Name <String> [[-InputObject] <Site[]>] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Remove-BrokerSiteMetadata cmdlet deletes Metadata from the Site objects.

## Related topics

## Parameters

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-InputObject**<Site[]>

Specifies the Site object's instance whose Metadata is to be deleted.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerSite You can pipe the Site to delete the metadata.

#### Return Values

None

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Remove-BrokerSiteMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the Site

# Remove-BrokerTag

May 10, 2016

Removes a tag from the system or clears a tag to object association.

## Syntax

```
Remove-BrokerTag [-InputObject] <Tag[]> [-Desktop <Desktop>] [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-BrokerTag [-Name] <String> [-Desktop <Desktop>] [-DesktopGroup <DesktopGroup>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Clears tag to object associations or delete tags from the system altogether.

To clear an association, supply one of the Application, DesktopGroup or PrivateDesktop parameters.

To delete the tag, and any associations between the tag and other objects in the database, specify the tag without any associated object parameters.

## Related topics

[Add-BrokerTag](#)

[Get-BrokerTag](#)

[New-BrokerTag](#)

[Rename-BrokerTag](#)

## Parameters

**-InputObject**<Tag[]>

Specifies one or more tag objects.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies a tag by name.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-Desktop**<Desktop>

Clears the association between the given tag and Desktop.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-DesktopGroup**<DesktopGroup>

Clears the association between the given tag and desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
-----------	-------



Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.Tag Tags may be specified through pipeline input.

## Return Values

None

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Remove-BrokerTag $tag -Desktop $desktop
```

Clears the association between a tag and a desktop. Note that the tag itself continues to exist as an object in the database.

### ----- EXAMPLE 2 -----

```
C:\PS> Remove-BrokerTag $tag
```

Deletes the tag object from the database and clears any associations that may exist between that tag and other objects.

# Remove-BrokerTagMetadata

May 10, 2016

Deletes Tag Metadata from the Tag objects

## Syntax

```
Remove-BrokerTagMetadata [-InputObject] <Tag[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Remove-BrokerTagMetadata cmdlet deletes Metadata from the Tag objects.

## Related topics

## Parameters

**-InputObject**<Tag[]>

Specifies the Tag object's instance whose Metadata is to be deleted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata to be deleted

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerTag You can pipe the Tag to delete the metadata.

#### Return Values

None

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Remove-BrokerTagMetadata -InputObject $obj-Uid -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for the Tag whose instance is pointed by \$obj-Uid

##### ----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerTag | Remove-BrokerTagMetadata -Name "MyMetadataName"
```

This command deletes the Metadata "MyMetadataName" key-value pair for all the Tag in the site

# Remove-BrokerUser

May 10, 2016

Remove broker user objects from another broker object

## Syntax

```
Remove-BrokerUser [-InputObject] <User[]> [-Application <Application>] [-SessionLinger <SessionLinger>] [-SessionPreLaunch <SessionPreLaunch>] [-Machine <Machine>] [-PrivateDesktop <PrivateDesktop>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-BrokerUser [-Name] <String> [-Application <Application>] [-SessionLinger <SessionLinger>] [-SessionPreLaunch <SessionPreLaunch>] [-Machine <Machine>] [-PrivateDesktop <PrivateDesktop>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-BrokerUser cmdlet removes broker user objects from another specified object, such as a broker private desktop, to which the user had previously been added.

## Related topics

[Add-BrokerUser](#)

[Get-BrokerUser](#)

## Parameters

**-InputObject**<User[]>

Specifies the user objects to remove.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the user objects to remove, based on their Name property.

Required?	true
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

### **-Application**<Application>

The application from which to remove the user

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByValue)

### **-SessionLinger**<SessionLinger>

The desktop group session linger setting from which to remove the user.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByValue)

### **-SessionPreLaunch**<SessionPreLaunch>

The desktop group session pre-launch setting from which to remove the user.

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByValue)

### **-Machine**<Machine>

The machine from which to remove the user

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByValue)

### **-PrivateDesktop**<PrivateDesktop>

The desktop from which to remove the user

Required?	false
Default Value	null
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.User You can pipe the users to be removed to Remove-BrokerUser.

#### Return Values

None

#### Notes

Specify one of the -Machine or -PrivateDesktop parameters only.

#### Examples

----- **EXAMPLE 1** -----

```
Remove-BrokerUser "DOMAIN\UserName" -PrivateDesktop "DOMAIN\MachineName"
```

Remove the assignment of the specified private desktop to the specified user.

# Rename-BrokerAccessPolicyRule

May 10, 2016

Renames a rule in the site's access policy.

## Syntax

```
Rename-BrokerAccessPolicyRule [-InputObject] <AccessPolicyRule[]> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Rename-BrokerAccessPolicyRule [-Name] <String> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Rename-BrokerAccessPolicyRule cmdlet renames a rule in the site's access policy. The Name property of the rule is changed.

An access policy rule defines a set of connection filters and access control rights relating to a desktop group. These allow fine-grained control of what access is granted to a desktop group based on details of, for example, a user's endpoint device, its address, and the user's identity.

## Related topics

[New-BrokerAccessPolicyRule](#)

[Get-BrokerAccessPolicyRule](#)

[Set-BrokerAccessPolicyRule](#)

[Remove-BrokerAccessPolicyRule](#)

## Parameters

**-InputObject**<AccessPolicyRule[]>

The access policy rule to be renamed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

The existing name of the access policy rule to be renamed.

Required?	true
-----------	------



Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-NewName**<String>

The new name for the access policy rule being renamed. The new name must not match that of any other existing rules in the policy.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host

name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.AccessPolicyRule The access policy rule to be renamed.

## Return Values

None or Citrix.Broker.Admin.SDK.AccessPolicyRule

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.AccessPolicyRule object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Rename-BrokerAccessPolicyRule 'Sales' -NewName 'TeleSales'
```

Renames the access policy rule called Sales to TeleSales. The new name of the rule must be unique in the access policy.

# Rename-BrokerAdminFolder

May 10, 2016

Renames a folder

## Syntax

```
Rename-BrokerAdminFolder [-InputObject <AdminFolder[]> [-NewName <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Rename-BrokerAdminFolder [-Name <String> [-NewName <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Rename-BrokerAdminFolder cmdlet renames a folder for organising objects for administration purposes (for example, Applications).

The following special characters are not allowed in the new FolderName: \ / ; : # . \* ? = < > | [ ] ( ) " ' `

## Related topics

[Get-BrokerAdminFolder](#)

[New-BrokerAdminFolder](#)

## Parameters

**-InputObject**<AdminFolder[]>

The folder(s) to be renamed

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

A pattern matching the names of folders to be renamed

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-NewName<String>**

The name the new folder(s) should have.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-PassThru<SwitchParameter>**

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-LoggingId<Guid>**

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.

Accept Pipeline Input?	false
------------------------	-------

## Input Type

Depends on parameter Parameters can be piped by property name.

## Return Values

None or Citrix.Broker.Admin.SDK.AdminFolder

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.AdminFolder object.

## Examples

### ----- **EXAMPLE 1** -----

Rename-BrokerAdminFolder F1\XXX\YYY

Renames the folder called XXX within the folder F1\ to YYY

# Rename-BrokerAppAssignmentPolicyRule

May 10, 2016

Renames an application rule in the site's assignment policy.

## Syntax

```
Rename-BrokerAppAssignmentPolicyRule [-InputObject] <AppAssignmentPolicyRule[]> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Rename-BrokerAppAssignmentPolicyRule [-Name] <String> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Rename-BrokerAppAssignmentPolicyRule cmdlet renames an application rule in the site's assignment policy. The Name property of the rule is changed.

An application rule in the assignment policy defines the users who are entitled to a self-service persistent machine assignment from the rule's desktop group; once assigned the machine can run one or more applications published from the group.

## Related topics

[New-BrokerAppAssignmentPolicyRule](#)

[Get-BrokerAppAssignmentPolicyRule](#)

[Set-BrokerAppAssignmentPolicyRule](#)

[Remove-BrokerAppAssignmentPolicyRule](#)

## Parameters

**-InputObject** <AppAssignmentPolicyRule[]>

The application rule in the assignment policy to be renamed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name** <String>

The existing name of the application rule in the assignment policy to be renamed.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-NewName**<String>

The new name of the application rule in the assignment policy being renamed. The new name must not match that of any other existing rule in the policy (irrespective of whether it is a desktop or application rule).

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host

name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.AppAssignmentPolicyRule The application rule in the assignment policy being renamed.

## Return Values

None or Citrix.Broker.Admin.SDK.AppAssignmentPolicyRule

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.AppAssignmentPolicyRule object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Rename-BrokerAppAssignmentPolicyRule 'Offshore' -NewName 'Remote Workers'
```

Renames the application rule in the assignment policy called Offshore to Remote Workers. The new name of the rule must be unique in the assignment policy.



# Rename-BrokerAppEntitlementPolicyRule

May 10, 2016

Renames an application rule in the site's entitlement policy.

## Syntax

```
Rename-BrokerAppEntitlementPolicyRule [-InputObject] <AppEntitlementPolicyRule[]> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Rename-BrokerAppEntitlementPolicyRule [-Name] <String> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Rename-BrokerAppEntitlementPolicyRule cmdlet renames an application rule in the site's entitlement policy. The Name property of the rule is changed.

An application rule in the entitlement policy defines the users who are allowed per-session access to a machine to run one or more applications published from the rule's desktop group.

## Related topics

[New-BrokerAppEntitlementPolicyRule](#)

[Get-BrokerAppEntitlementPolicyRule](#)

[Set-BrokerAppEntitlementPolicyRule](#)

[Remove-BrokerAppEntitlementPolicyRule](#)

## Parameters

**-InputObject** <AppEntitlementPolicyRule[]>

The application rule in the entitlement policy to be renamed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name** <String>

The existing name of the application rule in the entitlement policy to be renamed.

Required?	true

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-NewName**<String>

The new name of the application rule in the entitlement policy being renamed. The new name must not match that of any other existing rule in the policy (irrespective of whether it is a desktop or application rule).

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.AppEntitlementPolicyRule The application rule in the entitlement policy being renamed.

## Return Values

None or Citrix.Broker.Admin.SDK.AppEntitlementPolicyRule

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.AppEntitlementPolicyRule object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Rename-BrokerAppEntitlementPolicyRule 'Prod Dev' -NewName 'Product Development'
```

Renames the application rule in the entitlement policy called Prod Dev to Product Development. The new name of the rule must be unique in the entitlement policy.

# Rename-BrokerApplication

May 10, 2016

Renames an application.

## Syntax

```
Rename-BrokerApplication [-InputObject] <Application[]> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Rename-BrokerApplication [-Name] <String> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Rename-BrokerApplication cmdlet changes the administrative name of an application. An application cannot have the same name as another application.

Renaming an application does not alter its published name. To change the name with which this application appears to end-users, set a new value for the PublishedName property using the Set-BrokerApplication cmdlet.

Renaming an application does not alter its BrowserName. If the BrowserName property also needs to be changed, use the Set-BrokerApplication cmdlet to modify it.

## Related topics

[New-BrokerApplication](#)

[Set-BrokerApplication](#)

[Get-BrokerApplication](#)

[Remove-BrokerApplication](#)

## Parameters

**-InputObject** <Application[]>

Specifies the application to rename.

Required?	true
Default Value	null
Accept Pipeline Input?	true (ByValue)

**-Name** <String>

Specifies the name of the application to rename.

Required?	true
Default Value	null
Accept Pipeline Input?	true (ByPropertyName)

#### **-NewName**<String>

Specifies the new name for the application.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host

name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.Application You can pipe applications to Rename-BrokerApplication.

## Return Values

None or Citrix.Broker.Admin.SDK.Application

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.Application object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Rename-BrokerApplication -Name "Old Name" -NewName "New Name"
```

Renames the application with name "Old Name" to "New Name".

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerApplication -Uid 1 | Rename-BrokerApplication -NewName "New Name" -PassThru
```

Renames application with the Uid 1 to "New Name", showing the result.

# Rename-BrokerAssignmentPolicyRule

May 10, 2016

Renames a desktop rule in the site's assignment policy.

## Syntax

```
Rename-BrokerAssignmentPolicyRule [-InputObject] <AssignmentPolicyRule[]> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Rename-BrokerAssignmentPolicyRule [-Name] <String> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Rename-BrokerAssignmentPolicyRule cmdlet renames a desktop rule in the site's assignment policy. The Name property of the rule is changed.

A desktop rule in the assignment policy defines the users who are entitled to self-service persistent machine assignments from the rule's desktop group. A rule defines how many machines a user is allowed from the group for delivery of full desktop sessions.

## Related topics

[New-BrokerAssignmentPolicyRule](#)

[Get-BrokerAssignmentPolicyRule](#)

[Set-BrokerAssignmentPolicyRule](#)

[Remove-BrokerAssignmentPolicyRule](#)

## Parameters

**-InputObject** <AssignmentPolicyRule[]>

The desktop rule in the assignment policy to be renamed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name** <String>

The existing name of the desktop rule in the assignment policy to be renamed.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-NewName**<String>

The new name of the desktop rule in the assignment policy being renamed. The new name must not match that of any other existing rule in the policy (irrespective of whether it is a desktop or application rule).

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host



name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.AssignmentPolicyRule The desktop rule in the assignment policy being renamed.

### Return Values

None, or Citrix.Broker.Admin.SDK.AssignmentPolicyRule

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.AssignmentPolicyRule object.

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Rename-BrokerAssignmentPolicyRule 'Offshore' -NewName 'Remote Workers'
```

Renames the desktop rule in the assignment policy called Offshore to Remote Workers. The new name of the rule must be unique in the assignment policy.

# Rename-BrokerCatalog

May 10, 2016  
Renames a catalog.

## Syntax

Rename-BrokerCatalog [-InputObject] <Catalog[]> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Rename-BrokerCatalog [-Name] <String> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The Rename-BrokerCatalog cmdlet changes the name of a catalog. A catalog cannot have the same name as another catalog.

The following special characters are not allowed in a catalog name: \ / ; : # . \* ? = < > | [ ] ( ) " ' `

## Related topics

- [Get-BrokerCatalog](#)
- [New-BrokerCatalog](#)
- [Remove-BrokerCatalog](#)
- [Set-BrokerCatalog](#)

## Parameters

**-InputObject**<Catalog[]>

Specifies the catalog to rename.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the catalog to rename.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-NewName**<String>

Specifies the new name of the catalog.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
-----------	-------

Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.Catalog You can pipe catalogs to Rename-BrokerCatalog.

## Return Values

None or Citrix.Broker.Admin.SDK.Catalog

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.Catalog object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Rename-BrokerCatalog -Name "Old Name" -NewName "New Name"
Renames the catalog with the name "Old Name" to "New Name".
```

### ----- EXAMPLE 2 -----

```
C:\PS> c:\$catalog = Get-BrokerCatalog -Name "Old Name"
C:\PS> Rename-BrokerCatalog -InputObject $catalog -NewName "New Name"
Renames the catalog with the name "Old Name" to "New Name".
```

# Rename-BrokerDesktopGroup

May 10, 2016

Renames a desktop group.

## Syntax

```
Rename-BrokerDesktopGroup [-InputObject] <DesktopGroup[]> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Rename-BrokerDesktopGroup [-Name] <String> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Rename-BrokerDesktopGroup cmdlet changes the name of a desktop group. A desktop group cannot have the same name as another desktop group.

## Related topics

[Get-BrokerDesktopGroup](#)

[New-BrokerDesktopGroup](#)

[Set-BrokerDesktopGroup](#)

[Remove-BrokerDesktopGroup](#)

## Parameters

**-InputObject** <DesktopGroup[]>

Specifies the desktop group to rename.

Required?	true
Default Value	null
Accept Pipeline Input?	true (ByValue)

**-Name** <String>

Specifies the name of the desktop group to rename.

Required?	true
Default Value	null

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

### **-NewName**<String>

Specifies the new name that the desktop group will have.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.DesktopGroup You can pipe desktop groups to Rename-BrokerDesktopGroup.

## Return Values

None or Citrix.Broker.Admin.SDK.DesktopGroup

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.DesktopGroup object.

## Notes

Renaming a desktop group does not alter its published name. If you need to change the name with which this desktop group appears to end-users, set a new value for the PublishedName property using the Set-BrokerDesktopGroup cmdlet.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Rename-BrokerDesktopGroup -Name "Old Name" -NewName "New Name"
```

Renames desktop group with the name "Old Name" to "New Name".

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerDesktopGroup -Uid 1 | Rename-BrokerDesktopGroup -NewName "New Name" -PassThru
```

Renames desktop group with the Uid 1 to "New Name", showing the result.

# Rename-BrokerEntitlementPolicyRule

May 10, 2016

Renames a desktop rule in the site's entitlement policy.

## Syntax

```
Rename-BrokerEntitlementPolicyRule [-InputObject] <EntitlementPolicyRule[]> [-NewName] <String> [-PassThru] [-LoggingId
<Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Rename-BrokerEntitlementPolicyRule [-Name] <String> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-
AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Rename-BrokerEntitlementPolicyRule cmdlet renames a desktop rule in the site's entitlement policy. The Name property of the rule is changed.

A desktop rule in the entitlement policy defines the users who are allowed per-session access to a machine from the rule's associated desktop group to run a full desktop session.

## Related topics

[New-BrokerEntitlementPolicyRule](#)

[Get-BrokerEntitlementPolicyRule](#)

[Set-BrokerEntitlementPolicyRule](#)

[Remove-BrokerEntitlementPolicyRule](#)

## Parameters

**-InputObject**<EntitlementPolicyRule[]>

The desktop rule in the entitlement policy to be renamed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

The existing name of the desktop rule in the entitlement policy to be renamed.

Required?	true



Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-NewName**<String>

The new name of the desktop rule in the entitlement policy being renamed. The new name must not match that of any other existing rule in the policy (irrespective of whether it is a desktop or application rule).

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.EntitlementPolicyRule The desktop rule in the entitlement policy being renamed.

## Return Values

None or Citrix.Broker.Admin.SDK.EntitlementPolicyRule

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.EntitlementPolicyRule object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Rename-BrokerEntitlementPolicyRule 'Prod Dev' -NewName 'Product Development'
```

Renames the desktop rule in the entitlement policy called Prod Dev to Product Development. The new name of the rule must be unique in the entitlement policy.

# Rename-BrokerMachineConfiguration

May 10, 2016

Renames a machine configuration.

## Syntax

```
Rename-BrokerMachineConfiguration [-InputObject] <MachineConfiguration[]> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Rename-BrokerMachineConfiguration [-Name] <String> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Rename-MachineConfiguration cmdlet changes the name of a machine configuration. A machine configuration cannot have the same name as another machine configuration associated with the same slot.

## Related topics

[New-BrokerMachineConfiguration](#)

[Get-BrokerMachineConfiguration](#)

[Set-BrokerMachineConfiguration](#)

[Remove-BrokerMachineConfiguration](#)

[Add-BrokerMachineConfiguration](#)

## Parameters

**-InputObject** <MachineConfiguration[]>

Machine configuration to rename.

Required?	true
Default Value	None
Accept Pipeline Input?	true (ByValue)

**-Name** <String>

Current name of machine configuration.

Required?	true
Default Value	None

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-NewName**<String>

New name for machine configuration. This may have the form "ConfigurationSlotName\MachineConfigurationName" or "MachineConfigurationName". If the "ConfigurationSlotName" is provided it must match the name of the configuration slot that the machine configuration is associated with.

Required?	true
Default Value	None
Accept Pipeline Input?	false

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.MachineConfiguration Machine configuration to rename.

## Return Values

None or Citrix.Broker.Admin.SDK.MachineConfiguration

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.MachineConfiguration object.

## Notes

The configuration slot can not be changed. Thus the left term of the Name and NewName must match.

## Examples

### ----- EXAMPLE 1 -----

Rename-BrokerMachineConfiguration -Name "UPM\All Departments" -NewName "UPM\Finance Department"  
 Renames the machine configuration named "UPM\All Departments" to "UPM\Finance Department".

### ----- EXAMPLE 2 -----

Rename-BrokerMachineConfiguration -Name "UPM\All Departments" -NewName "Finance Department"  
 Renames the machine configuration named "UPM\All Departments" to "UPM\Finance Department".

# Rename-BrokerPowerTimeScheme

May 10, 2016

Changes the name of an existing power time scheme.

## Syntax

```
Rename-BrokerPowerTimeScheme [-InputObject] <PowerTimeScheme[]> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Rename-BrokerPowerTimeScheme [-Name] <String> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Rename-BrokerPowerTimeScheme cmdlet renames a particular power time scheme.

Each power time scheme is associated with a particular desktop group, and covers one or more days of the week, defining which hours of those days are considered peak times and which are off-peak times. In addition, the time scheme defines a pool size value for each hour of the day for the days of the week covered by the time scheme. No one desktop group can be associated with two or more time schemes that cover the same day of the week.

For more information about the power policy mechanism and pool size management, see 'help about\_Broker\_PowerManagement'.

## Related topics

[Get-BrokerPowerTimeScheme](#)

[Set-BrokerPowerTimeScheme](#)

[New-BrokerPowerTimeScheme](#)

[Remove-BrokerPowerTimeScheme](#)

## Parameters

**-InputObject**<PowerTimeScheme[]>

The power time scheme to be renamed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

The current name of the power time scheme to be renamed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-NewName**<String>

The new name to be applied to the power time scheme.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.PowerTimeScheme The power time scheme to be renamed.

### Return Values

None or Citrix.Broker.Admin.SDK.PowerTimeScheme

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.PowerTimeScheme object.

### Examples

#### ----- **EXAMPLE 1** -----

```
C:\PS> Rename-BrokerPowerTimeScheme -Name 'Development Weekdays' -NewName 'Dev Week'
```

Renames the power time scheme named 'Development Weekdays' to 'Dev Week'.



# Rename-BrokerTag

May 10, 2016

Rename one or more tags.

## Syntax

```
Rename-BrokerTag [-InputObject] <Tag[]> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Rename-BrokerTag [-Name] <String> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Renames one or more tags with the supplied name.

## Related topics

[Add-BrokerTag](#)

[Get-BrokerTag](#)

[New-BrokerTag](#)

[Remove-BrokerTag](#)

## Parameters

**-InputObject**<Tag[]>

Specifies one or more tag objects to rename.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Identifies tags to be renamed by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-NewName**<String>

Specifies new name for the tags.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.

Accept Pipeline Input?	false
------------------------	-------

## Input Type

Citrix.Broker.Admin.SDK.Tag The tag to rename can be piped into this cmdlet.

## Return Values

None or Citrix.Broker.Admin.SDK.Tag

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.Tag object.

## Notes

Note that when renaming a tag, its UUID remains the same and any associations are maintained.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Rename-BrokerTag -Name 'OldName' -NewName 'ReplacementName'
```

Renames tags with the name 'OldName' to 'ReplacementName'.

# Reset-BrokerLicensingConnection

May 10, 2016

Resets the broker's license server connection.

## Syntax

```
Reset-BrokerLicensingConnection [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Reset-BrokerLicensingConnection cmdlet resets the broker's connection to the license server.

Licensing changes resulting from new license files or alterations to the site-level licensing properties don't become effective immediately. There will typically be a delay as the changes are propagated across the site based on the scheduling of refresh logic built into the controllers and the license server.

Resetting the connection causes the list of available licenses for the connection to be updated. After adding licenses or changing the site-level licensing properties you can run Reset-BrokerLicensingConnection to ensure that the broker can access the new licenses immediately.

Each broker service instance holds its own connection to the license server. In order for the licensing changes to be applied immediately throughout the XenDesktop site this command needs to be run on every controller in the site.

## Related topics

[Get-BrokerSite](#)

[Set-BrokerSite](#)

## Parameters

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

None You cannot pipe input into this cmdlet.

### Return Values

None

### Examples

#### ----- **EXAMPLE 1** -----

C:\PS> Reset-BrokerLicensingConnection  
Reset the broker's license server connection.

# Reset-BrokerServiceGroupMembership

May 10, 2016  
Reloads the access permissions and configuration service locations for the Broker Service.

## Syntax

```
Reset-BrokerServiceGroupMembership -ConfigServiceInstance <PSObject[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Enables you to reload Broker Service access permissions and configuration service locations. The Reset-BrokerServiceGroupMembership command must be run on at least one instance of the service type (Broker) after installation and registration with the configuration service. Without this operation, the Broker services will be unable to communicate with other services in the XenDesktop deployment. When the command is run, the services are updated when additional services are added to the deployment, provided that the configuration service is not stopped. The Reset-BrokerServiceGroupMembership command can be run again to refresh this information if automatic updates do not occur when new services are added to the deployment. If more than one configuration service instance is passed to the command, the first instance that meets the expected service type requirements is used.

## Related topics

- [Get-BrokerServiceInstance](#)
- [Get-BrokerServiceStatus](#)

## Parameters

### -ConfigServiceInstance<PSObject[]>

Specifies the configuration service instance object that represents the service instance for the type 'InterService' that references a configuration service for the deployment.

Required?	true
Default Value	LocalHost
Accept Pipeline Input?	true (ByValue)

### -LoggingId<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### -AdminAddress<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

--	--

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

`Citrix.Broker.Sdk.ServiceInstance[]` Service instances containing a `ServiceInstance` object that refers to the central configuration service interservice interface can be piped to the `Reset-BrokerServiceGroupMembership` command.

## Return Values

`Citrix.Broker.Sdk.ServiceInstance`

`Reset-BrokerServiceGroupMembership` returns opaque objects containing Configuration Service instances that are used by the Broker Service instance.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### NoSuitableServiceInstance

None of the supplied service instance objects were suitable for resetting service group membership.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config | Reset-BrokerServiceGroupMembership
```

Reset the service group membership for a service in a deployment where the configuration service is configured and running on the same machine as the service.

----- **EXAMPLE 2** -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config -AdminAddress OtherServer.example.com | Reset-BrokerServiceGroupmembership
```

Reset the service group membership for a service in a deployment where the configuration service that is configured and running on a machine named 'OtherServer.example.com'.



# Send-BrokerSessionMessage

May 10, 2016

Sends a message to a session.

## Syntax

```
Send-BrokerSessionMessage [-InputObject] <Session[]> [-MessageStyle] <SendMessageStyle> [-Title] <String> [-Text] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Generates a message box in the target session(s).

## Related topics

[Get-BrokerSession](#)

## Parameters

**-InputObject** <Session[]>

The target session(s) to send the message to.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-MessageStyle** <SendMessageStyle>

The style of message box to use (valid values are Critical, Question, Exclamation, or Information).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Title** <String>

Text to display in the messagebox title bar.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByValue)
------------------------	----------------

#### **-Text**<String>

The message to display.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.Session The session to which to send the message can be piped in.

#### Return Values

None

#### Notes

Sessions can be passed as the InputObject parameter as either session objects or their numeric Uids.

This operation is non-blocking and returns before it completes. The operation, however, is unlikely to fail unless there are communication problems between controller and machine, if bad arguments are passed to the cmdlet itself or if the machine cannot successfully execute the operation.

The transient nature of sessions means that the list of session objects or UIDs supplied to Send-BrokerSessionMessage could consist of valid and invalid sessions. Invalid sessions are detected and disregarded and the send message operation is invoked on the machines running valid sessions.

The system can fail to invoke the operation if the machine is not in an appropriate state or if there are problems in communicating with the machine. When an operation is invoked the system detects if the operation was initiated successfully or not by the session. As this operation is non-blocking the system doesn't detect or report whether the operation ultimately succeeded or failed after its successful initialization in the session.

Operation failures are reported through the broker SDK error handling mechanism (see about\_Broker\_ErrorHandling). In the event of errors the SdkErrorRecord error status code is set to SessionOperationFailed and its error data dictionary is populated with the following entries:

- o OperationsAttemptedCount: The number of operations attempted.
- o OperationsFailedCount - The number of failed operations.
- o OperationsSucceededCount - The number of successfully executed operations.
- o UnresolvedSessionFailuresCount - The number of operations that failed due to invalid sessions being supplied.
- o OperationInvocationFailuresCount - The number of operations that failed because they could not be invoked in the session.
- o DesktopExecutionFailuresCount - The number of operations that failed because they could not be successfully executed in the session.

The SdkErrorRecord message will also display the number of attempted, failed and successful operations in the following format:

```
"Session operation error - attempted:<OperationsAttemptedCount>, failed:<OperationsFailedCount>, succeeded:
<OperationsSucceededCount>"
```

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> $sessions = Get-BrokerSession -UserName MYDOMAIN*
C:\PS> Send-BrokerSessionMessage $sessions -MessageStyle Information -Title TestTitle -Text TestMessage
Sends a message to all the sessions for any user in MYDOMAIN.
```

### ----- EXAMPLE 2 -----

```
C:\PS> $desktop = Get-BrokerDesktop -Uid 1
C:\PS> Send-BrokerSessionMessage $desktop.SessionUid -MessageStyle Information -Title TestTitle -Text TestMessage
Sends a message to the session on the desktop with Uid 1.
```

### ----- EXAMPLE 3 -----

```
C:\PS> trap [Citrix.Broker.Admin.SDK.SdkOperationException]
```

```
C:\PS> {
C:\PS> write $("Exception name = " + $_.Exception.GetType().FullName)
C:\PS> write $("SdkOperationException.Status = " + $_.Exception.Status)
C:\PS> write $("SdkOperationException.ErrorData=")
C:\PS> $_.Exception.ErrorData
C:\PS>
C:\PS> write $("SdkOperationException.InnerException = " + $_.Exception.InnerException)
C:\PS> $_.Exception.InnerException
C:\PS> continue
C:\PS> }
C:\PS>
C:\PS> Send-BrokerSessionMessage -InputObject 10,11,12 -MessageStyle Information -Title "message title" -Text "message text"
Trap and display error information.
```

# Set-BrokerAccessPolicyRule

May 10, 2016  
Modifies an existing rule in the site's access policy.

## Syntax

```
Set-BrokerAccessPolicyRule [-InputObject] <AccessPolicyRule[]> [-PassThru] [-AddExcludedClientIPs <IPAddressRange[]>] [-AddExcludedClientNames <String[]>] [-AddExcludedSmartAccessTags <String[]>] [-AddExcludedUsers <User[]>] [-AddIncludedClientIPs <IPAddressRange[]>] [-AddIncludedClientNames <String[]>] [-AddIncludedSmartAccessTags <String[]>] [-AddIncludedUsers <User[]>] [-AllowedConnections <AllowedConnection>] [-AllowedProtocols <String[]>] [-AllowedUsers <AllowedUser>] [-AllowRestart <Boolean>] [-Description <String>] [-Enabled <Boolean>] [-ExcludedClientIPFilterEnabled <Boolean>] [-ExcludedClientIPs <IPAddressRange[]>] [-ExcludedClientNameFilterEnabled <Boolean>] [-ExcludedClientNames <String[]>] [-ExcludedSmartAccessFilterEnabled <Boolean>] [-ExcludedSmartAccessTags <String[]>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-HdxSslEnabled <Boolean>] [-IncludedClientIPFilterEnabled <Boolean>] [-IncludedClientIPs <IPAddressRange[]>] [-IncludedClientNameFilterEnabled <Boolean>] [-IncludedClientNames <String[]>] [-IncludedSmartAccessFilterEnabled <Boolean>] [-IncludedSmartAccessTags <String[]>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-RemoveExcludedClientIPs <IPAddressRange[]>] [-RemoveExcludedClientNames <String[]>] [-RemoveExcludedSmartAccessTags <String[]>] [-RemoveExcludedUsers <User[]>] [-RemoveIncludedClientIPs <IPAddressRange[]>] [-RemoveIncludedClientNames <String[]>] [-RemoveIncludedSmartAccessTags <String[]>] [-RemoveIncludedUsers <User[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAccessPolicyRule [-Name] <String> [-PassThru] [-AddExcludedClientIPs <IPAddressRange[]>] [-AddExcludedClientNames <String[]>] [-AddExcludedSmartAccessTags <String[]>] [-AddExcludedUsers <User[]>] [-AddIncludedClientIPs <IPAddressRange[]>] [-AddIncludedClientNames <String[]>] [-AddIncludedSmartAccessTags <String[]>] [-AddIncludedUsers <User[]>] [-AllowedConnections <AllowedConnection>] [-AllowedProtocols <String[]>] [-AllowedUsers <AllowedUser>] [-AllowRestart <Boolean>] [-Description <String>] [-Enabled <Boolean>] [-ExcludedClientIPFilterEnabled <Boolean>] [-ExcludedClientIPs <IPAddressRange[]>] [-ExcludedClientNameFilterEnabled <Boolean>] [-ExcludedClientNames <String[]>] [-ExcludedSmartAccessFilterEnabled <Boolean>] [-ExcludedSmartAccessTags <String[]>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-HdxSslEnabled <Boolean>] [-IncludedClientIPFilterEnabled <Boolean>] [-IncludedClientIPs <IPAddressRange[]>] [-IncludedClientNameFilterEnabled <Boolean>] [-IncludedClientNames <String[]>] [-IncludedSmartAccessFilterEnabled <Boolean>] [-IncludedSmartAccessTags <String[]>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-RemoveExcludedClientIPs <IPAddressRange[]>] [-RemoveExcludedClientNames <String[]>] [-RemoveExcludedSmartAccessTags <String[]>] [-RemoveExcludedUsers <User[]>] [-RemoveIncludedClientIPs <IPAddressRange[]>] [-RemoveIncludedClientNames <String[]>] [-RemoveIncludedSmartAccessTags <String[]>] [-RemoveIncludedUsers <User[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerAccessPolicyRule cmdlet modifies an existing rule in the site's access policy.

An access policy rule defines a set of connection filters and access control rights relating to a desktop group. These allow fine-grained control of what access is granted to a desktop group based on details of, for example, a user's endpoint device, its address, and the user's identity.

Changing a rule does not affect existing user sessions, but it may result in users being unable to launch new sessions, or reconnect to disconnected sessions if the change removes access to the desktop group delivering those sessions.

## Related topics

[New-BrokerAccessPolicyRule](#)

[Get-BrokerAccessPolicyRule](#)

[Rename-BrokerAccessPolicyRule](#)

[Remove-BrokerAccessPolicyRule](#)

## Parameters

**-InputObject**<AccessPolicyRule[]>

The access policy rule to be modified.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

The name of the access policy rule to be modified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-AddExcludedClientIPs**<IPAddressRange[]>

Adds the specified user device IP addresses to the excluded client IP address filter of the rule.

See the ExcludedClientIPs parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AddExcludedClientNames**<String[]>

Adds the specified user device names to the excluded client names filter of the rule.

See the ExcludedClientNames parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AddExcludedSmartAccessTags**<String[]>

Adds the specified SmartAccess tags to the excluded SmartAccess tags filter of the rule.

See the ExcludedSmartAccessTags parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AddExcludedUsers<User[]>**

Adds the specified users and groups to the excluded users filter of the rule.

See the ExcludedUsers parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AddIncludedClientIPs<IPAddressRange[]>**

Adds the specified user device IP addresses to the included client IP address filter of the rule.

See the IncludedClientIPs parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AddIncludedClientNames<String[]>**

Adds the specified user device names to the included client names filter of the rule.

See the IncludedClientNames parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AddIncludedSmartAccessTags<String[]>**

Adds the specified SmartAccess tags to the included SmartAccess tags filter of the rule.

See the IncludedSmartAccessTags parameter for more information.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-AddIncludedUsers<User[]>**

Adds the specified users and groups to the included users filter of the rule.

See the IncludedUsers parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AllowedConnections<AllowedConnection>**

Changes whether connections must be local or via Access Gateway, and if so whether specified SmartAccess tags must be provided by Access Gateway with the connection. This property forms part of the included SmartAccess tags filter.

Valid values are Filtered, NotViaAG, and ViaAG.

For a detailed description of this property see "help about\_Broker\_AccessPolicy".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AllowedProtocols<String[]>**

Changes the protocols (for example HDX, RDP) available to the user for sessions delivered from the rule's desktop group. If the user gains access to a desktop group by multiple rules, the allowed protocol list is the combination of the protocol lists from all those rules.

If the protocol list is empty, access to the desktop group is implicitly denied.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AllowedUsers<AllowedUser>**

Changes the behavior of the included users filter of the rule. This can restrict access to a list of named users or groups, allow access to any authenticated user, any user (whether authenticated or not), or only non-authenticated users. For a detailed description of this property see "help about\_Broker\_AccessPolicy".

Valid values are Filtered, AnyAuthenticated, Any, AnonymousOnly and FilteredOrAnonymous.

Required?	false
Default Value	
Accept Pipeline Input?	false



--	--

#### **-AllowRestart**<Boolean>

Changes whether the user can restart sessions delivered from the rule's desktop group. Session restart is handled as follows: For sessions on single-session power-managed machines, the machine is powered off, and a new session launch request made; for sessions on multi-session machines, a logoff request is issued to the session, and a new session launch request made; otherwise the property is ignored.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Description**<String>

Changes the description of the rule. The text is purely informational for the administrator; it is never visible to the end user.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Enabled**<Boolean>

Changes whether the rule is enabled or disabled. A disabled rule is ignored when evaluating the site's access policy.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedClientIPFilterEnabled**<Boolean>

Changes whether the excluded client IP address filter is enabled or disabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedClientIPs**<IPAddressRange[]>

Changes the IP addresses of user devices explicitly denied access to the rule's desktop group. Addresses can be specified as simple numeric addresses or as subnet masks (for example, 10.40.37.5 or 10.40.0.0/16). This property forms part of the excluded client IP address filter.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ExcludedClientNameFilterEnabled**<Boolean>

Changes whether the excluded client names filter is enabled or disabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedClientNames**<String[]>

Changes which names of user devices are explicitly denied access to the rule's desktop group. This property forms part of the excluded client names filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedSmartAccessFilterEnabled**<Boolean>

Changes whether the excluded SmartAccess tags filter is enabled or disabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedSmartAccessTags**<String[]>

Changes which SmartAccess tags explicitly deny access to the rule's desktop group if any occur in those provided by Access Gateway with the user's connection. This property forms part of the excluded SmartAccess tags filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedUserFilterEnabled**<Boolean>

Changes whether the excluded users filter is enabled or disabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ExcludedUsers<User[]>**

Changes which users and groups are explicitly denied access to the rule's desktop group. This property forms part of the excluded users filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-HdxSslEnabled<Boolean>**

Indicates whether SSL encryption is enabled for sessions delivered from the rule's desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedClientIPFilterEnabled<Boolean>**

Changes whether the included client IP address filter is enabled or disabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedClientIPs<IPAddressRange[]>**

Changes which IP addresses of user devices allowed access to the rule's desktop group. Addresses can be specified as simple numeric addresses or as subnet masks (for example, 10.40.37.5 or 10.40.0.0/16). This property forms part of the included client IP address filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedClientNameFilterEnabled<Boolean>**

Changes whether the included client name filter is enabled or disabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-IncludedClientNames<String[]>**

Changes which names of user devices are allowed access to the rule's desktop group. This property forms part of the included client names filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedSmartAccessFilterEnabled<Boolean>**

Changes whether the included SmartAccess tags filter is enabled or disabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedSmartAccessTags<String[]>**

Changes which SmartAccess tags grant access to the rule's desktop group if any occur in those provided by Access Gateway with the user's connection. This property forms part of the excluded SmartAccess tags filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedUserFilterEnabled<Boolean>**

Changes whether the included users filter is enabled or disabled. If the filter is disabled, it is ignored when the access policy rule is evaluated.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedUsers<User[]>**

Changes which users and groups are granted access to the rule's desktop group. This property forms part of the included users filter.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-RemoveExcludedClientIPs**<IPAddressRange[]>

Removes the specified user device IP addresses from the excluded client IP address filter of the rule.

See the ExcludedClientIPs parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RemoveExcludedClientNames**<String[]>

Removes the specified user device names from the excluded client names filter of the rule.

See the ExcludedClientNames parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RemoveExcludedSmartAccessTags**<String[]>

Removes the specified SmartAccess tags from the excluded SmartAccess tags filter of the rule.

See the ExcludedSmartAccessTags parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RemoveExcludedUsers**<User[]>

Removes the specified users and groups from the excluded users filter of the rule.

See the ExcludedUsers parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RemoveIncludedClientIPs**<IPAddressRange[]>

Removes the specified user device IP addresses from the included client IP address filter of the rule.

See the IncludedClientIPs parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-RemoveIncludedClientNames<String[]>**

Removes the specified client names from the included client names filter of the rule.

See the IncludedClientNames parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-RemoveIncludedSmartAccessTags<String[]>**

Removes the specified SmartAccess tags from the included SmartAccess tags filter of the rule.

See the IncludedSmartAccessTags parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-RemoveIncludedUsers<User[]>**

Removes the specified users and groups from the included users filter of the rule.

See the IncludedUsers parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LoggingId<Guid>**

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.AccessPolicyRule The access policy rule to be modified.

#### Return Values

None, or Citrix.Broker.Admin.SDK.AccessPolicyRule

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.AccessPolicyRule object.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Set-BrokerAccessPolicyRule 'Temp Staff' -AddIncludedUsers office\contractors
```

Adds user group OFFICE\contractors to the Temp Staff access policy rule. The resources that the group can access are dependent on the existing properties of the rule in addition to the site's assignment and entitlement policies.

##### ----- **EXAMPLE 2** -----

```
C:\PS> Set-BrokerAccessPolicyRule 'Temp Staff' -ExcludedClientIPFilterEnabled $true -AddExcludedClientIPs '10.15.0.0/16' -AllowedConnections ViaAG
```

Modifies the Temp Staff access policy rule to remove access to any user device with an IP address matching 10.15.0.0/16, and requires that all connections by the rule must come through Access Gateway (assuming that the included SmartAccess tags filter is enabled).

# Set-BrokerAccessPolicyRuleMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for AccessPolicyRule

## Syntax

```
Set-BrokerAccessPolicyRuleMetadata [-AccessPolicyRuleId] <Int32> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAccessPolicyRuleMetadata [-AccessPolicyRuleId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAccessPolicyRuleMetadata [-AccessPolicyRuleId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAccessPolicyRuleMetadata [-InputObject] <AccessPolicyRule[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAccessPolicyRuleMetadata [-InputObject] <AccessPolicyRule[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAccessPolicyRuleMetadata [-AccessPolicyRuleName] <String> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAccessPolicyRuleMetadata [-AccessPolicyRuleName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerAccessPolicyRuleMetadata cmdlet creates/updates metadata key-value pairs for AccessPolicyRule. The AccessPolicyRule can be specified by InputObject or piping.

## Related topics

## Parameters

**-AccessPolicyRuleId**<Int32>

Specifies the AccessPolicyRule object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<AccessPolicyRule[]>



Specifies the AccessPolicyRule objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-AccessPolicyRuleName**<String>

Specifies the AccessPolicyRule object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

Citrix.Broker.Admin.SDK.BrokerAccessPolicyRule You can pipe the AccessPolicyRule to hold the new or updated metadata.

## Return Values

None or Citrix.Broker.Admin.SDK.BrokerAccessPolicyRule

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerAccessPolicyRule object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerAccessPolicyRuleMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the AccessPolicyRule whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerAccessPolicyRule | Set-BrokerAccessPolicyRuleMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the AccessPolicyRule in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerAccessPolicyRuleMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the AccessPolicyRule in the site whose name is 'objname'

# Set-BrokerAdminFolderMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for AdminFolder

## Syntax

```
Set-BrokerAdminFolderMetadata [-AdminFolderId] <Int32> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAdminFolderMetadata [-AdminFolderId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAdminFolderMetadata [-AdminFolderId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAdminFolderMetadata [-InputObject] <AdminFolder[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAdminFolderMetadata [-InputObject] <AdminFolder[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAdminFolderMetadata [-AdminFolderName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAdminFolderMetadata [-AdminFolderName] <String> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerAdminFolderMetadata cmdlet creates/updates metadata key-value pairs for AdminFolder. The AdminFolder can be specified by InputObject or piping.

## Related topics

## Parameters

**-AdminFolderId**<Int32>

Specifies the AdminFolder object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<AdminFolder[]>

Specifies the AdminFolder objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-AdminFolderName**<String>

Specifies the AdminFolder object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

Citrix.Broker.Admin.SDK.BrokerAdminFolder You can pipe the AdminFolder to hold the new or updated metadata.

## Return Values

None or Citrix.Broker.Admin.SDK.BrokerAdminFolder

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerAdminFolder object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerAdminFolderMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the AdminFolder whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerAdminFolder | Set-BrokerAdminFolderMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the AdminFolder in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerAdminFolderMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the AdminFolder in the site whose name is 'objname'

# Set-BrokerAppAssignmentPolicyRule

May 10, 2016

Modifies an existing application rule in the site's assignment policy.

## Syntax

```
Set-BrokerAppAssignmentPolicyRule [-InputObject] <AppAssignmentPolicyRule[]> [-PassThru] [-AddExcludedUsers <User[]>] [-AddIncludedUsers <User[]>] [-Description <String>] [-Enabled <Boolean>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-RemoveExcludedUsers <User[]>] [-RemoveIncludedUsers <User[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAppAssignmentPolicyRule [-Name] <String> [-PassThru] [-AddExcludedUsers <User[]>] [-AddIncludedUsers <User[]>] [-Description <String>] [-Enabled <Boolean>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-RemoveExcludedUsers <User[]>] [-RemoveIncludedUsers <User[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerAppAssignmentPolicyRule cmdlet modifies an existing application rule in the site's assignment policy.

An application rule in the assignment policy defines the users who are entitled to a self-service persistent machine assignment from the rule's desktop group; once assigned the machine can run one or more applications published from the group.

Changing an application rule does not alter machine assignments that have already been made by the rule, nor does it affect active sessions to those machines in any way.

## Related topics

[New-BrokerAppAssignmentPolicyRule](#)

[Get-BrokerAppAssignmentPolicyRule](#)

[Rename-BrokerAppAssignmentPolicyRule](#)

[Remove-BrokerAppAssignmentPolicyRule](#)

## Parameters

**-InputObject** <AppAssignmentPolicyRule[]>

The application rule in the assignment policy to be modified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)



#### **-Name**<String>

Specifies the name of the application rule in the assignment policy to be modified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-AddExcludedUsers**<User[]>

Adds the specified users to the excluded users filter of the application rule, that is, the users and groups who are explicitly denied an entitlement to a machine assignment from this rule.

See the ExcludedUsers parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AddIncludedUsers**<User[]>

Adds the specified users to the included users filter of the application rule, that is, the users and groups who are granted an entitlement to a machine assignment by the rule.

See the IncludedUsers parameter for more information.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-Description**<String>

Changes the description of the application rule. The text is purely informational for the administrator, it is never visible to the end user.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Enabled**<Boolean>

Enables or disables the application rule. A disabled rule is ignored when evaluating the site's assignment policy.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedUserFilterEnabled**<Boolean>

Enables or disables the excluded users filter. If the filter is disabled then any user entries in the filter are ignored when assignment policy rules are evaluated.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedUsers**<User[]>

Changes the excluded users filter of the application rule, that is, the users and groups who are explicitly denied an entitlement to a machine assignment from the rule.

This can be used to exclude users or groups who would otherwise gain access by groups specified in the included users filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedUserFilterEnabled**<Boolean>

Enables or disables the included users filter. If the filter is disabled then any user who satisfies the requirements of the access policy is implicitly granted an entitlement to a machine assignment by the application rule.

Users who would be implicitly granted access when the filter is disabled can still be explicitly denied access using the excluded users filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedUsers**<User[]>

Changes the included users filter of the application rule, that is, the users and groups who are granted an entitlement to a machine assignment by the rule.

If a user appears explicitly in the excluded users filter of the rule, or is a member of a group that appears in the excluded users filter, no entitlement is granted whether or not the user appears in the included users filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RemoveExcludedUsers**<User[]>

Removes the specified users from the excluded users filter of the application rule, that is, the users and groups who are explicitly denied an entitlement to a machine assignment from this rule.

See the ExcludedUsers parameter for more information.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-RemoveIncludedUsers<User[]>**

Removes the specified users from the included users filter of the application rule, that is, the users and groups who are granted an entitlement to a machine assignment by the rule.

See the IncludedUsers parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId<Guid>**

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### **Input Type**

Citrix.Broker.Admin.SDK.AppAssignmentPolicyRule The application rule within the assignment policy to be modified.

## Return Values

None or Citrix.Broker.Admin.SDK.AppAssignmentPolicyRule

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.AppAssignmentPolicyRule object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerAppAssignmentPolicyRule 'Temp Staff' -AddIncludedUsers office\interns
```

Adds the user group OFFICE\interns to the Temp Staff application rule in the assignment policy. This grants all members of that user group an entitlement to a machine in the rule's desktop group. The machines can run applications published from the group. The application session properties obtained using the rule are determined by the rule's other properties.

### ----- EXAMPLE 2 -----

```
C:\PS> Set-BrokerAppAssignmentPolicyRule 'Temp Staff' -Enabled $false
```

Disables the Temp Staff application rule in the assignment policy. This prevents further machine assignments being made using this rule until it is re-enabled. However, access to machines already assigned using the rule is not impacted.

# Set-BrokerAppEntitlementPolicyRule

May 10, 2016  
Modifies an existing application rule in the site's entitlement policy.

## Syntax

Set-BrokerAppEntitlementPolicyRule [-InputObject] <AppEntitlementPolicyRule[]> [-PassThru] [-AddExcludedUsers <User[]>] [-AddIncludedUsers <User[]>] [-Description <String>] [-Enabled <Boolean>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-RemoveExcludedUsers <User[]>] [-RemoveIncludedUsers <User[]>] [-SessionReconnection <SessionReconnection>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-BrokerAppEntitlementPolicyRule [-Name] <String> [-PassThru] [-AddExcludedUsers <User[]>] [-AddIncludedUsers <User[]>] [-Description <String>] [-Enabled <Boolean>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-RemoveExcludedUsers <User[]>] [-RemoveIncludedUsers <User[]>] [-SessionReconnection <SessionReconnection>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The Set-BrokerAppEntitlementPolicyRule cmdlet modifies an existing application rule in the site's entitlement policy.

An application rule in the entitlement policy defines the users who are allowed per-session access to a machine to run one or more applications published from the rule's desktop group.

Changing a rule does not affect existing sessions launched using the rule, but if the change removes an entitlement to a machine that was previously granted, users may be unable to reconnect to a disconnected session on that machine.

## Related topics

- [New-BrokerAppEntitlementPolicyRule](#)
- [Get-BrokerAppEntitlementPolicyRule](#)
- [Rename-BrokerAppEntitlementPolicyRule](#)
- [Remove-BrokerAppEntitlementPolicyRule](#)

## Parameters

**-InputObject**<AppEntitlementPolicyRule[]>  
The application rule in the entitlement policy to be modified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

--	--

#### **-Name**<String>

The name of the application rule in the entitlement policy to be modified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-AddExcludedUsers**<User[]>

Adds the specified users to the excluded users filter of the rule, that is, the users and groups who are explicitly denied entitlements to run applications published from the desktop group.

See the ExcludedUsers parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AddIncludedUsers**<User[]>

Adds the specified users to the included users filter of the rule, that is, the users and groups who are granted an entitlement to an application session by the rule.

See the IncludedUsers parameter for more information.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-Description**<String>

Changes the description of the application rule. The text is purely informational for the administrator, it is never visible to the end user.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Enabled**<Boolean>

Enables or disables the application rule. A disabled rule is ignored when evaluating the site's entitlement policy.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedUserFilterEnabled**<Boolean>

Enables or disables the excluded users filter. If the filter is disabled then any user entries in the filter are ignored when entitlement policy rules are evaluated.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedUsers**<User[]>

Changes the excluded users filter of the rule, that is, the users and groups who are explicitly denied entitlements to run applications published from the desktop group.

This can be used to exclude users or groups or users who would otherwise gain access by groups specified in the included



users filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedUserFilterEnabled**<Boolean>

Enables or disables the included users filter. If the filter is disabled then any user who satisfies the requirements of the access policy is implicitly granted an entitlement to an application session by the application rule.

Users who would be implicitly granted access when the filter is disabled can still be explicitly denied access using the excluded users filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedUsers**<User[]>

Changes the included users filter of the rule, that is, the users and groups who are granted an entitlement to an application session by the rule.

If a user appears explicitly in the excluded users filter of the rule or is a member of a group that appears in the excluded users filter, no entitlement is granted whether or not the user appears in the included users filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RemoveExcludedUsers**<User[]>

Removes the specified users from the excluded users filter of the application rule, that is, the users and groups who are explicitly denied entitlements to run applications published from the desktop group.

See the ExcludedUsers parameter for more information.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-RemoveIncludedUsers**<User[]>

Removes the specified users from the included users filter of the rule, that is, the users and groups who are granted an entitlement to an application session by the rule.

See the IncludedUsers parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionReconnection**<SessionReconnection>

Defines reconnection (roaming) behavior for sessions launched using this rule. Session reconnection control is an experimental and unsupported feature.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.AppEntitlementPolicyRule The application rule in the entitlement policy rule to be modified.

## Return Values

None or Citrix.Broker.Admin.SDK.AppEntitlementPolicyRule

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.AppEntitlementPolicyRule object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerAppEntitlementPolicyRule 'Temp Workers' -AddIncludedUsers office\contractors
```

Adds the user group OFFICE\contractors to those entitled to run applications from the rule's associated desktop group. This grants all members of that group an entitlement to an application session from that group.

### ----- EXAMPLE 2 -----

```
C:\PS> Set-BrokerAppEntitlementPolicyRule 'Temp Workers' -Enabled $false
```

Disables the Temp Workers application rule in the entitlement policy. This prevents further application sessions being launched using this rule until it is re-enabled. However, access to existing application sessions is not affected.

# Set-BrokerApplication

May 10, 2016

Changes the settings of an application to the value specified in the command.

## Syntax

```
Set-BrokerApplication [-InputObject] <Application[]> [-PassThru] [-BrowserName <String>] [-ClientFolder <String>] [-CommandLineArguments <String>] [-CommandLineExecutable <String>] [-CpuPriorityLevel <CpuPriorityLevel>] [-Description <String>] [-Enabled <Boolean>] [-IconFromClient <Boolean>] [-IconUid <Int32>] [-PublishedName <String>] [-SecureCmdLineArgumentsEnabled <Boolean>] [-ShortcutAddedToDesktop <Boolean>] [-ShortcutAddedToStartMenu <Boolean>] [-StartMenuFolder <String>] [-UserFilterEnabled <Boolean>] [-Visible <Boolean>] [-WaitForPrinterCreation <Boolean>] [-WorkingDirectory <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerApplication [-Name] <String> [-PassThru] [-BrowserName <String>] [-ClientFolder <String>] [-CommandLineArguments <String>] [-CommandLineExecutable <String>] [-CpuPriorityLevel <CpuPriorityLevel>] [-Description <String>] [-Enabled <Boolean>] [-IconFromClient <Boolean>] [-IconUid <Int32>] [-PublishedName <String>] [-SecureCmdLineArgumentsEnabled <Boolean>] [-ShortcutAddedToDesktop <Boolean>] [-ShortcutAddedToStartMenu <Boolean>] [-StartMenuFolder <String>] [-UserFilterEnabled <Boolean>] [-Visible <Boolean>] [-WaitForPrinterCreation <Boolean>] [-WorkingDirectory <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerApplication cmdlet changes the value of one or more properties of an application, such as its CpuPriorityLevel or its CommandLineArguments, to the value specified in the command.

This cmdlet lets you change only the settings of the Application object, and not the relationships to other objects. For instance, it does not let you change which users can access this application, or change which desktop groups this application is published to. To do this, you need to remove the existing association, and then add a new association. The following example shows how to change the desktop group that an application is associated with from \$group1 to \$group2:

```
Remove-BrokerApplication -DesktopGroup $group1
```

```
Add-RemoveApplication -DesktopGroup $group2
```

You can change properties of both HostedOnDesktop and InstalledOnClient applications but it is not possible to change the ApplicationType. Also, the Name cannot be changed using this cmdlet; to do this, use the Rename-BrokerApplication cmdlet.

## Related topics

[New-BrokerApplication](#)

[Add-BrokerApplication](#)

[Get-BrokerApplication](#)

[Remove-BrokerApplication](#)

[Rename-BrokerApplication](#)

## Move-BrokerApplication

### Parameters

**-InputObject**<Application[]>

Specifies the application to modify. The Uid of the application can also be substituted for the object.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the application to be modified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-BrowserName**<String>

Specifies the name of the application to modify. Note that the BrowserName cannot be changed in this manner; to modify the BrowserName of an application, use the Rename-BrokerApplication cmdlet.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ClientFolder**<String>

Specifies the folder that the application belongs to as the user sees it. This is the application folder displayed in the Citrix Online Plug-in, in Web Services, and also in the user's start menu. Subdirectories can be specified with '\' character. The following special characters are not allowed: / \* ? < > | " :. Note that this property cannot be set for applications of type InstalledOnClient.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-CommandLineArguments**<String>

Specifies the command-line arguments to use when launching the executable.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-CommandLineExecutable**<String>

Specifies the name of the executable file to launch.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-CpuPriorityLevel**<CpuPriorityLevel>

Specifies the CPU priority for the launched executable. Valid values are: Low, BelowNormal, Normal, AboveNormal, and High. Note that this property cannot be set for applications of type InstalledOnClient.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-Description**<String>

Specifies the description of the application.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Enabled**<Boolean>

Specifies whether or not this application can be launched.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IconFromClient**<Boolean>

Specifies if the app icon should be retrieved from the application on the client. This is reserved for possible future use, and all applications of type HostedOnDesktop cannot set or change this value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IconUid**<Int32>

Specifies which icon to use for this application. This application is visible both to the administrator (in the consoles) and also to the user. If no icon is specified, then a generic built-in application icon is used.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-PublishedName<String>**

Specifies the name seen by end users who have access to this application.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecureCmdLineArgumentsEnabled<Boolean>**

Specifies whether the command-line arguments should be secured. This is reserved for possible future use, and all applications of type HostedOnDesktop can only have this value set to true.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ShortcutAddedToDesktop<Boolean>**

Specifies whether or not a shortcut to the application should be placed on the user device.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ShortcutAddedToStart Menu<Boolean>**

Specifies whether a shortcut to the application should be placed in the user's start menu on their user device.

Required?	false



Default Value	
Accept Pipeline Input?	false

#### **-StartMenuFolder**<String>

Specifies the name of the start menu folder that holds the application shortcut (if any). This is valid only for the Citrix Online Plug-in. Subdirectories can be specified with '\' character. The following special characters are not allowed: / \* ? < > | " .:

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UserFilterEnabled**<Boolean>

Specifies whether the application's user filter is enabled or disabled. Where the user filter is enabled, the application is only visible to users who appear in the filter (either explicitly or by virtue of group membership).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Visible**<Boolean>

Specifies whether or not this application is visible to users. Note that it's possible for an application to be disabled and still visible.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-WaitForPrinterCreation**<Boolean>

Specifies whether or not the session waits for the printers to be created before allowing the end-user to interact with the session. Note that this property cannot be set for applications of type InstalledOnClient.

--	--

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-WorkingDirectory**<String>

Specifies from which working directory the executable is launched from.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.Application, or depends on parameter You can pipe the application to be added to Set-

BrokerApplication. You can also pipe some of the other parameters by name.

## Return Values

None or Citrix.Broker.Admin.SDK.Application

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.Application object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerApplication -Name "Notepad" -Description 'Windows Notepad'
```

Modifies the application that has a Name of "Notepad" so that its description reads Windows Notepad.

### ----- EXAMPLE 2 -----

```
C:\PS> $app = Get-BrokerApplication -BrowserName "Calculator"
```

```
C:\PS> Set-BrokerApplication -InputObject $app -Enabled $false
```

First gets the application with a BrowserName of "Calculator", then modifies that application (by supplying the application object in the first position) so that it is disabled for users.

# Set-BrokerApplicationInstanceMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for ApplicationInstance

## Syntax

```
Set-BrokerApplicationInstanceMetadata [-ApplicationInstanceId] <Int64> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerApplicationInstanceMetadata [-ApplicationInstanceId] <Int64> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerApplicationInstanceMetadata [-ApplicationInstanceId] <Int64> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerApplicationInstanceMetadata [-InputObject] <ApplicationInstance[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerApplicationInstanceMetadata [-InputObject] <ApplicationInstance[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerApplicationInstanceMetadata cmdlet creates/updates metadata key-value pairs for ApplicationInstance. The ApplicationInstance can be specified by InputObject or piping.

## Related topics

### Parameters

**-ApplicationInstanceId**<Int64>

Specifies the ApplicationInstance object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-InputObject**<ApplicationInstance[]>

Specifies the ApplicationInstance objects whose Metadata is to be created/updated.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByValue)
------------------------	----------------

#### **-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False

Accept Pipeline Input?	false
------------------------	-------

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.BrokerApplicationInstance You can pipe the ApplicationInstance to hold the new or updated metadata.

### Return Values

None or Citrix.Broker.Admin.SDK.BrokerApplicationInstance

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerApplicationInstance object.

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerApplicationInstanceMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the ApplicationInstance whose instance is pointed by \$obj-Uid

#### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerApplicationInstance | Set-BrokerApplicationInstanceMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the ApplicationInstance in the site

# Set-BrokerApplicationMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for Application

## Syntax

```
Set-BrokerApplicationMetadata [-ApplicationId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerApplicationMetadata [-ApplicationId] <Int32> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerApplicationMetadata [-ApplicationId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerApplicationMetadata [-InputObject] <Application[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerApplicationMetadata [-InputObject] <Application[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerApplicationMetadata [-ApplicationName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerApplicationMetadata [-ApplicationName] <String> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerApplicationMetadata cmdlet creates/updates metadata key-value pairs for Application. The Application can be specified by InputObject or piping.

## Related topics

## Parameters

**-ApplicationId**<Int32>

Specifies the Application object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-InputObject**<Application[]>



Specifies the Application objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-ApplicationName**<String>

Specifies the Application object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

Citrix.Broker.Admin.SDK.BrokerApplication You can pipe the Application to hold the new or updated metadata.

## Return Values

None or Citrix.Broker.Admin.SDK.BrokerApplication

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerApplication object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerApplicationMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the Application whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerApplication | Set-BrokerApplicationMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the Application in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerApplicationMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the Application in the site whose name is 'objname'

# Set-BrokerAssignmentPolicyRule

May 10, 2016

Modifies an existing desktop rule in the site's assignment policy.

## Syntax

```
Set-BrokerAssignmentPolicyRule [-InputObject] <AssignmentPolicyRule[]> [-PassThru] [-AddExcludedUsers <User[]>] [-AddIncludedUsers <User[]>] [-ColorDepth <ColorDepth>] [-Description <String>] [-Enabled <Boolean>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-IconUid <Int32>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-MaxDesktops <Int32>] [-PublishedName <String>] [-RemoveExcludedUsers <User[]>] [-RemoveIncludedUsers <User[]>] [-SecureIcaRequired <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAssignmentPolicyRule [-Name] <String> [-PassThru] [-AddExcludedUsers <User[]>] [-AddIncludedUsers <User[]>] [-ColorDepth <ColorDepth>] [-Description <String>] [-Enabled <Boolean>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-IconUid <Int32>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-MaxDesktops <Int32>] [-PublishedName <String>] [-RemoveExcludedUsers <User[]>] [-RemoveIncludedUsers <User[]>] [-SecureIcaRequired <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerAssignmentPolicyRule cmdlet modifies an existing desktop rule in the site's assignment policy.

A desktop rule in the assignment policy defines the users who are entitled to self-service persistent machine assignments from the rule's desktop group. A rule defines how many machines a user is allowed from the group for delivery of full desktop sessions.

Changing a desktop rule does not alter machine assignments that have already been made by the rule, nor does it affect active sessions to those machines in any way.

## Related topics

[New-BrokerAssignmentPolicyRule](#)

[Get-BrokerAssignmentPolicyRule](#)

[Rename-BrokerAssignmentPolicyRule](#)

[Remove-BrokerAssignmentPolicyRule](#)

## Parameters

**-InputObject** <AssignmentPolicyRule[]>

The desktop rule in the assignment policy to be modified.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByValue)
------------------------	----------------

#### **-Name**<String>

Specifies the name of the desktop rule in the assignment policy to be modified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-AddExcludedUsers**<User[]>

Adds the specified users to the excluded users filter of the rule, that is, the users and groups who are explicitly denied an entitlement to a machine assignment from this rule.

See the ExcludedUsers parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AddIncludedUsers**<User[]>

Adds the specified users to the included users filter of the rule, that is, the users and groups who are granted an entitlement to a machine assignment by the rule.

See the IncludedUsers parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ColorDepth**<ColorDepth>

Changes the color depth of any desktop sessions to machines assigned by the rule.

Valid values are \$null, FourBit, EightBit, SixteenBit, and TwentyFourBit.

The default null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Description**<String>

Changes the description of the desktop rule. The text may be visible to the end user, for example, as a tooltip associated with the desktop entitlement.

A null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Enabled**<Boolean>

Enables or disables the desktop rule. A disabled rule is ignored when evaluating the site's assignment policy.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedUserFilterEnabled**<Boolean>

Enables or disables the excluded users filter. If the filter is disabled then any user entries in the filter are ignored when assignment policy rules are evaluated.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedUsers**<User[]>

Changes the excluded users filter of the desktop rule, that is, the users and groups who are explicitly denied an entitlement to a machine assignment from this rule.

This can be used to exclude users or groups who would otherwise gain access by groups specified in the included users filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IconUid**<Int32>

Changes the unique ID of the icon used to display the machine assignment entitlement to the user, and of the assigned desktop itself following the assignment.

The default null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedUserFilterEnabled**<Boolean>

Enables or disables the included users filter. If the filter is disabled then any user who satisfies the requirements of the access policy is implicitly granted an entitlement to a machine assignment by the rule.

Users who would be implicitly granted access when the filter is disabled can still be explicitly denied access using the excluded users filter.

For rules that relate to RemotePC desktop groups however, if the included user filter is disabled, the rule is effectively disabled.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedUsers**<User[]>

Changes the included users filter of the rule, that is, the users and groups who are granted an entitlement to a machine assignment by the rule.

If a user appears explicitly in the excluded users filter of the rule, or is a member of a group that appears in the excluded users filter, no entitlement is granted whether or not the user appears in the included users filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxDesktops**<Int32>

The number of machines from the rule's desktop group to which a user is entitled. Where an entitlement is granted to a user group rather than an individual, the number of machines applies to each member of the user group independently.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PublishedName**<String>

Changes the published name of the machine assignment entitlement as seen by the user. Changing the published name does not affect the names of desktops that have already been assigned by the rule.

The default null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	
Accept Pipeline Input?	



Default Value	
Accept Pipeline Input?	false

#### **-RemoveExcludedUsers<User[]>**

Removes the specified users from the excluded users filter of the rule, that is, the users and groups who are explicitly denied an entitlement to a machine assignment from this rule.

See the ExcludedUsers parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RemoveIncludedUsers<User[]>**

Removes the specified users from the included users filter of the desktop rule, that is, the users and groups who are granted an entitlement to a machine assignment by the rule.

See the IncludedUsers parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecureIcaRequired<Boolean>**

Changes whether the desktop rule requires the SecureICA protocol for desktop sessions to machines assigned using the entitlement.

The default null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.AssignmentPolicyRule The desktop rule within the assignment policy to be modified.

### Return Values

None or Citrix.Broker.Admin.SDK.AssignmentPolicyRule

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.AssignmentPolicyRule object.

### Examples

#### ----- **EXAMPLE 1** -----

```
C:\PS> Set-BrokerAssignmentPolicyRule 'Temp Staff' -AddIncludedUsers office\interns
```

Adds the user group OFFICE\interns to the Temp Staff desktop rule in the assignment policy. This grants all members of that user group entitlements to machines in the rule's desktop group. The number of machine entitlements per user and the session properties of the desktops obtained using the rule are determined by the rule's other properties.

#### ----- **EXAMPLE 2** -----

```
C:\PS> Set-BrokerAssignmentPolicyRule 'Temp Staff' -Enabled $false
```

Disables the Temp Staff desktop rule in the assignment policy. This prevents further machine assignments being made using

this rule until it is re-enabled. However, access to machines already assigned using the rule is not impacted.

# Set-BrokerAssignmentPolicyRuleMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for AssignmentPolicyRule

## Syntax

```
Set-BrokerAssignmentPolicyRuleMetadata [-AssignmentPolicyRuleId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAssignmentPolicyRuleMetadata [-AssignmentPolicyRuleId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAssignmentPolicyRuleMetadata [-AssignmentPolicyRuleId] <Int32> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAssignmentPolicyRuleMetadata [-InputObject] <AssignmentPolicyRule[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAssignmentPolicyRuleMetadata [-InputObject] <AssignmentPolicyRule[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAssignmentPolicyRuleMetadata [-AssignmentPolicyRuleName] <String> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerAssignmentPolicyRuleMetadata [-AssignmentPolicyRuleName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerAssignmentPolicyRuleMetadata cmdlet creates/updates metadata key-value pairs for AssignmentPolicyRule. The AssignmentPolicyRule can be specified by InputObject or piping.

## Related topics

## Parameters

**-AssignmentPolicyRuleId**<Int32>

Specifies the AssignmentPolicyRule object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-InputObject**<AssignmentPolicyRule[]>

Specifies the AssignmentPolicyRule objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Assignment PolicyRuleName**<String>

Specifies the AssignmentPolicyRule object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true

Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerAssignmentPolicyRule You can pipe the AssignmentPolicyRule to hold the new or updated metadata.

#### Return Values

None or Citrix.Broker.Admin.SDK.BrokerAssignmentPolicyRule

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerAssignmentPolicyRule object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerAssignmentPolicyRuleMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the AssignmentPolicyRule whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerAssignmentPolicyRule | Set-BrokerAssignmentPolicyRuleMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the AssignmentPolicyRule in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerAssignmentPolicyRuleMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the AssignmentPolicyRule in the site whose name is 'objname'

# Set-BrokerCatalog

May 10, 2016

Sets the properties of a catalog.

## Syntax

```
Set-BrokerCatalog [-InputObject] <Catalog[]> [-PassThru] [-Description <String>] [-IsRemotePC <Boolean>] [-MinimumFunctionalLevel <FunctionalLevel>] [-ProvisioningSchemeld <Guid>] [-PvsAddress <String>] [-PvsDomain <String>] [-RemotePCHypervisorConnectionUid <Int32>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerCatalog [-Name] <String> [-PassThru] [-Description <String>] [-IsRemotePC <Boolean>] [-MinimumFunctionalLevel <FunctionalLevel>] [-ProvisioningSchemeld <Guid>] [-PvsAddress <String>] [-PvsDomain <String>] [-RemotePCHypervisorConnectionUid <Int32>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerCatalog cmdlet sets properties of a catalog or set of catalogs. The catalog can be specified by name, in which case only one catalog can be specified, or one or more catalog instances can be passed to the command either by piping or by using the -InputObject parameter.

## Related topics

[Get-BrokerCatalog](#)

[New-BrokerCatalog](#)

[Remove-BrokerCatalog](#)

[Rename-BrokerCatalog](#)

## Parameters

**-InputObject**<Catalog[]>

Specifies the catalog objects to modify.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Identifies the catalog to modify.

Required?	true
-----------	------



Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-Description**<String>

Supplies the new value of the Description property.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IsRemotePC**<Boolean>

Supplies a new value for IsRemotePC.

IsRemotePC can only be enabled when:

- o SessionSupport is SingleSession
- o MachinesArePhysical is true.

IsRemotePC can only be set from true to false when no RemotePCAccount references this catalog, and when no Remote PC relationship exists between this catalog and a desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MinimumFunctionalLevel**<FunctionalLevel>

The new minimum FunctionalLevel required for machines to work successfully in the catalog. If this is higher than the FunctionalLevel of any machines already in the catalog, they will immediately cease to function.

Valid values are L5, L7, L7\_6

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ProvisioningSchemeld**<Guid>

Specifies the identity of the MCS provisioning scheme the catalog is associated with (can only be specified for new catalogs with a ProvisioningType of MCS; once set can never be changed).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PvsAddress**<String>

Supplies the new value of the PvsAddress property. Can only be set if CatalogKind is Pvs or PvsPvd.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PvsDomain**<String>

Supplies the new value of the PvsDomain property. Can only be set if CatalogKind is PvsPvd.

Required?	false
Default Value	
Accept Pipeline Input?	false

--	--

#### **-RemotePCHypervisorConnectionUid<Int32>**

Supplies the new hypervisor connection to use for powering on remote PCs in this catalog (only allowed when IsRemotePC is true); this will affect all machines already in the catalog as well as those created later.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId<Guid>**

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### **Input Type**

Citrix.Broker.Admin.SDK.Catalog You can pipe the catalogs to be modified to Set-BrokerCatalog.

#### **Return Values**

None or Citrix.Broker.Admin.SDK.Catalog

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a

Citrix.Broker.Admin.SDK.Catalog object.

## Notes

A catalog's Name property cannot be changed by Set-BrokerCatalog. To rename a catalog use Rename-BrokerCatalog.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerCatalog -Name "MyCatalog" -Description "New Description"
```

This example specifies a catalog by name and sets its description.

### ----- EXAMPLE 2 -----

```
C:\PS> $permCatalogs = Get-BrokerCatalog -AllocationType Static
```

```
C:\PS> Set-BrokerCatalog -InputObject $permCatalogs -Description "Permanently assigned machines"
```

This example sets the description for all catalogs with AllocationType 'Static'.

# Set-BrokerCatalogMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for Catalog

## Syntax

```
Set-BrokerCatalogMetadata [-CatalogId] <Int32> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [-<CommonParameters>]
```

```
Set-BrokerCatalogMetadata [-CatalogId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [-<CommonParameters>]
```

```
Set-BrokerCatalogMetadata [-CatalogId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [-<CommonParameters>]
```

```
Set-BrokerCatalogMetadata [-InputObject] <Catalog[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [-<CommonParameters>]
```

```
Set-BrokerCatalogMetadata [-InputObject] <Catalog[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [-<CommonParameters>]
```

```
Set-BrokerCatalogMetadata [-CatalogName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [-<CommonParameters>]
```

```
Set-BrokerCatalogMetadata [-CatalogName] <String> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [-<CommonParameters>]
```

## Detailed Description

The Set-BrokerCatalogMetadata cmdlet creates/updates metadata key-value pairs for Catalog. The Catalog can be specified by InputObject or piping.

## Related topics

## Parameters

**-CatalogId**<Int32>

Specifies the Catalog object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Catalog[]>

Specifies the Catalog objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-CatalogName**<String>

Specifies the Catalog object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

Citrix.Broker.Admin.SDK.BrokerCatalog You can pipe the Catalog to hold the new or updated metadata.

## Return Values

None or Citrix.Broker.Admin.SDK.BrokerCatalog

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerCatalog object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerCatalogMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the Catalog whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerCatalog | Set-BrokerCatalogMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the Catalog in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerCatalogMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the Catalog in the site whose name is 'objname'



# Set-BrokerConfigurationSlotMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for ConfigurationSlot

## Syntax

```
Set-BrokerConfigurationSlotMetadata [-ConfigurationSlotId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerConfigurationSlotMetadata [-ConfigurationSlotId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerConfigurationSlotMetadata [-ConfigurationSlotId] <Int32> -Map <PObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerConfigurationSlotMetadata [-InputObject] <ConfigurationSlot[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerConfigurationSlotMetadata [-InputObject] <ConfigurationSlot[]> -Map <PObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerConfigurationSlotMetadata [-ConfigurationSlotName] <String> -Map <PObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerConfigurationSlotMetadata [-ConfigurationSlotName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerConfigurationSlotMetadata cmdlet creates/updates metadata key-value pairs for ConfigurationSlot. The ConfigurationSlot can be specified by InputObject or piping.

## Related topics

## Parameters

**-ConfigurationSlotId**<Int32>

Specifies the ConfigurationSlot object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-InputObject**<ConfigurationSlot[]>

Specifies the ConfigurationSlot objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-ConfigurationSlotName**<String>

Specifies the ConfigurationSlot object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

Citrix.Broker.Admin.SDK.BrokerConfigurationSlot You can pipe the ConfigurationSlot to hold the new or updated metadata.

## Return Values

None or Citrix.Broker.Admin.SDK.BrokerConfigurationSlot

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerConfigurationSlot object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerConfigurationSlotMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the ConfigurationSlot whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerConfigurationSlot | Set-BrokerConfigurationSlotMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the ConfigurationSlot in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerConfigurationSlotMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the ConfigurationSlot in the site whose name is 'objname'

# Set-BrokerControllerMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for Controller

## Syntax

```
Set-BrokerControllerMetadata [-ControllerId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerControllerMetadata [-ControllerId] <Int32> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerControllerMetadata [-ControllerId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerControllerMetadata [-InputObject] <Controller[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerControllerMetadata [-InputObject] <Controller[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerControllerMetadata [-ControllerName] <String> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerControllerMetadata [-ControllerName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerControllerMetadata cmdlet creates/updates metadata key-value pairs for Controller. The Controller can be specified by InputObject or piping.

## Related topics

## Parameters

**-ControllerId**<Int32>

Specifies the Controller object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-InputObject**<Controller[]>

Specifies the Controller objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-ControllerName**<String>

Specifies the Controller object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

Citrix.Broker.Admin.SDK.BrokerController You can pipe the Controller to hold the new or updated metadata.

## Return Values

None or Citrix.Broker.Admin.SDK.BrokerController

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerController object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerControllerMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the Controller whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerController | Set-BrokerControllerMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the Controller in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerControllerMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the Controller in the site whose name is 'objname'



# Set-BrokerDBConnection

May 10, 2016

Configures a database connection for the Broker Service.

## Syntax

```
Set-BrokerDBConnection [-DBConnection] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [[-Force]]
[<CommonParameters>]
```

## Detailed Description

Specifies the database connection string for use by the currently selected Citrix Broker Service instance.

The service records the connection string and attempts to contact the specified database. If the database cannot initially be contacted the service retries the connection at intervals until contact with the database is successfully established.

Is is not possible to set a new database connection string for the service if one is already recorded. The connection string must first be set to \$null. This action causes the service to reset and return to its idle state, at which point a new connection string can be set.

When a database connection string is successfully set for the service, a status of OK is returned by the cmdlet. If the database connection string is set to \$null, a DBUnconfigured status is returned.

A syntactically invalid connection string is not recorded.

Only use of Windows authentication within the connection string is supported; a connection string containing SQL authentication credentials is always rejected as invalid.

The current service instance is the one on the local machine, or the one most recently specified using the -AdminAddress parameter of a Broker SDK cmdlet.

## Related topics

[Get-BrokerServiceStatus](#)

[Get-BrokerDBConnection](#)

[Test-BrokerDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be used by the Broker Service. Passing in \$null will clear any existing database connection configured.

Required?	true
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### **-Force**<SwitchParameter>

If present, allows the local administrator to set the connection string to null when there are problems contacting the database or other services.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Set-BrokerDBConnection cmdlet returns an object describing the status of the Broker Service together with extra diagnostics information. Possible values are:

-- OK:

The Broker Service instance is configured with a valid database and service schema. The service is operational.

-- DBUnconfigured:

No database connection string is set for the Broker Service instance.

-- DBRejectedConnection:

The database rejected the logon attempt from the Broker Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

-- InvalidDBConfigured:

The specified database does not exist, is not visible to the Broker Service instance, or the service's schema within the database is invalid.

-- DBNotFound:

The specified database could not be located with the configured connection string.

-- DBNewerVersionThanService:

The version of the Broker Service currently in use is newer than, and incompatible with, the version of the Broker Service schema on the database. Upgrade the Broker Service to a more recent version.

-- DBOlderVersionThanService:

The version of the Broker Service schema on the database is newer than, and incompatible with, the version of the Broker Service currently in use. Upgrade the database schema to a more recent version.

-- DBVersionChangeInProgress:

A database schema upgrade is in progress.

-- PendingFailure:

Connectivity between the Broker Service instance and the database has been lost. This may be a transitory network error, but may indicate a loss of connectivity that requires administrator intervention.

-- Failed:

Connectivity between the Broker Service instance and the database has been lost for an extended period of time, or has failed due to a configuration problem. The service instance cannot operate while its connection to the database is unavailable.

-- Unknown:

The status of the Broker Service cannot be determined.

Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

##### InvalidDBConnectionString

The database connection string has an invalid format.

##### DatabaseConnectionDetailsAlreadyConfigured

There was already a database connection configured. After a configuration is set, it can only be set to \$null.

##### PermissionDenied

You do not have permission to execute this command.

##### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

##### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

##### CommunicationError

There was a problem communicating with the remote service.

##### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS>Set-BrokerDBConnection "Server=dbserver\SQLEXPRESS;Database=XDDB;Trusted_Connection=True"
```

Configures the service instance to use a database called XDDB on an SQL Server Express database running on the machine called dbserver. Integrated Windows authentication is required.

##### ----- EXAMPLE 2 -----

```
C:\PS>Set-BrokerDBConnection -DBConnection $null
```

Resets the service instance's database connection string. The service instance resets and returns to an idle state until a valid new database connection string is specified.

# Set-BrokerDesktopGroup

May 10, 2016

Adjusts the settings of a broker desktop group.

## Syntax

```
Set-BrokerDesktopGroup [-InputObject] <DesktopGroup[]> [-PassThru] [-AutomaticPowerOnForAssigned <Boolean>] [-AutomaticPowerOnForAssignedDuringPeak <Boolean>] [-ColorDepth <ColorDepth>] [-DeliveryType <DeliveryType>] [-Description <String>] [-Enabled <Boolean>] [-IconUid <Int32>] [-InMaintenanceMode <Boolean>] [-IsRemotePC <Boolean>] [-MinimumFunctionalLevel <FunctionalLevel>] [-OffPeakBufferSizePercent <Int32>] [-OffPeakDisconnectAction <SessionChangeHostingAction>] [-OffPeakDisconnectTimeout <Int32>] [-OffPeakExtendedDisconnectAction <SessionChangeHostingAction>] [-OffPeakExtendedDisconnectTimeout <Int32>] [-OffPeakLogOffAction <SessionChangeHostingAction>] [-OffPeakLogOffTimeout <Int32>] [-PeakBufferSizePercent <Int32>] [-PeakDisconnectAction <SessionChangeHostingAction>] [-PeakDisconnectTimeout <Int32>] [-PeakExtendedDisconnectAction <SessionChangeHostingAction>] [-PeakExtendedDisconnectTimeout <Int32>] [-PeakLogOffAction <SessionChangeHostingAction>] [-PeakLogOffTimeout <Int32>] [-ProtocolPriority <String[]>] [-PublishedName <String>] [-SecureIcaRequired <Boolean>] [-SettlementPeriodBeforeAutoShutdown <TimeSpan>] [-ShutdownDesktopsAfterUse <Boolean>] [-TimeZone <String>] [-TurnOnAddedMachine <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerDesktopGroup [-Name] <String> [-PassThru] [-AutomaticPowerOnForAssigned <Boolean>] [-AutomaticPowerOnForAssignedDuringPeak <Boolean>] [-ColorDepth <ColorDepth>] [-DeliveryType <DeliveryType>] [-Description <String>] [-Enabled <Boolean>] [-IconUid <Int32>] [-InMaintenanceMode <Boolean>] [-IsRemotePC <Boolean>] [-MinimumFunctionalLevel <FunctionalLevel>] [-OffPeakBufferSizePercent <Int32>] [-OffPeakDisconnectAction <SessionChangeHostingAction>] [-OffPeakDisconnectTimeout <Int32>] [-OffPeakExtendedDisconnectAction <SessionChangeHostingAction>] [-OffPeakExtendedDisconnectTimeout <Int32>] [-OffPeakLogOffAction <SessionChangeHostingAction>] [-OffPeakLogOffTimeout <Int32>] [-PeakBufferSizePercent <Int32>] [-PeakDisconnectAction <SessionChangeHostingAction>] [-PeakDisconnectTimeout <Int32>] [-PeakExtendedDisconnectAction <SessionChangeHostingAction>] [-PeakExtendedDisconnectTimeout <Int32>] [-PeakLogOffAction <SessionChangeHostingAction>] [-PeakLogOffTimeout <Int32>] [-ProtocolPriority <String[]>] [-PublishedName <String>] [-SecureIcaRequired <Boolean>] [-SettlementPeriodBeforeAutoShutdown <TimeSpan>] [-ShutdownDesktopsAfterUse <Boolean>] [-TimeZone <String>] [-TurnOnAddedMachine <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerDesktopGroup cmdlet is used to disable or enable an existing broker desktop group or to alter its settings.

## Related topics

[Get-BrokerDesktopGroup](#)

[New-BrokerDesktopGroup](#)

[Rename-BrokerDesktopGroup](#)

[Remove-BrokerDesktopGroup](#)

## Parameters

### **-InputObject**<DesktopGroup[]>

Specifies the desktop groups to adjust.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name**<String>

Specifies the desktop groups to adjust, based on their Name property.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

### **-AutomaticPowerOnForAssigned**<Boolean>

Specifies whether assigned desktops in the desktop group should be automatically started at the start of peak time periods. Only relevant for groups whose DesktopKind is Private.

Required?	false
Default Value	
Accept Pipeline Input?	false

--	--

#### **-AutomaticPowerOnForAssignedDuringPeak**<Boolean>

Specifies whether assigned desktops in the desktop group should be automatically started throughout peak time periods. Only relevant for groups whose DesktopKind is Private and which have AutomaticPowerOnForAssigned set to true.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ColorDepth**<ColorDepth>

Specifies the color depth that the ICA session should use for desktops in this group. Valid values are FourBit, EightBit, SixteenBit, and TwentyFourBit.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DeliveryType**<DeliveryType>

Specifies whether desktops, applications, or both, can be delivered from machines contained within the desktop group. Desktop groups with a DesktopKind of Private cannot be used to deliver both desktops and applications.

When changing the delivery type to desktops only, there must be no remaining desktop-hosted applications associated with the group, or application-specific assignment/entitlement policy rules for the group.

When changing the delivery type to applications only, there must be no remaining client-hosted applications associated with the group, or desktop-specific assignment/entitlement policy rules for the group.

Valid values are DesktopsOnly, AppsOnly, and DesktopsAndApps.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Description**<String>

A description for this desktop group useful for administrators of the site.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Enabled**<Boolean>

Whether the desktop group should be in the enabled state; disabled desktop groups do not appear to users.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IconUid**<Int32>

The UID of the broker icon to be displayed to users for their desktop(s) in this desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-InMaintenanceMode**<Boolean>

Whether the desktop should be put into maintenance mode; a desktop group in maintenance mode will not allow users to connect or reconnect to their desktops.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IsRemotePC**<Boolean>



Supplies a new value for IsRemotePC.

IsRemotePC can only be enabled when:

- o SessionSupport is SingleSession

- o DeliveryType is DesktopsOnly

- o DesktopKind is Private

IsRemotePC can be switched from true to false only if no RemotePC relationship exists between a catalog and this desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MinimumFunctionalLevel**<FunctionalLevel>

The new minimum FunctionalLevel required for machines to work successfully in the desktop group. If this is higher than the FunctionalLevel of any machines already in the desktop group, they will immediately cease to function.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OffPeakBufferSizePercent**<Int32>

The percentage of machines in the desktop group that should be kept available in an idle state outside peak hours.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OffPeakDisconnectAction**<SessionChangeHostingAction>

The action to be performed after a configurable period of a user session disconnecting outside peak hours. Possible values are Nothing, Suspend, or Shutdown.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OffPeakDisconnectTimeout**<Int32>

The number of minutes before the configured action should be performed after a user session disconnects outside peak hours.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OffPeakExtendedDisconnectAction**<SessionChangeHostingAction>

The action to be performed after a second configurable period of a user session disconnecting outside peak hours. Possible values are Nothing, Suspend, or Shutdown.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OffPeakExtendedDisconnectTimeout**<Int32>

The number of minutes before the second configured action should be performed after a user session disconnects outside peak hours.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OffPeakLogOffAction**<SessionChangeHostingAction>

The action to be performed after a configurable period of a user session ending outside peak hours. Possible values are Nothing, Suspend, or Shutdown.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-OffPeakLogOffTimeout**<Int32>

The number of minutes before the configured action should be performed after a user session ends outside peak hours.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PeakBufferSizePercent**<Int32>

The percentage of machines in the desktop group that should be kept available in an idle state in peak hours.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PeakDisconnectAction**<SessionChangeHostingAction>

The action to be performed after a configurable period of a user session disconnecting in peak hours. Possible values are Nothing, Suspend, or Shutdown.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PeakDisconnectTimeout**<Int32>

The number of minutes before the configured action should be performed after a user session disconnects in peak hours.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PeakExtendedDisconnectAction**<SessionChangeHostingAction>

The action to be performed after a second configurable period of a user session disconnecting in peak hours. Possible values are Nothing, Suspend, or Shutdown.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PeakExtendedDisconnectTimeout**<Int32>

The number of minutes before the second configured action should be performed after a user session disconnects in peak hours.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PeakLogOffAction**<SessionChangeHostingAction>

The action to be performed after a configurable period of a user session ending in peak hours. Possible values are Nothing, Suspend, or Shutdown.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PeakLogOffTimeout<Int32>**

The number of minutes before the configured action should be performed after a user session ends in peak hours.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ProtocolPriority<String[]>**

A list of protocol names in the order in which they should be attempted for use during connection.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PublishedName<String>**

The name that will be displayed to users for their desktop(s) in this desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-SecureIcaRequired<Boolean>**

Whether HDX connections to desktops in the new desktop group require the use of a secure protocol.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-SettlementPeriodBeforeAutoShutdown<TimeSpan>**

Time after a session ends during which automatic shutdown requests (for example, shutdown after use, idle pool management) are deferred. Any outstanding shutdown request takes effect after the settlement period expires. This is typically used to configure time to allow logoff scripts to complete.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ShutdownDesktopsAfterUse**<Boolean>

Whether desktops in this desktop group should be automatically shut down when each user session completes (only relevant to power-managed desktops).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-TimeZone**<String>

The time zone in which this desktop group's machines reside.

The time zone must be specified for any of the group's automatic power management settings to take effect. Automatic power management operations include pool management (power time schemes), reboot schedules, session disconnect and logoff actions, and powering on assigned machines etc.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-TurnOnAddedMachine**<Boolean>

This flag specifies whether the Broker Service should attempt to power on machines when they are added to the desktop group.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.DesktopGroup You can pipe the desktop groups to be adjusted to Set-BrokerDesktopGroup.

#### Return Values

None or Citrix.Broker.Admin.SDK.DesktopGroup

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.DesktopGroup object.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerDesktopGroup EMEA* -InMaintenanceMode $true -PassThru
```

Sets all desktop groups with names starting "EMEA" into maintenance mode, returning the set of desktop groups.

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerDesktopGroup -InMaintenanceMode $true | Set-BrokerDesktopGroup -Enabled $false
```

Disable all desktop groups that are in maintenance mode.



# Set-BrokerDesktopGroupMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for DesktopGroup

## Syntax

```
Set-BrokerDesktopGroupMetadata [-DesktopGroupId] <Int32> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerDesktopGroupMetadata [-DesktopGroupId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerDesktopGroupMetadata [-DesktopGroupId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerDesktopGroupMetadata [-InputObject] <DesktopGroup[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerDesktopGroupMetadata [-InputObject] <DesktopGroup[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerDesktopGroupMetadata [-DesktopGroupName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerDesktopGroupMetadata [-DesktopGroupName] <String> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerDesktopGroupMetadata cmdlet creates/updates metadata key-value pairs for DesktopGroup. The DesktopGroup can be specified by InputObject or piping.

## Related topics

## Parameters

**-DesktopGroupId**<Int32>

Specifies the DesktopGroup object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<DesktopGroup[]>

Specifies the DesktopGroup objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-DesktopGroupName**<String>

Specifies the DesktopGroup object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

Citrix.Broker.Admin.SDK.BrokerDesktopGroup You can pipe the DesktopGroup to hold the new or updated metadata.

## Return Values

None or Citrix.Broker.Admin.SDK.BrokerDesktopGroup

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerDesktopGroup object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerDesktopGroupMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the DesktopGroup whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerDesktopGroup | Set-BrokerDesktopGroupMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the DesktopGroup in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerDesktopGroupMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the DesktopGroup in the site whose name is 'objname'

# Set-BrokerEntitlementPolicyRule

May 10, 2016

Modifies an existing desktop rule in the site's entitlement policy.

## Syntax

```
Set-BrokerEntitlementPolicyRule [-InputObject] <EntitlementPolicyRule[]> [-PassThru] [-AddExcludedUsers <User[]>] [-AddIncludedUsers <User[]>] [-ColorDepth <ColorDepth>] [-Description <String>] [-Enabled <Boolean>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-IconUid <Int32>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-PublishedName <String>] [-RemoveExcludedUsers <User[]>] [-RemoveIncludedUsers <User[]>] [-SecureIcaRequired <Boolean>] [-SessionReconnection <SessionReconnection>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerEntitlementPolicyRule [-Name] <String> [-PassThru] [-AddExcludedUsers <User[]>] [-AddIncludedUsers <User[]>] [-ColorDepth <ColorDepth>] [-Description <String>] [-Enabled <Boolean>] [-ExcludedUserFilterEnabled <Boolean>] [-ExcludedUsers <User[]>] [-IconUid <Int32>] [-IncludedUserFilterEnabled <Boolean>] [-IncludedUsers <User[]>] [-PublishedName <String>] [-RemoveExcludedUsers <User[]>] [-RemoveIncludedUsers <User[]>] [-SecureIcaRequired <Boolean>] [-SessionReconnection <SessionReconnection>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerEntitlementPolicyRule cmdlet modifies an existing desktop rule in the site's entitlement policy.

A desktop rule in the entitlement policy defines the users who are allowed per-session access to a machine from the rule's associated desktop group to run a full desktop session.

Changing a rule does not affect existing sessions launched using the rule, but if the change removes an entitlement to a machine that was previously granted, users may be unable to reconnect to a disconnected session on that machine.

## Related topics

[New-BrokerEntitlementPolicyRule](#)

[Get-BrokerEntitlementPolicyRule](#)

[Rename-BrokerEntitlementPolicyRule](#)

[Remove-BrokerEntitlementPolicyRule](#)

## Parameters

**-InputObject** <EntitlementPolicyRule[]>

The desktop rule in the entitlement policy to be modified.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByValue)
------------------------	----------------

#### **-Name**<String>

The name of the desktop rule in the entitlement policy to be modified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-AddExcludedUsers**<User[]>

Adds the specified users to the excluded users filter of the rule, that is, the users and groups who are explicitly denied an entitlement to a desktop session from this rule.

See the ExcludedUsers parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AddIncludedUsers**<User[]>

Adds the specified users to the included users filter of the rule, that is, the users and groups who are granted an entitlement to a desktop session by the rule.

See the IncludedUsers parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ColorDepth**<ColorDepth>

Changes the color depth of any desktop sessions launched by a user from this entitlement. Existing sessions are not affected.

Valid values are \$null, FourBit, EightBit, SixteenBit, and TwentyFourBit.

A null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Description**<String>

Changes the description of the desktop rule. The text may be visible to the end user, for example, as a tooltip associated with the desktop entitlement.

A null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Enabled**<Boolean>

Enables or disables the desktop rule. A disabled rule is ignored when evaluating the site's entitlement policy.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedUserFilterEnabled**<Boolean>

Enables or disables the excluded users filter. If the filter is disabled then any user entries in the filter are ignored when entitlement policy rules are evaluated.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ExcludedUsers**<User[]>

Changes the excluded users filter of the desktop rule, that is, the users and groups who are explicitly denied an entitlement to a desktop session from this rule.

This can be used to exclude users or groups who would otherwise gain access by groups specified in the included users filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IconUid**<Int32>

Changes the icon (identified by its unique ID) for the published desktop entitlement as seen by the user.

A null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedUserFilterEnabled**<Boolean>

Enables or disables the included users filter. If the filter is disabled then any user who satisfies the requirements of the access policy is implicitly granted an entitlement to a desktop session by the rule.

Users who would be implicitly granted access when the filter is disabled can still be explicitly denied access using the excluded users filter.



Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-IncludedUsers**<User[]>

Changes the included users filter of the desktop rule, that is, the users and groups who are granted an entitlement to a desktop session by the rule.

If a user appears explicitly in the excluded users filter of the rule or is a member of a group that appears in the excluded users filter, no entitlement is granted whether or not the user appears in the included users filter.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PublishedName**<String>

Changes the name of the published desktop entitlement as seen by the user.

A null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RemoveExcludedUsers**<User[]>

Removes the specified users from the excluded users filter of the rule, that is, the users and groups who are explicitly denied an entitlement to a desktop session from this rule.

See the ExcludedUsers parameter for more information.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-RemoveIncludedUsers**<User[]>

Removes the specified users from the included users filter of the desktop rule, that is, the users and groups who are granted an entitlement to a desktop session by the rule.

See the IncludedUsers parameter for more information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecureIcaRequired**<Boolean>

Changes whether the desktop rule requires the SecureICA protocol for desktop sessions launched using the entitlement.

A null value indicates that the equivalent setting from the rule's desktop group is used.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SessionReconnection**<SessionReconnection>

Defines reconnection (roaming) behavior for sessions launched using this rule. Session reconnection control is an experimental and unsupported feature.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.EntitlementPolicyRule The desktop rule in the entitlement policy to be modified.

### Return Values

None or Citrix.Broker.Admin.SDK.EntitlementPolicyRule

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.EntitlementPolicyRule object.

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerEntitlementPolicyRule 'Temp Workers' -AddIncludedUsers office\contractors
```

Adds the user group OFFICE\contractors to the Temp Workers desktop rule of the entitlement policy. This grants all members of that group an entitlement to a desktop session in the rule's associated desktop group. The session properties of the desktops obtained using the rule are determined by the rule's other properties.

#### ----- EXAMPLE 2 -----

```
C:\PS> Set-BrokerEntitlementPolicyRule 'Temp Workers' -Enabled $false
```

Disables the Temp Workers desktop rule in the entitlement policy. This prevents further desktop sessions being launched using this rule until it is re-enabled. However, access to existing desktop sessions is not affected.

# Set-BrokerEntitlementPolicyRuleMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for EntitlementPolicyRule

## Syntax

```
Set-BrokerEntitlementPolicyRuleMetadata [-EntitlementPolicyRuleId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerEntitlementPolicyRuleMetadata [-EntitlementPolicyRuleId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerEntitlementPolicyRuleMetadata [-EntitlementPolicyRuleId] <Int32> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerEntitlementPolicyRuleMetadata [-InputObject] <EntitlementPolicyRule[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerEntitlementPolicyRuleMetadata [-InputObject] <EntitlementPolicyRule[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerEntitlementPolicyRuleMetadata [-EntitlementPolicyRuleName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerEntitlementPolicyRuleMetadata [-EntitlementPolicyRuleName] <String> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerEntitlementPolicyRuleMetadata cmdlet creates/updates metadata key-value pairs for EntitlementPolicyRule. The EntitlementPolicyRule can be specified by InputObject or piping.

## Related topics

## Parameters

**-EntitlementPolicyRuleId**<Int32>

Specifies the EntitlementPolicyRule object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-InputObject**<EntitlementPolicyRule[]>

Specifies the EntitlementPolicyRule objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-EntitlementPolicyRuleName**<String>

Specifies the EntitlementPolicyRule object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerEntitlementPolicyRule You can pipe the EntitlementPolicyRule to hold the new or updated metadata.

## Return Values

None or Citrix.Broker.Admin.SDK.BrokerEntitlementPolicyRule

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerEntitlementPolicyRule object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerEntitlementPolicyRuleMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the EntitlementPolicyRule whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerEntitlementPolicyRule | Set-BrokerEntitlementPolicyRuleMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the EntitlementPolicyRule in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerEntitlementPolicyRuleMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the EntitlementPolicyRule in the site whose name is 'objname'

# Set-BrokerHostingPowerAction

May 10, 2016  
Changes the priority of one or more pending power actions.

## Syntax

Set-BrokerHostingPowerAction [-InputObject] <HostingPowerAction[]> [-PassThru] [-ActualPriority <Int32>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-BrokerHostingPowerAction [-MachineName] <String> [-PassThru] [-ActualPriority <Int32>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The Set-BrokerHostingPowerAction cmdlet modifies an existing power action in the site's power action queue. The only property of power actions you can change, is the current priority of the action.

For a detailed description of the queuing mechanism, see 'help about\_Broker\_PowerManagement'.

## Related topics

- [Get-BrokerHostingPowerAction](#)
- [New-BrokerHostingPowerAction](#)
- [Remove-BrokerHostingPowerAction](#)

## Parameters

**-InputObject**<HostingPowerAction[]>

The power action whose priority is to be changed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-MachineName**<String>

Changes the priority of actions that are for machines whose name (of the form domain\machine) matches the specified string.

Required?	true
Default Value	



Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-ActualPriority**<Int32>

Specifies a new priority value for the action in the queue.

This priority is the current action priority; the 'base' or original priority for actions cannot be altered. Numerically lower priority values indicate more important actions that are processed in preference to actions with numerically higher priority settings.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host

name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.HostingPowerAction The power action whose priority is to be changed.

### Return Values

None or Citrix.Broker.Admin.SDK.HostingPowerAction

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.HostingPowerAction object.

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerHostingPowerAction -MachineName 'XD_VDA1' -ActualPriority 25
```

Sets the current priority of actions for the machine called 'XD\_VDA1' to 25. Numerically lower priority values indicate more important actions that will be processed in preference to actions with numerically higher priority settings.

# Set-BrokerHostingPowerActionMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for HostingPowerAction

## Syntax

```
Set-BrokerHostingPowerActionMetadata [-HostingPowerActionId] <Int64> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerHostingPowerActionMetadata [-HostingPowerActionId] <Int64> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerHostingPowerActionMetadata [-HostingPowerActionId] <Int64> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerHostingPowerActionMetadata [-InputObject] <HostingPowerAction[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerHostingPowerActionMetadata [-InputObject] <HostingPowerAction[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerHostingPowerActionMetadata [-HostingPowerActionName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerHostingPowerActionMetadata [-HostingPowerActionName] <String> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerHostingPowerActionMetadata cmdlet creates/updates metadata key-value pairs for HostingPowerAction. The HostingPowerAction can be specified by InputObject or piping.

## Related topics

## Parameters

**-HostingPowerActionId**<Int64>

Specifies the HostingPowerAction object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<HostingPowerAction[]>

Specifies the HostingPowerAction objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-HostingPowerActionName**<String>

Specifies the HostingPowerAction object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

Citrix.Broker.Admin.SDK.BrokerHostingPowerAction You can pipe the HostingPowerAction to hold the new or updated metadata.

## Return Values

None or Citrix.Broker.Admin.SDK.BrokerHostingPowerAction

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerHostingPowerAction object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerHostingPowerActionMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the HostingPowerAction whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerHostingPowerAction | Set-BrokerHostingPowerActionMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the HostingPowerAction in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerHostingPowerActionMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the HostingPowerAction in the site whose name is 'objname'

# Set-BrokerHypervisorAlertMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for HypervisorAlert

## Syntax

```
Set-BrokerHypervisorAlertMetadata [-HypervisorAlertId] <Int64> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerHypervisorAlertMetadata [-HypervisorAlertId] <Int64> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerHypervisorAlertMetadata [-HypervisorAlertId] <Int64> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerHypervisorAlertMetadata [-InputObject] <HypervisorAlert[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerHypervisorAlertMetadata [-InputObject] <HypervisorAlert[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerHypervisorAlertMetadata cmdlet creates/updates metadata key-value pairs for HypervisorAlert. The HypervisorAlert can be specified by InputObject or piping.

## Related topics

## Parameters

**-HypervisorAlertId**<Int64>

Specifies the HypervisorAlert object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-InputObject**<HypervisorAlert[]>

Specifies the HypervisorAlert objects whose Metadata is to be created/updated.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByValue)
------------------------	----------------

#### **-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False



Accept Pipeline Input?	false
------------------------	-------

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

Citrix.Broker.Admin.SDK.BrokerHypervisorAlert You can pipe the HypervisorAlert to hold the new or updated metadata.

Return Values

None or Citrix.Broker.Admin.SDK.BrokerHypervisorAlert

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerHypervisorAlert object.

Examples

----- **EXAMPLE 1** -----

C:\PS> Set-BrokerHypervisorAlertMetadata -InputObject \$obj-Uid -Name "MyMetadataName" -Value "1234"  
This command creates/updates the Metadata "MyMetadataName" key-value pair for the HypervisorAlert whose instance is pointed by \$obj-Uid

----- **EXAMPLE 2** -----

C:\PS> Get-BrokerHypervisorAlert | Set-BrokerHypervisorAlertMetadata -Name "MyMetadataName" -Value "1234"

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the HypervisorAlert in the site

# Set-BrokerHypervisorConnection

May 10, 2016

Sets the properties of a hypervisor connection.

## Syntax

```
Set-BrokerHypervisorConnection [-InputObject] <HypervisorConnection[]> [-PassThru] [-PreferredController <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerHypervisorConnection [-Name] <String> [-PassThru] [-PreferredController <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerHypervisorConnection cmdlet sets the properties on a hypervisor connection.

## Related topics

[Get-BrokerHypervisorConnection](#)

[New-BrokerHypervisorConnection](#)

[Remove-BrokerHypervisorConnection](#)

## Parameters

**-InputObject**<HypervisorConnection[]>

Specifies the hypervisor connection object to adjust.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the hypervisor connection object to adjust.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-PreferredController**<String>

Supplies the new value of the PreferredController property.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.

Accept Pipeline Input?	false
------------------------	-------

## Input Type

Citrix.Broker.Admin.SDK.HypervisorConnection You can pipe the hypervisor connection to be modified to Set-BrokerHypervisorConnection.

## Return Values

None or Citrix.Broker.Admin.SDK.HypervisorConnection

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.HypervisorConnection object.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS> $hvConn = Get-BrokerHypervisorConnection -PreferredController "oldController" -Name "Xen Server Connection"
```

```
c:\PS> Set-BrokerHypervisorConnection -InputObject $hvConn -PreferredController "newController"
```

Updates the preferred controller for a hypervisor connection.

# Set-BrokerHypervisorConnectionMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for HypervisorConnection

## Syntax

```
Set-BrokerHypervisorConnectionMetadata [-HypervisorConnectionId] <Int32> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-BrokerHypervisorConnectionMetadata [-HypervisorConnectionId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-BrokerHypervisorConnectionMetadata [-HypervisorConnectionId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-BrokerHypervisorConnectionMetadata [-InputObject] <HypervisorConnection[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-BrokerHypervisorConnectionMetadata [-InputObject] <HypervisorConnection[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-BrokerHypervisorConnectionMetadata [-HypervisorConnectionName] <String> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-BrokerHypervisorConnectionMetadata [-HypervisorConnectionName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerHypervisorConnectionMetadata cmdlet creates/updates metadata key-value pairs for HypervisorConnection. The HypervisorConnection can be specified by InputObject or piping.

## Related topics

## Parameters

**-HypervisorConnectionId**<Int32>

Specifies the HypervisorConnection object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<HypervisorConnection[]>

Specifies the HypervisorConnection objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-HypervisorConnectionName**<String>

Specifies the HypervisorConnection object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerHypervisorConnection You can pipe the HypervisorConnection to hold the new or updated metadata.



## Return Values

None or Citrix.Broker.Admin.SDK.BrokerHypervisorConnection

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerHypervisorConnection object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerHypervisorConnectionMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the HypervisorConnection whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerHypervisorConnection | Set-BrokerHypervisorConnectionMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the HypervisorConnection in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerHypervisorConnectionMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the HypervisorConnection in the site whose name is 'objname'

# Set-BrokerIconMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for Icon

## Syntax

```
Set-BrokerIconMetadata [-IconId] <Int32> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-BrokerIconMetadata [-IconId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Set-BrokerIconMetadata [-IconId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Set-BrokerIconMetadata [-InputObject] <Icon[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-
AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerIconMetadata [-InputObject] <Icon[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Set-BrokerIconMetadata cmdlet creates/updates metadata key-value pairs for Icon. The Icon can be specified by InputObject or piping.

## Related topics

## Parameters

**-IconId**<Int32>

Specifies the Icon object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Icon[]>

Specifies the Icon objects whose Metadata is to be created/updated.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByValue)
------------------------	----------------

#### **-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False

Accept Pipeline Input?	false
------------------------	-------

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

Citrix.Broker.Admin.SDK.BrokerIcon You can pipe the Icon to hold the new or updated metadata.

Return Values

None or Citrix.Broker.Admin.SDK.BrokerIcon

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerIcon object.

Examples

----- **EXAMPLE 1** -----

C:\PS> Set-BrokerIconMetadata -InputObject \$obj-Uid -Name "MyMetadataName" -Value "1234"  
This command creates/updates the Metadata "MyMetadataName" key-value pair for the Icon whose instance is pointed by \$obj-Uid

----- **EXAMPLE 2** -----

C:\PS> Get-BrokerIcon | Set-BrokerIconMetadata -Name "MyMetadataName" -Value "1234"

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the Icon in the site

# Set-BrokerLeaseMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for Lease

## Syntax

```
Set-BrokerLeaseMetadata [-LeaseId] <Int64> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerLeaseMetadata [-LeaseId] <Int64> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerLeaseMetadata [-LeaseId] <Int64> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerLeaseMetadata [-InputObject] <Lease[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerLeaseMetadata [-InputObject] <Lease[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerLeaseMetadata [-LeaseName] <String> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerLeaseMetadata [-LeaseName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerLeaseMetadata cmdlet creates/updates metadata key-value pairs for Lease. The Lease can be specified by InputObject or piping.

## Related topics

## Parameters

**-LeaseId**<Int64>

Specifies the Lease object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Lease[]>

Specifies the Lease objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LeaseName**<String>

Specifies the Lease object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type



Citrix.Broker.Admin.SDK.BrokerLease You can pipe the Lease to hold the new or updated metadata.

## Return Values

None or Citrix.Broker.Admin.SDK.BrokerLease

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerLease object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerLeaseMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the Lease whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerLease | Set-BrokerLeaseMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the Lease in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerLeaseMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the Lease in the site whose name is 'objname'

# Set-BrokerMachine

May 10, 2016

Sets properties on a machine.

## Syntax

```
Set-BrokerMachine [-InputObject] <Machine[]> [-PassThru] [-AssignedClientName <String>] [-AssignedIPAddress <String>] [-HostedMachineId <String>] [-HypervisorConnectionUid <Int32>] [-InMaintenanceMode <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachine [-MachineName] <String> [-PassThru] [-AssignedClientName <String>] [-AssignedIPAddress <String>] [-HostedMachineId <String>] [-HypervisorConnectionUid <Int32>] [-InMaintenanceMode <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerMachine cmdlet sets properties on a machine or set of machines. You can specify a single machine by name or multiple machine instances can be passed to the command by piping or using the -InputObject parameter.

## Related topics

[Get-BrokerMachine](#)

[New-BrokerMachine](#)

## Parameters

**-InputObject**<Machine[]>

The machine instances whose properties you want to set.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-MachineName**<String>

The machine whose properties you want to set.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

### **-AssignedClientName**<String>

Changes the client name assignment of the machine. Set this to \$null to remove the assignment. You can assign machines to multiple users, a single client IP address, or a single client name, but only to one of these categories at a time.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AssignedIPAddress**<String>

Changes the client IP address assignment of the machine. Set this to \$null to remove the assignment. You can assign machines to multiple users, a single client IP address, or a single client name, but only to one of these categories at a time.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-HostedMachineId**<String>

The unique ID by which the hypervisor recognizes the machine. This may only be set for VMs which are not provisioned by MCS.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-HypervisorConnectionUid<Int32>**

The hypervisor connection that runs the machine. This may only be set for VMs which are not provisioned by MCS.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-InMaintenanceMode<Boolean>**

Sets whether the machine is in maintenance mode or not. A machine in maintenance mode is not available for new sessions, and for managed machines all automatic power management is disabled.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId<Guid>**

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
-----------	-------

Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.Machine You can pipe in the machines whose properties you want to set.

## Return Values

None or Citrix.Broker.Admin.SDK.Machine

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.Machine object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerMachine -MachineName 'MyDomain\MyMachine' -HypervisorConnectionUid 3
```

This example specifies a machine by name and sets its hypervisor connection.

### ----- EXAMPLE 2 -----

```
C:\PS> $machines = Get-BrokerMachine -MachineName 'MyDomain*'
C:\PS> Set-BrokerMachine -InputObject $machines -HypervisorConnectionUid 3
```

This example finds all machines in domain MyDomain and sets their hypervisor connections.

### ----- EXAMPLE 3 -----

```
C:\PS> Get-BrokerMachine -MachineName 'MyDomain*' | Set-BrokerMachine -HypervisorConnectionUid 3
```

This example also finds all machines in domain MyDomain and sets their hypervisor connections.

# Set-BrokerMachineCatalog

May 10, 2016

Moves one or more machines into a different catalog.

## Syntax

```
Set-BrokerMachineCatalog [-InputObject] <Machine[]> [-CatalogUid] <Int32> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Set-BrokerMachineCatalog cmdlet is used to move machines into a different catalog. The following properties of the destination catalog must exactly match those of the machine's current catalog otherwise the command fails:

- o AllocationType
- o ProvisioningType
- o PersistUserChanges
- o SessionSupport
- o IsRemotePC
- o MinimumFunctionalLevel
- o PhysicalMachines

Changing a machine's catalog does not change the machine's desktop group membership. There is no effect on user sessions present on a machine if its catalog is changed.

## Related topics

[Set-BrokerMachine](#)

[New-BrokerCatalog](#)

## Parameters

**-InputObject**<Machine[]>

The machine instances that are being moved into a different catalog.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-CatalogUid<Int32>**

The unique identifier of the catalog into which the machines are being moved.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId<Guid>**

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### **Input Type**

Citrix.Broker.Admin.SDK.Machine You can pipe in the machines that are to be moved into a new catalog.

### **Return Values**

None

### **Examples**

----- **EXAMPLE 1** -----

```
C:\PS> $catalog = Get-BrokerCatalog MarketingMachines
```

```
C:\PS> $machines = Get-BrokerMachine -MachineName 'Marketing**'
```

```
C:\PS> Set-BrokerMachineCatalog $machines -CatalogUid $cat.Uid
```

This example finds all machines in domain Marketing and moves them into a catalog called MarketingMachines.



# Set-BrokerMachineCommandMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for MachineCommand

## Syntax

```
Set-BrokerMachineCommandMetadata [-MachineCommandId] <Int64> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineCommandMetadata [-MachineCommandId] <Int64> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineCommandMetadata [-MachineCommandId] <Int64> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineCommandMetadata [-InputObject] <MachineCommand[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineCommandMetadata [-InputObject] <MachineCommand[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerMachineCommandMetadata cmdlet creates/updates metadata key-value pairs for MachineCommand. The MachineCommand can be specified by InputObject or piping.

## Related topics

## Parameters

**-MachineCommandId**<Int64>

Specifies the MachineCommand object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<MachineCommand[]>

Specifies the MachineCommand objects whose Metadata is to be created/updated.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByValue)
------------------------	----------------

#### **-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False

Accept Pipeline Input?	false
------------------------	-------

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

Citrix.Broker.Admin.SDK.BrokerMachineCommand You can pipe the MachineCommand to hold the new or updated metadata.

Return Values

None or Citrix.Broker.Admin.SDK.BrokerMachineCommand

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerMachineCommand object.

Examples

----- **EXAMPLE 1** -----

C:\PS> Set-BrokerMachineCommandMetadata -InputObject \$obj-Uid -Name "MyMetadataName" -Value "1234"  
This command creates/updates the Metadata "MyMetadataName" key-value pair for the MachineCommand whose instance is pointed by \$obj-Uid

----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerMachineCommand | Set-BrokerMachineCommandMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the MachineCommand in the site

# Set-BrokerMachineConfiguration

May 10, 2016

Sets the properties of a machine configuration.

## Syntax

Set-BrokerMachineConfiguration [-InputObject] <MachineConfiguration[]> [-PassThru] [-Description <String>] [-Policy <Byte[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-BrokerMachineConfiguration [-Name] <String> [-PassThru] [-Description <String>] [-Policy <Byte[]>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Sets the properties of a machine configuration. The encoded settings data must only contain settings that match the SettingsGroup of the associated configuration slot. Use the SDK snap-in that matches the SettingsGroup of the associated configuration slot to generate new encoded settings data or modify existing settings values.

## Related topics

[New-BrokerMachineConfiguration](#)

[Get-BrokerMachineConfiguration](#)

[Rename-BrokerMachineConfiguration](#)

[Remove-BrokerMachineConfiguration](#)

[Add-BrokerMachineConfiguration](#)

## Parameters

**-InputObject**<MachineConfiguration[]>

Machine configuration to modify.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Name of machine configuration to modify.

Required?	true

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-Description**<String>

New description for the machine configuration.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Policy**<Byte[]>

New binary array of encoded settings data.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.MachineConfiguration Machine configuration to modify.

#### Return Values

None or Citrix.Broker.Admin.SDK.MachineConfiguration

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.MachineConfiguration object.

#### Examples

##### ----- **EXAMPLE 1** -----

Set-BrokerMachineConfiguration -Name "UPM\Finance Department" -Policy \$newPolicy  
Use the encoded settings binary data in \$newPolicy to update the machine configuration.

##### ----- **EXAMPLE 2** -----

Get-BrokerMachineConfiguration -Name "UPM\\*" | Set-BrokerMachineConfiguration -Description "User Profile Management"  
Assign the description "User Profile Management" to every machine configuration associated with the UPM slot.

# Set-BrokerMachineConfigurationMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for MachineConfiguration

## Syntax

```
Set-BrokerMachineConfigurationMetadata [-MachineConfigurationId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineConfigurationMetadata [-MachineConfigurationId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineConfigurationMetadata [-MachineConfigurationId] <Int32> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineConfigurationMetadata [-InputObject] <MachineConfiguration[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineConfigurationMetadata [-InputObject] <MachineConfiguration[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineConfigurationMetadata [-MachineConfigurationName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineConfigurationMetadata [-MachineConfigurationName] <String> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerMachineConfigurationMetadata cmdlet creates/updates metadata key-value pairs for MachineConfiguration. The MachineConfiguration can be specified by InputObject or piping.

## Related topics

## Parameters

**-MachineConfigurationId**<Int32>

Specifies the MachineConfiguration object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-InputObject**<MachineConfiguration[]>

Specifies the MachineConfiguration objects whose Metadata is to be created/updated.



Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-MachineConfigurationName**<String>

Specifies the MachineConfiguration object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerMachineConfiguration You can pipe the MachineConfiguration to hold the new or updated metadata.

## Return Values

None or Citrix.Broker.Admin.SDK.BrokerMachineConfiguration

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerMachineConfiguration object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerMachineConfigurationMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the MachineConfiguration whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerMachineConfiguration | Set-BrokerMachineConfigurationMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the MachineConfiguration in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerMachineConfigurationMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the MachineConfiguration in the site whose name is 'objname'

# Set-BrokerMachineMaintenanceMode

May 10, 2016

Sets whether the specified machine(s) are in maintenance mode.

## Syntax

```
Set-BrokerMachineMaintenanceMode [-InputObject] <Machine[]> [-MaintenanceMode] <Boolean> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The cmdlet can be used to set whether a machine is in maintenance mode or not. A machine in maintenance mode is not available for new sessions, and for managed machines all automatic power management is disabled.

There are times when it is necessary to disable desktops. You can do this by setting the `InMaintenanceMode` property of a desktop to `$true`. This puts it into maintenance mode. The broker excludes single-session desktops in maintenance mode from brokering decisions and does not start new sessions on them. Existing sessions are unaffected. For multi-session desktops in maintenance mode, reconnections to existing sessions are allowed, but no new sessions are created on the machine.

Desktops in maintenance mode are also excluded from automatic power management, although explicit power actions are still performed.

This cmdlet is equivalent to using the `Set-BrokerMachine` cmdlet to set the value of only the `InMaintenanceMode` property.

## Related topics

[Set-BrokerMachine](#)

## Parameters

**-InputObject**<Machine[]>

The machine instances whose `InMaintenanceMode` property you want to set.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-MaintenanceMode**<Boolean>

Sets whether the machine is in maintenance mode or not. A machine in maintenance mode is not available for new sessions, and for managed machines all automatic power management is disabled.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.Machine You can pipe in the machines whose properties you want to set.

#### Return Values

None

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> $machines = Get-BrokerMachine -MachineName 'MyDomain*'
```

```
C:\PS> Set-BrokerMachineMaintenanceMode -InputObject $machines $false
```

This example finds all machines in domain MyDomain and removes them from maintenance mode by setting their InMaintenanceMode property to false.

# Set-BrokerMachineMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for Machine

## Syntax

```
Set-BrokerMachineMetadata [-MachineId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineMetadata [-MachineId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineMetadata [-MachineId] <Int32> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineMetadata [-InputObject] <Machine[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineMetadata [-InputObject] <Machine[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineMetadata [-MachineName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerMachineMetadata [-MachineName] <String> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerMachineMetadata cmdlet creates/updates metadata key-value pairs for Machine. The Machine can be specified by InputObject or piping.

## Related topics

## Parameters

**-MachineId**<Int32>

Specifies the Machine object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-InputObject**<Machine[]>

Specifies the Machine objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-MachineName**<String>

Specifies the Machine object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type



Citrix.Broker.Admin.SDK.BrokerMachine You can pipe the Machine to hold the new or updated metadata.

## Return Values

None or Citrix.Broker.Admin.SDK.BrokerMachine

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerMachine object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerMachineMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the Machine whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerMachine | Set-BrokerMachineMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the Machine in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerMachineMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the Machine in the site whose name is 'objname'

# Set-BrokerPowerTimeScheme

May 10, 2016  
Modifies an existing power time scheme.

## Syntax

Set-BrokerPowerTimeScheme [-InputObject] <PowerTimeScheme[]> [-PassThru] [-DaysOfWeek <TimeSchemeDays>] [-DisplayName <String>] [-PeakHours <Boolean[]>] [-PoolSize <Int32[]>] [-PoolUsingPercentage <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-BrokerPowerTimeScheme [-Name] <String> [-PassThru] [-DaysOfWeek <TimeSchemeDays>] [-DisplayName <String>] [-PeakHours <Boolean[]>] [-PoolSize <Int32[]>] [-PoolUsingPercentage <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The Set-BrokerPowerTimeScheme cmdlet modifies an existing time scheme.

Each power time scheme is associated with a particular desktop group, and covers one or more days of the week, defining which hours of those days are considered peak times and which are off-peak times. In addition, the time scheme defines a pool size value for each hour of the day for the days of the week covered by the time scheme. No one desktop group can be associated with two or more time schemes that cover the same day of the week.

For more information about the power policy mechanism and pool size management, see 'help about\_Broker\_PowerManagement'.

## Related topics

- [Get-BrokerPowerTimeScheme](#)
- [Rename-BrokerPowerTimeScheme](#)
- [New-BrokerPowerTimeScheme](#)
- [Remove-BrokerPowerTimeScheme](#)

## Parameters

**-InputObject**<PowerTimeScheme[]>

The power time scheme to be changed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Identifies the power time scheme to be changed by name.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-DaysOfWeek**<TimeSchemeDays>

Changes the pattern of days of the week that the time scheme applies to. The pattern of days is specified as a single value or a list of values, where each value refers to either a single day or defined group of days.

Valid values are: Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Weekend and Weekdays.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DisplayName**<String>

Changes the name that is used by the DesktopStudio console when showing the time scheme.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PeakHours**<Boolean[]>

Changes the pattern of hours considered 'peak' vs 'off-peak' for the days covered by the time scheme. A 24-entry array of boolean truth values is expected, where the zeroth entry of the array relates to the time period between midnight and 0:59, the first relates to 1am to 1:59 and so on, with the last array element relating to 11 PM to 11:59. If the flag value is \$true it means the associated hour of the day is considered a peak time; if \$false it means that it is considered off-peak.

If fewer than 24 values are supplied, the final missing values are assumed to be 'false', and if more than 24 values are supplied, only the first 24 are used.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PoolSize<Int32[]>**

Changes the requested pool size of running machines at the various hours of the day for the days covered by the time scheme. A 24-entry array of integer values is expected, where the zeroth entry of the array relates to the time period between midnight and 0:59, the first relates to 1am to 1:59 and so on, with the last array element relating to 11 PM to 11:59. The pool size array entry values are either absolute numbers of machines that should be running or are a percentage of the machines in the desktop group that should be running during the associated hour of the day. A value of -1 in the array signifies that no management of the number of running machines should be attempted during the associated hour of the day.

If fewer than 24 values are supplied, the final missing values are assumed to be -1, and if more than 24 values are supplied, only the first 24 are used.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PoolUsingPercentage<Boolean>**

Changes whether pool size values from the 'PoolSize' array are evaluated as absolute numbers of running machines or as a percentage of machines in the desktop group that are to be maintained as running.

A value of \$true indicates that the pool size array values are percentages of total machines in the desktop group and a value of \$false indicates that the pool size array values are absolute numbers of machines to maintain as running.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId<Guid>**

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.PowerTimeScheme The power time scheme to be changed.

#### Return Values

None or Citrix.Broker.Admin.SDK.PowerTimeScheme

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.PowerTimeScheme object.

#### Examples

##### ----- EXAMPLE 1 -----

C:\PS> Set-BrokerPowerTimeScheme -Name 'Development Weekdays' -PoolSize ( 0..23 | %{ if (\$\_ -lt 8 -or \$\_ -gt 19) { 5 } else { 20 } } )  
Sets the pool size for the power time scheme named 'Development Weekdays' to be 20 for the time between 8am to 7:59pm, and 5 for other times.

# Set-BrokerPowerTimeSchemeMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for PowerTimeScheme

## Syntax

```
Set-BrokerPowerTimeSchemeMetadata [-PowerTimeSchemeId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerPowerTimeSchemeMetadata [-PowerTimeSchemeId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerPowerTimeSchemeMetadata [-PowerTimeSchemeId] <Int32> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerPowerTimeSchemeMetadata [-InputObject] <PowerTimeScheme[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerPowerTimeSchemeMetadata [-InputObject] <PowerTimeScheme[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerPowerTimeSchemeMetadata [-PowerTimeSchemeName] <String> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerPowerTimeSchemeMetadata [-PowerTimeSchemeName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerPowerTimeSchemeMetadata cmdlet creates/updates metadata key-value pairs for PowerTimeScheme. The PowerTimeScheme can be specified by InputObject or piping.

## Related topics

## Parameters

**-PowerTimeSchemeId**<Int32>

Specifies the PowerTimeScheme object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-InputObject**<PowerTimeScheme[]>

Specifies the PowerTimeScheme objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-PowerTimeSchemeName**<String>

Specifies the PowerTimeScheme object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type



Citrix.Broker.Admin.SDK.BrokerPowerTimeScheme You can pipe the PowerTimeScheme to hold the new or updated metadata.

## Return Values

None or Citrix.Broker.Admin.SDK.BrokerPowerTimeScheme

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerPowerTimeScheme object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerPowerTimeSchemeMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the PowerTimeScheme whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerPowerTimeScheme | Set-BrokerPowerTimeSchemeMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the PowerTimeScheme in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerPowerTimeSchemeMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the PowerTimeScheme in the site whose name is 'objname'

# Set-BrokerPrivateDesktop

May 10, 2016

Change the settings of a private desktop.

## Syntax

```
Set-BrokerPrivateDesktop [-InputObject] <PrivateDesktop[]> [-PassThru] [-AssignedClientName <String>] [-AssignedIPAddress <String>] [-ColorDepth <ColorDepth>] [-Description <String>] [-IconUid <Int32>] [-InMaintenanceMode <Boolean>] [-PublishedName <String>] [-SecureIcaRequired <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerPrivateDesktop [-MachineName] <String> [-PassThru] [-AssignedClientName <String>] [-AssignedIPAddress <String>] [-ColorDepth <ColorDepth>] [-Description <String>] [-IconUid <Int32>] [-InMaintenanceMode <Boolean>] [-PublishedName <String>] [-SecureIcaRequired <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Private desktops are automatically created when a machine is added to a desktop group with a DesktopKind of 'Private', and these inherit default properties. Use Set-BrokerPrivateDesktop to change the configuration settings of an existing private desktop.

To specify private desktops, you can choose whether to update by machine name, or by passing a PrivateDesktop or an array of PrivateDesktop objects. You can also use the Uid or an array of Uids instead.

You cannot modify many properties of a private desktop as these contain status information; for example DNSName, RegistrationState, and OSVersion.

Use Add- and Remove- cmdlets to update relationships between private desktops and other objects. For example, you can add a tag to a private desktop with:

```
Add-BrokerTag $tag -Desktop $desktop.Uid
```

Similarly, assign users to private desktops with:

```
Add-BrokerUser $user -PrivateDesktop $desktop
```

Many of the fields that can be set with this cmdlet can also be set with Set-BrokerMachine, such as MaintenanceMode. Using Set-BrokerMachine is preferred in these cases.

For more information about desktops, see about\_Broker\_Desktops; for more information about machines, see about\_Broker\_Machines.

## Related topics

[Get-BrokerMachine](#)

## Parameters

**-InputObject** <PrivateDesktop[]>

Specifies the desktop or array of desktops to modify. You can also use an integer Uid of the desktop instead.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-MachineName**<String>

Specifies the desktop to modify using its machine name (in the form 'domain\machine').

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-AssignedClientName**<String>

Changes the client name assignment of the desktop. Set this to \$null to remove the assignment. Desktops can be assigned to multiple users, a single IP address, or a single client name, but only to one of these categories at one time.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AssignedIPAddress**<String>

Changes the IP address assignment of the desktop. Set this to \$null to remove the assignment. Desktops can be assigned to multiple users, a single IP address, or a single client name, but only to one of these categories at one time.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ColorDepth**<ColorDepth>

Changes the color depth connections to this desktop should use.

Valid values are \$null, FourBit, EightBit, SixteenBit, and TwentyFourBit. A value of \$null results in the desktop group value being used instead.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Description**<String>

Changes the description of the desktop. This is seen only by Citrix Administrators and is not visible to users.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IconUid**<Int32>

Changes the icon displayed for this desktop. When this setting is \$null, the icon displayed is determined by the desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-InMaintenanceMode**<Boolean>

Changes the maintenance mode setting of a desktop. When a desktop is in maintenance mode, it is not included as a candidate when brokering new sessions, and it does not participate in automatic power management (idle pool); however, it still responds to explicit power operations.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PublishedName**<String>

Changes the name displayed to the user for this desktop. When this setting is \$null, the name displayed is determined by the PublishedName of the desktop group.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecureIcaRequired**<Boolean>

Changes whether or not SecureICA is required for connections to this desktop. When this setting is \$null, the SecureIcaRequired setting from the desktop group is used.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.PrivateDesktop You can pipe PrivateDesktop objects into this cmdlet instead of on the command line with the -InputObject parameter.

#### Return Values

None or Citrix.Broker.Admin.SDK.PrivateDesktop

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.PrivateDesktop object.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Set-BrokerPrivateDesktop DOMAIN\Machine1 -ColorDepth SixteenBit
```

Change the color depth of Machine1 to be 16-bit.

##### ----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerPrivateDesktop -InMaintenanceMode $true | Set-BrokerPrivateDesktop -InMaintenanceMode $false
```

Bring all private desktops currently in maintenance mode back into normal service.

# Set-BrokerRebootCycleMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for RebootCycle

## Syntax

```
Set-BrokerRebootCycleMetadata [-RebootCycleId] <Int64> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerRebootCycleMetadata [-RebootCycleId] <Int64> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerRebootCycleMetadata [-RebootCycleId] <Int64> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerRebootCycleMetadata [-InputObject] <RebootCycle[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerRebootCycleMetadata [-InputObject] <RebootCycle[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerRebootCycleMetadata cmdlet creates/updates metadata key-value pairs for RebootCycle. The RebootCycle can be specified by InputObject or piping.

## Related topics

## Parameters

**-RebootCycleId**<Int64>

Specifies the RebootCycle object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-InputObject**<RebootCycle[]>

Specifies the RebootCycle objects whose Metadata is to be created/updated.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByValue)
------------------------	----------------

#### **-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False



Accept Pipeline Input?	false
------------------------	-------

#### **-LoggingId<Guid>**

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.BrokerRebootCycle You can pipe the RebootCycle to hold the new or updated metadata.

#### Return Values

None or Citrix.Broker.Admin.SDK.BrokerRebootCycle

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerRebootCycle object.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerRebootCycleMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the RebootCycle whose instance is pointed by \$obj-Uid

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerRebootCycle | Set-BrokerRebootCycleMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the RebootCycle in the site

# Set-BrokerRebootSchedule

May 10, 2016  
Updates the values of one or more desktop group reboot schedules.

## Syntax

Set-BrokerRebootSchedule [-InputObject <RebootSchedule[]>] [-PassThru] [-Day <RebootScheduleDays>] [-Enabled <Boolean>] [-Frequency <RebootScheduleFrequency>] [-RebootDuration <Int32>] [-StartTime <TimeSpan>] [-WarningDuration <Int32>] [-WarningMessage <String>] [-WarningTitle <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-BrokerRebootSchedule [-DesktopGroupName <String>] [-PassThru] [-Day <RebootScheduleDays>] [-Enabled <Boolean>] [-Frequency <RebootScheduleFrequency>] [-RebootDuration <Int32>] [-StartTime <TimeSpan>] [-WarningDuration <Int32>] [-WarningMessage <String>] [-WarningTitle <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The Set-BrokerRebootSchedule cmdlet is used to alter the settings of an existing desktop group reboot schedule.

## Related topics

[Get-BrokerRebootSchedule](#)

[New-BrokerRebootSchedule](#)

[Remove-BrokerRebootSchedule](#)

## Parameters

**-InputObject**<RebootSchedule[]>

The reboot schedule to be modified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-DesktopGroupName**<String>

The name of the desktop group whose reboot schedule is to be modified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-Day**<RebootScheduleDays>

For weekly schedules, the day of the week on which the scheduled reboot-cycle starts (one of Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday).

--	--

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Enabled**<Boolean>

Boolean that indicates if the reboot schedule is to be enabled or disabled.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Frequency**<RebootScheduleFrequency>

Frequency with which this schedule runs (either Weekly or Daily).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RebootDuration**<Int32>

Approximate maximum number of minutes over which the scheduled reboot cycle runs.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-StartTime**<TimeSpan>

Time of day at which the scheduled reboot cycle starts (HH:MM).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-WarningDuration**<Int32>

Time prior to the initiation of a machine reboot at which warning message is displayed in all user sessions on that machine. If the warning duration is zero then no message is displayed.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-WarningMessage<String>**

Warning message displayed in user sessions on a machine scheduled for reboot. If the message is blank then no message is displayed.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-WarningTitle<String>**

The window title used when showing the warning message in user sessions on a machine scheduled for reboot.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LoggingId<Guid>**

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress<String>**

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

**Input Type**

Citrix.Broker.Admin.SDK.RebootSchedule Reboot schedules may be specified through pipeline input.

**Return Values**

None

**Examples**

----- **EXAMPLE 1** -----

C:\PS> Set-BrokerRebootSchedule -DesktopGroupName Accounting -WarningMessage "Save your work" -WarningDuration 10 -WarningTitle "WARNING: Reboot pending"  
Sets the reboot schedule for the desktop group named Accounting to display a message with the title "WARNING: Reboot pending" and body "Save your work" ten minutes prior to rebooting each machine. The message is displayed in every user session on that machine.

----- **EXAMPLE 2** -----

C:\PS> Get-BrokerRebootSchedule -Frequency Weekly | Set-BrokerRebootSchedule -Day Friday  
Sets all weekly reboot schedules to run on Friday.

# Set-BrokerRemotePCAccount

May 10, 2016

Modify one or more RemotePCAccounts.

## Syntax

```
Set-BrokerRemotePCAccount [-InputObject] <RemotePCAccount[]> [-PassThru] [-AllowSubfolderMatches <Boolean>] [-MachinesExcluded <String[]>] [-MachinesIncluded <String[]>] [-OU <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Modify one or more RemotePCAccounts.

## Related topics

[Get-BrokerRemotePCAccount](#)

[New-BrokerRemotePCAccount](#)

[Remove-BrokerRemotePCAccount](#)

## Parameters

**-InputObject**<RemotePCAccount[]>

Specifies the RemotePCAccounts to modify.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-AllowSubfolderMatches**<Boolean>

When true a machine matches this RemotePCAccount if the AD computer is in the container specified by the OU property, or within a child container of the OU.

When false the AD computer object only matches if it is directly in the AD container specified by the OU property.

This property is not meaningful when OU has the special value 'any'.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachinesExcluded**<String[]>

MachinesExcluded specifies a set of strings that can include asterisk wildcards. If a machine name matches any entries in MachinesExcluded then it cannot match with this RemotePCAccount regardless of whether there is a MachinesIncluded match.

Matches are performed against the domain name joined with the machine name by a backslash (DOMAIN\MACHINE), e.g.:

DOMAIN1\M\*

DOMAIN\*\M\*

\*\M\*

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MachinesIncluded**<String[]>

MachinesIncluded specifies a set of strings that can include asterisk wildcards. A machine may only match with this RemotePCAccount if it matches a MachinesIncluded entry and does not match any MachinesExcluded entries.

Matches are performed against the domain name joined with the machine name by a backslash (DOMAIN\MACHINE), e.g.:

DOMAIN1\M\*

DOMAIN\*\M\*

\*\M\*

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-OU**<String>

Specifies the DN of an AD container, or has the special value 'any'.

When an AD container is specified a machine may only match with the RemotePCAccount when the AD computer object is located relative to the OU.

When 'any' is specified the location of the AD computer object is ignored for purposes of matching this RemotePCAccount. The machine must still meet the MachinesIncluded and MachinesExcluded filters for a match to occur.

In the event that a machine matches with multiple RemotePCAccounts then the RemotePCAccount OU with the longest canonical name takes precedence. The special 'any' OU is treated as lowest priority.

Note that the OU value of every RemotePCAccount must be unique, and this includes only one 'any' entry being permitted.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
-----------	-------



Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.RemotePCAccount You can pipe the RemotePCAccounts to be modified into this cmdlet.

## Return Values

None or Citrix.Broker.Admin.SDK.RemotePCAccount

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.RemotePCAccount object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerRemotePCAccount | Set-BrokerRemotePCAccount -MachinesExcluded @('DOMAIN42*')
Make all RemotePCAccounts filter out machines from DOMAIN42.
```

# Set-BrokerSession

May 10, 2016

Sets properties of a session.

## Syntax

```
Set-BrokerSession [-InputObject] <Session[]> [-PassThru] [-Hidden <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-BrokerSession [-SessionKey] <Guid> [-PassThru] [-Hidden <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Set-BrokerSession cmdlet sets properties on a session or set of sessions. You can specify a single session by Uid or SessionKey (Guid) or multiple session instances can be passed to the command by piping or using the -InputObject parameter.

## Related topics

[Get-BrokerSession](#)

## Parameters

**-InputObject**<Session[]>

The session instances whose properties you want to set.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-SessionKey**<Guid>

The session key (Guid) of the session whose properties you want to set.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-Hidden**<Boolean>

Changes whether the session is hidden or not. Hidden sessions are treated as though they do not exist when brokering sessions; a hidden session cannot be reconnected to, but a new session may be launched using the same entitlement.

A session may be hidden automatically by the system when an attempt is made to reconnect to a session that is unreachable due to, for example, the failure of a VM's hypervisor. This allows a new session to be brokered provided that other machines in the same desktop group are still available. If the original session still exists after the hypervisor is restored then it must be unhidden to allow the user to reconnect to it.

Only sessions for shared desktops or applications can be hidden or unhidden. Private desktop or application sessions can never be hidden.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host

name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.Session You can pipe in the sessions whose properties you want to set.

## Return Values

None or Citrix.Broker.Admin.SDK.Session

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.Session object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> $sessions = Get-BrokerSession -User ACCOUNTS\JohnSmith -Hidden $true
```

```
C:\PS> $sessions | Set-BrokerSession -Hidden $false
```

Finds all sessions in the site for user John Smith that have been hidden, and makes them available for reconnection again.

# Set-BrokerSessionLinger

May 10, 2016

Updates the values of one or more desktop group session linger settings.

## Syntax

```
Set-BrokerSessionLinger [-InputObject] <SessionLinger[]> [-PassThru] [-Enabled <Boolean>] [-MaxAverageLoadThreshold <Int32>] [-MaxLoadPerMachineThreshold <Int32>] [-MaxTimeBeforeDisconnect <TimeSpan>] [-MaxTimeBeforeTerminate <TimeSpan>] [-UserFilterEnabled <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerSessionLinger [-DesktopGroupName] <String> [-PassThru] [-Enabled <Boolean>] [-MaxAverageLoadThreshold <Int32>] [-MaxLoadPerMachineThreshold <Int32>] [-MaxTimeBeforeDisconnect <TimeSpan>] [-MaxTimeBeforeTerminate <TimeSpan>] [-UserFilterEnabled <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerSessionLinger cmdlet is used to alter the settings of an existing desktop group session linger setting.

## Related topics

[New-BrokerSessionLinger](#)

[Get-BrokerSessionLinger](#)

[Remove-BrokerSessionLinger](#)

## Parameters

**-InputObject** <SessionLinger[]>

The session linger to be modified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-DesktopGroupName** <String>

The name of the desktop group whose session linger setting is to be modified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-Enabled**<Boolean>

Boolean that indicates if the session linger setting is to be enabled or disabled.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxAverageLoadThreshold**<Int32>

Specifies the average load threshold across the desktop group. When the threshold hits, lingering sessions across the group be terminated to reduce load. Sessions that have been lingering the longest will be chosen first.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxLoadPerMachineThreshold**<Int32>

Specifies the maximum load threshold per machine in the desktop group. When the threshold hits, lingering sessions on each loaded machine will be terminated to reduce load. Sessions that have been lingering the longest will be chosen first.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-MaxTimeBeforeDisconnect**<TimeSpan>

Specifies the time by which a lingering session will be disconnected. The disconnect time cannot be greater than the terminate timer (if enabled). When the disconnect and terminate times are the same, the terminate time takes precedence.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxTimeBeforeTerminate**<TimeSpan>

Specifies the time by which a lingering session will be terminated. When the disconnect and terminate times are the same, the terminate time takes precedence.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UserFilterEnabled**<Boolean>

Specifies whether the session linger's user filter is enabled or disabled. Where the user filter is enabled, lingering is enabled only to users who appear in the filter (either explicitly or by virtue of group membership).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.SessionLinger Session linger settings may be specified through pipeline input.

#### Return Values

None or Citrix.Broker.Admin.SDK.SessionLinger

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.SessionLinger object.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Set-BrokerSessionLinger -DesktopGroupName Accounting -MaxTimeBeforeDisconnect 0:10
```

Sets the disconnect time for the session linger setting associated with desktop group named Accounting.



# Set-BrokerSessionMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for Session

## Syntax

```
Set-BrokerSessionMetadata [-SessionId] <Int64> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerSessionMetadata [-SessionId] <Int64> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerSessionMetadata [-SessionId] <Int64> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerSessionMetadata [-InputObject] <Session[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerSessionMetadata [-InputObject] <Session[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerSessionMetadata [-SessionName] <Guid> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerSessionMetadata [-SessionName] <Guid> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerSessionMetadata cmdlet creates/updates metadata key-value pairs for Session. The Session can be specified by InputObject or piping.

## Related topics

## Parameters

**-SessionId**<Int64>

Specifies the Session object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Session[]>

Specifies the Session objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-SessionName**<Guid>

Specifies the Session object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

Citrix.Broker.Admin.SDK.BrokerSession You can pipe the Session to hold the new or updated metadata.

## Return Values

None or Citrix.Broker.Admin.SDK.BrokerSession

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerSession object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerSessionMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the Session whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerSession | Set-BrokerSessionMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the Session in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerSessionMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the Session in the site whose name is 'objname'

# Set-BrokerSessionPreLaunch

May 10, 2016

Updates the values of one or more desktop group session pre-launch settings.

## Syntax

```
Set-BrokerSessionPreLaunch [-InputObject] <SessionPreLaunch[]> [-PassThru] [-Enabled <Boolean>] [-MaxAverageLoadThreshold <Int32>] [-MaxLoadPerMachineThreshold <Int32>] [-MaxTimeBeforeDisconnect <TimeSpan>] [-MaxTimeBeforeTerminate <TimeSpan>] [-UserFilterEnabled <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerSessionPreLaunch [-DesktopGroupName] <String> [-PassThru] [-Enabled <Boolean>] [-MaxAverageLoadThreshold <Int32>] [-MaxLoadPerMachineThreshold <Int32>] [-MaxTimeBeforeDisconnect <TimeSpan>] [-MaxTimeBeforeTerminate <TimeSpan>] [-UserFilterEnabled <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerSessionPreLaunch cmdlet is used to alter the settings of an existing desktop group session pre-launch setting.

## Related topics

[New-BrokerSessionPreLaunch](#)

[Get-BrokerSessionPreLaunch](#)

[Remove-BrokerSessionPreLaunch](#)

## Parameters

**-InputObject**<SessionPreLaunch[]>

The session pre-launch to be modified.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-DesktopGroupName**<String>

The name of the desktop group whose session pre-launch setting is to be modified.

Required?	true

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-Enabled**<Boolean>

Boolean that indicates if the session pre-launch setting is to be enabled or disabled.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxAverageLoadThreshold**<Int32>

Specifies the average load threshold across the desktop group. When the threshold hits, pre-launched sessions across the group be terminated to reduce load. Sessions that have been pre-launched the longest will be chosen first.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxLoadPerMachineThreshold**<Int32>

Specifies the maximum load threshold per machine in the desktop group. When the threshold hits, pre-launched sessions on each loaded machine will be terminated to reduce load. Sessions that have been pre-launched the longest will be chosen first.

--	--

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxTimeBeforeDisconnect**<TimeSpan>

Specifies the time by which a pre-launched session will be disconnected. The disconnect time cannot be greater than the terminate timer (if enabled). When the disconnect and terminate times are the same, the terminate time takes precedence.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-MaxTimeBeforeTerminate**<TimeSpan>

Specifies the time by which a pre-launched session will be terminated. When the disconnect and terminate times are the same, the terminate time takes precedence.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-UserFilterEnabled**<Boolean>

Specifies whether the session pre-launch's user filter is enabled or disabled. Where the user filter is enabled, pre-launch is enabled only to users who appear in the filter (either explicitly or by virtue of group membership).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop

Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.SessionPreLaunch Session pre-launch settings may be specified through pipeline input.

#### Return Values

None or Citrix.Broker.Admin.SDK.SessionPreLaunch

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.SessionPreLaunch object.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Set-BrokerSessionPreLaunch -DesktopGroupName Accounting -MaxTimeBeforeDisconnect 0:10
```

Sets the disconnect time for the session pre-launch setting associated with desktop group named Accounting.



# Set-BrokerSharedDesktop

May 10, 2016  
Change the settings of a shared desktop.

## Syntax

Set-BrokerSharedDesktop [-InputObject] <SharedDesktop[]> [-PassThru] [-InMaintenanceMode <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-BrokerSharedDesktop [-MachineName] <String> [-PassThru] [-InMaintenanceMode <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Shared desktops are automatically created when a machine is added to a desktop group with a DesktopKind of 'Shared', and these inherit default properties. Use Set-BrokerSharedDesktop to change the configuration settings of an existing shared desktop.

To specify shared desktops, you can choose whether to update by machine name or by passing a SharedDesktop or an array of SharedDesktop objects. You can also use the Uid or an array of Uids instead.

Most properties of a shared desktop cannot be modified as these contain status information; for example DNSName, RegistrationState, and OSVersion. You can change only the maintenance mode setting with this cmdlet.

Many of the properties that can be set with Set-BrokerSharedDesktop can be set by using Set-BrokerMachine (e.g. InMaintenanceMode). Using the Set-BrokerMachine cmdlet, where possible, is the preferred behaviour.

See about\_Broker\_Desktops for more information about desktops.

## Related topics

[Get-BrokerSharedDesktop](#)

## Parameters

**-InputObject**<SharedDesktop[]>

Specifies the desktop or array of desktops to modify. You can also use an integer Uid of the desktop instead.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-MachineName**<String>

Specifies the desktop to modify using its machine name (in the form 'domain\machine').

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-InMaintenanceMode**<Boolean>

Changes the maintenance mode setting of a desktop. When a desktop is in maintenance mode, it is not included as a candidate when brokering new sessions, and it does not participate in automatic power management (idle pool); however, it still responds to explicit power operations.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.SharedDesktop You can pipe SharedDesktop objects into this cmdlet instead of on the command line with the -InputObject parameter.

## Return Values

None or Citrix.Broker.Admin.SDK.SharedDesktop

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.SharedDesktop object.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Set-BrokerSharedDesktop DOMAIN\Machine1 -InMaintenanceMode $true
```

Put the Machine1 shared desktop into maintenance mode.

### ----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerSharedDesktop -InMaintenanceMode $true | Set-BrokerSharedDesktop -InMaintenanceMode $false
```

Bring all shared desktops currently in maintenance mode back into normal service.

# Set-BrokerSite

May 10, 2016

Changes the overall settings of the current XenDesktop broker site.

## Syntax

```
Set-BrokerSite [-PassThru] [-BaseOU <Guid>] [-ColorDepth <ColorDepth>] [-ConnectionLeasingEnabled <Boolean>] [-DesktopGroupIconUid <Int32>] [-DnsResolutionEnabled <Boolean>] [-SecureIcaRequired <Boolean>] [-TrustRequestsSentToTheXmlServicePort <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerSite cmdlet modifies properties of the current broker site.

The broker site is a top-level, logical representation of the XenDesktop site, from the perspective of the brokering services running within the site. It defines various site-wide default attributes used by the brokering services.

A XenDesktop installation has only a single broker site instance.

## Related topics

[Get-BrokerSite](#)

[New-BrokerIcon](#)

## Parameters

**-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-BaseOU**<Guid>

Changes the objectGUID property identifying the base OU in Active Directory used for desktop registrations. For sites using only registry-based discovery (the default) this value is \$null.

Any desktop attempting to register through a different OU from the one specified here is rejected. Note that desktops configured for registry-based discovery can register with the site, even if a BaseOU value is specified.

Information held in Active Directory is not modified by changing this value.

Typically, this property is changed only by using the Set-ADControllerDiscovery.ps1 script.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ColorDepth**<ColorDepth>

Changes the default color depth for new desktop groups, if no color depth is specified explicitly when a group is created. Changing this default has no impact on the color depths used already by existing groups.

Valid values are FourBit, EightBit, SixteenBit, and TwentyFourBit.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ConnectionLeasingEnabled**<Boolean>

Enabled or disable connection leasing on the site.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DesktopGroupIconUid**<Int32>

Changes the default desktop icon used for new desktop groups if no icon is specified explicitly when a group is created. Changing this default has no impact on the icons used already by existing groups.

The specified icon must already have been added to the site using New-BrokerIcon.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DnsResolutionEnabled**<Boolean>

Changes whether ICA files returned by a broker service to a user device contain the numeric IP address or the DNS name of the desktop machine to which a session should be established.

With the default value (\$false), ICA files will always contain a numeric IP address. To have DNS names appear in the ICA files, set the value to \$true.

Even when DNS resolution is enabled (\$true), IP addresses may still appear in ICA files. The reasons for this include, for example, that the broker service is unable to obtain a DNS name for the target machine, or that Web Interface is configured to always use numeric IP addresses in this context.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SecureIcaRequired**<Boolean>

Changes the default SecureICA usage requirements for new desktop groups if no SecureICA setting is specified explicitly when a group is created. Changing this default has no impact on the SecureICA usage requirements of existing groups.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-TrustRequestsSentToTheXmlServicePort**<Boolean>

Changes whether the XML Service (as used by Web Interface) implicitly trusts the originator of requests it receives, or whether it fully authenticates them.

With the default value (\$false), full authentication checks are performed. However, you must enable this setting (\$true) to allow support for "Pass-through" authentication, and/or connections routed through Access Gateway.

If this setting is enabled, you must ensure that controllers running the brokering services are securely firewalled.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

None You cannot pipe input into this cmdlet.

### Return Values

None or Citrix.Broker.Admin.SDK.Site

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.Site object.

### Examples

#### ----- **EXAMPLE 1** -----

```
C:\PS> Set-BrokerSite -ColorDepth SixteenBit
```

Specifies that any new desktop groups created, where a color depth value is not specified, default to using 16-bit color depth for user sessions to desktops or applications within that group.

# Set-BrokerSiteMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for Site

## Syntax

```
Set-BrokerSiteMetadata -Name <String> -Value <String> [[-InputObject] <Site[]>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerSiteMetadata -Map <PSObject> [[-InputObject] <Site[]>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerSiteMetadata -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerSiteMetadata -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerSiteMetadata -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerSiteMetadata cmdlet creates/updates metadata key-value pairs for Site. The Site can be specified by InputObject or piping.

## Related topics

## Parameters

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	



Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-InputObject**<Site[]>

Specifies the Site objects whose Metadata is to be created/updated.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.BrokerSite You can pipe the Site to hold the new or updated metadata.

### Return Values

None or Citrix.Broker.Admin.SDK.BrokerSite

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerSite object.

### Examples

#### ----- **EXAMPLE 1** -----

```
C:\PS> Set-BrokerSiteMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the Site

#### ----- **EXAMPLE 2** -----

```
C:\PS> Get-BrokerSite | Set-BrokerSiteMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234"

# Set-BrokerTagMetadata

May 10, 2016

Creates/Updates metadata key-value pairs for Tag

## Syntax

```
Set-BrokerTagMetadata [-TagId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerTagMetadata [-TagId] <Int32> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerTagMetadata [-TagId] <Int32> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerTagMetadata [-InputObject] <Tag[]> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerTagMetadata [-InputObject] <Tag[]> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerTagMetadata [-TagName] <String> -Map <PSObject> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-BrokerTagMetadata [-TagName] <String> -Name <String> -Value <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-BrokerTagMetadata cmdlet creates/updates metadata key-value pairs for Tag. The Tag can be specified by InputObject or piping.

## Related topics

## Parameters

**-TagId**<Int32>

Specifies the Tag object whose Metadata is to be created/updated by ID.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-InputObject**<Tag[]>

Specifies the Tag objects whose Metadata is to be created/updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-TagName**<String>

Specifies the Tag object whose Metadata is to be created/updated by name.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Name**<String>

Specifies the name of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Value**<String>

Specifies the value of the Metadata member to be created/updated

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Map**<PSObject>

Specifies a hashtable containing name/value pairs to be used to create or update Metadata members

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-PassThru**<SwitchParameter>

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it returns the affected record.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

Input Type

Citrix.Broker.Admin.SDK.BrokerTag You can pipe the Tag to hold the new or updated metadata.

## Return Values

None or Citrix.Broker.Admin.SDK.BrokerTag

This cmdlet does not generate any output, unless you use the PassThru parameter, in which case it generates a Citrix.Broker.Admin.SDK.BrokerTag object.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Set-BrokerTagMetadata -InputObject $obj-Uid -Name "MyMetadataName" -Value "1234"
```

This command creates/updates the Metadata "MyMetadataName" key-value pair for the Tag whose instance is pointed by \$obj-Uid

### ----- EXAMPLE 2 -----

```
C:\PS> Get-BrokerTag | Set-BrokerTagMetadata -Name "MyMetadataName" -Value "1234"
```

This command creates/updates metadata key "MyMetadataName" with the value "1234" for all the Tag in the site

### ----- EXAMPLE 3 -----

```
C:\PS> @{ 'name1' = 'value1'; 'name2' = 'value2' } | Set-BrokerTagMetadata 'objname'
```

This command creates/updates two metadata keys "name1" and "name2" with the values "value1" and "value2" respectively for the Tag in the site whose name is 'objname'

# Start-BrokerCatalogPvdImagePrepare

May 10, 2016

Start the PVD Image prepare process in the Broker for the machines in the specified catalog(s).

## Syntax

```
Start-BrokerCatalogPvdImagePrepare [-InputObject] <Catalog[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Start-BrokerCatalogPvdImagePrepare cmdlet instructs the Broker to request the Personal VDisk (PVD) preparation process for all of the machines in the specified catalog(s). The process begins when each machine is subsequently powered off, at which point brokering of user desktop sessions is suspended as well as regular machine power operations until the process completes. Only catalogs with a PersistUserChanges value of OnPvd are supported by this cmdlet.

## Related topics

[Start-BrokerMachinePvdImagePrepare](#)

## Parameters

**-InputObject**<Catalog[]>

The catalog(s) holding the machines on which to start the PVD Image Preparation process.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.Catalog You can pipe in the catalogs on which to start the PVD image preparation process.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerCatalog 'MyCatalog' | Start-BrokerCatalogPvdImagePrepare
```

Instruct the Broker that the Personal VDisk preparation process will start the next time the machines in the specified catalog are powered off.



# Start-BrokerMachinePvdImagePrepare

May 10, 2016

Start the PVD Image prepare process in the Broker for the specified machine(s).

## Syntax

```
Start-BrokerMachinePvdImagePrepare [-InputObject] <Machine[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Start-BrokerMachinePvdImagePrepare cmdlet instructs the Broker to request the Personal VDisk (PVD) preparation process for the specified machine(s). The process begins when each machine is subsequently powered off, at which point brokering of user desktop sessions is suspended as well as regular machine power operations until the process completes. Only machines in catalogs with a PersistUserChanges value of OnPvd are supported by this cmdlet.

## Related topics

[Start-BrokerCatalogPvdImagePrepare](#)

## Parameters

**-InputObject**<Machine[]>

The machine(s) to start the PVD Image Preparation process on.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.Machine You can pipe in the machines to start the PVD image preparation process on.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerMachine 'MyMachine' | Start-BrokerMachinePvdImagePrepare
```

Instruct the Broker that the Personal VDisk preparation process will start the next time the specified machine(s) are powered off.

# Start-BrokerNaturalRebootCycle

May 10, 2016

Reboots all machines from the specified catalog when they are not in use.

## Syntax

```
Start-BrokerNaturalRebootCycle [-InputObject] <Catalog[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Creating a natural reboot cycle for catalog ensures that all machines in the catalog are running the most recent image for the catalog and that all PvD image updates have been performed.

The machines are rebooted in a non disruptive manner, allowing machines that are in use to continue working and be restarted only after they become idle.

## Related topics

None

## Parameters

**-InputObject**<Catalog[]>

Reboots all machines from this input catalog. The catalogs can be specified using UID values, name values (including wildcards) or catalog objects.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.Catalog Catalogs may be specified through pipeline input. The catalogs can be specified using UID values, name values (including wildcards) or catalog objects

## Return Values

None

## Notes

Natural reboot cycles do not apply to non-power managed and multi session catalogs

## Examples

### ----- **EXAMPLE 1** -----

```
$c = Get-BrokerCatalog -Uid 1
```

```
Start-BrokerNaturalRebootCycle -InputObject $c
```

The above code applies a natural reboot cycle on catalog with Id 1

# Start-BrokerRebootCycle

May 10, 2016

Creates and starts a reboot cycle for each desktop group that contains machines from the specified catalog.

## Syntax

```
Start-BrokerRebootCycle [-InputObject] <Catalog[]> -RebootDuration <Int32> [-WarningDuration <Int32>] [-WarningTitle <String>] [-WarningMessage <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Start-BrokerRebootCycle cmdlet is used to create and start a reboot cycle for each desktop group that contains machines from the specified catalog. For a given desktop group, only the machines from the target catalog are rebooted and any machines from other catalogs are not rebooted.

Creating a reboot cycle for catalog ensures that all machines in the catalog are running the most recent image for the catalog and that all Pvd image updates have been performed.

## Related topics

[Stop-BrokerRebootCycle](#)

[Get-BrokerRebootCycle](#)

## Parameters

**-InputObject**<Catalog[]>

Creates a reboot cycle for each desktop group that contains machines from this input catalog SDK object. The catalogs can be specified using UID values, name values (including wildcards) or catalog objects.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-RebootDuration**<Int32>

Approximate maximum duration in minutes over which the reboot cycle runs.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-WarningDuration**<Int32>

Time in minutes prior to a machine reboot at which a warning message is displayed in all user sessions on that machine. If the warning duration value is zero then no message is displayed.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-WarningTitle**<String>

The window title used when showing the warning message in user sessions on a machine scheduled for reboot.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-WarningMessage**<String>

Warning message displayed in user sessions on a machine scheduled for reboot. If the message is blank then no message is displayed.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Broker.Admin.SDK.Catalog Catalogs may be specified through pipeline input. The catalogs can be specified using UID values, name values (including wildcards) or catalog objects

#### Return Values

none

#### Examples

----- **EXAMPLE 1** -----

```
C:\PS> Get-BrokerCatalog -Name "SampleCatalog" | Start-BrokerRebootCycle -RebootDuration 240 -WarningMessage "Save your work" -WarningDuration 15
```

Starts a new reboot cycle for each desktop group containing machines from the catalog "SampleCatalog". Each reboot cycle will have a duration of six hours. Fifteen minutes prior to rebooting a machine, the message "Save your work" will be displayed in each active user session.

# Stop-BrokerRebootCycle

May 10, 2016

Cancels the specified reboot cycle.

## Syntax

```
Stop-BrokerRebootCycle [-InputObject] <RebootCycle[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Stop-BrokerRebootCycle cmdlet is used to cancel the specified reboot cycle.

## Related topics

[Get-BrokerRebootCycle](#)

[Start-BrokerRebootCycle](#)

## Parameters

**-InputObject**<RebootCycle[]>

Cancels this reboot cycle.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host



name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Citrix.Broker.Admin.SDK.RebootCycle Reboot cycles may be specified through pipeline input.

### Return Values

none

### Examples

#### ----- **EXAMPLE 1** -----

```
C:\PS> Get-BrokerRebootCycle -CatalogUid 7 | Stop-BrokerRebootCycle
```

Cancels every reboot cycle for the catalog that has the Uid of 7.

# Stop-BrokerSession

May 10, 2016  
Stop or log off a session.

## Syntax

Stop-BrokerSession [-InputObject] <Session[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Stops or logs off sessions.

## Related topics

[Disconnect-BrokerSession](#)

[Get-BrokerSession](#)

## Parameters

**-InputObject**<Session[]>

Identifies the session(s) to terminate. This can be expressed as either a session Uid or a session object.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Broker.Admin.SDK.Session The sessions to stop can be piped into this cmdlet.

## Return Values

None

## Notes

This operation is non-blocking and returns before it completes. The operation, however, is unlikely to fail unless there are communication problems between the controller and the machine, if bad arguments are passed to the cmdlet itself or if the machine cannot successfully execute the operation.

The transient nature of sessions means that the list of session objects or UIDs supplied to Stop-BrokerSession could consist of valid and invalid sessions. Invalid sessions are detected and disregarded and the stop session operation is invoked on the valid sessions.

The system can fail to invoke the operation if the machine is not in an appropriate state or if there are problems in communicating with the machine. When an operation is invoked the system detects if the operation was initiated successfully or not by the machine. As this operation is non-blocking the system doesn't detect or report whether the operation ultimately succeeded or failed after its successful initialization on the machine.

Operation failures are reported through the broker SDK error handling mechanism (see about\_Broker\_ErrorHandling). In the event of errors the SdkErrorRecord error status is set to SessionOperationFailed and its error data dictionary is populated with the following entries:

- o OperationsAttemptedCount - The number of operations attempted.
- o OperationsFailedCount - The number of failed operations.
- o OperationsSucceededCount - The number of successfully executed operations.
- o UnresolvedSessionFailuresCount - The number of operations that failed due to invalid sessions being supplied.
- o OperationInvocationFailuresCount - The number of operations that failed because they could not be invoked on the desktop.
- o DesktopExecutionFailuresCount - The number of operations that failed because they could not be successfully executed by the desktop.

The SdkErrorRecord message will also display the number of attempted, failed and successful operations in the following format:

"Session operation error - attempted:<OperationsAttemptedCount>, failed:<OperationsFailedCount>, succeeded:<OperationsSucceededCount>"

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-BrokerSession -UserName MyDomain\MyAccount | Stop-BrokerSession
```

Stops all sessions for the user MyDomain\MyAccount.

### ----- EXAMPLE 2 -----

```
C:\PS> $desktop = Get-BrokerDesktop -DNSName MyMachine.MyDomain.com
C:\PS> Stop-BrokerSession $desktop.SessionUid
```

Stops the session on MyMachine.

### ----- EXAMPLE 3 -----

```
C:\PS> Get-BrokerSession -Filter { SessionState -eq 'Disconnected' -and SessionStateChangeTime -lt '-1' } | Stop-BrokerSession
```

Stop sessions that have been disconnected for more than one day.

### ----- EXAMPLE 4 -----

```
C:\PS> trap [Citrix.Broker.Admin.SDK.SdkOperationException]
C:\PS> {
C:\PS> write $("Exception name = " + $_.Exception.GetType().FullName)
C:\PS> write $("SdkOperationException.Status = " + $_.Exception.Status)
C:\PS> write $("SdkOperationException.ErrorData=")
C:\PS> $_.Exception.ErrorData
C:\PS>
C:\PS> write $("SdkOperationException.InnerException = " + $_.Exception.InnerException)
C:\PS> $_.Exception.InnerException
C:\PS> continue
C:\PS> }
C:\PS>
C:\PS> Stop-BrokerSession -InputObject 10,11,12
```

Trap and display error information.

# Test-BrokerAccessPolicyRuleNameAvailable

May 10, 2016

Determine whether the proposed AccessPolicyRule Name is available for use.

## Syntax

```
Test-BrokerAccessPolicyRuleNameAvailable [-Name] <String[]> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet checks whether proposed AccessPolicyRule Name is available for use. It returns a record for each Name indicating the availability of that Name, with \$true indicating that the Name is unused and available for use, or \$false if it is not available.

## Related topics

[Get-BrokerAccessPolicyRule](#)

[New-BrokerAccessPolicyRule](#)

[Rename-BrokerAccessPolicyRule](#)

## Parameters

**-Name**<String[]>

The AccessPolicyRule Name to be tested.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

System.String You can pipe a string that contains the Name to test.

## Return Values

Citrix.Broker.Admin.SDK.NameAvailability

The cmdlet returns a result for each Name specified. An availability of "True" indicates the Name is available for use, and "False" if it is not available.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Test-BrokerAccessPolicyRuleNameAvailable -Name Test1
```

Checks whether the Name "Test1" is available.

### ----- **EXAMPLE 2** -----

```
C:\PS> Test-BrokerAccessPolicyRuleNameAvailable @("Test1","Test2","Test3")
```

Checks whether each of the specified names is available.

# Test-BrokerAppAssignmentPolicyRuleNameAvailable

May 10, 2016

Determine whether the proposed AppAssignmentPolicyRule Name is available for use.

## Syntax

```
Test-BrokerAppAssignmentPolicyRuleNameAvailable [-Name] <String[]> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet checks whether proposed AppAssignmentPolicyRule Name is available for use. It returns a record for each Name indicating the availability of that Name, with \$true indicating that the Name is unused and available for use, or \$false if it is not available.

## Related topics

[Get-BrokerAppAssignmentPolicyRule](#)

[New-BrokerAppAssignmentPolicyRule](#)

[Rename-BrokerAppAssignmentPolicyRule](#)

## Parameters

**-Name**<String[]>

The AppAssignmentPolicyRule Name to be tested.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

System.String You can pipe a string that contains the Name to test.

## Return Values

Citrix.Broker.Admin.SDK.NameAvailability

The cmdlet returns a result for each Name specified. An availability of "True" indicates the Name is available for use, and "False" if it is not available.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Test-BrokerAppAssignmentPolicyRuleNameAvailable -Name Test1
```

Checks whether the Name "Test1" is available.

### ----- EXAMPLE 2 -----

```
C:\PS> Test-BrokerAppAssignmentPolicyRuleNameAvailable @("Test1","Test2","Test3")
```

Checks whether each of the specified names is available.



# Test-BrokerAppEntitlementPolicyRuleNameAvailable

May 10, 2016

Determine whether the proposed AppEntitlementPolicyRule Name is available for use.

## Syntax

Test-BrokerAppEntitlementPolicyRuleNameAvailable [-Name] <String[]> [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

This cmdlet checks whether proposed AppEntitlementPolicyRule Name is available for use. It returns a record for each Name indicating the availability of that Name, with \$true indicating that the Name is unused and available for use, or \$false if it is not available.

## Related topics

[Get-BrokerAppEntitlementPolicyRule](#)

[New-BrokerAppEntitlementPolicyRule](#)

[Rename-BrokerAppEntitlementPolicyRule](#)

## Parameters

**-Name**<String[]>

The AppEntitlementPolicyRule Name to be tested.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

System.String You can pipe a string that contains the Name to test.

## Return Values

Citrix.Broker.Admin.SDK.NameAvailability

The cmdlet returns a result for each Name specified. An availability of "True" indicates the Name is available for use, and "False" if it is not available.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Test-BrokerAppEntitlementPolicyRuleNameAvailable -Name Test1
```

Checks whether the Name "Test1" is available.

### ----- **EXAMPLE 2** -----

```
C:\PS> Test-BrokerAppEntitlementPolicyRuleNameAvailable @("Test1","Test2","Test3")
```

Checks whether each of the specified names is available.

# Test-BrokerApplicationNameAvailable

May 10, 2016

Determine whether the proposed Application Name is available for use.

## Syntax

```
Test-BrokerApplicationNameAvailable [-Name] <String[]> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet checks whether proposed Application Name is available for use. It returns a record for each Name indicating the availability of that Name, with \$true indicating that the Name is unused and available for use, or \$false if it is not available.

## Related topics

[Get-BrokerApplication](#)

[New-BrokerApplication](#)

[Rename-BrokerApplication](#)

## Parameters

**-Name**<String[]>

The Application Name to be tested.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

System.String You can pipe a string that contains the Name to test.

## Return Values

Citrix.Broker.Admin.SDK.NameAvailability

The cmdlet returns a result for each Name specified. An availability of "True" indicates the Name is available for use, and "False" if it is not available.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Test-BrokerApplicationNameAvailable -Name Test1
```

Checks whether the Name "Test1" is available.

### ----- EXAMPLE 2 -----

```
C:\PS> Test-BrokerApplicationNameAvailable @("Test1","Test2","Test3")
```

Checks whether each of the specified names is available.

# Test-BrokerAssignmentPolicyRuleNameAvailable

May 10, 2016

Determine whether the proposed AssignmentPolicyRule Name is available for use.

## Syntax

Test-BrokerAssignmentPolicyRuleNameAvailable [-Name] <String[]> [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

This cmdlet checks whether proposed AssignmentPolicyRule Name is available for use. It returns a record for each Name indicating the availability of that Name, with \$true indicating that the Name is unused and available for use, or \$false if it is not available.

## Related topics

[Get-BrokerAssignmentPolicyRule](#)

[New-BrokerAssignmentPolicyRule](#)

[Rename-BrokerAssignmentPolicyRule](#)

## Parameters

**-Name**<String[]>

The AssignmentPolicyRule Name to be tested.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

System.String You can pipe a string that contains the Name to test.

## Return Values

Citrix.Broker.Admin.SDK.NameAvailability

The cmdlet returns a result for each Name specified. An availability of "True" indicates the Name is available for use, and "False" if it is not available.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Test-BrokerAssignmentPolicyRuleNameAvailable -Name Test1
```

Checks whether the Name "Test1" is available.

### ----- EXAMPLE 2 -----

```
C:\PS> Test-BrokerAssignmentPolicyRuleNameAvailable @("Test1","Test2","Test3")
```

Checks whether each of the specified names is available.

# Test-BrokerCatalogNameAvailable

May 10, 2016

Determine whether the proposed Catalog Name is available for use.

## Syntax

```
Test-BrokerCatalogNameAvailable [-Name] <String[]> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet checks whether proposed Catalog Name is available for use. It returns a record for each Name indicating the availability of that Name, with \$true indicating that the Name is unused and available for use, or \$false if it is not available.

## Related topics

[Get-BrokerCatalog](#)

[New-BrokerCatalog](#)

[Rename-BrokerCatalog](#)

## Parameters

**-Name**<String[]>

The Catalog Name to be tested.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

System.String You can pipe a string that contains the Name to test.

## Return Values

Citrix.Broker.Admin.SDK.NameAvailability

The cmdlet returns a result for each Name specified. An availability of "True" indicates the Name is available for use, and "False" if it is not available.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Test-BrokerCatalogNameAvailable -Name Test1
```

Checks whether the Name "Test1" is available.

### ----- EXAMPLE 2 -----

```
C:\PS> Test-BrokerCatalogNameAvailable @("Test1","Test2","Test3")
```

Checks whether each of the specified names is available.



# Test-BrokerDBConnection

May 10, 2016

Tests whether a database is suitable for use by the Citrix Broker Service.

## Syntax

```
Test-BrokerDBConnection [-DBConnection] <String> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Tests whether the database specified in the given connection string is suitable for use by the currently selected Citrix Broker Service instance.

The service attempts to contact the specified database and returns a status indicating whether the database is both contactable and usable. The test does not impact any currently established connection from the service instance to another database in any way. The tested connection string is not recorded.

Only use of Windows authentication within the connection string is supported; a connection string containing SQL authentication credentials is always rejected as invalid.

The current service instance is the one on the local machine, or the one most recently specified using the -AdminAddress parameter of a Broker SDK cmdlet.

## Related topics

[Get-BrokerServiceStatus](#)

[Get-BrokerDBConnection](#)

[Set-BrokerDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be tested by the currently selected Citrix Broker Service instance.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Test-BrokerDBConnection cmdlet returns an object describing the status of the selected Broker Service instance that would result if the connection string were used with the Set-BrokerDBConnection cmdlet together with extra diagnostics information for the specified connection string. The actual current status of the service is not changed. Possible diagnostic values are:

-- OK:

The Set-BrokerDBConnection command would succeed if it were executed with the supplied connection string.

-- DBUnconfigured:

No database connection string is set for the service instance.

-- DBRejectedConnection:

The database rejected the logon attempt from the Broker Service instance. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

-- InvalidDBConfigured:

The specified database does not exist, is not visible to the Broker Service instance, or the service's schema within the database is invalid.

-- DBNotFound:

The specified database could not be located with the given connection string.

-- DBNewerVersionThanService:

The Broker Service instance is older than, and incompatible with, the service's schema in the database. The service instance needs upgrading.

-- DBOlderVersionThanService:

The Broker Service instance is newer than, and incompatible with, the service's schema in the database. The database schema needs upgrading.

-- DBVersionChangeInProgress:

A database schema upgrade is currently in progress.

-- PendingFailure:

Connectivity between the Broker Service instance and the database has been lost. This may be a transitory network error, but may indicate a loss of connectivity that requires administrator intervention.

-- Failed:

Connectivity between the Broker Service instance and the database has been lost for an extended period of time, or has failed due to a configuration problem. The service instance cannot operate while its connection to the database is unavailable.

-- Unknown:

Service status cannot be determined.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidDBConnectionString

The database connection string has an invalid format.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS>Test-BrokerDBConnection "Server=dbserver\SQLEXPRESS;Database=XDDB;Trusted_Connection=True"
```

Tests whether the service instance could use a database called XDDB on an SQL Server Express database running on the machine called dbserver. Integrated Windows authentication is required.

# Test-BrokerDesktopGroupNameAvailable

May 10, 2016

Determine whether the proposed DesktopGroup Name is available for use.

## Syntax

```
Test-BrokerDesktopGroupNameAvailable [-Name] <String[]> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet checks whether proposed DesktopGroup Name is available for use. It returns a record for each Name indicating the availability of that Name, with \$true indicating that the Name is unused and available for use, or \$false if it is not available.

## Related topics

[Get-BrokerDesktopGroup](#)

[New-BrokerDesktopGroup](#)

[Rename-BrokerDesktopGroup](#)

## Parameters

**-Name**<String[]>

The DesktopGroup Name to be tested.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

System.String You can pipe a string that contains the Name to test.

## Return Values

Citrix.Broker.Admin.SDK.NameAvailability

The cmdlet returns a result for each Name specified. An availability of "True" indicates the Name is available for use, and "False" if it is not available.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Test-BrokerDesktopGroupNameAvailable -Name Test1
```

Checks whether the Name "Test1" is available.

### ----- **EXAMPLE 2** -----

```
C:\PS> Test-BrokerDesktopGroupNameAvailable @("Test1","Test2","Test3")
```

Checks whether each of the specified names is available.

# Test-BrokerEntitlementPolicyRuleNameAvailable

May 10, 2016

Determine whether the proposed EntitlementPolicyRule Name is available for use.

## Syntax

```
Test-BrokerEntitlementPolicyRuleNameAvailable [-Name] <String[]> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet checks whether proposed EntitlementPolicyRule Name is available for use. It returns a record for each Name indicating the availability of that Name, with \$true indicating that the Name is unused and available for use, or \$false if it is not available.

## Related topics

[Get-BrokerEntitlementPolicyRule](#)

[New-BrokerEntitlementPolicyRule](#)

[Rename-BrokerEntitlementPolicyRule](#)

## Parameters

**-Name**<String[]>

The EntitlementPolicyRule Name to be tested.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

System.String You can pipe a string that contains the Name to test.

## Return Values

Citrix.Broker.Admin.SDK.NameAvailability

The cmdlet returns a result for each Name specified. An availability of "True" indicates the Name is available for use, and "False" if it is not available.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Test-BrokerEntitlementPolicyRuleNameAvailable -Name Test1
```

Checks whether the Name "Test1" is available.

### ----- **EXAMPLE 2** -----

```
C:\PS> Test-BrokerEntitlementPolicyRuleNameAvailable @("Test1","Test2","Test3")
```

Checks whether each of the specified names is available.

# Test-BrokerLicenseServer

May 10, 2016

Tests whether or not a license server can be used by the broker.

## Syntax

```
Test-BrokerLicenseServer [-ComputerName] <String> [-AdminAddress <String>] [[-Port] <Int32>] [<CommonParameters>]
```

## Detailed Description

Tests whether or not a given license server can be used by the broker.

## Related topics

[Get-BrokerSite](#)

[Set-BrokerSite](#)

## Parameters

**-ComputerName**<String>

The name of the license server to test (machine.domain).

Required?	true
Default Value	None
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

**-Port**<Int32>

The port number to use on the server.

Required?	false
-----------	-------



Default Value	27000
Accept Pipeline Input?	false

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

System.string

Test-BrokerLicenseServer returns:

- o 'Compatible' - the server is a compatible license server that can be used.
- o 'Incompatible' - the server is an incompatible license server that can't be used.
- o 'Inaccessible' - the server cannot be accessed. The server may be down, unreachable, or non-existent.
- o 'InternalError' - the server can't be used due to an internal error. A required licensing component on the server may not be installed, configured, or working correctly.

## Examples

### ----- EXAMPLE 1 -----

C:\PS> Test-BrokerLicenseServer -LicenseServerAddress "machine.domain" 1234  
Tests whether or not the license server "machine.domain" with port number 1234 can be used.

### ----- EXAMPLE 2 -----

C:\PS> Test-BrokerLicenseServer -LicenseServerAddress "machine.domain"  
Tests whether or not the license server "machine.domain" with port number 2700 can be used.

# Test-BrokerMachineNameAvailable

May 10, 2016

Determine whether the proposed Machine MachineName is available for use.

## Syntax

```
Test-BrokerMachineNameAvailable [-MachineName] <String[]> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet checks whether proposed Machine MachineName is available for use. It returns a record for each MachineName indicating the availability of that MachineName, with \$true indicating that the MachineName is unused and available for use, or \$false if it is not available.

## Related topics

[Get-BrokerMachine](#)

[New-BrokerMachine](#)

## Parameters

**-MachineName**<String[]>

The Machine MachineName to be tested.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

System.String You can pipe a string that contains the MachineName to test.

## Return Values

Citrix.Broker.Admin.SDK.NameAvailability

The cmdlet returns a result for each MachineName specified. An availability of "True" indicates the MachineName is available for use, and "False" if it is not available.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Test-BrokerMachineNameAvailable -MachineName Test1
```

Checks whether the MachineName "Test1" is available.

### ----- EXAMPLE 2 -----

```
C:\PS> Test-BrokerMachineNameAvailable @("Test1","Test2","Test3")
```

Checks whether each of the specified names is available.

# Test-BrokerPowerTimeSchemeNameAvailable

May 10, 2016

Determine whether the proposed PowerTimeScheme Name is available for use.

## Syntax

```
Test-BrokerPowerTimeSchemeNameAvailable [-Name] <String[]> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet checks whether proposed PowerTimeScheme Name is available for use. It returns a record for each Name indicating the availability of that Name, with \$true indicating that the Name is unused and available for use, or \$false if it is not available.

## Related topics

[Get-BrokerPowerTimeScheme](#)

[New-BrokerPowerTimeScheme](#)

[Rename-BrokerPowerTimeScheme](#)

## Parameters

**-Name**<String[]>

The PowerTimeScheme Name to be tested.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

System.String You can pipe a string that contains the Name to test.

## Return Values

Citrix.Broker.Admin.SDK.NameAvailability

The cmdlet returns a result for each Name specified. An availability of "True" indicates the Name is available for use, and "False" if it is not available.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Test-BrokerPowerTimeSchemeNameAvailable -Name Test1
```

Checks whether the Name "Test1" is available.

### ----- **EXAMPLE 2** -----

```
C:\PS> Test-BrokerPowerTimeSchemeNameAvailable @"Test1","Test2","Test3"
```

Checks whether each of the specified names is available.

# Test-BrokerRemotePCAccountNameAvailable

May 10, 2016

Determine whether the proposed RemotePCAccount OU is available for use.

## Syntax

```
Test-BrokerRemotePCAccountNameAvailable [-OU] <String[]> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet checks whether proposed RemotePCAccount OU is available for use. It returns a record for each OU indicating the availability of that OU, with \$true indicating that the OU is unused and available for use, or \$false if it is not available.

## Related topics

[Get-BrokerRemotePCAccount](#)

[New-BrokerRemotePCAccount](#)

## Parameters

**-OU**<String[]>

The RemotePCAccount OU to be tested.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

System.String You can pipe a string that contains the OU to test.

## Return Values

Citrix.Broker.Admin.SDK.NameAvailability

The cmdlet returns a result for each OU specified. An availability of "True" indicates the OU is available for use, and "False" if it is not available.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Test-BrokerRemotePCAccountNameAvailable -OU Test1
```

Checks whether the OU "Test1" is available.

### ----- EXAMPLE 2 -----

```
C:\PS> Test-BrokerRemotePCAccountNameAvailable @("Test1","Test2","Test3")
```

Checks whether each of the specified names is available.

# Update-BrokerImportedFTA

May 10, 2016

Imports or updates all of the file type associations for the specified worker.

## Syntax

```
Update-BrokerImportedFTA -DesktopUids <Int32[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Imports or updates the file type associations from a specified worker machine.

File type association associates a file extension (such as ".txt") with an application (such as Notepad). In a Citrix environment file type associations on a user device can be configured so that when an user clicks on a document it launches the appropriate published application. This is known as "content redirection".

Imported file type associations are different from configured file type associations. Imported file type associations are lists of known file type associations for a given desktop group. Configured file type associations are those that are actually associated with published applications for the purposes of content redirection.

Initially the system is not aware of any extensions, and they must be imported by the Citrix administrator. To import or update file type associations from a worker machine, two criteria must be met:

- o The worker machine must not be in use
- o The worker machine must be in maintenance mode

For more information about putting a worker machine in maintenance mode, see the Set-BrokerPrivateDesktop and Set-BrokerSharedDesktop cmdlets.

Imported file type associations are grouped together based on the desktop group of the machine from which they were imported. All file types for a desktop group are deleted. There is no mechanism for deleting a subset imported file type associations for a specific desktop group.

If file type associations are imported more than once for a desktop group, for example, if this cmdlet is run twice for two workers belonging to the same desktop group, all existing imported file type associations for that desktop group are deleted and imported again.

## Related topics

[Get-BrokerImportedFTA](#)

[Remove-BrokerImportedFTA](#)

## Parameters

**-DesktopUids**<Int32[]>

Imports or updates the file type associations from the specified desktop. The desktop must belong to a desktop group of the Private or Shared desktop kind.



Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

Int32[] An array of Uids for desktops can be supplied as input.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> $desktop = Get-BrokerSharedDesktop -MachineName "ACME\Worker1"
C:\PS> Set-BrokerSharedDesktop $desktop -InMaintenanceMode $true
C:\PS> Update-BrokerImportedFTA -DesktopUids $desktop.Uid
C:\PS> Set-BrokerSharedDesktop $desktop -InMaintenanceMode $false
```

Gets an object for the worker machine named "Worker1" in the "ACME" domain, and ensures no users can connect to it, before importing the file type associations from that desktop and re-enabling it.

# Update-BrokerLocalLeaseCache

May 10, 2016

Flushes the local lease cache.

## Syntax

Update-BrokerLocalLeaseCache [-Workers] [-Applications] [-Icons] [-Desktops] [-Leases] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Removes all local cached lease data and any state information stored in the registry.

## Related topics

[Remove-BrokerLease](#)

## Parameters

**-Workers**<SwitchParameter>

Removes all locally cached workers on the controller. The worker cache will be repopulated from the current site database contents after a short delay.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-Applications**<SwitchParameter>

Removes all locally cached applications on the controller. The application cache will be repopulated from the current site database contents after a short delay.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-Icons**<SwitchParameter>

Removes all locally cached icons on the controller. The icon cache will be repopulated from the current site database contents after a short delay.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### **-Desktops**<SwitchParameter>

Removes all locally cached desktops on the controller. The desktop cache will be repopulated from the current site database contents after a short delay.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### **-Leases**<SwitchParameter>

Removes all locally cached leases on the controller. The lease cache will be repopulated from the current site database contents after a short delay.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high level operation that this cmdlet call forms a part of. Desktop Studio and Desktop Director typically create High Level Operations. PowerShell scripts can also wrap a series of cmdlet calls in a High Level Operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

### Input Type

None

### Return Values

None

### Notes

The local cache for lease and other data like worker, desktop, application and icon information is removed and will be repopulated from the current site database contents after a short delay.

### Examples

#### ----- **EXAMPLE 1** -----

```
C:\PS> Update-BrokerLocalLeaseCache
```

Flushes the local lease cache for all objects and deletes any state information stored in the registry.

#### ----- **EXAMPLE 2** -----

```
C:\PS> Update-BrokerLocalLeaseCache -Workers
```

Flushes the local lease cache for all workers and deletes any state information stored in the registry.

# Update-BrokerNameCache

May 10, 2016

Performs administrative operations on the user and machine name cache.

## Syntax

```
Update-BrokerNameCache [-Machines] [-Users] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Triggers an immediate asynchronous refresh of the name cache. This may be useful to ensure up-to-date name information is present in the cache after user and/or machine accounts are known to have changed and you need to see those changes immediately instead of waiting for the periodic automatic refresh.

The Broker Service maintains a cache of the names of users and machines in use by the site. By default, name information is obtained periodically from Active Directory and the cache refreshed automatically.

During normal usage, you should not need to perform administrative operations on the name cache.

For users/groups, the following name information is cached:

Windows name (DOMAIN\user)

User Principal Name or 'UPN' (user@upndomain)

Full Name or 'Common Name' (typically a user's full name)

For machines, the following name information is cached:

Windows name (DOMAIN\machine)

DNS name (machine.dnsdomain)

## Related topics

## Parameters

**-Machines**<SwitchParameter>

Triggers an asynchronous refresh of all cached machine name information.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Users**<SwitchParameter>

Triggers an asynchronous refresh of all cached user name information.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snapin will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

None

#### Notes

For some user accounts, for example, the built-in domain administrator, the UPN and/or Full Name values may not be available because they are not typically specified within Active Directory.

For group accounts, UPN and Full Name values are not available because they are not applicable or not specified within Active Directory.

The DNS name information for a machine is obtained from Active Directory and not from the DNS sub-system. If a machine has only recently been configured, the DNS information may not be available initially.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Update-BrokerNameCache -Machines
```

Triggers an immediate asynchronous refresh of all machine name information held within the name cache.

##### ----- **EXAMPLE 2** -----

```
C:\PS> Update-BrokerNameCache -Machines -Users
```

Triggers an immediate asynchronous refresh of all machine and user name information held within the name cache.



# Citrix.Configuration.Admin.V2

May 10, 2016

## Overview

Name	Description
<a href="#">ConfigConfigurationSnapin</a>	The Configuration service PowerShell snap-in provides administrative
<a href="#">Config Filtering</a>	Describes the common filtering options for XenDesktop cmdlets.

## Cmdlets

Name	Description
<a href="#">Add-ConfigRegisteredServiceInstanceMetadata</a>	Adds metadata on the given ServiceInstance.
<a href="#">Add-ConfigServiceGroupMetadata</a>	Adds metadata on the given ServiceGroup.
<a href="#">Export-ConfigFeatureTable</a>	Returns the current feature table.
<a href="#">Get-ConfigDBConnection</a>	Gets the database string for the specified data store used by the Configuration Service.
<a href="#">Get-ConfigDBSchema</a>	Gets a script that creates the Configuration Service database schema for the specified data store.
<a href="#">Get-ConfigDBVersionChangeScript</a>	Gets a script that updates the Configuration Service database schema.
<a href="#">Get-ConfigEnabledFeature</a>	Lists features of the site that are enabled.
<a href="#">Get-ConfigInstalledDBVersion</a>	Gets a list of all available database schema versions for the Configuration Service.
<a href="#">Get-ConfigLicensingModel</a>	Lists the supported licensing models.
<a href="#">Get-ConfigLocalData</a>	Gets the service local data.
<a href="#">Get-ConfigProduct</a>	Lists the site's supported product names and codes.
<a href="#">Get-ConfigProductEdition</a>	Lists the supported product editions.

<b>Name</b>	<b>Description</b>
<a href="#">Get-ConfigProductFeature</a>	Lists the supported features.
<a href="#">Get-ConfigProductVersion</a>	Lists the supported product versions.
<a href="#">Get-ConfigRegisteredServiceInstance</a>	Gets the service instances that are registered in the directory.
<a href="#">Get-ConfigService</a>	Gets the service record entries for the Configuration Service.
<a href="#">Get-ConfigServiceAddedCapability</a>	Gets any added capabilities for the Configuration Service on the controller.
<a href="#">Get-ConfigServiceGroup</a>	Gets the service groups that match the parameters supplied.
<a href="#">Get-ConfigServiceInstance</a>	Gets the service instance entries for the Configuration Service.
<a href="#">Get-ConfigServiceStatus</a>	Gets the current status of the Configuration Service on the controller.
<a href="#">Get-ConfigSite</a>	Gets the site.
<a href="#">Import-ConfigFeatureTable</a>	Sets the feature table of the site.
<a href="#">Register-ConfigServiceInstance</a>	Allows the registration of a service instance.
<a href="#">Remove-ConfigRegisteredServiceInstanceMetadata</a>	Removes metadata from the given ServiceInstance.
<a href="#">Remove-ConfigServiceGroup</a>	Removes service groups.
<a href="#">Remove-ConfigServiceGroupMetadata</a>	Removes metadata from the given ServiceGroup.
<a href="#">Remove-ConfigServiceMetadata</a>	Removes metadata from the given Service.
<a href="#">Remove-ConfigSiteMetadata</a>	Removes metadata from the given Site.
<a href="#">Reset-ConfigServiceGroupMembership</a>	Reloads the access permissions and configuration service locations for the Configuration Service.
<a href="#">Set-ConfigDBConnection</a>	Configures a database connection for the Configuration Service.
<a href="#">Set-ConfigRegisteredServiceInstance</a>	Updates a service instance.

Name	Description
Set-ConfigRegisteredServiceInstanceMetadata	Adds or updates metadata on the given ServiceInstance.
Set-ConfigServiceGroupMetadata	Adds or updates metadata on the given ServiceGroup.
Set-ConfigServiceMetadata	Adds or updates metadata on the given Service.
Set-ConfigSite	Changes the overall settings of the site.
Set-ConfigSiteMetadata	Adds or updates metadata on the Site.
Test-ConfigDBConnection	Tests a database connection for the Configuration Service.
Test-ConfigServiceInstanceAvailability	Tests whether the supplied service instances are responding to requests.
Unregister-ConfigRegisteredServiceInstance	Removes a service instance from the Configuration Service registry.

# about\_ConfigConfigurationSnapin

May 10, 2016

## TOPIC

about\_ConfigConfigurationSnapin

## SHORT DESCRIPTION

The Configuration service PowerShell snap-in provides administrative functions for the Configuration service.

## COMMAND PREFIX

All commands in this snap-in have 'Config' in their name.

## LONG DESCRIPTION

The Configuration service PowerShell snap-in enables both local and remote administration of the Configuration service. It provides facilities to store details about other services that are used in the XenDesktop deployment.

All the services in the XenDesktop deployment use the Configuration service as a directory to locate other services with which they need to communicate. The directory publishes a list of all the services, the communication types they accept, and the facilities they offer.

The snap-in provides the following main entities:

### Site Metadata

Metadata for the entire XenDesktop deployment. This metadata is not used directly by any of the XenDesktop services, but can be used by third parties to store information for other purposes.

### Registered Service Instances

Service instance items that have been retrieved from the available services in the XenDesktop deployment and held in a directory in the Configuration service. Each physical service can host a collection of service instances to provide different facilities.

### Service Groups

A collection of service instances that are considered equivalent. Service instances that are registered in the same service group provide the same state and offer the same functionality.

Only one service group of each type is supported. For example, there

must not be more than one service group with a ServiceType of 'Config'.

## Site and Features

The configuration site is a top-level, logical representation of the XenDesktop site, from the perspective of the configuration services running within the site.

# about\_Config\_Filtering

May 10, 2016

## TOPIC

XenDesktop - Advanced Dataset Filtering

## SHORT DESCRIPTION

Describes the common filtering options for XenDesktop cmdlets.

## LONG DESCRIPTION

Some cmdlets operate on large quantities of data and, to reduce the overhead of sending all of that data over the network, many of the Get- cmdlets support server-side filtering of the results.

The conventional way of filtering results in PowerShell is to pipeline them into Where-Object, Select-Object, and Sort-Object, for example:

```
Get-<Noun> | Where { $_.Size = 'Small' } | Sort 'Date' | Select -First 10
```

However, for most XenDesktop cmdlets the data is stored remotely and it would be slow and inefficient to retrieve large amounts of data over the network and then discard most of it. Instead, many of the Get- cmdlets provide filtering parameters that allow results to be processed on the server, returning only the required results.

You can filter results by most object properties using parameters derived from the property name. You can also sort results or limit them to a specified number of records:

```
Get-<Noun> -Size 'Small' -SortBy 'Date' -MaxRecordCount 10
```

You can express more complex filter conditions using a syntax and set of operators very similar to those used by PowerShell expressions.

Those cmdlets that support filtering have the following common parameters:

-MaxRecordCount <int>

Specifies the maximum number of results to return.  
For example, to return only the first nine results use:

```
Get-<Noun> -MaxRecordCount 9
```

If not specified, only the first 250 records are returned, and if more are available, a warning is produced:

WARNING: Only first 250 records returned. Use -MaxRecordCount to retrieve more.

You can suppress this warning by using -WarningAction or by specifying a value for -MaxRecordCount.

To retrieve all records, specify a large number for -MaxRecordCount. As the value is an integer, you can use the following:

```
Get-<Noun> -MaxRecordCount [int]::MaxValue
```

-ReturnTotalRecordCount [<SwitchParameter>]

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. For example:

```
Get-<Noun> -MaxRecordCount 9 -ReturnTotalRecordCount
```

....

```
Get-<Noun> : Returned 9 of 10 items
```

```
At line:1 char:18
```

```
+ Get-<Noun> <<<< -MaxRecordCount 9 -ReturnTotalRecordCount
```

```
+ CategoryInfo : OperationStopped: (:) [Get-<Noun>], PartialDataException
```

```
+ FullyQualifiedErrorId : PartialData,Citrix.<SDKName>.SDK.Get<Noun>
```

The count can be accessed using the TotalAvailableResultCount property:

```
$count = $error[0].TotalAvailableResultCount
```

-Skip <int>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

-SortBy <string>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order, respectively. Ascending order is assumed if no prefix is present.

Sorting occurs before -MaxRecordCount and -Skip parameters are applied. For example, to sort by Name and then by Count (largest first) use:

```
-SortBy 'Name,-Count'
```

By default, sorting by an enumeration property uses the numeric value of the elements. You can specify a different sort order by qualifying the name with an ordered list of elements or their numeric values, or <null> to indicate the placement of null values.

Elements not mentioned are placed at the end in their numeric order. For example, to sort by two different enums and then by the object id:

```
-SortBy 'MyState(StateC,<null>,StateA,StateB),Another(0,3,2,1),Id'
```

-Filter <String>

This parameter lets you specify advanced filter expressions, and supports combination of conditions with -and and -or, and grouping with braces. For example:

```
Get-<Noun> -Filter 'Name -like "High*" -or (Priority -eq 1 -and Severity -ge 2)'
```

The syntax is close enough to PowerShell syntax that you can use script blocks in most cases. This can be easier to read as it reduces quoting:

```
Get-<Noun> -Filter { Count -ne $null }
```

The full -Filter syntax is provided below.

## EXAMPLES

Filtering by strings performs a case-insensitive wildcard match.

Separate parameters are combined with an implicit -and operator.

Normal PowerShell quoting rules apply, so you can use single or double quotes, and omit the quotes altogether for many strings. The order of parameters does not make any difference. The following are equivalent:

```
Get-<Noun> -Company Citrix -Product Xen*
```

```
Get-<Noun> -Company "citrix" -Product '[X]EN*'
```

```
Get-<Noun> -Product "Xen*" -Company "CITRIX"
```

```
Get-<Noun> -Filter { Company -eq 'Citrix' -and Product -like 'Xen*' }
```



See [about\\_Quoting\\_Rules](#) and [about\\_Wildcards](#) for details about PowerShell handling of quotes and wildcards.

To avoid wildcard matching or include quote characters, you can escape the wildcards using the normal PowerShell escape mechanisms (see [about\\_Escape\\_Characters](#)), or switch to a filter expression and the `-eq` operator:

```
Get-<Noun> -Company "Abc[*]" # Matches Abc*
Get-<Noun> -Company "Abc`*" # Matches Abc*
Get-<Noun> -Filter { Company -eq "Abc*" } # Matches Abc*
Get-<Noun> -Filter { Company -eq "A`"B`C`" } # Matches A"B'C
```

Simple filtering by numbers, booleans, and TimeSpans perform direct equality comparisons, although if the value is nullable you can also search for null values. Here are some examples:

```
Get-<Noun> -Uid 123
Get-<Noun> -Enabled $true
Get-<Noun> -Duration 1:30:40
Get-<Noun> -NullableProperty $null
```

More comparisons are possible using advanced filtering with `-Filter`:

```
Get-<Noun> -Filter 'Capacity -ge 10gb'
Get-<Noun> -Filter 'Age -ge 20 -and Age -lt 40'
Get-<Noun> -Filter 'VolumeLevel -like "[123]"'
Get-<Noun> -Filter 'Enabled -ne $false'
Get-<Noun> -Filter 'NullableProperty -ne $null'
```

You can check boolean values without an explicit comparison operator, and you can also combine them with `-not`:

```
Get-<Noun> -Filter 'Enabled' # Equivalent to 'Enabled -eq $true'
Get-<Noun> -Filter '-not Enabled' # Equivalent to 'Enabled -eq $false'
```

See [about\\_Comparison\\_Operators](#) for an explanation of the operators, but note that only a subset of PowerShell operators are supported (`-eq`, `-ne`, `-gt`, `-ge`, `-lt`, `-le`, `-like`, `-notlike`, `-in`, `-notin`, `-contains`, `-notcontains`).

Enumeration values can either be specified using typed values or the string name of the enumeration value:

```
Get-<Noun> -Shape [Shapes]::Square
Get-<Noun> -Shape Circle
```

With filter expressions, typed values can be specified with simple variables or quoted strings. They also support enumerations with wildcards:

```
$s = [Shapes]::Square
Get-<Noun> -Filter { Shape -eq $s -or Shape -eq "Circle" }
Get-<Noun> -Filter { Shape -like 'C*' }
```

By their nature, floating point values, DateTime values, and TimeSpan values are best suited to relative comparisons rather than just equality. DateTime strings are converted using the locale and time zone of the user device, but you can use ISO8601 format strings (YYYY-MM-DDThh:mm:ss.sTZD) to avoid ambiguity. You can also use standard PowerShell syntax to create these values:

```
Get-<Noun> -Filter { StartTime -ge "2010-08-23T12:30:00.0Z" }
$d = [DateTime]"2010-08-23T12:30:00.0Z"
Get-<Noun> -Filter { StartTime -ge $d }
$d = (Get-Date).AddDays(-1)
Get-<Noun> -Filter { StartTime -ge $d }
```

Relative times are quite common and, when using filter expressions, you can also specify DateTime values using a relative format:

```
Get-<Noun> -Filter { StartTime -ge '-2' } # Two days ago
Get-<Noun> -Filter { StartTime -ge '-1:30' } # Hour and a half ago
Get-<Noun> -Filter { StartTime -ge '-0:0:30' } # 30 seconds ago
```

## ARRAY PROPERTIES

When filtering against list or array properties, simple parameters perform a case-insensitive wildcard match against each of the members. With filter expressions, you can use the -contains and -notcontains operators. Unlike PowerShell, these perform wildcard matching on strings.

Note that for array properties the naming convention is for the returned property to be plural, but the parameter used to search for any match is singular. The following are equivalent (assuming Users is an array property):

```
Get-<Noun> -User Fred*
Get-<Noun> -Filter { User -like "Fred*" }
Get-<Noun> -Filter { Users -contains "Fred*" }
```

You can also use the singular form with -Filter to search using other operators:

```
Match if any user in the list is called "Frederick"
Get-<Noun> -Filter { User -eq "Frederick" }
Match if any user in the list has a name alphabetically below 'F'
Get-<Noun> -Filter { User -lt 'F' }
```

## COMPLEX EXPRESSIONS

When matching against multiple values, you can use a sequence of comparisons joined with -or operators, or you can use -in and -notin:

```
Get-<Noun> -Filter { Shape -eq 'Circle' -or Shape -eq 'Square' }
$shapes = 'Circle','Square'
Get-<Noun> -Filter { Shape -in $shapes }
$sides = 1..4
Get-<Noun> -Filter { Sides -notin $sides }
```

Braces can be used to group complex expressions, and override the default left-to-right evaluation of -and and -or. You can also use -not to invert the sense of any sub-expression:

```
Get-<Noun> -Filter { Size -gt 4 -or (Color -eq 'Blue' -and Shape -eq 'Circle') }
Get-<Noun> -Filter { Sides -lt 5 -and -not (Color -eq 'Blue' -and Shape -eq 'Circle') }
```

## PAGING

The simplest way to page through data is to use the -Skip and -MaxRecordCount parameters. So, to read the first three pages of data with 10 records per page, use:

```
Get-<Noun> -Skip 0 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 10 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 20 -MaxRecordCount 10 <other filtering criteria>
```

You must include the same filtering criteria on each call, and ensure that the data is sorted consistently.

The above approach is often acceptable, but as each call performs an independent query, data changes can result in records being skipped or appearing twice. One approach to improve this is to sort by a unique id field and then start the search for the next page at the unique id after the last unique id of the previous page. For example:

```
Get the first page
Get-<Noun> -MaxRecordCount 10 -SortBy SerialNumber

SerialNumber ...

A120004
A120007
... 7 other records ...
A120900

Get the next page
Get-<Noun> -MaxRecordCount 10 -Filter { FirstName -gt 'A120900' }

SerialNumber ...
```

-----  
A120901  
B220000  
...

## FILTER SYNTAX DEFINITION

<Filter> ::= <ScriptBlock> | <ComponentList>

<ScriptBlock> ::= "{" <ComponentList> "}"

<ComponentList> ::= <Component> <AndOrOperator> <ComponentList> |

<Component>

<Component> ::= <NotOperator> <Factor> |

<Factor>

<Factor> ::= "(" <ComponentList> ")" |

<PropertyName> <ComparisonOperator> <Value> |  
<PropertyName>

<AndOrOperator> ::= "-and" | "-or"

<NotOperator> ::= "-not" | "!"

<ComparisonOperator>

::= "-eq" | "-ne" | "-le" | "-ge" | "-lt" | "-gt" |  
"-like" | "-notlike" | "-contains" | "-notcontains" |  
"-in" | "-notin"

<PropertyName> ::= <simple name of property>

<Value> ::= <string literal> | <numeric literal> |

<scalar variable> | <array variable> |  
"\$null" | "\$true" | "\$false"

Numeric literals support decimal and hexadecimal literals, with optional multiplier suffixes (kb, mb, gb, tb, pb).

Dates and times can be specified as string literals. The current culture determines what formats are accepted. To avoid any ambiguity, use strings formatted to the ISO8601 standard. If not specified, the current time zone is used.

Relative date-time string literals are also supported, using a minus sign followed by a TimeSpan. For example, "-1:30" means 1 hour and 30 minutes ago.

# Add-ConfigRegisteredServiceInstanceMetadata

May 10, 2016  
Adds metadata on the given ServiceInstance.

## Syntax

Add-ConfigRegisteredServiceInstanceMetadata [-ServiceInstanceId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Add-ConfigRegisteredServiceInstanceMetadata [-ServiceInstanceId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Add-ConfigRegisteredServiceInstanceMetadata [-InputObject] <ServiceInstance[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Add-ConfigRegisteredServiceInstanceMetadata [-InputObject] <ServiceInstance[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Provides the ability for additional custom data to be stored against given ServiceInstance objects. This cmdlet will not overwrite existing metadata on an object - use the Set-ConfigRegisteredServiceInstanceMetadata cmdlet instead.

## Related topics

[Set-ConfigRegisteredServiceInstanceMetadata](#)

[Remove-ConfigRegisteredServiceInstanceMetadata](#)

## Parameters

**-ServiceInstanceId**<Guid>

Id of the ServiceInstance

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<ServiceInstance[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the ServiceInstance specified. The property cannot contain any of the following characters \/:#. \*?=<> | []()''

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	false

#### **-Value<String>**

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### **Return Values**

Citrix.Configuration.Sdk.Metadata

Add-ConfigRegisteredServiceInstanceMetadata returns an array of objects containing the new definition of the metadata.

\n Property <string>

\n Specifies the name of the property.

\n Value <string>

\n Specifies the value for the property.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DuplicateObject

One of the specified metadata already exists.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

c:\PS>Add-ConfigRegisteredServiceInstanceMetadata -ServiceInstanceUid 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value

Property

Value



-----

property

-----

value

Add metadata with a name of 'property' and a value of 'value' to the ServiceInstance with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Add-ConfigServiceGroupMetadata

May 10, 2016

Adds metadata on the given ServiceGroup.

## Syntax

```
Add-ConfigServiceGroupMetadata [-ServiceGroupUid] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Add-ConfigServiceGroupMetadata [-ServiceGroupUid] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Add-ConfigServiceGroupMetadata [-InputObject] <ServiceGroup[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Add-ConfigServiceGroupMetadata [-InputObject] <ServiceGroup[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability for additional custom data to be stored against given ServiceGroup objects. This cmdlet will not overwrite existing metadata on an object - use the Set-ConfigServiceGroupMetadata cmdlet instead.

## Related topics

[Set-ConfigServiceGroupMetadata](#)

[Remove-ConfigServiceGroupMetadata](#)

## Parameters

**-ServiceGroupUid**<Guid>

Id of the ServiceGroup

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<ServiceGroup[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with `@{"name1" = "val1"; "name2" = "val2"}`) or a string dictionary (created with `new-object "System.Collections.Generic.Dictionary[String,String]"`).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name<String>**

Specifies the property name of the metadata to be added. The property must be unique for the ServiceGroup specified. The property cannot contain any of the following characters `\/:#.*?=<>|[]()`

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Value<String>**

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the `Start-LogHighLevelOperation` and `Stop-LogHighLevelOperation` cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP

address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Configuration.Sdk.Metadata

Add-ConfigServiceGroupMetadata returns an array of objects containing the new definition of the metadata.

\n Property <string>

\n Specifies the name of the property.

\n Value <string>

\n Specifies the value for the property.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DuplicateObject

One of the specified metadata already exists.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Add-ConfigServiceGroupMetadata -ServiceGroupUid 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value

Property	Value
property	value

Add metadata with a name of 'property' and a value of 'value' to the ServiceGroup with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Export-ConfigFeatureTable

May 10, 2016  
Returns the current feature table.

## Syntax

Export-ConfigFeatureTable [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

## Related topics

- [Set-ConfigSite](#)
- [Import-ConfigFeatureTable](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.String

## Examples

----- **EXAMPLE 1** -----

```
C:\PS> Export-ConfigFeatureTable
<?xml version="1.0" encoding="utf-8">
<Products xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="FeatureTable.xsd">
 <Product Code="XDT" Name="XenDesktop">
 <Editions>
 <Edition>PLT</Edition>
 <Edition>ENT</Edition>
 <Edition>APP</Edition>
 <Edition>STD</Edition>
 </Editions>
 <DefaultEdition>PLT</DefaultEdition>
 <VersionToBurninDates>
 <VersionToBurninDate Version="7.0" BurninDate="2013.0522" />
 </VersionToBurninDates>
 <LicensingModels>
 <LicensingModel>Concurrent</LicensingModel>
 <LicensingModel>UserDevice</LicensingModel>
 </LicensingModels>
 <DefaultLicensingModel>UserDevice</DefaultLicensingModel>
 <Features>
 <Feature Name="CustomRole">
```

```

 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="ENT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="APP" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
 <Feature Name="ConfigurationLogging">
 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="ENT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="APP" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
 <Feature Name="SingleUserMode">
 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="ENT" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
 <Feature Name="MultiSessionDesktops">
 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="ENT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="APP" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
 <Feature Name="SingleSessionDesktops">
 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="ENT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="STD" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
 <Feature Name="MultiSessionApplications">
 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="ENT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="APP" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
 <Feature Name="SingleSessionApplications">
 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="ENT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="APP" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
 <Feature Name="HistoricalMonitorData">
 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
 <Feature Name="CloudHostedMachines">
 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
 <Feature Name="HdxInsightIntegration">
 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
 <Feature Name="RemotePC">
 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="ENT" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
</Features>
</Product>
<Product Code="MPS" Name="XenApp">
 <Editions>
 <Edition>PLT</Edition>
 <Edition>ENT</Edition>
 </Editions>
 <DefaultEdition>PLT</DefaultEdition>
 <VersionToBurninDates>
 <VersionToBurninDate Version="7.0" BurninDate="2013.0522" />
 </VersionToBurninDates>

```

```

<LicensingModels>
 <LicensingModel>Concurrent</LicensingModel>
</LicensingModels>
<DefaultLicensingModel>Concurrent</DefaultLicensingModel>
<Features>
 <Feature Name="CustomRole">
 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="ENT" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
 <Feature Name="ConfigurationLogging">
 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="ENT" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
 <Feature Name="SingleUserMode">
 </Feature>
 <Feature Name="MultiSessionDesktops">
 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="ENT" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
 <Feature Name="SingleSessionDesktops">
 </Feature>
 <Feature Name="MultiSessionApplications">
 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="ENT" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
 <Feature Name="SingleSessionApplications">
 <BurninDate ForProductEdition="PLT" MustBe="AtLeast">2013.0522</BurninDate>
 <BurninDate ForProductEdition="ENT" MustBe="AtLeast">2013.0522</BurninDate>
 </Feature>
 <Feature Name="HistoricalMonitorData">
 </Feature>
 <Feature Name="CloudHostedMachines">
 </Feature>
 <Feature Name="HdxInsightIntegration">
 </Feature>
 <Feature Name="RemotePC">
 </Feature>
</Features>
</Product>
</Products>

```

Returns the current feature table.



# Get-ConfigDBConnection

May 10, 2016

Gets the database string for the specified data store used by the Configuration Service.

## Syntax

```
Get-ConfigDBConnection [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the database connection string for the specified data store.

If the returned string is blank, no valid connection string has been specified. In this case the service is running, but is idle and awaiting specification of a valid connection string.

## Related topics

[Get-ConfigServiceStatus](#)

[Set-ConfigDBConnection](#)

[Test-ConfigDBConnection](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

system.string

The database connection string configured for the Configuration Service.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

NoDBConnections

The database connection string for the Configuration Service has not been specified.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Get-ConfigDBConnection

Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True  
Get the database connection string for the Configuration Service.

# Get-ConfigDBSchema

May 10, 2016

Gets a script that creates the Configuration Service database schema for the specified data store.

## Syntax

```
Get-ConfigDBSchema [-DatabaseName <String>] [-ServiceGroupName <String>] [-ScriptType <ScriptTypes>] [-LocalDatabase] [-Sid <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets SQL scripts that can be used to create a new Configuration Service database schema, add a new Configuration Service to an existing site, remove a Configuration Service from a site, or create a database server logon for a Configuration Service. If no Sid parameter is provided, the scripts obtained relate to the currently selected Configuration Service instance, otherwise the scripts relate to Configuration Service instance running on the machine identified by the Sid provided. When obtaining the Evict script, a Sid parameter must be supplied. The current service instance is that on the local machine, or that explicitly specified by the last usage of the -AdminAddress parameter to a Configuration SDK cmdlet. The service instance used to obtain the scripts does not need to be a member of a site or to have had its database connection configured. The database scripts support only Microsoft SQL Server, or SQL Server Express, and require Windows integrated authentication to be used. They can be run using SQL Server's SQLCMD utility, or by copying the script into an SQL Server Management Studio (SSMS) query window and executing the query. If using SSMS, the query must be executed in 'SMDCMD mode'. The ScriptType parameter determines which script is obtained. If ScriptType is not specified, or is FullDatabase, the script contains:

- o Creation of service schema
- o Creation of database server logon
- o Creation of database user
- o Addition of database user to Configuration Service roles

If ScriptType is Instance, the returned script contains:

- o Creation of database server logon
- o Creation of database user
- o Addition of database user to Configuration Service roles

If ScriptType is Evict, the returned script contains:

- o Removal of Configuration Service instance from database
- o Removal of database user

If ScriptType is Login, the returned script contains:

- o Creation of database server logon only

If the service uses two data stores they can exist in the same database. You do not need to configure a database before

using this command.

## Related topics

[Set-ConfigDBConnection](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database for which the schema will be generated.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ServiceGroupName**<String>

Specifies the name of the service group to be used when creating the database schema. The service group is a collection of all the Configuration services that share the same database instance and are considered equivalent; that is, all the services within a service group can be used interchangeably.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ScriptType**<ScriptTypes>

Specifies the type of database script returned. Available script types are:

### Database

Returns a full database script that can be used to create a database schema for the Configuration Service in a database instance that does not already contain a schema for this service. The DatabaseName and ServiceGroupName parameters must be specified to create a script of this type.

### Instance

Returns a permissions script that can be used to add further Configuration services to an existing database instance that already contains the full Configuration service schema, associating the services to the Service Group. The Sid parameter can optionally be specified to create a script of this type.

### Login

Returns a database logon script that can be used to add the required logon accounts to an existing database instance that contains the Configuration Service schema. This is used primarily when creating a mirrored database environment. The DatabaseName parameter must be specified to create a script of this type.

#### Evict

Returns a script that can be used to remove the specified Configuration Service from the database entirely. The DatabaseName and Sid parameters must be specified to create a script of this type.

Required?	false
Default Value	Database
Accept Pipeline Input?	false

#### **-LocalDatabase**<SwitchParameter>

Specifies whether the database script is to be used in a database instance run on the same controller as other services in the service group. Including this parameter ensures the script creates only the required permissions for local services to access the database schema for Configuration services.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Sid**<String>

Specifies the SID of the controller on which the Configuration Service instance to remove from the database is running.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
-----------	-------

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

### Systemstring

A string containing the required SQL script for application to a database.

## Notes

The scripts returned support Microsoft SQL Server Express Edition, Microsoft SQL Server Standard Edition, and Microsoft SQL Server Enterprise Edition databases only, and are generated on the assumption that integrated authentication will be used.

If the ScriptType parameter is not included or set to 'FullDatabase', the full database script is returned, which will:

Create the database schema.

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist).

If the ScriptType parameter is set to 'Instance', the script will:

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist) and associate it with a user.

If the ScriptType parameter is set to 'Login', the script will:

Create the logon (providing the schema does not already exist) and associate it with a pre-existing user of the same name.

If the LocalDatabase parameter is included, the NetworkService account will be added to the list of accounts permitted to access the database. This is required only if the database is run on a controller.

If the command fails, the following errors can be returned.

### Error Codes

-----

#### GetSchemasFailed

The database schema could not be found.

#### ActiveDirectoryAccountResolutionFailed

The specified Active Directory account or Group could not be found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-ConfigDBSchema -DatabaseName MyDB -ServiceGroupName MyServiceGroup > c:\ConfigSchema.sql
Get the full database schema for site data store of the Configuration Service and copy it to a file called
'c:\ConfigSchema.sql'.
```

This script can then be used to create the schema in a pre-existing database named 'MyDB' that does not already contain a Configuration Service site schema.

----- **EXAMPLE 2** -----

```
c:\PS>Get-ConfigDBSchema -DatabaseName MyDB -scriptType Login > c:\ConfigurationLogins.sql
Get the logon scripts for the Configuration Service.
```

# Get-ConfigDBVersionChangeScript

May 10, 2016

Gets a script that updates the Configuration Service database schema.

## Syntax

```
Get-ConfigDBVersionChangeScript -DatabaseName <String> -TargetVersion <Version> [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Returns a database script that can be used to upgrade or downgrade the site or secondary schema for the Configuration Service from the current schema version to a different version.

## Related topics

[Get-ConfigInstalledDBVersion](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database instance to which the update applies.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-TargetVersion**<Version>

Specifies the version of the database you want to update to.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

--	--



Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.Management.Automation.PSObject

A PSObject containing the required SQL script for application to a database.

## Notes

The PSObject returned by this cmdlet contains the following properties:

-- Script The raw text of the SQL script to apply the update, or null in the case when no upgrade path to the specified target version exists.

-- NeedExclusiveAccess Indicates whether all services in the service group must be shut down during the update or not.

-- CanUndo Indicates whether the generated script allows the updated schema to be reverted to the state prior to the update.

Scripts to update the schema version are stored in the database so any service in the service group can obtain these scripts. Extreme caution should be exercised when using update scripts. Citrix recommends backing up the database before attempting to upgrade the schema. Database update scripts may require exclusive use of the schema and so may not be able to execute while any Configuration services are running. However, this depends on the specific update being carried out.

After a schema update has been carried out, services that require the previous version of the schema may cease to operate. The ServiceState parameter reported by the Get-ConfigServiceStatus command provides information about service compatibility. For example, if the schema has been upgraded to a more recent version that a service cannot use, the service reports "DBNewerVersionThanService".

If the command fails, the following errors can be returned.

## Error Codes

-----

### NoOp

The operation was successful but had no effect.

### NoDBConnections

The database connection string for the Configuration Service has not been specified.

### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> $update = Get-ConfigDBVersionChangeScript -DatabaseName MyDb -TargetVersion 1.0.75.0
```

```
C:\PS> $update.Script > update_75.sql
```

Gets an SQL update script to update the current schema to version 1.0.75.0. The resulting update\_75.sql script is suitable for direct use with the SQL Server SQLCMD utility.

# Get-ConfigEnabledFeature

May 10, 2016

Lists features of the site that are enabled.

## Syntax

```
Get-ConfigEnabledFeature [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

## Related topics

[Set-ConfigSite](#)

[Import-ConfigFeatureTable](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.String

## Examples

----- **EXAMPLE 1** -----

```
C:\PS> Get-ConfigEnabledFeature
```

Retrieves the list of enabled features for the site's current configuration.

# Get-ConfigInstalledDBVersion

May 10, 2016

Gets a list of all available database schema versions for the Configuration Service.

## Syntax

```
Get-ConfigInstalledDBVersion [-Upgrade] [-Downgrade] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the current version of the Configuration Service database schema, if no flags are set, otherwise returns versions for which upgrade or downgrade scripts are available and have been stored in the database.

## Related topics

## Parameters

**-Upgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be updated should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Downgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be reverted should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

### System.Version

The Get-ConfigInstalledDbVersion command returns objects containing the new definition of the Configuration Service database schema version.

Major <Integer>

Minor <Integer>

Build <Integer>

Revision <Integer>

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

Both the Upgrade and Downgrade flags were specified.

#### NoOp

The operation was successful but had no effect.

#### NoDBConnections

The database connection string for the Configuration Service has not been specified.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ConfigInstalledDBVersion
```

Major	Minor	Build	Revision
-------	-------	-------	----------

-----	-----	-----	-----
-------	-------	-------	-------

5	6	0	0
---	---	---	---

Get the currently installed version of the Configuration Service database schema.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-ConfigInstalledDBVersion -Upgrade
```

Major	Minor	Build	Revision
-------	-------	-------	----------

-----	-----	-----	-----
-------	-------	-------	-------

6	0	0	0
---	---	---	---

Get the versions of the Configuration Service database schema for which upgrade scripts are supplied.

# Get-ConfigLicensingModel

May 10, 2016

Lists the supported licensing models.

## Syntax

```
Get-ConfigLicensingModel -ProductCode <String> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

## Related topics

[Get-ConfigProduct](#)

[Get-ConfigSite](#)

[Set-ConfigSite](#)

[Import-ConfigFeatureTable](#)

## Parameters

**-ProductCode**<String>

The product code

Required?	true
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.String

The list of supported licensing models for the specified product code.

## Notes

The Get-ConfigProduct cmdlet lists the available product codes.

The site object returned by the Get-ConfigSite cmdlet contains the currently configured product code.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Get-ConfigLicensingModel -ProductCode "XDS"
```

Retrieves the list of supported licensing models for product code "XDS".



# Get-ConfigLocalData

May 10, 2016

Gets the service local data.

## Syntax

```
Get-ConfigLocalData [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Get-ConfigLocalData returns service-and-controller-specific local data. This information is not site-wide, but rather controller-specific. The Configuration Service currently stores the controller product version as local data. The overall site version used by Studio is an aggregate of the product versions from each controller.

## Related topics

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

This cmdlet does not accept pipeline input

## Return Values

Citrix.Configuration.DataModel.LocalData

Contains the ControllerProductVersion field

## Examples

----- **EXAMPLE 1** -----

```
C:\PS> Get-ConfigLocalData
```

ControllerProductVersion

-----

7.1

Gets the service local data.



# Get-ConfigProduct

May 10, 2016

Lists the site's supported product names and codes.

## Syntax

```
Get-ConfigProduct [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

## Related topics

[Get-ConfigSite](#)

[Set-ConfigSite](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

PSObject

## Examples

----- **EXAMPLE 1** -----

```
C:\PS> Get-ConfigProduct
```

Lists the supported products by name and code.

# Get-ConfigProductEdition

May 10, 2016

Lists the supported product editions.

## Syntax

```
Get-ConfigProductEdition [-ProductCode] <String> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

## Related topics

[Get-ConfigProduct](#)

[Get-ConfigSite](#)

[Set-ConfigSite](#)

[Import-ConfigFeatureTable](#)

## Parameters

**-ProductCode**<String>

The product code

Required?	true
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.String

The list of supported licensing models for the specified product code.

## Notes

The Get-ConfigProduct cmdlet lists the available product codes.

The site object returned by Get-ConfigSite cmdlet contains the currently configured product code.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Get-ConfigProductEdition -ProductCode "XDS"
```

Retrieves the list of supported editions for XenDesktop.

# Get-ConfigProductFeature

May 10, 2016

Lists the supported features.

## Syntax

```
Get-ConfigProductFeature [-ProductCode] <String> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Lists the supported features. Use the Get-ConfigEnabledFeature command to determine which features are currently enabled.

## Related topics

[Get-ConfigProduct](#)

[Get-ConfigSite](#)

[Set-ConfigSite](#)

[Import-ConfigFeatureTable](#)

## Parameters

**-ProductCode**<String>

The product code

Required?	true
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.String

The list of supported licensing models for the specified product code.

## Notes

The Get-ConfigProduct cmdlet lists the available product codes.

The site object returned by Get-ConfigSite cmdlet contains the currently configured product code.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Get-ConfigProductFeature -ProductCode "XDS"
```

Retrieves the list of supported features for XenDesktop.

# Get-ConfigProductVersion

May 10, 2016

Lists the supported product versions.

## Syntax

```
Get-ConfigProductVersion [-ProductCode] <String> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

## Related topics

[Get-ConfigProduct](#)

[Get-ConfigSite](#)

[Set-ConfigSite](#)

[Import-ConfigFeatureTable](#)

## Parameters

**-ProductCode**<String>

The product code

Required?	true
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.String



The list of supported licensing models for the specified product code.

## Notes

The Get-ConfigProduct cmdlet lists the available product codes.

The site object returned by Get-ConfigSite cmdlet contains the currently configured product code.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Get-ConfigProductVersion -ProductCode "XDS"
```

Retrieves the list of supported versions for XenDesktop.

# Get-ConfigRegisteredServiceInstance

May 10, 2016

Gets the service instances that are registered in the directory.

## Syntax

```
Get-ConfigRegisteredServiceInstance [-ServiceInstanceId <Guid>] [-ServiceGroupUid <Guid>] [-ServiceGroupName <String>] [-ServiceType <String>] [-Address <String>] [-Binding <String>] [-Version <Int32>] [-ServiceAccountSid <String>] [-InterfaceType <String>] [-Metadata <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Use this cmdlet to retrieve the service instances currently registered with the Configuration Service that match the parameters supplied. If no parameters are supplied, all the service instances are returned.

## Related topics

[Register-ConfigServiceInstance](#)

[Unregister-ConfigRegisteredServiceInstance](#)

[Set-ConfigRegisteredServiceInstance](#)

## Parameters

**-ServiceInstanceId**<Guid>

The unique identifier for the service instance.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ServiceGroupUid**<Guid>

The unique identifier for the service group to which the service instance belongs.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ServiceGroupName<String>**

The name for the service group to which the service instance belongs.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ServiceType<String>**

The service type for the service instance.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Address<String>**

The connection address for the service instance.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Binding<String>**

The binding for the service instance.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Version<Int32>**

The service instance version.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ServiceAccountSid**<String>

The AD account SID for the computer account that the computer hosting the service instance is running as.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-InterfaceType**<String>

The interface type for the service instance.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ReturnTotalRecordCount**<SwitchParameter>

See about\_Config\_Filtering for details.

Required?	false
Default Value	false
Accept Pipeline Input?	false

### **-MaxRecordCount**<Int32>

See about\_Config\_Filtering for details.

Required?	false
Default Value	250
Accept Pipeline Input?	false

### **-Skip**<Int32>

See about\_Config\_Filtering for details.

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy**<String>

See about\_Config\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Filter**<String>

See about\_Config\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.Configuration.Sdk.ServiceInstance

This represents a service instance and has the following parameters;

ServiceGroupUid <Guid>

The unique identifier for the service group to which the service instance belongs.

ServiceGroupName <string>

The name of the service group to which the service instance belongs.

ServiceInstanceUid <Guid>

The unique identifier for the service instance.

ServiceType <string>

The type of the service group.

Address <string>

The contact address for the service instance.

Binding <string>

The binding to use for connections to the service instance.

Version <int>

The version of the service instance.

ServiceAccount <string>

The AD computer account for the computer that is providing the service instance.

ServiceAccountSid <string>

The AD computer account SID for the computer that is providing the service instance.

InterfaceType <string>

The interface type for the service instance.

Metadata <Citrix.Configuration.Sdk.Metadata[]>

The metadata for the service instance.

## Notes

In the case of failure, the following errors can result.

Error Codes ----- PartialData Only a subset of the available data was returned. CouldNotQueryDatabase The query required to get the database was not defined. CommunicationError An error occurred while communicating with the service. DatabaseNotConfigured The operation could not be completed because the database for the service is not configured. InvalidFilter A filtering expression was supplied that could not be interpreted for this cmdlet. ExceptionThrown An unexpected error occurred. To locate more details see the Windows event logs on the controller being used, or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

C:\>Get-ConfigRegisteredServiceInstance

Address : http://MyServer.com:80/Citrix/MyContract/v1  
Binding : wcf\_HTTP\_kerb

InterfaceType : SDK  
Metadata : {}  
MetadataMap : {}  
ServiceAccount : ENG\MyAccount\$  
ServiceAccountSid : S-1-5-21-1155438255-2213498043-2452000591-1104  
ServiceGroupName : MyService  
ServiceGroupUid : 2b990d5a-bba9-413b-aa08-e104e67f89bc  
ServiceInstanceUid : 8dc38b5a-3fbb-457c-b326-6c41c94c18d5  
ServiceType : MySnapIn  
Version : 1

Address : http://MyServer.com:80/Citrix/MyContract/PeerAPI/v1  
Binding : wcf\_HTTP\_kerb  
InterfaceType : Peer  
Metadata : {}  
MetadataMap : {}  
ServiceAccount : ENG\MyAccount\$  
ServiceAccountSid : S-1-5-21-1155438255-2213498043-2452000591-1104  
ServiceGroupName : MyService  
ServiceGroupUid : 2b990d5a-bba9-413b-aa08-e104e67f89bc  
ServiceInstanceUid : 8f822ed6-42f3-4a26-911a-a4a6a87c0ef2  
ServiceType : MySnapIn  
Version : 1

Address : http://MyServer.com:80/Citrix/MyContract/MyServiceEnvTestAPI/v1  
Binding : wcf\_HTTP\_kerb  
InterfaceType : EnvironmentTest  
Metadata : {}  
MetadataMap : {}  
ServiceAccount : ENG\MyAccount\$  
ServiceAccountSid : S-1-5-21-1155438255-2213498043-2452000591-1104  
ServiceGroupName : MyService  
ServiceGroupUid : 2b990d5a-bba9-413b-aa08-e104e67f89bc  
ServiceInstanceUid : d2d40d9b-2a5d-4c5a-b9ca-a7f73cffe4f2  
ServiceType : MySnapIn  
Version : 1

Address : http://MyServer.com:80/Citrix/MyContract/MyServiceAPI/v1  
Binding : wcf\_HTTP\_kerb  
InterfaceType : InterService  
Metadata : {}  
MetadataMap : {}  
ServiceAccount : ENG\MyAccount\$  
ServiceAccountSid : S-1-5-21-1155438255-2213498043-2452000591-1104  
ServiceGroupName : MyService  
ServiceGroupUid : 2b990d5a-bba9-413b-aa08-e104e67f89bc  
ServiceInstanceUid : 5d428970-2ba1-4336-b8d0-f3aa961b8983  
ServiceType : MySnapIn



Version : 1

Return all the service instances that are registered in the Configuration Service.

----- **EXAMPLE 2** -----

C:\>Get-ConfigRegisteredServiceInstance -InterfaceType "SDK"

Address : http://MyServer.com:80/Citrix/MyContract/v1

Binding : wcf\_HTTP\_kerb

InterfaceType : SDK

Metadata : {}

MetadataMap : {}

ServiceAccount : ENG\MyAccount\$

ServiceAccountSid : S-1-5-21-1155438255-2213498043-2452000591-1104

ServiceGroupName : MyService

ServiceGroupUid : 2b990d5a-bba9-413b-aa08-e104e67f89bc

ServiceInstanceUid : 8dc38b5a-3fbb-457c-b326-6c41c94c18d5

ServiceType : MySnapIn

Version : 1

Return all the service instances that are registered in the Configuration Service and are of type 'SDK'.

# Get-ConfigService

May 10, 2016

Gets the service record entries for the Configuration Service.

## Syntax

```
Get-ConfigService [-Metadata <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns instances of the Configuration Service that the service publishes. The service records contain account security identifier information that can be used to remove each service from the database.

A database connection for the service is required to use this command.

## Related topics

### Parameters

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See `about_Config_Filtering` for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by `-ReturnTotalRecordCount`.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Config\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.Configuration.Sdk.Service

The Get-ConfigServiceInstance command returns an object containing the following properties.

Uid <Integer>

Specifies the unique identifier for the service in the group. The unique identifier is an index number.

ServiceHostId <Guid>

Specifies the unique identifier for the service instance.

DNSName <String>

Specifies the domain name of the host on which the service runs.

MachineName <String>

Specifies the short name of the host on which the service runs.

CurrentState <Citrix.Fma.Sdk.ServiceCore.ServiceState>

Specifies whether the service is running, started but inactive, stopped, or failed.

LastStartTime <DateTime>

Specifies the date and time at which the service was last restarted.

LastActivityTime <DateTime>

Specifies the date and time at which the service was last stopped or restarted.

OSType

Specifies the operating system installed on the host on which the service runs.

OSVersion

Specifies the version of the operating system installed on the host on which the service runs.

ServiceVersion

Specifies the version number of the service instance. The version number is a string that reflects the full build version of the service.

DatabaseUserName <string>

Specifies for the service instance the Active Directory account name with permissions to access the database. This will be either the machine account or, if the database is running on a controller, the NetworkService account.

Sid <string>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

ActiveSiteServices <string[]>

Specifies the names of active site services currently running in the service. Site services are components that perform long-running background processing in some services. This field is empty for services that do not contain site services.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

UnknownObject

One of the specified objects was not found.

PartialData

Only a subset of the available data was returned.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

CouldNotQueryDatabase

The query required to get the database was not defined.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-ConfigService
```

```
Uid : 1
ServiceHostId : aef6f464-f1ee-4042-a523-66982e0cecd0
DNSName : MyServer.company.com
MachineName : MYSERVER
CurrentState : On
LastStartTime : 04/04/2011 15:25:38
LastActivityTime : 04/04/2011 15:33:39
OSType : Win32NT
OSVersion : 6.1.7600.0
ServiceVersion : 5.1.0.0
DatabaseUserName : NT AUTHORITY\NETWORK SERVICE
SID : S-1-5-21-2316621082-1546847349-2782505528-1165
ActiveSiteServices : {MySiteService1, MySiteService2...}
Get all the instances of the Configuration Service running in the current service group.
```

# Get-ConfigServiceAddedCapability

May 10, 2016

Gets any added capabilities for the Configuration Service on the controller.

## Syntax

```
Get-ConfigServiceAddedCapability [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables updates to the Configuration Service on the controller to be detected.

You do not need to configure a database connection before using this command.

## Related topics

### Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.String

String containing added capabilities.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ConfigServiceAddedCapability
```

Get the added capabilities of the Configuration Service.



# Get-ConfigServiceGroup

May 10, 2016

Gets the service groups that match the parameters supplied.

## Syntax

```
Get-ConfigServiceGroup [-ServiceGroupUid <Guid>] [-ServiceGroupName <String>] [-ServiceType <String>] [-Metadata <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Use this cmdlet to retrieve existing service groups that match the parameters supplied. If no parameters are supplied, all the service groups are returned.

## Related topics

## Parameters

### **-ServiceGroupUid**<Guid>

The unique identifier for the service group.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ServiceGroupName**<String>

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ServiceType**<String>

The service type for the service group.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

See about\_Config\_Filtering for details.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

See about\_Config\_Filtering for details.

Required?	false
-----------	-------

Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

See about\_Config\_Filtering for details.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

See about\_Config\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Filter**<String>

See about\_Config\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false

Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Configuration.Sdk.ServiceGroup

This represents a service instance and has the following parameters;

ServiceGroupUid <Guid>

The unique identifier for the service group.

ServiceGroupName <string>

The name of the service group.

ServiceType <string>

The type of the service group.

Metadata <Citrix.Configuration.Sdk.Metadata[]>

The metadata for the service group.

## Notes

In the case of failure, the following errors can result.

Error Codes -----

DatabaseNotConfigured The operation could not be completed because the database for the service is not configured. PartialData Only a subset of the available data was returned. CouldNotQueryDatabase The Query required to get the database was not defined. CommunicationError An error occurred while communicating with the service. InvalidFilter A filtering expression was supplied that could not be interpreted for this cmdlet. ExceptionThrown An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

C:\>Get-ConfigServiceGroup

Return all the service groups that are registered in the Configuration Service.

### ----- EXAMPLE 2 -----

C:\>Get-ConfigRegisteredServiceInstance -ServiceType "config"

Return all the service groups that are registered in the Configuration Service and are of type 'config' (i.e. the service groups for the Configuration Service).

# Get-ConfigServiceInstance

May 10, 2016

Gets the service instance entries for the Configuration Service.

## Syntax

```
Get-ConfigServiceInstance [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns service interfaces published by the instance of the Configuration Service. Each instance of a service publishes multiple interfaces with distinct interface types, and each of these interfaces is represented as a ServiceInstance object. Service instances can be used to register the service with a central configuration service so that other services can use the functionality.

You do not need to configure a database connection to use this command.

## Related topics

## Parameters

**-AdminAddress** <String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Configuration.Sdk.ServiceInstance

The Get-ConfigServiceInstance command returns an object containing the following properties.

ServiceGroupUid <Guid>

Specifies the unique identifier for the service group of which the service is a member.

ServiceGroupName <String>

Specifies the name of the service group of which the service is a member.

ServiceInstanceUID <Guid>

Specifies the unique identifier for registered service instances, which are service instances held by and obtained from a

central configuration service. Unregistered service instances do not have unique identifiers.

ServiceType <String>

Specifies the service instance type. For this service, the service instance type is always Config.

Address

Specifies the address of the service instance. The address can be used to access the service and, when registered in the central configuration service, can be used by other services to access the service.

Binding

Specifies the binding type that must be used to communicate with the service instance. In this release of XenDesktop, the binding type is always 'wcf\_HTTP\_kerb'. This indicates that the service provides a Windows Communication Foundation endpoint that uses HTTP binding with integrated authentication.

Version

Specifies the version of the service instance. The version number is used to ensure that the correct versions of the services are used for communications.

ServiceAccount <String>

Specifies the Active Directory account name for the machine on which the service instance is running. The account name is used to provide information about the permissions required for interservice communications.

ServiceAccountSid <String>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

InterfaceType <String>

Specifies the interface type. Each service can provide multiple service instances, each for a different purpose, and the interface defines the purpose. Available interfaces are:

SDK - for PowerShell operations

InterService - for operations between different services

Peer - for communications between services of the same type

Metadata <Citrix.Configuration.Sdk.Metadata[]>

The collection of metadata associated with registered service instances, which are service instances held by and obtained from a central configuration service. Metadata is not stored for unregistered service instances.

Notes

If the command fails, the following errors can be returned.

Error Codes

-----

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ConfigServiceInstance
```

```
Address : http://MyServer.com:80/Citrix/ConfigurationService
Binding : wcf_HTTP_kerb
InterfaceType : SDK
Metadata :
MetadataMap :
ServiceAccount : ENG\MyAccount$
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164
ServiceGroupName : MyServiceGroup
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000
ServiceType : Config
Version : 1
```

```
Address : http://MyServer.com:80/Citrix/ConfigurationService/IServiceApi
Binding : wcf_HTTP_kerb
InterfaceType : InterService
Metadata :
MetadataMap :
```

ServiceAccount : ENG\MyAccount  
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164  
ServiceGroupName : MyServiceGroup  
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d  
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000  
ServiceType : Config  
Version : 1

Get all instances of the Configuration Service running on the specified machine. For remote services, use the AdminAddress parameter to define the service for which the interfaces are required. If the AdminAddress parameter has not been specified for the runspace, service instances running on the local machine are returned.



# Get-ConfigServiceStatus

May 10, 2016

Gets the current status of the Configuration Service on the controller.

## Syntax

```
Get-ConfigServiceStatus [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables the status of the Configuration Service on the controller to be monitored. If the service has multiple data stores it will return the overall state as an aggregate of all the data store states. For example, if the site data store status is OK and the secondary data store status is DBUnconfigured then it will return DBUnconfigured.

## Related topics

[Set-ConfigDBConnection](#)

[Test-ConfigDBConnection](#)

[Get-ConfigDBConnection](#)

[Get-ConfigDBSchema](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Get-ConfigServiceStatus command returns an object containing the status of the Configuration Service together with extra diagnostics information.

DBUnconfigured

The Configuration Service does not have a database connection configured.

DBRejectedConnection

The database rejected the logon attempt from the Configuration Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

#### InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the Configuration Service schema has not been added to the database.

#### DBNotFound

The specified database could not be located with the configured connection string.

#### DBNewerVersionThanService

The version of the Configuration Service currently in use is incompatible with the version of the Configuration Service schema on the database. Upgrade the Configuration Service to a more recent version.

#### DBOlderVersionThanService

The version of the Configuration Service schema on the database is incompatible with the version of the Configuration Service currently in use. Upgrade the database schema to a more recent version.

#### DBVersionChangeInProgress

A database schema upgrade is currently in progress.

#### OK

The Configuration Service is running and is connected to a database containing a valid schema.

#### Failed

The Configuration Service has failed.

#### Unknown

(0) The service status cannot be determined.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ConfigServiceStatus
```

DBUnconfigured

Get the current status of the Configuration Service.

# Get-ConfigSite

May 10, 2016

Gets the site.

## Syntax

```
Get-ConfigSite [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Get-ConfigSite cmdlet gets the site.

A XenDesktop installation has only a single site instance.

## Related topics

[Set-ConfigSite](#)

[Import-ConfigFeatureTable](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Get-ConfigSite returns the site instance.

## Examples

----- **EXAMPLE 1** -----

```
C:\PS> Get-ConfigSite
```

Gets the site.

# Import-ConfigFeatureTable

May 10, 2016

Sets the feature table of the site.

## Syntax

```
Import-ConfigFeatureTable [-Path] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Import-ConfigFeatureTable -Content <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

## Related topics

[Export-ConfigFeatureTable](#)

[Get-ConfigSite](#)

[Set-ConfigSite](#)

[Get-ConfigProduct](#)

[Get-ConfigProductEdition](#)

[Get-ConfigProductFeature](#)

[Get-ConfigProductVersion](#)

[Get-ConfigLicensingModel](#)

## Parameters

**-Path**<String>

The path to the file containing the feature table

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Content**<String>

Set the site's feature table.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByValue)
------------------------	----------------

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Import-ConfigFeatureTable $xml
```

Specifies the use of a Platinum edition license. A suitable license must be available on the site's license server.

# Register-ConfigServiceInstance

May 10, 2016

Allows the registration of a service instance.

## Syntax

```
Register-ConfigServiceInstance -ServiceGroupUid <Guid> -ServiceGroupName <String> -ServiceType <String> -Address <String> -Binding <String> -Version <Int32> -ServiceAccountSid <String> -InterfaceType <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Register-ConfigServiceInstance -ServiceGroupUid <Guid> -ServiceGroupName <String> -ServiceType <String> -Address <String> -Binding <String> -Version <Int32> -ServiceAccount <String> -InterfaceType <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Register-ConfigServiceInstance -ServiceInstance <ServiceInstance[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Use this cmdlet to register service instance items in the Configuration Service. Service instances can be registered either by retrieving the data directly from other services or by manually entering the details into this command.

If the service group specified by the service instance already exists, the service is added to the service group, otherwise a new service group is created to hold the service instance.

## Related topics

[Unregister-ConfigRegisteredServiceInstance](#)

[Add-ConfigRegisteredServiceInstanceMetadata](#)

[Set-ConfigRegisteredServiceInstanceMetadata](#)

[Remove-ConfigRegisteredServiceInstanceMetadata](#)

[Add-ConfigServiceGroupMetadata](#)

[Set-ConfigServiceGroupMetadata](#)

[Remove-ConfigServiceGroupMetadata](#)

## Parameters

**-ServiceGroupUid**<Guid>

The Service Group Unique Identifier

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

--	--

#### **-ServiceGroupName<String>**

The name of the service group

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-ServiceType<String>**

The type of the service group

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Address<String>**

The address that is used to access the service instance.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Binding<String>**

The binding type that must be used to access the service instance.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Version<Int32>**



The version of the service instance.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ServiceAccountSid**<String>

The AD computer account Sid for the computer on which the service instance is hosted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-InterfaceType**<String>

The type of interface that the service provides (i.e. SDK, InterService, Peer).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ServiceAccount**<String>

The AD computer account name (domain qualified) for the computer on which the service instance is hosted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ServiceInstance**<ServiceInstance[]>

The service instances to register.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the logging id of the high-level operation that this cmdlet is part of.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.Configuration.Sdk.ServiceInstance An object with the following parameters can be used to register a service instance. Address, ServiceGroupUid, ServiceGroupName, ServiceType, Binding, Version, InterfaceType.

#### Return Values

Citrix.Configuration.Sdk.ServiceInstance

This represents a service instance and has the following parameters;

ServiceGroupUid <Guid>

The unique identifier for the service group to which the service instance belongs.

ServiceGroupName <string>

The name of the service group to which the service instance belongs.

ServiceInstanceId <Guid>

The unique identifier for the service instance.

ServiceType <string>

The type of the service group.

Address <string>

The contact address for the service instance.

Binding <string>

The binding to use for connections to the service instance.

Version <int>

The version of the service instance.

ServiceAccount <string>

The AD computer account for the computer that is providing the service instance.

ServiceAccountSid <string>

The AD computer account SID for the computer that is providing the service instance.

InterfaceType <string>

The interface type for the service instance.

Metadata <Citrix.Configuration.Sdk.Metadata[]>

The metadata for the service instance.

## Notes

In the case of failure, the following errors can result.

Error Codes ----- ActiveDirectoryAccountResolutionFailed The account name provided could not be found in Active Directory.

ServiceGroupWithSameUidExistsForDifferentServiceGroupNameOrSameUidExistsForDifferentServiceGroupNameOrServiceType The service group name or service type do not match the service group found with the specified uid. TypeAlreadyExists A different service group with the same type is registered already in the Configuration Service. DatabaseError An error occurred in the service while attempting a database operation. DatabaseNotConfigured The operation could not be completed because the database for the service is not configured. DataStoreException An error occurred in the service while attempting a database operation - communication with the database failed for various reasons. CommunicationError An error occurred while communicating with the service. ExceptionThrown An unexpected error occurred. For more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS>Get-ConfigServiceInstance | Register-ConfigServiceInstance
```

Gets the service instances for the Configuration Service and registers them.

# Remove-ConfigRegisteredServiceInstanceMetadata

May 10, 2016

Removes metadata from the given ServiceInstance.

## Syntax

```
Remove-ConfigRegisteredServiceInstanceMetadata [-ServiceInstanceId] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ConfigRegisteredServiceInstanceMetadata [-ServiceInstanceId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ConfigRegisteredServiceInstanceMetadata [-InputObject] <ServiceInstance[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ConfigRegisteredServiceInstanceMetadata [-InputObject] <ServiceInstance[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given ServiceInstance.

## Related topics

[Add-ConfigRegisteredServiceInstanceMetadata](#)

[Set-ConfigRegisteredServiceInstanceMetadata](#)

## Parameters

**-ServiceInstanceId**<Guid>

Id of the ServiceInstance

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<ServiceInstance[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByValue)
------------------------	----------------

#### **-Name**<String>

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
-----------	-------

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ConfigServiceInstance | Remove-ConfigRegisteredServiceInstanceMetadata
```

Remove all metadata from all ServiceInstance objects.

# Remove-ConfigServiceGroup

May 10, 2016

Removes service groups.

## Syntax

```
Remove-ConfigServiceGroup [-ServiceGroupUid] <Guid> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Use this cmdlet to remove a service group and all the service instances that it contains.

## Related topics

[Register-ConfigServiceInstance](#)

[Add-ConfigServiceGroupMetadata](#)

[Set-ConfigServiceGroupMetadata](#)

[Remove-ConfigServiceGroupMetadata](#)

## Parameters

**-ServiceGroupUid**<Guid>

The unique identifier for the service group.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the logging id of the high-level operation this cmdlet invocation is part of.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>



Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

In the case of failure, the following errors can result.

Error Codes ----- ServiceGroupObjectNotFound The service group specified could not be located. DatabaseError An error occurred in the service while attempting a database operation. DatabaseNotConfigured The operation could not be completed because the database for the service is not configured. DataStoreException An error occurred in the service while attempting a database operation - communication with the database failed for various reasons. CommunicationError An error occurred while communicating with the service. ExceptionThrown An unexpected error occurred. For more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS>Remove-ConfigServiceGroup -ServiceGroupUid 31951f6f-7703-4abb-938a-2861d76ecfea
```

Removes the service group (and all contained service instances) for the service group with a unique identifier of '31951f6f-7703-4abb-938a-2861d76ecfea'.

### ----- EXAMPLE 2 -----

```
C:\PS> Get-configServiceGroup | remove-ConfigServiceGroup
```

Removes all the service groups (and all contained service instances) for the XenDesktop deployment.

# Remove-ConfigServiceGroupMetadata

May 10, 2016

Removes metadata from the given ServiceGroup.

## Syntax

```
Remove-ConfigServiceGroupMetadata [-ServiceGroupUid] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ConfigServiceGroupMetadata [-ServiceGroupUid] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ConfigServiceGroupMetadata [-InputObject] <ServiceGroup[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ConfigServiceGroupMetadata [-InputObject] <ServiceGroup[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given ServiceGroup.

## Related topics

[Add-ConfigServiceGroupMetadata](#)

[Set-ConfigServiceGroupMetadata](#)

## Parameters

**-ServiceGroupUid**<Guid>

Id of the ServiceGroup

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<ServiceGroup[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByValue)
------------------------	----------------

#### **-Name**<String>

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
-----------	-------

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- **EXAMPLE 1** -----

```
c:\PS>Get-ConfigServiceGroup | % { Remove-ConfigServiceGroupMetadata -Map $_.MetadataMap }
```

Remove all metadata from all ServiceGroup objects.

# Remove-ConfigServiceMetadata

May 10, 2016

Removes metadata from the given Service.

## Syntax

```
Remove-ConfigServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-ConfigServiceMetadata [-ServiceHostId] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-ConfigServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-ConfigServiceMetadata [-InputObject] <Service[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Service.

## Related topics

[Set-ConfigServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name<String>**

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples



----- **EXAMPLE 1** -----

```
c:\PS>Get-ConfigService | % { Remove-ConfigServiceMetadata -Map $_.MetadataMap }
```

Remove all metadata from all Service objects.

# Remove-ConfigSiteMetadata

May 10, 2016

Removes metadata from the given Site.

## Syntax

```
Remove-ConfigSiteMetadata -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ConfigSiteMetadata -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Site.

## Related topics

[Set-ConfigSiteMetadata](#)

## Parameters

**-Name**<String>

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-

LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

##### InvalidParameterCombination

The cmdlet parameters are inconsistent.

##### UnknownObject

One of the specified objects was not found.

##### DatabaseError

An error occurred in the service while attempting a database operation.

##### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

##### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

##### PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-ConfigSite | % { Remove-ConfigSiteMetadata -Map $_.MetadataMap }
Remove all metadata from all Site objects.
```

# Reset-ConfigServiceGroupMembership

May 10, 2016

Reloads the access permissions and configuration service locations for the Configuration Service.

## Syntax

```
Reset-ConfigServiceGroupMembership [-ConfigServiceInstance] <ServiceInstance[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Enables you to reload Configuration Service access permissions and configuration service locations. The Reset-ConfigServiceGroupMembership command must be run on at least one instance of the service type (Config) after installation and registration with the configuration service. Without this operation, the Configuration services will be unable to communicate with other services in the XenDesktop deployment. When the command is run, the services are updated when additional services are added to the deployment, provided that the configuration service is not stopped. The Reset-ConfigServiceGroupMembership command can be run again to refresh this information if automatic updates do not occur when new services are added to the deployment. If more than one configuration service instance is passed to the command, the first instance that meets the expected service type requirements is used.

## Related topics

### Parameters

**-ConfigServiceInstance**<ServiceInstance[]>

Specifies the configuration service instance object that represents the service instance for the type 'InterService' that references a configuration service for the deployment.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Input Type

Citrix.Configuration.Sdk.ServiceInstance[] Service instances containing a ServiceInstance object that refers to the central configuration service interservice interface can be piped to the Reset-ConfigServiceGroupMembership command.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### NoSuitableServiceInstance

None of the supplied service instance objects were suitable for resetting service group membership.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config | Reset-ConfigServiceGroupMembership
```

Reset the service group membership for a service in a deployment where the configuration service is configured and running on the same machine as the service.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config -AdminAddress OtherServer.example.com | Reset-ConfigServiceGroupmembership
```

Reset the service group membership for a service in a deployment where the configuration service that is configured and running on a machine named 'OtherServer.example.com'.



# Set-ConfigDBConnection

May 10, 2016  
Configures a database connection for the Configuration Service.

## Syntax

Set-ConfigDBConnection [-DBConnection] <String> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Configures a connection to a database in which the Configuration Service can store its state. The service will attempt to connect and start using the database immediately after the connection is configured. The database connection string is updated to the specified value regardless of whether it is valid or not. Specifying an invalid connection string prevents a service from functioning until the error is corrected.

After a connection is configured, you cannot alter it without first clearing it (by setting the connection to \$null).

You do not need to configure a database connection to use this command.

## Related topics

[Get-ConfigServiceStatus](#)

[Get-ConfigDBConnection](#)

[Test-ConfigDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be used by the Configuration Service. Passing in \$null will clear any existing database connection configured.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Force**<SwitchParameter>

If present, allows the local administrator to set the connection string to null when there are problems contacting the database or other services.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.



Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Set-ConfigDBConnection command returns an object containing the status of the Configuration Service together with extra diagnostics information.

DBUnconfigured

The Configuration Service does not have a database connection configured.

DBRejectedConnection

The database rejected the logon attempt from the Configuration Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the Configuration Service schema has not been added to the database.

DBNotFound

The specified database could not be located with the configured connection string.

DBNewerVersionThanService

The version of the Configuration Service currently in use is incompatible with the version of the Configuration Service schema on the database. Upgrade the Configuration Service to a more recent version.

DBOlderVersionThanService

The version of the Configuration Service schema on the database is incompatible with the version of the Configuration Service currently in use. Upgrade the database schema to a more recent version.

DBVersionChangeInProgress

A database schema upgrade is currently in progress.

OK

The Configuration Service is running and is connected to a database containing a valid schema.

## Failed

The Configuration Service has failed.

## Unknown

The status of the Configuration Service cannot be determined.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidDBConnectionString

The database connection string has an invalid format.

#### DatabaseConnectionDetailsAlreadyConfigured

There was already a database connection configured. After a configuration is set, it can only be set to \$null.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Set-ConfigDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

#### OK

Configures a database connection string for the Configuration Service.

### ----- EXAMPLE 2 -----

```
c:\PS>Set-ConfigDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Configures an invalid database connection string for the Configuration Service.

# Set-ConfigRegisteredServiceInstance

May 10, 2016  
Updates a service instance.

## Syntax

Set-ConfigRegisteredServiceInstance -ServiceInstanceId <Guid> -Address <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>]  
[<CommonParameters>]

## Detailed Description

Use this cmdlet to change the address property of an existing service instance that is registered in the Configuration Service.

## Related topics

- [Get-ConfigRegisteredServiceInstance](#)
- [Register-ConfigServiceInstance](#)
- [Unregister-ConfigRegisteredServiceInstance](#)

## Parameters

**-ServiceInstanceId**<Guid>

The unique identifier for the service instance to be updated.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Address**<String>

The new address for the service instance.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-PassThru**<SwitchParameter>

Defines whether or not the command returns a result showing the new state of the updated service instance.

Required?	false
Default Value	true
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the logging id of the high-level operation this cmdlet invocation is part of.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

Citrix.Configuration.Sdk.ServiceInstance

This represents a service instance and has the following parameters:

**ServiceGroupUid** <Guid>

The unique identifier for the service group to which the service instance belongs.

**ServiceGroupName** <string>

The name of the service group to which the service instance belongs.

**ServiceInstanceUid** <Guid>

The unique identifier for the service instance.

**ServiceType** <string>

The type of the service group.

**Address** <string>

The contact address for the service instance.

**Binding** <string>

The binding to use for connections to the service instance.

**Version** <int>

The version of the service instance.

**ServiceAccount** <string>

The AD computer account for the computer that is providing the service instance.

**ServiceAccountSid** <string>

The AD computer account SID for the computer that is providing the service instance.

**InterfaceType** <string>

The interface type for the service instance.

**Metadata** <Citrix.Configuration.Sdk.Metadata[]>

The metadata for the service instance.

#### Notes

In the case of failure, the following errors can result.

Error Codes ----- ObjectToUpdateDoesNotExist The service instance specified could not be located. DatabaseError An error occurred in the service while attempting a database operation. DatabaseNotConfigured The operation could not be completed because the database for the service is not configured. DataStoreException An error occurred in the service while attempting a database operation - communication with the database failed for various reasons. CommunicationError An error occurred while communicating with the service. ExceptionThrown An unexpected error occurred. For more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS>Set-ConfigResgisteredService -ServiceInstanceUid "9805f39d-99eb-44f0-8f63-9d8e3f1228e0" -Address "http://myServer.com/Citrix/sdkHostingUnitService"
```

Update the service instance with the unique identifier of '9805f39d-99eb-44f0-8f63-9d8e3f1228e0' to use the new address.

# Set-ConfigRegisteredServiceInstanceMetadata

May 10, 2016  
Adds or updates metadata on the given ServiceInstance.

## Syntax

```
Set-ConfigRegisteredServiceInstanceMetadata [-ServiceInstanceUid] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Set-ConfigRegisteredServiceInstanceMetadata [-ServiceInstanceUid] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Set-ConfigRegisteredServiceInstanceMetadata [-InputObject] <ServiceInstance[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Set-ConfigRegisteredServiceInstanceMetadata [-InputObject] <ServiceInstance[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability for additional custom data to be stored against given ServiceInstance objects.

## Related topics

- [Add-ConfigRegisteredServiceInstanceMetadata](#)
- [Remove-ConfigRegisteredServiceInstanceMetadata](#)

## Parameters

**-ServiceInstanceUid**<Guid>

Id of the ServiceInstance

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<ServiceInstance[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the ServiceInstance specified. The property cannot contain any of the following characters \;#.\*?=<>|[]{}"

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-ConfigRegisteredServiceInstanceMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Set-ConfigRegisteredServiceInstanceMetadata -ServiceInstanceId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
property	value

Add metadata with a name of 'property' and a value of 'value' to the ServiceInstance with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.



# Set-ConfigServiceGroupMetadata

May 10, 2016

Adds or updates metadata on the given ServiceGroup.

## Syntax

```
Set-ConfigServiceGroupMetadata [-ServiceGroupUid] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-ConfigServiceGroupMetadata [-ServiceGroupUid] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-ConfigServiceGroupMetadata [-InputObject] <ServiceGroup[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-ConfigServiceGroupMetadata [-InputObject] <ServiceGroup[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability for additional custom data to be stored against given ServiceGroup objects.

## Related topics

[Add-ConfigServiceGroupMetadata](#)

[Remove-ConfigServiceGroupMetadata](#)

## Parameters

**-ServiceGroupUid**<Guid>

Id of the ServiceGroup

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<ServiceGroup[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the ServiceGroup specified. The property cannot contain any of the following characters \/:;#.\*?=<>|[]()''

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.Collections.Generic.Dictionary[String,String]

Set-ConfigServiceGroupMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Set-ConfigServiceGroupMetadata -ServiceGroupUid 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
---	-----
property	value

Add metadata with a name of 'property' and a value of 'value' to the ServiceGroup with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Set-ConfigServiceMetadata

May 10, 2016

Adds or updates metadata on the given Service.

## Syntax

```
Set-ConfigServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-ConfigServiceMetadata [-ServiceHostId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Set-ConfigServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-ConfigServiceMetadata [-InputObject] <Service[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

## Detailed Description

Allows you to store additional custom data against given Service objects.

## Related topics

[Remove-ConfigServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with `@{"name1" = "val1"; "name2" = "val2"}`) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name<String>**

Specifies the property name of the metadata to be added. The property must be unique for the Service specified. The property cannot contain any of the following characters `\;/:;. *?=<>|[]()`

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Value<String>**

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the `Start-LogHighLevelOperation` and `Stop-LogHighLevelOperation` cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.Collections.Generic.Dictionary[String,String]

Set-ConfigServiceMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

InvalidParameterCombination

The cmdlet parameters are inconsistent.

UnknownObject

One of the specified objects was not found.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various

reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Set-ConfigServiceMetadata -ServiceHostId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
---	----
property	value

Add metadata with a name of 'property' and a value of 'value' to the Service with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.



# Set-ConfigSite

May 10, 2016

Changes the overall settings of the site.

## Syntax

```
Set-ConfigSite [-SiteName <String>] [-ProductCode <String>] [-ProductEdition <String>] [-ProductVersion <String>] [-LicensingModel <String>] [-LicenseServerName <String>] [-LicenseServerPort <Int32>] [-LicenseServerUri <Uri>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-ConfigSite cmdlet modifies properties of the site.

The site is a top-level, logical representation of the XenDesktop site, from the perspective of the configuration services running within the site.

A XenDesktop installation has only a single site instance.

Modifications to the product code, product edition, product version and licensing model properties are successful only if their values are consistent with the feature table. Use the Get-ConfigProduct, Get-ConfigProductEdition, Get-ConfigProductVersion and Get-ConfigLicensingModel cmdlets to determine consistent values.

To configure the site, first import the feature table using the Import-ConfigFeatureTable cmdlet.

## Related topics

[Export-ConfigFeatureTable](#)

[Get-ConfigSite](#)

[Get-ConfigProduct](#)

[Get-ConfigProductEdition](#)

[Get-ConfigProductFeature](#)

[Get-ConfigProductVersion](#)

[Get-ConfigLicensingModel](#)

## Parameters

**-SiteName**<String>

Changes the name of the site.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ProductCode<String>**

Changes the product code.

The Get-ConfigProduct cmdlet returns a list of supported values.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ProductEdition<String>**

Changes the license edition. A license matching the specified edition must be available within the site's license server.

The Get-ConfigProductEdition returns a list of supported values.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ProductVersion<String>**

Changes the product version.

The Get-ConfigProductVersion returns a list of supported values.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LicensingModel<String>**

Changes the license model. A license matching the specified model must be available within the site's license server.

The Get-ConfigLicensingModel returns a list of supported values.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LicenseServerName<String>**

Changes the machine used by the brokering services to obtain licenses for desktop and application session brokering. The specified machine must be running a Citrix license server and have suitable licenses installed.

The license server machine can be specified by its DNS name ('machine.domain') or its numeric IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LicenseServerPort<Int32>**

Changes the port number on the license server machine used by the brokering services to contact the Citrix license server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LicenseServerUri<Uri>**

Changes the Uri of the web server for licensing. The hostname component of this Uri must match the Site's LicenseServerName property.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

Site

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Set-ConfigSite -ProductEdition PLT
```

Specifies the use of a Platinum edition license. A suitable license must be available on the site's license server.

# Set-ConfigSiteMetadata

May 10, 2016

Adds or updates metadata on the Site.

## Syntax

```
Set-ConfigSiteMetadata -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-ConfigSiteMetadata -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability for additional custom data to be stored against the Site.

## Related topics

[Remove-ConfigSiteMetadata](#)

## Parameters

**-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the Site specified. The property cannot contain any of the following characters \/:;#.\*?=<>|[]()"

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object

"System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-ConfigSiteMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Set-ConfigSiteMetadata -Name property -Value value
```

Key	Value
-----	-------

---

-----

property

value

Add metadata with a name of 'property' and a value of 'value' to the Site.



# Test-ConfigDBConnection

May 10, 2016

Tests a database connection for the Configuration Service.

## Syntax

```
Test-ConfigDBConnection [-DBConnection] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Tests a connection to the database in which the Configuration Service can store its state. The service will attempt to connect to the database without affecting the current connection to the database.

You do not have to clear the connection to use this command.

## Related topics

[Get-ConfigServiceStatus](#)

[Get-ConfigDBConnection](#)

[Set-ConfigDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be tested by the Configuration Service.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Test-ConfigDBConnection command returns an object containing the status of the Configuration Service if the connection string of the specified data store were to be set to the string being tested, together with extra diagnostics information for the specified connection string.

DBRejectedConnection

The database rejected the logon attempt from the Configuration Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the Configuration Service schema has not been added to the database.

DBNotFound

The specified database could not be located with the configured connection string.

DBNewerVersionThanService

The version of the Configuration Service currently in use is incompatible with the version of the Configuration Service schema on the database. Upgrade the Configuration Service to a more recent version.

DBOlderVersionThanService

The version of the Configuration Service schema on the database is incompatible with the version of the Configuration Service currently in use. Upgrade the database schema to a more recent version.

DBVersionChangeInProgress

A database schema upgrade is currently in progress.

OK

The Set-ConfigDBConnection command would succeed if it were executed with the supplied connection string.

Failed

The Configuration Service has failed.

Unknown

The status of the Configuration Service cannot be determined.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

InvalidDBConnectionString

The database connection string has an invalid format.

PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Test-ConfigDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Tests a database connection string for the Configuration Service.

##### ----- EXAMPLE 2 -----

```
c:\PS>Test-ConfigDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Tests an invalid database connection string for the Configuration Service.

# Test-ConfigServiceInstanceAvailability

May 10, 2016

Tests whether the supplied service instances are responding to requests.

## Syntax

```
Test-ConfigServiceInstanceAvailability [-ServiceInstance] <ServiceInstance[]> [-MaxDelaySeconds <Int32>] [-ForceWaitForOneOfEachType] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

## Related topics

[Register-ConfigServiceInstance](#)

[Unregister-ConfigRegisteredServiceInstance](#)

[Get-ConfigRegisteredServiceInstance](#)

## Parameters

**-ServiceInstance**<ServiceInstance[]>

The service instances to test.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-MaxDelaySeconds**<Int32>

The timeout period to wait before concluding that services are unresponsive.

Required?	false
Default Value	Infinite
Accept Pipeline Input?	false

**-ForceWaitForOneOfEachType**<SwitchParameter>

If at least one of each type of service responds, finish immediately.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the host name or IP address of the controller to which the PowerShell snap-in connects.

Required?	false
Default Value	'LocalHost'. Once a value is specified by any command, this value becomes the new default.
Accept Pipeline Input?	false

#### Return Values

System.Management.Automation.PSObject

This represents a service instance and has the following parameters;

ServiceGroupUid <Guid>

The unique identifier for the service group to which the service instance belongs.

ServiceGroupName <string>

The name of the service group that the service instance is part of.

ServiceInstanceUid <Guid>

The unique identifier for the service instance.

ServiceType <string>

The type of the service group.

Address <string>

The contact address for the service instance.

Binding <string>

The binding to use for connections to the service instance.

Version <int>

The version of the service instance.

ServiceAccount <string>

The AD computer account for the computer that is providing the service instance.

ServiceAccountSid <string>

The AD computer account SID for the computer that is providing the service instance.

InterfaceType <string>

The interface type for the service instance.

Metadata <Citrix.Configuration.Sdk.Metadata[]>

The metadata for the service instance.

Status <Citrix.Configuration.Sdk.Commands.Availability>

An enumeration value indicating whether the service is Responding, NotResponding, Unknown, or BadBindingType.

ResponseTime <System.TineSpan>

The interval elapsed between hailing the service and getting a definite response

## Notes

The Availability Status Codes are o Responding: Got a positive response o NotResponding: Got a response, but it was negative or the connection was refused o Unknown: Did not respond in time / timed-out o BadBindingType: Binding parameter in ServiceInstance is not wcf\_HTTP\_kerb

## Examples

### ----- EXAMPLE 1 -----

C:\>Get-ConfigRegisteredServiceInstance | Test-ConfigServiceInstanceAvailability -ForceWaitForOneOfEachType  
Test all the service instances that are registered in the Configuration Service, returning when one of each type is responding.

### ----- EXAMPLE 2 -----

C:\>Test-ConfigServiceInstanceAvailability -ServiceInstance \$services -MaxDelaySeconds 5  
Test each of the given services, allowing a 5 second time-out.

# Unregister-ConfigRegisteredServiceInstance

May 10, 2016

Removes a service instance from the Configuration Service registry.

## Syntax

```
Unregister-ConfigRegisteredServiceInstance [-ServiceInstanceId <Guid>] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Use this cmdlet to remove a service instance from the Configuration Service registry. This does not remove any service groups (if all service instances for a Service Group are removed, an empty service group remains and must be removed using the Remove-ConfigServiceGroup command).

## Related topics

[Register-ConfigServiceInstance](#)

## Parameters

**-ServiceInstanceId**<Guid>

The unique identifier for the service instance to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the logging id of the high-level operation this cmdlet invocation is part of.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Configuration.Sdk.ServiceInstance An object with a parameter called 'ServiceInstanceUid' can be used to unregister service instances.

## Notes

In the case of failure, the following errors can result.

Error Codes ----- ServiceInstanceObjectNotFound The service instance specified could not be located. DatabaseError An error occurred in the service while attempting a database operation. DatabaseNotConfigured The operation could not be completed because the database for the service is not configured. DataStoreException An error occurred in the service while attempting a database operation - communication with the database failed for various reasons. CommunicationError An error occurred while communicating with the service. ExceptionThrown An unexpected error occurred. For more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

C:\PS>Get-ConfigRegisteredServiceInstance -ServiceType "Config" | Unregister-ConfigRegisteredServiceInstance  
Unregisters all the service instances that are for a service type of 'Config' from the Configuration Service instance register.



# Citrix.ConfigurationLogging.Admin.V1

May 10, 2016

## Overview

Name	Description
<a href="#">LogConfigurationLoggingSnapin</a>	The Configuration Logging Service PowerShell snap-in provides administrative
<a href="#">Log Filtering</a>	Describes the common filtering options for XenDesktop cmdlets.
<a href="#">logical operators</a>	Describes the operators that connect statements in Windows PowerShell.

## Cmdlets

Name	Description
<a href="#">Export-LogReportCsv</a>	Exports Configuration Logging data into a CSV file.
<a href="#">Export-LogReportHtml</a>	Exports Configuration Logging data into a HTML report.
<a href="#">Get-LogDataStore</a>	Gets details for each of the ConfigurationLogging data stores.
<a href="#">Get-LogDBConnection</a>	Gets the database string for the specified data store used by the ConfigurationLogging Service.
<a href="#">Get-LogDBSchema</a>	Gets a script that creates the ConfigurationLogging Service database schema for the specified data store.
<a href="#">Get-LogDBVersionChangeScript</a>	Gets a script that updates the ConfigurationLogging Service database schema.
<a href="#">Get-LogHighLevelOperation</a>	Gets high level operations
<a href="#">Get-LogInstalledDBVersion</a>	Gets a list of all available database schema versions for the ConfigurationLogging Service.
<a href="#">Get-LogLowLevelOperation</a>	Gets low level operations
<a href="#">Get-LogService</a>	Gets the service record entries for the ConfigurationLogging Service.
<a href="#">Get-LogServiceAddedCapability</a>	Gets any added capabilities for the ConfigurationLogging Service on the controller.

Name	Description
<a href="#">Get-LogServiceInstance</a>	Gets the service instance entries for the ConfigurationLogging Service.
<a href="#">Get-LogServiceStatus</a>	Gets the current status of the ConfigurationLogging Service on the controller.
<a href="#">Get-LogSite</a>	Gets global configuration logging settings.
<a href="#">Get-LogSummary</a>	Gets operations logged within time intervals inside a date range.
<a href="#">Remove-LogOperation</a>	Deletes configuration logs
<a href="#">Remove-LogServiceMetadata</a>	Removes metadata from the given Service.
<a href="#">Remove-LogSiteMetadata</a>	Removes metadata from the given Site.
<a href="#">Reset-LogDataStore</a>	Refreshes the database string currently being used by the Log service.
<a href="#">Reset-LogServiceGroupMembership</a>	Reloads the access permissions and configuration service locations for the ConfigurationLogging Service.
<a href="#">Set-LogDBConnection</a>	Configures a database connection for the ConfigurationLogging Service.
<a href="#">Set-LogServiceMetadata</a>	Adds or updates metadata on the given Service.
<a href="#">Set-LogSite</a>	Sets global configuration logging settings.
<a href="#">Set-LogSiteMetadata</a>	Adds or updates metadata on the given Site.
<a href="#">Start-LogHighLevelOperation</a>	Logs the start of a high level operation.
<a href="#">Stop-LogHighLevelOperation</a>	Logs the completion of a previously started high level operation.
<a href="#">Test-LogDBConnection</a>	Tests a database connection for the ConfigurationLogging Service.

# about\_LogConfigurationLoggingSnapin

May 10, 2016

## TOPIC

about\_LogConfigurationLoggingSnapin

## SHORT DESCRIPTION

The Configuration Logging Service PowerShell snap-in provides administrative functions for the Configuration Logging Service.

## COMMAND PREFIX

All commands in this snap-in have the noun prefixed with 'Log'.

## LONG DESCRIPTION

The Configuration Logging Service PowerShell snap-in enables both local and remote administration of the Configuration Logging Service.

The Configuration Logging Service logs configuration changes or administrator requested state changes made to the site. Configuration Logging can be configured, site wide, to be mandatory or optional. If mandatory logging is selected, then any attempts to change site configuration or state when the logging mechanism is unavailable are denied.

The Configuration Logging Service stores information about the logged changes in a database which can be configured to be separate from the site database.

The snap-in provides storage and configuration of these entities:

### Site

The Configuration Logging Site object holds global settings which control the behaviour of the Configuration Logging Service. The site object can be configured by the Set-LogSite cmdlet. The properties of the site object are returned by the Get-LogSite cmdlet.

## High Level Operations

A high level operation object represents a logged configuration change performed from Desktop Studio, Desktop Director or a PowerShell Script.

The XenDesktop consoles log high level operations when:

- 1) Executing operations which performs configuration changes.
- 2) Executing operations which performs administration related activities which may affect site configuration.

PowerShell scripts which carry out customized configuration changes can also log high level operations via cmdlets `Start-LogHighLevelOperation` and `Stop-LogHighLevelOperation`.

## Low Level Operations

A low level operation object represents a logged configuration change performed by a service. One or more low level operation objects are used to record the actions performed by a services in order to fulfil a high level operation initiated from the consoles, or from PowerShell scripts.

Low level operations in the system are returned by cmdlet `Get-LogLowLevelOperation`.

## Operation Details

A low level operation performed by a service can affect a number of individual objects, or a number of properties on an object. An operation detail log records each individual change to an object. This includes the creation and deletion of the object, as well as changes to individual properties of the object.

One or more operation detail objects are used to record specific changes to each object that is affected by a low level service operation.

Operation details are included in the data returned from the `Get-LogLowLevelOperation` cmdlet.

High Level Operations, Low Level Operations and Operation Details are arranged in a hierarchy. A High Level Operation can have multiple Low Level Operations, and each Low Level Operation can have multiple Operation Details.

Setting up a separate logging database ----- After creating the database on the database server, the logging database can be setup and configured for use by:

- 1) Generating the database schema, and applying it the logging database
- 2) Configuring the configuration logging service to use the new logging database.

The logging database schema can be generated from the `Get-LogDBSchem` cmdlet, as illustrated below:

```
Get-LogDBSchema -DatabaseName "loggingDB"
-ServiceGroupName "service group name"
-ScriptType Database
-LocalDatabase:$LocalDB
-DataStore Logging
```

The configuration logging service can be configured to use the logging database with the Set-LogDBConnection cmdlet, as illustrated below:

```
Set-LogDBConnection -DataStore Logging -DBConnection $null
Set-LogDBConnection -DataStore Logging -DBConnection "new logging db connection string"
```

Configuration Logging site settings -----

On the Site object:

- o The 'State' setting allows configuration logging to be disabled, enabled, or made mandatory.
- o The 'Locale' setting specifies the language in which configuration logging data text will be stored.

See Get-LogSite cmdlet help for further information on these settings.

This locale setting applies to the text description that is associated with each log, e.g. **Create Catalog**. It doesn't apply to other textual information in the log like the names of parameters passed to operations, e.g. **CatalogName**.

This localisation is applied when the data is logged, and not when the logs are viewed later. For example, logs which are created in English will be displayed in English on an end user system which may be configured with a different locale.

# about\_Log\_Filtering

May 10, 2016

## TOPIC

XenDesktop - Advanced Dataset Filtering

## SHORT DESCRIPTION

Describes the common filtering options for XenDesktop cmdlets.

## LONG DESCRIPTION

Some cmdlets operate on large quantities of data and, to reduce the overhead of sending all of that data over the network, many of the Get- cmdlets support server-side filtering of the results.

The conventional way of filtering results in PowerShell is to pipeline them into Where-Object, Select-Object, and Sort-Object, for example:

```
Get-<Noun> | Where { $_.Size = 'Small' } | Sort 'Date' | Select -First 10
```

However, for most XenDesktop cmdlets the data is stored remotely and it would be slow and inefficient to retrieve large amounts of data over the network and then discard most of it. Instead, many of the Get- cmdlets provide filtering parameters that allow results to be processed on the server, returning only the required results.

You can filter results by most object properties using parameters derived from the property name. You can also sort results or limit them to a specified number of records:

```
Get-<Noun> -Size 'Small' -SortBy 'Date' -MaxRecordCount 10
```

You can express more complex filter conditions using a syntax and set of operators very similar to those used by PowerShell expressions.

Those cmdlets that support filtering have the following common parameters:

-MaxRecordCount <int>

Specifies the maximum number of results to return.  
For example, to return only the first nine results use:

```
Get-<Noun> -MaxRecordCount 9
```

If not specified, only the first 250 records are returned, and if more are available, a warning is produced:

WARNING: Only first 250 records returned. Use -MaxRecordCount to retrieve more.

You can suppress this warning by using -WarningAction or by specifying a value for -MaxRecordCount.

To retrieve all records, specify a large number for -MaxRecordCount. As the value is an integer, you can use the following:

```
Get-<Noun> -MaxRecordCount [int]::MaxValue
```

-ReturnTotalRecordCount [<SwitchParameter>]

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. For example:

```
Get-<Noun> -MaxRecordCount 9 -ReturnTotalRecordCount
....
```

```
Get-<Noun> : Returned 9 of 10 items
At line:1 char:18
+ Get-<Noun> <<<< -MaxRecordCount 9 -ReturnTotalRecordCount
+ CategoryInfo : OperationStopped: (:) [Get-<Noun>], PartialDataException
+ FullyQualifiedErrorId : PartialData,Citrix.<SDKName>.SDK.Get<Noun>
```

The count can be accessed using the TotalAvailableResultCount property:

```
$count = $error[0].TotalAvailableResultCount
```

-Skip <int>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

-SortBy <string>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order, respectively. Ascending order is assumed if no prefix is present.

Sorting occurs before -MaxRecordCount and -Skip parameters are applied. For example, to sort by Name and then by Count (largest first) use:

```
-SortBy 'Name,-Count'
```

By default, sorting by an enumeration property uses the numeric value of the elements. You can specify a different sort order by qualifying the name with an ordered list of elements or their numeric values, or <null> to indicate the placement of null values.

Elements not mentioned are placed at the end in their numeric order. For example, to sort by two different enums and then by the object id:

```
-SortBy 'MyState(StateC,<null>,StateA,StateB),Another(0,3,2,1),Id'
```

-Filter <String>

This parameter lets you specify advanced filter expressions, and supports combination of conditions with -and and -or, and grouping with braces. For example:

```
Get-<Noun> -Filter 'Name -like "High*" -or (Priority -eq 1 -and Severity -ge 2)'
```

The syntax is close enough to PowerShell syntax that you can use script blocks in most cases. This can be easier to read as it reduces quoting:

```
Get-<Noun> -Filter { Count -ne $null }
```

The full -Filter syntax is provided below.

## EXAMPLES

Filtering by strings performs a case-insensitive wildcard match.

Separate parameters are combined with an implicit -and operator.

Normal PowerShell quoting rules apply, so you can use single or double quotes, and omit the quotes altogether for many strings. The order of parameters does not make any difference. The following are equivalent:

```
Get-<Noun> -Company Citrix -Product Xen*
```

```
Get-<Noun> -Company "citrix" -Product '[X]EN*'
```

```
Get-<Noun> -Product "Xen*" -Company "CITRIX"
```

```
Get-<Noun> -Filter { Company -eq 'Citrix' -and Product -like 'Xen*' }
```



See [about\\_Quoting\\_Rules](#) and [about\\_Wildcards](#) for details about PowerShell handling of quotes and wildcards.

To avoid wildcard matching or include quote characters, you can escape the wildcards using the normal PowerShell escape mechanisms (see [about\\_Escape\\_Characters](#)), or switch to a filter expression and the `-eq` operator:

```
Get-<Noun> -Company "Abc[*]" # Matches Abc*
Get-<Noun> -Company "Abc`*" # Matches Abc*
Get-<Noun> -Filter { Company -eq "Abc*" } # Matches Abc*
Get-<Noun> -Filter { Company -eq "A`"B`C`" } # Matches A"B'C
```

Simple filtering by numbers, booleans, and TimeSpans perform direct equality comparisons, although if the value is nullable you can also search for null values. Here are some examples:

```
Get-<Noun> -Uid 123
Get-<Noun> -Enabled $true
Get-<Noun> -Duration 1:30:40
Get-<Noun> -NullableProperty $null
```

More comparisons are possible using advanced filtering with `-Filter`:

```
Get-<Noun> -Filter 'Capacity -ge 10gb'
Get-<Noun> -Filter 'Age -ge 20 -and Age -lt 40'
Get-<Noun> -Filter 'VolumeLevel -like "[123]"'
Get-<Noun> -Filter 'Enabled -ne $false'
Get-<Noun> -Filter 'NullableProperty -ne $null'
```

You can check boolean values without an explicit comparison operator, and you can also combine them with `-not`:

```
Get-<Noun> -Filter 'Enabled' # Equivalent to 'Enabled -eq $true'
Get-<Noun> -Filter '-not Enabled' # Equivalent to 'Enabled -eq $false'
```

See [about\\_Comparison\\_Operators](#) for an explanation of the operators, but note that only a subset of PowerShell operators are supported (`-eq`, `-ne`, `-gt`, `-ge`, `-lt`, `-le`, `-like`, `-notlike`, `-in`, `-notin`, `-contains`, `-notcontains`).

Enumeration values can either be specified using typed values or the string name of the enumeration value:

```
Get-<Noun> -Shape [Shapes]::Square
Get-<Noun> -Shape Circle
```

With filter expressions, typed values can be specified with simple variables or quoted strings. They also support enumerations with wildcards:

```
$s = [Shapes]::Square
Get-<Noun> -Filter { Shape -eq $s -or Shape -eq "Circle" }
Get-<Noun> -Filter { Shape -like 'C*' }
```

By their nature, floating point values, DateTime values, and TimeSpan values are best suited to relative comparisons rather than just equality. DateTime strings are converted using the locale and time zone of the user device, but you can use ISO8601 format strings (YYYY-MM-DDThh:mm:ss.sTZD) to avoid ambiguity. You can also use standard PowerShell syntax to create these values:

```
Get-<Noun> -Filter { StartTime -ge "2010-08-23T12:30:00.0Z" }
$d = [DateTime]"2010-08-23T12:30:00.0Z"
Get-<Noun> -Filter { StartTime -ge $d }
$d = (Get-Date).AddDays(-1)
Get-<Noun> -Filter { StartTime -ge $d }
```

Relative times are quite common and, when using filter expressions, you can also specify DateTime values using a relative format:

```
Get-<Noun> -Filter { StartTime -ge '-2' } # Two days ago
Get-<Noun> -Filter { StartTime -ge '-1:30' } # Hour and a half ago
Get-<Noun> -Filter { StartTime -ge '-0:0:30' } # 30 seconds ago
```

## ARRAY PROPERTIES

When filtering against list or array properties, simple parameters perform a case-insensitive wildcard match against each of the members. With filter expressions, you can use the -contains and -notcontains operators. Unlike PowerShell, these perform wildcard matching on strings.

Note that for array properties the naming convention is for the returned property to be plural, but the parameter used to search for any match is singular. The following are equivalent (assuming Users is an array property):

```
Get-<Noun> -User Fred*
Get-<Noun> -Filter { User -like "Fred*" }
Get-<Noun> -Filter { Users -contains "Fred*" }
```

You can also use the singular form with -Filter to search using other operators:

```
Match if any user in the list is called "Frederick"
Get-<Noun> -Filter { User -eq "Frederick" }
Match if any user in the list has a name alphabetically below 'F'
Get-<Noun> -Filter { User -lt 'F' }
```

## COMPLEX EXPRESSIONS

When matching against multiple values, you can use a sequence of comparisons joined with -or operators, or you can use -in and -notin:

```
Get-<Noun> -Filter { Shape -eq 'Circle' -or Shape -eq 'Square' }
$shapes = 'Circle','Square'
Get-<Noun> -Filter { Shape -in $shapes }
$sides = 1..4
Get-<Noun> -Filter { Sides -notin $sides }
```

Braces can be used to group complex expressions, and override the default left-to-right evaluation of -and and -or. You can also use -not to invert the sense of any sub-expression:

```
Get-<Noun> -Filter { Size -gt 4 -or (Color -eq 'Blue' -and Shape -eq 'Circle') }
Get-<Noun> -Filter { Sides -lt 5 -and -not (Color -eq 'Blue' -and Shape -eq 'Circle') }
```

## PAGING

The simplest way to page through data is to use the -Skip and -MaxRecordCount parameters. So, to read the first three pages of data with 10 records per page, use:

```
Get-<Noun> -Skip 0 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 10 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 20 -MaxRecordCount 10 <other filtering criteria>
```

You must include the same filtering criteria on each call, and ensure that the data is sorted consistently.

The above approach is often acceptable, but as each call performs an independent query, data changes can result in records being skipped or appearing twice. One approach to improve this is to sort by a unique id field and then start the search for the next page at the unique id after the last unique id of the previous page. For example:

```
Get the first page
Get-<Noun> -MaxRecordCount 10 -SortBy SerialNumber

SerialNumber ...

A120004
A120007
... 7 other records ...
A120900

Get the next page
Get-<Noun> -MaxRecordCount 10 -Filter { FirstName -gt 'A120900' }

SerialNumber ...
```

----- ---  
A120901  
B220000  
...

## FILTER SYNTAX DEFINITION

<Filter> ::= <ScriptBlock> | <ComponentList>

<ScriptBlock> ::= "{" <ComponentList> "}"

<ComponentList> ::= <Component> <AndOrOperator> <ComponentList> |

<Component>

<Component> ::= <NotOperator> <Factor> |

<Factor>

<Factor> ::= "(" <ComponentList> ")" |

<PropertyName> <ComparisonOperator> <Value> |  
<PropertyName>

<AndOrOperator> ::= "-and" | "-or"

<NotOperator> ::= "-not" | "!"

<ComparisonOperator>

::= "-eq" | "-ne" | "-le" | "-ge" | "-lt" | "-gt" |  
"-like" | "-notlike" | "-contains" | "-notcontains" |  
"-in" | "-notin"

<PropertyName> ::= <simple name of property>

<Value> ::= <string literal> | <numeric literal> |

<scalar variable> | <array variable> |  
"\$null" | "\$true" | "\$false"

Numeric literals support decimal and hexadecimal literals, with optional multiplier suffixes (kb, mb, gb, tb, pb).

Dates and times can be specified as string literals. The current culture determines what formats are accepted. To avoid any ambiguity, use strings formatted to the ISO8601 standard. If not specified, the current time zone is used.

Relative date-time string literals are also supported, using a minus sign followed by a TimeSpan. For example, "-1:30" means 1 hour and 30 minutes ago.

# about\_logical\_operators

May 10, 2016

## TOPIC

about\_Logical\_Operators

## SHORT DESCRIPTION

Describes the operators that connect statements in Windows PowerShell.

## LONG DESCRIPTION

The Windows PowerShell logical operators connect expressions and statements, allowing you to use a single expression to test for multiple conditions.

For example, the following statement uses the and operator and the or operator to connect three conditional statements. The statement is true only when the value of \$a is greater than the value of \$b, and either \$a or \$b is less than 20.

```
($a -gt $b) -and (($a -lt 20) -or ($b -lt 20))
```

Windows PowerShell supports the following logical operators.

Operator	Description	Example
-and	Logical and. TRUE only when both statements are TRUE.	(1 -eq 1) -and (1 -eq 2) False
-or	Logical or. TRUE when either or both statements are TRUE.	(1 -eq 1) -or (1 -eq 2) True
-xor	Logical exclusive or. TRUE only when one of the statements is TRUE and the other is FALSE.	(1 -eq 1) -xor (2 -eq 2) False
-not	Logical not. Negates the statement that follows it.	-not (1 -eq 1) False
!	Logical not. Negates the statement that follows it. (Same as -not)	!(1 -eq 1) False

Note: The previous examples also use the equal to comparison

operator (-eq). For more information, see [about\\_Comparison\\_Operators](#).  
The examples also use the Boolean values of integers. The integer 0 has a value of FALSE. All other integers have a value of TRUE.

The syntax of the logical operators is as follows:

```
<statement> {-AND | -OR | -XOR} <statement>
{! | -NOT} <statement>
```

Statements that use the logical operators return Boolean (TRUE or FALSE) values.

The Windows PowerShell logical operators evaluate only the statements required to determine the truth value of the statement. If the left operand in a statement that contains the and operator is FALSE, the right operand is not evaluated. If the left operand in a statement that contains the or statement is TRUE, the right operand is not evaluated. As a result, you can use these statements in the same way that you would use the If statement.

#### SEE ALSO

[about\\_Operators](#)

[Compare-Object](#)

[about\\_Comparison\\_operators](#)

[about\\_If](#)

# Export-LogReportCsv

May 10, 2016

Exports Configuration Logging data into a CSV file.

## Syntax

```
Export-LogReportCsv -OutputFile <String> [-StartDateRange <DateTime>] [-EndDateRange <DateTime>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

This cmdlet exports the Configuration Logging data into a CSV data file. The hierarchical logging data is flattened into a single CSV 'table'. The content of CSV file is not intended to be human-readable. It is meant to be input data for external reporting or manipulation tools (for example, a spread sheet application).

## Related topics

[Export-LogReportHtml](#)

## Parameters

### **-OutputFile**<String>

Specifies the path to a file where the CSV data will be saved.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-StartDateRange**<DateTime>

Specifies the start time of the earliest operation to include.

Required?	false
Default Value	DateTime.Min
Accept Pipeline Input?	false

### **-EndDateRange**<DateTime>

Specifies the end time of the latest operation to include.

Required?	false
Default Value	DateTime.UtcNow
Accept Pipeline Input?	false

### **-AdminAddress**<String>



Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Export-LogReportCsv -OutputFile "c:\MyReports\LoggingData.csv"
```

Export all logged operations to a csv file.

### ----- EXAMPLE 2 -----

```
C:\PS> Export-LogReportCsv -OutputFile "c:\MyReports\LoggingData.csv" -StartDateRange "2012-12-21 09:00"
```

Export to a CSV file logged operations started on or after a specified datetime..

### ----- EXAMPLE 3 -----

```
C:\PS> Export-LogReportCsv -OutputFile "c:\MyReports\LoggingData.csv" -StartDateRange "2012-12-21 09:00" -EndDateRange "2012-12-31 18:00"
```

Export to a CSV file logged operations started and completed between a date range.

# Export-LogReportHtml

May 10, 2016

Exports Configuration Logging data into a HTML report.

## Syntax

```
Export-LogReportHtml -OutputDirectory <String> [-StartDateRange <DateTime>] [-EndDateRange <DateTime>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

This cmdlet exports the Configuration Logging data into a HTML report. The report consists of two HTML files:

- o Summary.Html - this shows summary information from the high level operation logs.
- o Details.Html - this shows additional logging data from the low level operation and operation detail logs.

Hyperlinks in summary.html allow drill-down into the associated low level logging data contained within details.html.

## Related topics

[Export-LogReportCsv](#)

## Parameters

**-OutputDirectory**<String>

Specifies the path to a directory where the HTML report files will be saved.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-StartDateRange**<DateTime>

Specifies the start time of the earliest operation to include.

Required?	false
Default Value	DateTime.Min
Accept Pipeline Input?	false

**-EndDateRange**<DateTime>

Specifies the end time of the latest operation to include.

Required?	false
-----------	-------

Default Value	DateTime.UtcNow
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Export-LogReportHtml -OutputDirectory "c:\MyReports"
```

Export all logged operations to HTML.

##### ----- **EXAMPLE 2** -----

```
C:\PS> Export-LogReportHtml -OutputDirectory "c:\MyReports" -StartDateRange "2012-12-21 09:00"
```

Export to a HTML logged operations started on or after a specified datetime..

##### ----- **EXAMPLE 3** -----

```
C:\PS> Export-LogReportHtml -OutputDirectory "c:\MyReports" -StartDateRange "2012-12-21 09:00" -EndDateRange "2012-12-31 18:00"
```

Export to HTML logged operations started and completed between a date range.

# Get-LogDataStore

May 10, 2016

Gets details for each of the ConfigurationLogging data stores.

## Syntax

```
Get-LogDataStore [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns an object for each of the ConfigurationLogging data stores describing the connection string, data store name, db type, provider, schema name, and DB status.

A database connection must be configured in order for this command to be used if the service has a secondary data store. This is not required for the site data store.

## Related topics

[Reset-LogDataStore](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.ConfigurationLogging.Sdk.DataStoreConfiguration

An object describing the connection string, data store name, database type, provider, schema name and database status.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-LogDataStore
```

```
ConnectionString : Server=.\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True
DataStore : Site
DatabaseType : SqlServer
Provider : MSSQL
SchemaName : LogSiteSchema
Status : OK
```

```
ConnectionString :
DataStore : Secondary
DatabaseType : SqlServer
Provider : MSSQL
SchemaName : LogSecondarySchema
Status : DBUnconfigured
Get the database connection string for the ConfigurationLogging Service.
```

# Get-LogDBConnection

May 10, 2016

Gets the database string for the specified data store used by the ConfigurationLogging Service.

## Syntax

```
Get-LogDBConnection [[-DataStore] <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the database connection string for the specified data store.

If the returned string is blank, no valid connection string has been specified. In this case the service is running, but is idle and awaiting specification of a valid connection string.

## Related topics

[Get-LogServiceStatus](#)

[Get-LogDataStore](#)

[Set-LogDBConnection](#)

[Test-LogDBConnection](#)

## Parameters

**-DataStore**<String>

Specifies the logical name of the data store for the ConfigurationLogging Service. Can be either be 'Site' or the logical name of the secondary data store.

Required?	false
Default Value	Site
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

system.string

The database connection string configured for the ConfigurationLogging Service.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### NoDBConnections

The database connection string for the ConfigurationLogging Service has not been specified.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-LogDBConnection
```

Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True  
Get the database connection string for the ConfigurationLogging Service.



# Get-LogDBSchema

May 10, 2016

Gets a script that creates the ConfigurationLogging Service database schema for the specified data store.

## Syntax

```
Get-LogDBSchema [-DatabaseName <String>] [-ServiceGroupName <String>] [-ScriptType <ScriptTypes>] [-LocalDatabase] [-Sid <String>] [-DataStore <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets SQL scripts that can be used to create a new ConfigurationLogging Service database schema, add a new ConfigurationLogging Service to an existing site, remove a ConfigurationLogging Service from a site, or create a database server logon for a ConfigurationLogging Service. If no Sid parameter is provided, the scripts obtained relate to the currently selected ConfigurationLogging Service instance, otherwise the scripts relate to ConfigurationLogging Service instance running on the machine identified by the Sid provided. When obtaining the Evict script, a Sid parameter must be supplied. The current service instance is that on the local machine, or that explicitly specified by the last usage of the -AdminAddress parameter to a ConfigurationLogging SDK cmdlet. The service instance used to obtain the scripts does not need to be a member of a site or to have had its database connection configured. The database scripts support only Microsoft SQL Server, or SQL Server Express, and require Windows integrated authentication to be used. They can be run using SQL Server's SQLCMD utility, or by copying the script into an SQL Server Management Studio (SSMS) query window and executing the query. If using SSMS, the query must be executed in 'SMDCMD mode'. The ScriptType parameter determines which script is obtained. If ScriptType is not specified, or is FullDatabase, the script contains:

- o Creation of service schema
- o Creation of database server logon
- o Creation of database user
- o Addition of database user to ConfigurationLogging Service roles

If ScriptType is Instance, the returned script contains:

- o Creation of database server logon
- o Creation of database user
- o Addition of database user to ConfigurationLogging Service roles

If ScriptType is Evict, the returned script contains:

- o Removal of ConfigurationLogging Service instance from database
- o Removal of database user

If ScriptType is Login, the returned script contains:

- o Creation of database server logon only

If the service uses two data stores they can exist in the same database. You do not need to configure a database before using this command.

## Related topics

[Get-LogDataStore](#)

[Set-LogDBConnection](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database for which the schema will be generated.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ServiceGroupName**<String>

Specifies the name of the service group to be used when creating the database schema. The service group is a collection of all the ConfigurationLogging services that share the same database instance and are considered equivalent; that is, all the services within a service group can be used interchangeably.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ScriptType**<ScriptTypes>

Specifies the type of database script returned. Available script types are:

### Database

Returns a full database script that can be used to create a database schema for the ConfigurationLogging Service in a database instance that does not already contain a schema for this service. The DatabaseName and ServiceGroupName parameters must be specified to create a script of this type.

### Instance

Returns a permissions script that can be used to add further ConfigurationLogging services to an existing database instance that already contains the full ConfigurationLogging service schema, associating the services to the Service Group. The Sid parameter can optionally be specified to create a script of this type.

### Login

Returns a database logon script that can be used to add the required logon accounts to an existing database instance that contains the ConfigurationLogging Service schema. This is used primarily when creating a mirrored database environment. The DatabaseName parameter must be specified to create a script of this type.

## Evict

Returns a script that can be used to remove the specified ConfigurationLogging Service from the database entirely. The DatabaseName and Sid parameters must be specified to create a script of this type.

Required?	false
Default Value	Database
Accept Pipeline Input?	false

## **-LocalDatabase**<SwitchParameter>

Specifies whether the database script is to be used in a database instance run on the same controller as other services in the service group. Including this parameter ensures the script creates only the required permissions for local services to access the database schema for ConfigurationLogging services.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-Sid**<String>

Specifies the SID of the controller on which the ConfigurationLogging Service instance to remove from the database is running.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

## **-DataStore**<String>

Specifies the logical name of the data store for the ConfigurationLogging Service. Can be either be 'Site' or the logical name of the secondary data store.

Required?	false
Default Value	Site
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.string

A string containing the required SQL script for application to a database.

## Notes

The scripts returned support Microsoft SQL Server Express Edition, Microsoft SQL Server Standard Edition, and Microsoft SQL Server Enterprise Edition databases only, and are generated on the assumption that integrated authentication will be used.

If the ScriptType parameter is not included or set to 'FullDatabase', the full database script is returned, which will:

Create the database schema.

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist).

If the ScriptType parameter is set to 'Instance', the script will:

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist) and associate it with a user.

If the ScriptType parameter is set to 'Login', the script will:

Create the logon (providing the schema does not already exist) and associate it with a pre-existing user of the same name.

If the LocalDatabase parameter is included, the NetworkService account will be added to the list of accounts permitted to access the database. This is required only if the database is run on a controller.

If the command fails, the following errors can be returned.

## Error Codes

-----

GetSchemasFailed

The database schema could not be found.

ActiveDirectoryAccountResolutionFailed

The specified Active Directory account or Group could not be found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Get-LogDBSchema -DatabaseName MyDB -ServiceGroupName MyServiceGroup > c:\LogSchema.sql
```

Get the full database schema for site data store of the ConfigurationLogging Service and copy it to a file called 'c:\LogSchema.sql'.

This script can then be used to create the schema in a pre-existing database named 'MyDB' that does not already contain a ConfigurationLogging Service site schema.

##### ----- EXAMPLE 2 -----

```
c:\PS>Get-LogDBSchema -DatabaseName MyDB -scriptType Login > c:\ConfigurationLoggingLogins.sql
```

Get the logon scripts for the ConfigurationLogging Service.

##### ----- EXAMPLE 3 -----

```
c:\PS>Get-LogDBSchema -DatabaseName MyDB -ServiceGroupName MyServiceGroup -DataStore Secondary > c:\LogSchema.sql
```

Get the full database schema for the secondary data store of the ConfigurationLogging Service and copy it to a file called 'c:\LogSecondarySchema.sql'.

This script can then be used to create the schema in a pre-existing database named 'MyDB' that does not already contain a ConfigurationLogging Service secondary schema.

# Get-LogDBVersionChangeScript

May 10, 2016

Gets a script that updates the ConfigurationLogging Service database schema.

## Syntax

```
Get-LogDBVersionChangeScript -DatabaseName <String> -TargetVersion <Version> [-DataStore <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns a database script that can be used to upgrade or downgrade the site or secondary schema for the ConfigurationLogging Service from the current schema version to a different version.

## Related topics

[Get-LogInstalledDBVersion](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database instance to which the update applies.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-TargetVersion**<Version>

Specifies the version of the database you want to update to.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-DataStore**<String>

Specifies the logical name of the data store for the ConfigurationLogging Service. Can be either be 'Site' or the logical name of the secondary data store.

--	--

Required?	false
Default Value	Site
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

System.Management.Automation.PSObject

A PSObject containing the required SQL script for application to a database.

### Notes

The PSObject returned by this cmdlet contains the following properties:

-- Script The raw text of the SQL script to apply the update, or null in the case when no upgrade path to the specified target version exists.

-- NeedExclusiveAccess Indicates whether all services in the service group must be shut down during the update or not.

-- CanUndo Indicates whether the generated script allows the updated schema to be reverted to the state prior to the update.

Scripts to update the schema version are stored in the database so any service in the service group can obtain these scripts. Extreme caution should be exercised when using update scripts. Citrix recommends backing up the database before attempting to upgrade the schema. Database update scripts may require exclusive use of the schema and so may not be able to execute while any ConfigurationLogging services are running. However, this depends on the specific update being carried out.

After a schema update has been carried out, services that require the previous version of the schema may cease to operate. The ServiceState parameter reported by the Get-LogServiceStatus command provides information about service compatibility. For example, if the schema has been upgraded to a more recent version that a service cannot use, the service reports "DBNewerVersionThanService".

If the command fails, the following errors can be returned.

## Error Codes

-----

### NoOp

The operation was successful but had no effect.

### NoDBConnections

The database connection string for the ConfigurationLogging Service has not been specified.

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

### PermissionDenied

You do not have permission to execute this command.

### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

### CommunicationError

There was a problem communicating with the remote service.

### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> $update = Get-LogDBVersionChangeScript -DatabaseName MyDb -TargetVersion 1.0.75.0
```

```
C:\PS> $update.Script > update_75.sql
```

Gets an SQL update script to update the current schema to version 1.0.75.0. The resulting update\_75.sql script is suitable for direct use with the SQL Server SQLCMD utility.



# Get-LogHighLevelOperation

May 10, 2016

Gets high level operations

## Syntax

```
Get-LogHighLevelOperation [-Id <Guid>] [-Text <String>] [-StartTime <DateTime>] [-Source <String>] [-EndTime <DateTime>]
[-IsSuccessful <Boolean>] [-User <String>] [-AdminMachineIP <String>] [-OperationType <OperationType>] [-TargetType
<String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-
Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieves high level operations matching the specified criteria. If no parameters are specified this cmdlet returns all high level operations.

## Related topics

[Start-LogHighLevelOperation](#)

[Stop-LogHighLevelOperation](#)

[Get-LogLowLevelOperation](#)

## Parameters

**-Id**<Guid>

Gets the high level operation with the specified identifier.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Text**<String>

Gets high level operations with the specified text

Required?	false
Default Value	
Accept Pipeline Input?	false

**-StartTime**<DateTime>

Gets high level operations with the specified start time

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Source**<String>

Gets high level operations with the specified source.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-EndTime**<DateTime>

Gets high level operations with the specified end time.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IsSuccessful**<Boolean>

Gets high level operations with the specified success indicator.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-User**<String>

Gets high level operations logged by the specified user.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminMachineIP**<String>

Gets high level operations logged from the machine with the specified IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-OperationType**<OperationType>

Gets high level operations with the specified operation type. Values can be:

- o AdminActivity - to get operations which log administration activity.
- o ConfigurationChange - to get operations which log configuration changes.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-TargetType**<String>

Gets high level operations with the specified target type. The target type describes the type of object that was the target of the configuration change. For example, "Session" or "Machine".

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ReturnTotalRecordCount<SwitchParameter>**

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Log\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

### **-MaxRecordCount<Int32>**

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

### **-Skip<Int32>**

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	

Accept Pipeline Input?	false
------------------------	-------

### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Log\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.ConfigurationLogging.Sdk.HighLevelOperation

The returned HighLevelOperation object has the following properties:

- o Id - The unique identifier of the operation.
- o Text - A description of the operation.

- o StartTime - The date and time that the operation started.
- o EndTime - The date and time that the operation completed. This will be null if the operation is still in progress, or if the operation never completed.
- o IsSuccessful - Indicates whether the operation completed successfully or not. This will be null if the operation is still in progress, or if the operation never completed.
- o User - The identifier of the administrator who performed the operation.
- o AdminMachineIP - The IP address of the machine that the operation was initiated from.
- o Source - Identifies the XenDesktop console, or custom script, where the operation was initiated from. For example, "Desktop Studio", "Desktop Director", "My custom script".
- o OperationType - The operation type. This can be 'AdminActivity' or 'ConfigurationChange'.
- o TargetTypes - Identifies the type of objects that were affected by the operation. For example, "Catalog" or "Desktop","Machine".
- o Parameters - The names and values of the parameters that were supplied for the operation.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-LogHighLevelOperation
Get all high level operations
```

### ----- EXAMPLE 2 -----

```
C:\PS> Get-LogHighLevelOperation -OperationType ConfigurationChange
Get high level operations which log configuration changes.
```

### ----- EXAMPLE 3 -----

```
C:\PS> Get-LogHighLevelOperation -OperationType AdminActivity
Get high level operations which log administration activities.
```

### ----- EXAMPLE 4 -----

```
C:\PS> Get-LogHighLevelOperation -Filter{ StartTime -ge "2013-02-27 09:00:00" -and EndTime -le "2013-02-27 18:00:00"}
Use advanced filtering to get high level operations with a start time on or after "2013-02-27 09:00:00" and an end time on or before "2013-02-27 18:00:00".
```

### ----- EXAMPLE 5 -----

```
C:\PS> Get-LogHighLevelOperation -EndTime $null
C:\PS> Get-LogHighLevelOperation -IsSuccessful $null
Either of these commands will get high level operations which have not yet been completed.
```

### ----- EXAMPLE 6 -----

```
C:\PS> Get-LogHighLevelOperation -User "DOMAIN\UserName"
```

Get high level operations performed by user "DOMAIN\UserName".

# Get-LogInstalledDBVersion

May 10, 2016

Gets a list of all available database schema versions for the ConfigurationLogging Service.

## Syntax

```
Get-LogInstalledDBVersion [-Upgrade] [-Downgrade] [-DataStore <String>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Returns the current version of the ConfigurationLogging Service database schema, if no flags are set, otherwise returns versions for which upgrade or downgrade scripts are available and have been stored in the database.

## Related topics

## Parameters

**-Upgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be updated should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Downgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be reverted should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DataStore**<String>

Specifies the database connection logical name the schema script should be returned for. The parameter is optional.

Required?	false
Default Value	



Accept Pipeline Input?	false
------------------------	-------

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

System.Version

The Get-LogInstalledDbVersion command returns objects containing the new definition of the ConfigurationLogging Service database schema version.

Major <Integer>

Minor <Integer>

Build <Integer>

Revision <Integer>

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

InvalidParameterCombination

Both the Upgrade and Downgrade flags were specified.

NoOp

The operation was successful but had no effect.

NoDBConnections

The database connection string for the ConfigurationLogging Service has not been specified.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-LogInstalledDBVersion
```

Major	Minor	Build	Revision
-------	-------	-------	----------

----	----	----	-----
------	------	------	-------

5	6	0	0
---	---	---	---

Get the currently installed version of the ConfigurationLogging Service database schema.

----- **EXAMPLE 2** -----

```
c:\PS>Get-LogInstalledDBVersion -Upgrade
```

Major	Minor	Build	Revision
-------	-------	-------	----------

----	----	----	-----
------	------	------	-------

6	0	0	0
---	---	---	---

Get the versions of the ConfigurationLogging Service database schema for which upgrade scripts are supplied.

# Get-LogLowLevelOperation

May 10, 2016

Gets low level operations

## Syntax

```
Get-LogLowLevelOperation [-Id <Guid>] [-HighLevelOperationId <Guid>] [-StartTime <DateTime>] [-EndTime <DateTime>] [-IsSuccessful <Boolean>] [-User <String>] [-AdminMachineIP <String>] [-Text <String>] [-Source <String>] [-SourceSdk <String>] [-OperationType <OperationType>] [-TargetType <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieves low level operations matching the specified criteria. If no parameters are specified this cmdlet returns all low level operations.

## Related topics

[Get-LogLowLevelOperation](#)

## Parameters

**-Id**<Guid>

Gets the low level operation with the specified identifier.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-HighLevelOperationId**<Guid>

Gets low level operations for the high level operation with the specified identifier.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-StartTime**<DateTime>

Gets low level operations with the specified start time

Required?	false
Default Value	
Accept Pipeline Input?	false

**-EndTime**<DateTime>

Gets low level operations with the specified end time.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-IsSuccessful**<Boolean>

Gets low level operations with the specified success indicator.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-User**<String>

Gets low level operations logged by the specified administrator.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminMachineIP**<String>

Gets low level operations logged from the machine with the specified IP address.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Text**<String>

Gets low level operations with the specified text

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Source**<String>

Gets low level operations with the specified source.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-SourceSdk**<String>

Gets low level operations logged from the SDK with the specified identifier.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-OperationType**<OperationType>

Gets low level operations with the specified operation type. Values can be:

o AdminActivity - to get operations which log administration activity.

o ConfigurationChange - to get operations which log configuration changes.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-TargetType**<String>

Gets low level operations with the specified target type. The target type describes the type of object that was the target of the configuration change. For example, "Session" or "Machine".

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Log\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Log\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.ConfigurationLogging.Sdk.LowLevelOperation

The returned LowLevelOperation object has the following properties:

- o Id - The unique identifier of the operation.
- o Text - A description of the operation.
- o HighLevelOperationId - The unique identifier of the related high level operation.
- o StartTime - The date and time that the operation started.
- o EndTime - The date and time that the operation completed. This will be null if the operation is still in progress, or if the operation never completed.
- o IsSuccessful - Indicates whether the operation completed successfully or not. This will be null if the operation is still in progress, or if the operation never completed.
- o AdminSid - The identifier of the administrator who performed the operation.
- o AdminMachineIP - The IP address of the machine that the operation was performed on.
- o Source - The name of the XenDesktop service that the operation was performed on; for example, "MachineCreation", "DelegatedAdmin".
- o SourceSdk - The identifier of the XenDesktop service SDK through which the operation was performed; for example, "Prov", "Admin".
- o OperationType - The operation type. This can be 'AdminActivity' or 'ConfigurationChange'.
- o TargetTypes - Identifies the type of objects that were affected by the operation. For example, "Catalog" or "Desktop","Machine".
- o Parameters - The names and values of the parameters that were supplied for the operation.
- o Details - A collection of OperationDetail objects containing specific information about each object affected by the operation.



Each OperationDetail object in the 'Details' collection has the following properties:

- o TargetUid - The unique identifier of the target object affected by the operation.
- o TargetName - The name of the target object affected by the operation.
- o TargetType - The type of the target object.
- o Text - The description of operation performed on the target object.
- o StartTime - The date and time that the operation started.
- o EndTime - The date and time that the operation completed. This will be null if the operation is still in progress, or if the operation never completed.
- o IsSuccessful - Indicates whether the operation completed successfully or not. This will be null if the operation is still in progress, or if the operation didn't complete.

The following properties will be set if the operation changed a property on the object:

- o PropertyName - The name of the changed property.
- o NewValue - The new property value.
- o PreviousValue - The previous property value.
- o AddValue - If the object property contains a set of values, this specifies the new value which was added to the set.
- o RemoveValue - If the object property contains a set of values, this specifies the value which was removed from the set.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-LogLowLevelOperation
Get all low level operations
```

### ----- EXAMPLE 2 -----

```
C:\PS> Get-LogLowLevelOperation -OperationType ConfigurationChange
Get low level operations which log configuration changes.
```

### ----- EXAMPLE 3 -----

```
C:\PS> Get-LogLowLevelOperation -OperationType AdminActivity
Get low level operations which log administration activities.
```

### ----- EXAMPLE 4 -----

```
C:\PS> Get-LogLowLevelOperation -Filter{ StartTime -ge "2013-02-27 09:00:00" -and EndTime -le "2013-02-27 18:00:00"}
Use advanced filtering to get low level operations with a start time on or after "2013-02-27 09:00:00" and an end time on or before "2013-02-27 18:00:00".
```

### ----- EXAMPLE 5 -----

```
C:\PS> Get-LogLowLevelOperation -EndTime $null
```

```
C:\PS> Get-LogLowLevelOperation -IsSuccessful $null
```

Either of these commands will get low level operations which have not yet been completed.

----- **EXAMPLE 6** -----

```
C:\PS> Get-LogLowLevelOperation -User "DOMAIN\UserName"
```

Get low level operations performed by user "DOMAIN\UserName".

# Get-LogService

May 10, 2016

Gets the service record entries for the ConfigurationLogging Service.

## Syntax

```
Get-LogService [-Metadata <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns instances of the ConfigurationLogging Service that the service publishes. The service records contain account security identifier information that can be used to remove each service from the database.

A database connection for the service is required to use this command.

## Related topics

### Parameters

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Log\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-Sort By**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Log\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.ConfigurationLogging.Sdk.Service

The Get-LogServiceInstance command returns an object containing the following properties.

Uid <Integer>

Specifies the unique identifier for the service in the group. The unique identifier is an index number.

ServiceHostId <Guid>

Specifies the unique identifier for the service instance.

DNSName <String>

Specifies the domain name of the host on which the service runs.

MachineName <String>

Specifies the short name of the host on which the service runs.

CurrentState <Citrix.Fma.Sdk.ServiceCore.ServiceState>

Specifies whether the service is running, started but inactive, stopped, or failed.

LastStartTime <DateTime>

Specifies the date and time at which the service was last restarted.

LastActivityTime <DateTime>

Specifies the date and time at which the service was last stopped or restarted.

OSType

Specifies the operating system installed on the host on which the service runs.

OSVersion

Specifies the version of the operating system installed on the host on which the service runs.

ServiceVersion

Specifies the version number of the service instance. The version number is a string that reflects the full build version of the service.

DatabaseUserName <string>

Specifies for the service instance the Active Directory account name with permissions to access the database. This will be either the machine account or, if the database is running on a controller, the NetworkService account.

Sid <string>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

ActiveSiteServices <string[]>

Specifies the names of active site services currently running in the service. Site services are components that perform long-running background processing in some services. This field is empty for services that do not contain site services.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

UnknownObject

One of the specified objects was not found.

PartialData

Only a subset of the available data was returned.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

CouldNotQueryDatabase

The query required to get the database was not defined.

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-LogService
```

```
Uid : 1
ServiceHostId : aef6f464-f1ee-4042-a523-66982e0cecd0
DNSName : MyServer.company.com
MachineName : MYSERVER
CurrentState : On
LastStartTime : 04/04/2011 15:25:38
LastActivityTime : 04/04/2011 15:33:39
OSType : Win32NT
OSVersion : 6.1.7600.0
ServiceVersion : 5.1.0.0
DatabaseUserName : NT AUTHORITY\NETWORK SERVICE
SID : S-1-5-21-2316621082-1546847349-2782505528-1165
ActiveSiteServices : {MySiteService1, MySiteService2...}
Get all the instances of the ConfigurationLogging Service running in the current service group.
```

# Get-LogServiceAddedCapability

May 10, 2016

Gets any added capabilities for the ConfigurationLogging Service on the controller.

## Syntax

```
Get-LogServiceAddedCapability [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables updates to the ConfigurationLogging Service on the controller to be detected.

You do not need to configure a database connection before using this command.

## Related topics

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.String

String containing added capabilities.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.



## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-LogServiceAddedCapability
```

Get the added capabilities of the ConfigurationLogging Service.

# Get-LogServiceInstance

May 10, 2016

Gets the service instance entries for the ConfigurationLogging Service.

## Syntax

```
Get-LogServiceInstance [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns service interfaces published by the instance of the ConfigurationLogging Service. Each instance of a service publishes multiple interfaces with distinct interface types, and each of these interfaces is represented as a ServiceInstance object. Service instances can be used to register the service with a central configuration service so that other services can use the functionality.

You do not need to configure a database connection to use this command.

## Related topics

### Parameters

**-AdminAddress** <String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.ConfigurationLogging.Sdk.ServiceInstance

The Get-LogServiceInstance command returns an object containing the following properties.

ServiceGroupUid <Guid>

Specifies the unique identifier for the service group of which the service is a member.

ServiceGroupName <String>

Specifies the name of the service group of which the service is a member.

ServiceInstanceUID <Guid>

Specifies the unique identifier for registered service instances, which are service instances held by and obtained from a

central configuration service. Unregistered service instances do not have unique identifiers.

ServiceType <String>

Specifies the service instance type. For this service, the service instance type is always Log.

Address

Specifies the address of the service instance. The address can be used to access the service and, when registered in the central configuration service, can be used by other services to access the service.

Binding

Specifies the binding type that must be used to communicate with the service instance. In this release of XenDesktop, the binding type is always 'wcf\_HTTP\_kerb'. This indicates that the service provides a Windows Communication Foundation endpoint that uses HTTP binding with integrated authentication.

Version

Specifies the version of the service instance. The version number is used to ensure that the correct versions of the services are used for communications.

ServiceAccount <String>

Specifies the Active Directory account name for the machine on which the service instance is running. The account name is used to provide information about the permissions required for interservice communications.

ServiceAccountSid <String>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

InterfaceType <String>

Specifies the interface type. Each service can provide multiple service instances, each for a different purpose, and the interface defines the purpose. Available interfaces are:

SDK - for PowerShell operations

InterService - for operations between different services

Peer - for communications between services of the same type

Metadata <Citrix.ConfigurationLogging.Sdk.Metadata[]>

The collection of metadata associated with registered service instances, which are service instances held by and obtained from a central configuration service. Metadata is not stored for unregistered service instances.

Notes

If the command fails, the following errors can be returned.

Error Codes

-----

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-LogServiceInstance
```

```
Address : http://MyServer.com:80/Citrix/ConfigurationLoggingService
Binding : wcf_HTTP_kerb
InterfaceType : SDK
Metadata :
MetadataMap :
ServiceAccount : ENG\MyAccount$
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164
ServiceGroupName : MyServiceGroup
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000
ServiceType : Log
Version : 1
```

```
Address : http://MyServer.com:80/Citrix/ConfigurationLoggingService/IServiceApi
Binding : wcf_HTTP_kerb
InterfaceType : InterService
Metadata :
MetadataMap :
```

ServiceAccount : ENG\MyAccount  
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164  
ServiceGroupName : MyServiceGroup  
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d  
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000  
ServiceType : Log  
Version : 1

Get all instances of the ConfigurationLogging Service running on the specified machine. For remote services, use the AdminAddress parameter to define the service for which the interfaces are required. If the AdminAddress parameter has not been specified for the runspace, service instances running on the local machine are returned.

# Get-LogServiceStatus

May 10, 2016

Gets the current status of the ConfigurationLogging Service on the controller.

## Syntax

```
Get-LogServiceStatus [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables the status of the ConfigurationLogging Service on the controller to be monitored. If the service has multiple data stores it will return the overall state as an aggregate of all the data store states. For example, if the site data store status is OK and the secondary data store status is DBUnconfigured then it will return DBUnconfigured.

## Related topics

[Get-LogDataStore](#)

[Set-LogDBConnection](#)

[Test-LogDBConnection](#)

[Get-LogDBConnection](#)

[Get-LogDBSchema](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Get-LogServiceStatus command returns an object containing the status of the ConfigurationLogging Service together with extra diagnostics information.

DBUnconfigured

The ConfigurationLogging Service does not have a database connection configured.

#### DBRejectedConnection

The database rejected the logon attempt from the ConfigurationLogging Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

#### InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the ConfigurationLogging Service schema has not been added to the database.

#### DBNotFound

The specified database could not be located with the configured connection string.

#### DBNewerVersionThanService

The version of the ConfigurationLogging Service currently in use is incompatible with the version of the ConfigurationLogging Service schema on the database. Upgrade the ConfigurationLogging Service to a more recent version.

#### DBOlderVersionThanService

The version of the ConfigurationLogging Service schema on the database is incompatible with the version of the ConfigurationLogging Service currently in use. Upgrade the database schema to a more recent version.

#### DBVersionChangeInProgress

A database schema upgrade is currently in progress.

#### OK

The ConfigurationLogging Service is running and is connected to a database containing a valid schema.

#### Failed

The ConfigurationLogging Service has failed.

#### Unknown

(0) The service status cannot be determined.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-LogServiceStatus
```

## DBUnconfigured

Get the current status of the ConfigurationLogging Service.



# Get-LogSite

May 10, 2016

Gets global configuration logging settings.

## Syntax

```
Get-LogSite [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet retrieves the global configuration logging settings.

## Related topics

[Set-LogSite](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.ConfigurationLogging.Sdk.Site

Get-LogSite returns an object containing the following properties:

- o Locale - the current language that logs should be recorded in. Can be: 'en', 'ja', 'zh-CN', 'de', 'es' or 'fr'.
- o State - the current state of configuration logging. Can be: Enabled, Disabled, Mandatory or NotSupported.

## Examples

----- **EXAMPLE 1** -----

```
C:\PS> Get-LogSite
```

Gets configuration logging site settings.

# Get-LogSummary

May 10, 2016

Gets operations logged within time intervals inside a date range.

## Syntax

```
Get-LogSummary [-StartDateRange <DateTime>] [-EndDateRange <DateTime>] [-IntervalSeconds <Int64>] [-GetLowLevelOperations] [-IncludeIncomplete] [-OperationType <OperationType>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Get-LogSummary cmdlet retrieves summary counts of operations logged within time intervals inside a date range. The returned data indicates the rate at which configuration changes and activities were performed out within a time period.

## Related topics

[Get-LogHighLevelOperation](#)

[Get-LogLowLevelOperation](#)

## Parameters

**-StartDateRange**<DateTime>

Specifies the start of the summary period date range

Required?	false
Default Value	1900-01-01 00:00:00
Accept Pipeline Input?	false

**-EndDateRange**<DateTime>

Specifies the end of the summary period date range.

Required?	false
Default Value	DateTime.UtcNow
Accept Pipeline Input?	false

**-IntervalSeconds**<Int64>

Specifies the size, in seconds, of each time interval required within the summary date range. If this is not specified, is null, zero or exceeds the specified date range, it defaults to the total number of seconds between EndDateRange and StartDateRange.

Required?	false

Default Value	Total number of seconds in the EndDateRange and StartDateRange time span.
Accept Pipeline Input?	false

#### **-GetLowLevelOperations**<SwitchParameter>

Specifies if the cmdlet should return low level operation summary counts.

Required?	false
Default Value	\$false - high level operations counts are returned.
Accept Pipeline Input?	false

#### **-IncludeIncomplete**<SwitchParameter>

Specifies if incomplete operations should be included in the returned summary counts.

Required?	false
Default Value	\$false - incomplete operations are excluded.
Accept Pipeline Input?	false

#### **-OperationType**<OperationType>

Specifies the type of logged operations to include. Values can be 'AdminActivity' or 'ConfigurationChange'

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.Collections.Generic.Dictionary<string, int>

The summary data is returned as a collection of dictionary items. The 'Key' value of each dictionary item specifies the start of the time interval within the overall summary date range. The 'Value' data in each dictionary item contains the count of operations which were started within that time interval.

## Notes

If the specified summary date range and interval period will result in more than 50,000 intervals being returned Get-LogSummary will generate an error and abort the operation.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> $logSummary = Get-LogSummary
```

Get a summary of all completed high level operations. The returned log summary collection will contain a single item for the time period spanning the entire [1900-01-01 00:00:00]-[UtcNow] date range. e.g.

Key ==> Value

01/01/1900 00:00:00 ==> 41

### ----- EXAMPLE 2 -----

```
C:\PS> $daily = 60*60*24
```

```
C:\PS> [DateTime]$startRange = "2013-02-01 14:50:39"
```

```
C:\PS> [DateTime]$endRange = $startRange.AddDays(14)
```

```
C:\PS> Get-LogSummary -StartDateRange $startRange -EndDateRange $endRange -intervalSeconds $daily
```

Gets a summarised count of completed high level operations logged over two weeks, at daily intervals. The returned log summary collection will contain multiple items; one for each day in the summary date range. - e.g.

Key ==> Value

01/02/2013 14:50:39 ==> 0

02/02/2013 14:50:39 ==> 4

03/02/2013 14:50:39 ==> 21

04/02/2013 14:50:39 ==> 0

05/02/2013 14:50:39 ==> 0

06/02/2013 14:50:39 ==> 0

07/02/2013 14:50:39 ==> 5

08/02/2013 14:50:39 ==> 0

09/02/2013 14:50:39 ==> 0

10/02/2013 14:50:39 ==> 0

11/02/2013 14:50:39 ==> 0

12/02/2013 14:50:39 ==> 7

13/02/2013 14:50:39 ==> 0

14/02/2013 14:50:39 ==> 0

15/02/2013 14:50:39 ==> 12

----- **EXAMPLE 3** -----

C:\PS> \$hourly = 60\*60

C:\PS> [DateTime]\$startRange = "2013-02-03 00:00:00"

C:\PS> [DateTime]\$endRange = "2013-02-03 23:59:59"

C:\PS> Get-LogSummary -StartDateRange \$startRange -EndDateRange \$endRange -intervalSeconds \$hourly -GetLowLevelOperations

Gets a summarised count of completed low level operations logged during a day, at hourly intervals. The returned log summary collection will contain multiple items; one for each hour in the summary date range - e.g.

Key ==> Value

04/03/2013 00:00:00 ==> 12

04/03/2013 01:00:00 ==> 10

04/03/2013 02:00:00 ==> 5

.

.

.

04/03/2013 21:00:00 ==> 26

04/03/2013 22:00:00 ==> 0

04/03/2013 23:00:00 ==> 9

# Remove-LogOperation

May 10, 2016

Deletes configuration logs

## Syntax

```
Remove-LogOperation -DatabaseCredentials <PSCredential> [-StartDateRange <DateTime>] [-EndDateRange <DateTime>] [-IncludeIncomplete] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-LogOperation -UserName <String> [-Password <String>] [-StartDateRange <DateTime>] [-EndDateRange <DateTime>] [-IncludeIncomplete] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-LogOperation -UserName <String> -SecurePassword <SecureString> [-StartDateRange <DateTime>] [-EndDateRange <DateTime>] [-IncludeIncomplete] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Remove-LogOperation deletes logs from the Configuration Logging database. This cmdlet targets high level operation logs for deletion. The associated low level operations logs are deleted as part of this operation via cascade deletion functionality present in the configuration logging database schema.

## Related topics

### Parameters

**-DatabaseCredentials**<PSCredential>

Specifies the credentials of a database user with permission to delete records from the Configuration Logging database.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-UserName**<String>

Specifies the user name of a database user with permission to delete records from the Configuration Logging database.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-SecurePassword**<SecureString>

Specifies the password a database user, in secure string form, with permission to delete records from the Configuration Logging database.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Password**<String>

Specifies the password of a database user with permission to delete records from the Configuration Logging database.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-StartDateRange**<DateTime>

Specifies the start time of the earliest high level operation to delete

Required?	false
Default Value	DateTime.Min
Accept Pipeline Input?	false

#### **-EndDateRange**<DateTime>

Specifies the end time of the latest high level operation to delete

Required?	false
Default Value	DateTime.UtcNow
Accept Pipeline Input?	false

#### **-IncludeIncomplete**<SwitchParameter>

Specifies if incomplete high level operations should be deleted.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Remove-LogOperation -UserName "DOMAIN\User" -Password "UserPassword"
Remove all completed operation logs
```

### ----- EXAMPLE 2 -----

```
C:\PS> Remove-LogOperation -UserName "DOMAIN\User" -Password "UserPassword" -IncludeIncomplete
Remove all operations
```

### ----- EXAMPLE 3 -----

```
C:\PS> Remove-LogOperation -username "domain\userName" -password "password" -StartDateRange "2013-01-01 12:00:00" -EndDateRange "2013-01-31 12:00:00"
Delete logs started on or after "2013-01-01 12:00:00" and completed on or before "2013-01-31 12:00:00"
```

### ----- EXAMPLE 4 -----

```
C:\PS> $securePassword = ConvertTo-SecureString -String "UserPassword" -AsPlainText -Force
C:\PS> $credentials = New-Object -TypeName System.Management.Automation.PSCredential -ArgumentList "DOMAIN\UserName", $securePassword
C:\PS> Remove-LogOperation -StartDateRange -DatabaseCredentials $credentials
Delete logs by supplying database user credentials via a credentials object.
```



# Remove-LogServiceMetadata

May 10, 2016

Removes metadata from the given Service.

## Syntax

```
Remove-LogServiceMetadata [-ServiceHostId] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-LogServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-LogServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-LogServiceMetadata [-InputObject] <Service[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Service.

## Related topics

[Set-LogServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name**<String>

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-LogService | % { Remove-LogServiceMetadata -Map $_.MetadataMap }
```

Remove all metadata from all Service objects.

# Remove-LogSiteMetadata

May 10, 2016

Removes metadata from the given Site.

## Syntax

```
Remove-LogSiteMetadata -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-LogSiteMetadata -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-LogSiteMetadata -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-LogSiteMetadata -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Site.

## Related topics

[Set-LogSiteMetadata](#)

## Parameters

**-Name**<String>

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- **EXAMPLE 1** -----

```
c:\PS>Get-LogSite | % { Remove-LogSiteMetadata -Map $_.MetadataMap }
Remove all metadata from all Site objects.
```

# Reset-LogDataStore

May 10, 2016

Refreshes the database string currently being used by the Log service.

## Syntax

```
Reset-LogDataStore [-DataStore] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the string for the database connection currently being used by the ConfigurationLogging Service. Can only be called for secondary data stores.

There is no requirement for a database connection to be configured in order for this command to be used.

## Related topics

[Get-LogDataStore](#)

## Parameters

### **-DataStore**<String>

Specifies the database connection logical name to be used by the ConfigurationLogging Service. Can be either be 'Site' or the logical name of the secondary data store. Specifying the site data store will display an error because this operation is not supported for site data stores.

Required?	true
Default Value	Site
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false



## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.ConfigurationLogging.Sdk.ServiceStatus

The status of the specified data store.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

### PermissionDenied

You do not have permission to execute this command.

### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

### CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Reset-LogDataStore -DataStore Secondary
```

OK

Refresh the database connection string for the ConfigurationLogging Service.

# Reset-LogServiceGroupMembership

May 10, 2016

Reloads the access permissions and configuration service locations for the ConfigurationLogging Service.

## Syntax

```
Reset-LogServiceGroupMembership [-ConfigServiceInstance] <ServiceInstance[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Enables you to reload ConfigurationLogging Service access permissions and configuration service locations. The Reset-LogServiceGroupMembership command must be run on at least one instance of the service type (Log) after installation and registration with the configuration service. Without this operation, the ConfigurationLogging services will be unable to communicate with other services in the XenDesktop deployment. When the command is run, the services are updated when additional services are added to the deployment, provided that the configuration service is not stopped. The Reset-LogServiceGroupMembership command can be run again to refresh this information if automatic updates do not occur when new services are added to the deployment. If more than one configuration service instance is passed to the command, the first instance that meets the expected service type requirements is used.

## Related topics

### Parameters

**-ConfigServiceInstance**<ServiceInstance[]>

Specifies the configuration service instance object that represents the service instance for the type 'InterService' that references a configuration service for the deployment.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

Citrix.ConfigurationLogging.Sdk.ServiceInstance[] Service instances containing a ServiceInstance object that refers to the central configuration service interservice interface can be piped to the Reset-LogServiceGroupMembership command.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### NoSuitableServiceInstance

None of the supplied service instance objects were suitable for resetting service group membership.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config | Reset-LogServiceGroupMembership
```

Reset the service group membership for a service in a deployment where the configuration service is configured and running on the same machine as the service.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config -AdminAddress OtherServer.example.com | Reset-LogServiceGroupmembership
```

Reset the service group membership for a service in a deployment where the configuration service that is configured and running on a machine named 'OtherServer.example.com'.

# Set-LogDBConnection

May 10, 2016

Configures a database connection for the ConfigurationLogging Service.

## Syntax

```
Set-LogDBConnection [-DBConnection] <String> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [[-DataStore] <String>]
[<CommonParameters>]
```

## Detailed Description

Configures a connection to a database in which the ConfigurationLogging Service can store its state. The service will attempt to connect and start using the database immediately after the connection is configured. The database connection string is updated to the specified value regardless of whether it is valid or not. Specifying an invalid connection string prevents a service from functioning until the error is corrected.

After a connection is configured, you cannot alter it without first clearing it (by setting the connection to \$null).

You do not need to configure a database connection to use this command.

## Related topics

[Get-LogServiceStatus](#)

[Get-LogDBConnection](#)

[Test-LogDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be used by the ConfigurationLogging Service. Passing in \$null will clear any existing database connection configured.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Force**<SwitchParameter>

If present, allows the local administrator to set the connection string to null when there are problems contacting the database or other services.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level

operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### **-DataStore**<String>

Specifies the logical name of the data store for the ConfigurationLogging Service. Can be either be 'Site' or the logical name of the secondary data store.

Required?	false
Default Value	Site
Accept Pipeline Input?	false

#### Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Set-LogDBConnection command returns an object containing the status of the ConfigurationLogging Service together with extra diagnostics information.

DBUnconfigured

The ConfigurationLogging Service does not have a database connection configured.

DBRejectedConnection

The database rejected the logon attempt from the ConfigurationLogging Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the ConfigurationLogging Service schema has not been added to the database.

DBNotFound

The specified database could not be located with the configured connection string.

#### DBNewerVersionThanService

The version of the ConfigurationLogging Service currently in use is incompatible with the version of the ConfigurationLogging Service schema on the database. Upgrade the ConfigurationLogging Service to a more recent version.

#### DBOlderVersionThanService

The version of the ConfigurationLogging Service schema on the database is incompatible with the version of the ConfigurationLogging Service currently in use. Upgrade the database schema to a more recent version.

#### DBVersionChangeInProgress

A database schema upgrade is currently in progress.

#### OK

The ConfigurationLogging Service is running and is connected to a database containing a valid schema.

#### Failed

The ConfigurationLogging Service has failed.

#### Unknown

The status of the ConfigurationLogging Service cannot be determined.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### InvalidDBConnectionString

The database connection string has an invalid format.

#### DatabaseConnectionDetailsAlreadyConfigured

There was already a database connection configured. After a configuration is set, it can only be set to \$null.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.



CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Set-LogDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Configures a database connection string for the ConfigurationLogging Service.

----- **EXAMPLE 2** -----

```
c:\PS>Set-LogDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Configures an invalid database connection string for the ConfigurationLogging Service.

# Set-LogServiceMetadata

May 10, 2016

Adds or updates metadata on the given Service.

## Syntax

```
Set-LogServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-LogServiceMetadata [-ServiceHostId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Set-LogServiceMetadata [-InputObject] <Service[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Set-LogServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Allows you to store additional custom data against given Service objects.

## Related topics

[Remove-LogServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name<String>**

Specifies the property name of the metadata to be added. The property must be unique for the Service specified. The property cannot contain any of the following characters \/:;#. \*?=<> | []()

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Value<String>**

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	

Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-LogServiceMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

InvalidParameterCombination

The cmdlet parameters are inconsistent.

UnknownObject

One of the specified objects was not found.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Set-LogServiceMetadata -ServiceHostId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
---	-----
property	value

Add metadata with a name of 'property' and a value of 'value' to the Service with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Set-LogSite

May 10, 2016

Sets global configuration logging settings.

## Syntax

```
Set-LogSite [-State <LoggingState>] [-Locale <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

This cmdlet sets the global configuration logging settings.

## Related topics

[Get-LogSite](#)

## Parameters

**-State**<LoggingState>

Sets the state of configuration logging. Values can be:

o Enabled - state changes will be logged. If logging is unavailable, the state change will still be permitted. o Disabled - state changes will not be logged.

o Mandatory - state change will be logged. If logging is unavailable, the state change will not be permitted.

When the state is set to Enabled or Mandatory, XenDesktop services will attempt to log operations (which perform configuration changes) before performing them.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Locale**<String>

Sets the language that logs should be recorded in. Values can be: 'en', 'ja', 'zh-CN', 'de', 'es' or 'fr'.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-PassThru**<SwitchParameter>

Returns the affected record. By default, this cmdlet does not generate any output.

Required?	false
Default Value	False
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.ConfigurationLogging.Sdk.Site

Current logging settings

### Notes

Configuration logging will automatically transition to a 'NotSupported' state if the logging feature is not licensed. Set-LogSite will reject request to set logging to 'Enabled' or 'Mandatory' while the logging feature is not licensed.

### Examples

----- **EXAMPLE 1** -----

```
C:\PS> Set-LogSite -Locale "zh-CN"
```

Set the logging language to Chinese. The logging state will be unchanged.

----- **EXAMPLE 2** -----

```
C:\PS> Set-LogSite -State "Mandatory"
```

Set the logging state to mandatory. The logging locale will be unaffected. In this state, no configuration change will be allowed unless it is successfully logged.

----- **EXAMPLE 3** -----

```
C:\PS> Set-LogSite -Locale "de" -State "Enabled"
```

Set the logging language to German and the state to Enabled.



# Set-LogSiteMetadata

May 10, 2016  
Adds or updates metadata on the given Site.

## Syntax

Set-LogSiteMetadata -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-LogSiteMetadata -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-LogSiteMetadata -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-LogSiteMetadata -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Provides the ability for additional custom data to be stored against given Site objects.

## Related topics

[Remove-LogSiteMetadata](#)

## Parameters

**-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the Site specified. The property cannot contain any of the following characters \/:#.\*?=<> | []()''

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-LogSiteMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

## InvalidParameterCombination

The cmdlet parameters are inconsistent.

## UnknownObject

One of the specified objects was not found.

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Set-LogSiteMetadata -SiteDUMMY_IdProperty 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
-----	-------

---

----

property	value
----------	-------

Add metadata with a name of 'property' and a value of 'value' to the Site with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Start-LogHighLevelOperation

May 10, 2016  
Logs the start of a high level operation.

## Syntax

Start-LogHighLevelOperation -Text <String> -Source <String> [-StartTime <DateTime>] [-OperationType <OperationType>] [-TargetTypes <String[]>] [-Parameters <Hashtable>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Start-LogHighLevelOperation creates a log entry to record the start of a high level operation.

Start-LogHighLevelOperation can be called to log high level configuration changes which originate from customized configuration scripts. Start-LogHighLevelOperation should be called before the script invokes service SDK cmdlets which execute the configuration changes.

Start-LogHighLevelOperation returns a HighLevelOperation object which contains information about the started high level operation. The "Id" property of the returned HighLevelOperation uniquely identifies the started high level operation. This can be supplied into the "-LoggingId" parameter which is implemented in all service SDK cmdlets which execute loggable configuration changes.

High level operation logs are automatically created by the XenDesktop consoles when:

- o Initiating an operation which performs configuration changes.
- o Initiating an operation which performs an administration activity.

## Configuration Change and Administrator Activity

A configuration change is an operation which alters the configuration of the XenDesktop site. Examples of a configuration changes include:

- o Creating or editing a host.
- o Creating or editing a catalog.
- o Adding a user to a delivery group.
- o Deleting a machine.

An administrator activity operation doesn't directly alter site configuration, but it could be an operation carried out by an administrator as part of site management or helpdesk support. Examples of administrator activities include:

- o Shutdown/start/restart of a user desktop.
- o Studio or Director sending a message to a user.
- o Rebooting a user's desktop.

Once the change being logged has completed (whether successfully or not) the Stop-LogHighLevelOperation cmdlet should be called to log the completion of a started high level operation.

## Related topics

[Stop-LogHighLevelOperation](#)

[Get-LogHighLevelOperation](#)

## Parameters

**-Text**<String>

Specifies text to describe the high level operation.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Source**<String>

Specifies the source of the high level operation.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-StartTime**<DateTime>

Specifies the start time of the high level operation.

Required?	false
Default Value	DateTime.UtcNow
Accept Pipeline Input?	false

#### **-OperationType**<OperationType>

Specifies the type of operation being logged. Values can be:

- o AdminActivity - If the operation being logged performs an administration activity.
- o ConfigurationChange - If the operation being logged performs a configuration change.

Required?	false
Default Value	ConfigurationChange
Accept Pipeline Input?	false

#### **-TargetTypes**<String[]>

Specifies the names of the object types that will be affected by the operation being logged.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Parameters**<Hashtable>

Specifies the names and values of parameters that are supplied to the operation being logged.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.ConfigurationLogging.Sdk.HighLevelOperation

The newly logged high level operation start.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> $succeeded = $false #indicates if high level operation succeeded.
C:\PS> # Log high level operation start.
C:\PS> $highLevelOp = Start-LogHighLevelOperation -Text "Create catalog" -Source "My Custom Script"
C:\PS>
C:\PS> try
C:\PS> {
C:\PS> # Create catalog object
C:\PS> $catalog = New-BrokerCatalog -Name "MyCatalog" -ProvisioningType Manual -AllocationType Permanent -MinimumFunctionalLevel 'LMAX' -LoggingId $highLevelOp.Id
C:\PS>
C:\PS> # Add a machine to the catalog
C:\PS> $machine = New-BrokerMachine -CatalogUid $catalog.Uid -MachineName "DOMAIN\Machine" -LoggingId $highLevelOp.Id
C:\PS> $succeeded = $true
C:\PS> }
C:\PS> catch{ "Error encountered" }
C:\PS>
C:\PS> finally{
C:\PS> # Log high level operation stop, and indicate its success
C:\PS> Stop-LogHighLevelOperation -HighLevelOperationId $highLevelOp.Id -IsSuccessful $succeeded
C:\PS> }
```

Creates an unmanaged catalog and assigns a machine to it, within the scope of a high level operation start and stop. The identifier of the high level operation is passed into the "-LoggingId" parameter of the service SDK cmdlets. The execution of the cmdlets in the services will create the low level operation logs for the supplied high level operation.

# Stop-LogHighLevelOperation

May 10, 2016  
Logs the completion of a previously started high level operation.

## Syntax

Stop-LogHighLevelOperation -HighLevelOperationId <String> -IsSuccessful <Boolean> [-EndTime <DateTime>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Stop-LogHighLevelOperation logs the completion of a started high level operation.

## Related topics

[Start-LogHighLevelOperation](#)

[Get-LogHighLevelOperation](#)

## Parameters

**-HighLevelOperationId**<String>

Specifies the identifier of the high level operation being completed.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-IsSuccessful**<Boolean>

Specifies if the started high level operation completed successfully.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-EndTime**<DateTime>

Specifies the end time of the high level operation being completed.

Required?	false
Default Value	DateTime.UtcNow.
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Examples

----- **EXAMPLE 1** -----

```

C:\PS> $succeeded = $false #indicates if high level operation succeeded.
C:\PS> # Log high level operation start.
C:\PS> $highLevelOp = Start-LogHighLevelOperation -Text "Create unmanaged catalog" -Source "My Custom Script"
C:\PS>
C:\PS> try
C:\PS> {
C:\PS> # Create catalog object
C:\PS> $catalog = New-BrokerCatalog -Name "MyCatalog" -ProvisioningType Manual -AllocationType Permanent -MinimumFunctionalLevel 'LMAX' -LoggingId $highLevelOp.Id
C:\PS>
C:\PS> # Add a machine to the catalog
C:\PS> $machine = New-BrokerMachine -CatalogUid $catalog.Uid -MachineName "DOMAIN\Machine" -LoggingId $highLevelOp.Id
C:\PS> $succeeded = $true
C:\PS> }
C:\PS> catch{ "Error encountered" }
C:\PS>
C:\PS> finally{
C:\PS> # Log high level operation stop, and indicate its success
C:\PS> Stop-LogHighLevelOperation -HighLevelOperationId $highLevelOp.Id -IsSuccessful $succeeded
C:\PS> }

```

Creates an unmanaged catalog and assigns a machine to it, within the scope of a high level operation start and stop. The identifier of the high level operation is passed into the "-LoggingId" parameter of the service SDK cmdlets. The execution of the cmdlets in the services will create the low level operation logs for the supplied high level operation.



# Test-LogDBConnection

May 10, 2016

Tests a database connection for the ConfigurationLogging Service.

## Syntax

```
Test-LogDBConnection [-DBConnection] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [[-DataStore] <String>] [<CommonParameters>]
```

## Detailed Description

Tests a connection to the database in which the ConfigurationLogging Service can store its state. The service will attempt to connect to the database without affecting the current connection to the database.

You do not have to clear the connection to use this command.

## Related topics

[Get-LogServiceStatus](#)

[Get-LogDBConnection](#)

[Set-LogDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be tested by the ConfigurationLogging Service.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### **-DataStore<String>**

Specifies the logical name of the data store for the ConfigurationLogging Service. Can be either be 'Site' or the logical name of the secondary data store.

Required?	false
Default Value	Site
Accept Pipeline Input?	false

#### **Return Values**

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Test-LogDBConnection command returns an object containing the status of the ConfigurationLogging Service if the connection string of the specified data store were to be set to the string being tested, together with extra diagnostics information for the specified connection string.

#### **DBRejectedConnection**

The database rejected the logon attempt from the ConfigurationLogging Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

#### **InvalidDBConfigured**

The expected stored procedures are missing from the database. This may be because the ConfigurationLogging Service schema has not been added to the database.

#### **DBNotFound**

The specified database could not be located with the configured connection string.

#### **DBNewerVersionThanService**

The version of the ConfigurationLogging Service currently in use is incompatible with the version of the ConfigurationLogging Service schema on the database. Upgrade the ConfigurationLogging Service to a more recent version.

#### **DBOlderVersionThanService**

The version of the ConfigurationLogging Service schema on the database is incompatible with the version of the ConfigurationLogging Service currently in use. Upgrade the database schema to a more recent version.

#### **DBVersionChangeInProgress**

A database schema upgrade is currently in progress.

#### **OK**

The Set-LogDBConnection command would succeed if it were executed with the supplied connection string.

#### **Failed**

The ConfigurationLogging Service has failed.

## Unknown

The status of the ConfigurationLogging Service cannot be determined.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidDBConnectionString

The database connection string has an invalid format.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Test-LogDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Tests a database connection string for the ConfigurationLogging Service.

### ----- EXAMPLE 2 -----

```
c:\PS>Test-LogDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Tests an invalid database connection string for the ConfigurationLogging Service.

# Citrix.DelegatedAdmin.Admin.V1

May 10, 2016

## Overview

Name	Description
<a href="#">AdminDelegatedAdminSnapin</a>	The Delegated Administration Service PowerShell snap-in provides
<a href="#">Admin Filtering</a>	Describes the common filtering options for XenDesktop cmdlets.

## Cmdlets

Name	Description
<a href="#">Add-AdminPermission</a>	Add permissions to the set of permissions of a role.
<a href="#">Add-AdminRight</a>	Grants a given right to the specified administrator.
<a href="#">Get-AdminAdministrator</a>	Gets administrators configured for this site.
<a href="#">Get-AdminDBConnection</a>	Gets the database string for the specified data store used by the DelegatedAdmin Service.
<a href="#">Get-AdminDBSchema</a>	Gets a script that creates the DelegatedAdmin Service database schema for the specified data store.
<a href="#">Get-AdminDBVersionChangeScript</a>	Gets a script that updates the DelegatedAdmin Service database schema.
<a href="#">Get-AdminEffectiveAdministrator</a>	Retrieve the effective administrator objects for a user.
<a href="#">Get-AdminEffectiveRight</a>	Gets the set of Right objects associated with the current user.
<a href="#">Get-AdminInstalledDBVersion</a>	Gets a list of all available database schema versions for the DelegatedAdmin Service.
<a href="#">Get-AdminPermission</a>	Gets permissions configured for the site.
<a href="#">Get-AdminPermissionGroup</a>	Gets permission groups configured for the site.
<a href="#">Get-AdminRevision</a>	Gets the current revision of the delegated administration configuration data.

<b>Get-AdminRole</b> <b>Name</b>	<b>Gets roles configured for this site.</b> <b>Description</b>
Get-AdminRoleConfiguration	Gets role configurations for this site.
Get-AdminScope	Gets scopes configured for this site.
Get-AdminService	Gets the service record entries for the DelegatedAdmin Service.
Get-AdminServiceAddedCapability	Gets any added capabilities for the DelegatedAdmin Service on the controller.
Get-AdminServiceInstance	Gets the service instance entries for the DelegatedAdmin Service.
Get-AdminServiceStatus	Gets the current status of the DelegatedAdmin Service on the controller.
Import-AdminRoleConfiguration	Imports role configuration data into the Delegated Administration Service.
New-AdminAdministrator	Adds a new administrator to the site.
New-AdminRole	Adds a new custom role to the site.
New-AdminScope	Adds a new scope to the site.
Remove-AdminAdministrator	Removes administrators from the site.
Remove-AdminAdministratorMetadata	Removes metadata from the given Administrator.
Remove-AdminPermission	Remove permissions from the set of permissions of a role.
Remove-AdminRight	Removes rights from an administrator.
Remove-AdminRole	Removes a role from the site.
Remove-AdminRoleMetadata	Removes metadata from the given Role.
Remove-AdminScope	Removes a scope from the site.
Remove-AdminScopeMetadata	Removes metadata from the given Scope.

<b>Remove-AdminServiceMetadata</b> <b>Name</b>	<b>Description</b>
Rename-AdminRole	Rename a role
Rename-AdminScope	Rename a scope
Reset-AdminServiceGroupMembership	Reloads the access permissions and configuration service locations for the DelegatedAdmin Service.
Set-AdminAdministrator	Sets the properties of an administrator.
Set-AdminAdministratorMetadata	Adds or updates metadata on the given Administrator.
Set-AdminDBConnection	Configures a database connection for the DelegatedAdmin Service.
Set-AdminRole	Set the properties of a role.
Set-AdminRoleMetadata	Adds or updates metadata on the given Role.
Set-AdminScope	Set the properties of a scope.
Set-AdminScopeMetadata	Adds or updates metadata on the given Scope.
Set-AdminServiceMetadata	Adds or updates metadata on the given Service.
Test-AdminAccess	Retrieves the scopes where the specified operation is permitted.
Test-AdminDBConnection	Tests a database connection for the DelegatedAdmin Service.

# about\_AdminDelegatedAdminSnapin

May 10, 2016

## TOPIC

about\_AdminDelegatedAdminSnapin

## SHORT DESCRIPTION

The Delegated Administration Service PowerShell snap-in provides administrative functions for the Delegated Administration Service.

## COMMAND PREFIX

All commands in this snap-in are prefixed with 'Admin'.

## LONG DESCRIPTION

The Delegated Administration Service PowerShell snap-in enables both local and remote administration of the Delegated Administration Service.

The Delegated Administration Service (or DAS for short) stores information about Citrix administrators and the rights they have. Services in the XenDesktop deployment use the DAS to determine whether a particular user has the privilege to perform an operation or not.

The snap-in provides storage and configuration of these entities:

### Administrators

Each administrator object represents an individual person or a group of people identified by their Active Directory account. Administrators can be enabled and disabled.

The effective rights that a user has is the union of any rights that they have by looking at their Active Directory group membership. Disabled administrator entries are ignored for this calculation.

Once a site is setup, there must always be a full administrator and the Delegated Administration snap-in rejects requests to remove or disable the last full administrator.

### Roles

A role represents a job function. That is, anyone with a given role is expected to be able to use or perform the tasks, wizards, and actions associated with that role. Administrators may have multiple roles for a particular site.

Some roles are built-in, and some editions of the product allow custom roles to be created with different combinations of permissions.

## Scopes

Scopes represent a collection of objects, and are used to group objects for administrative purposes in a way that is relevant to the organisation. They can be used to represent both hierarchical and non-hierarchical relationships.

Objects can exist in multiple scopes at once. You may find it easier to think of scopes as labels, or a non-exclusive grouping such as a play-list.

All objects are implicitly in the built-in 'All' scope.

Some objects are not scoped, and access to them is through either the 'All' scope or indirectly through a scoped object. For example sessions are not directly scoped but can be accessed using the scope of the desktop group.

The DAS stores information about scopes, but the mapping between scopes and objects is stored and updated using the PowerShell snap-ins of each corresponding service. For example, Delivery Group scopes are managed using the Broker PowerShell snap-in.

## Rights

Rights determine what an administrator can do and where they can do it. They are expressed as a number of <role, scope> pairs associated with each administrator.

To gain access to any particular object, a person must match an administrator object that has an appropriate right that allows the required operation in a scope that the object is a member of.

## Permissions

Each task, wizard or action in the Citrix Studio or Director consoles represents a unit of functionality that an administrator can perform. Permissions are expressed at a high level and generally correspond directly to the labels in the consoles. For example: "Edit catalog", or "Create delivery group".



## Permission groups:

Permissions are grouped into related functionality when displayed by the console.

## Operations

Operations are the indivisible unit of functionality that each XenDesktop service can perform, and usually correspond to individual cmdlets. Internally, each permission requires a number of operations to be performed, possibly by different services.

# about\_Admin\_Filtering

May 10, 2016

## TOPIC

XenDesktop - Advanced Dataset Filtering

## SHORT DESCRIPTION

Describes the common filtering options for XenDesktop cmdlets.

## LONG DESCRIPTION

Some cmdlets operate on large quantities of data and, to reduce the overhead of sending all of that data over the network, many of the Get- cmdlets support server-side filtering of the results.

The conventional way of filtering results in PowerShell is to pipeline them into Where-Object, Select-Object, and Sort-Object, for example:

```
Get-<Noun> | Where { $_.Size = 'Small' } | Sort 'Date' | Select -First 10
```

However, for most XenDesktop cmdlets the data is stored remotely and it would be slow and inefficient to retrieve large amounts of data over the network and then discard most of it. Instead, many of the Get- cmdlets provide filtering parameters that allow results to be processed on the server, returning only the required results.

You can filter results by most object properties using parameters derived from the property name. You can also sort results or limit them to a specified number of records:

```
Get-<Noun> -Size 'Small' -SortBy 'Date' -MaxRecordCount 10
```

You can express more complex filter conditions using a syntax and set of operators very similar to those used by PowerShell expressions.

Those cmdlets that support filtering have the following common parameters:

-MaxRecordCount <int>

Specifies the maximum number of results to return.  
For example, to return only the first nine results use:

```
Get-<Noun> -MaxRecordCount 9
```

If not specified, only the first 250 records are returned, and if more are available, a warning is produced:

WARNING: Only first 250 records returned. Use -MaxRecordCount to retrieve more.

You can suppress this warning by using -WarningAction or by specifying a value for -MaxRecordCount.

To retrieve all records, specify a large number for -MaxRecordCount. As the value is an integer, you can use the following:

```
Get-<Noun> -MaxRecordCount [int]::MaxValue
```

-ReturnTotalRecordCount [<SwitchParameter>]

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. For example:

```
Get-<Noun> -MaxRecordCount 9 -ReturnTotalRecordCount
```

....

```
Get-<Noun> : Returned 9 of 10 items
```

```
At line:1 char:18
```

```
+ Get-<Noun> <<<< -MaxRecordCount 9 -ReturnTotalRecordCount
```

```
+ CategoryInfo : OperationStopped: (:) [Get-<Noun>], PartialDataException
```

```
+ FullyQualifiedErrorId : PartialData,Citrix.<SDKName>.SDK.Get<Noun>
```

The count can be accessed using the TotalAvailableResultCount property:

```
$count = $error[0].TotalAvailableResultCount
```

-Skip <int>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

-SortBy <string>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order, respectively. Ascending order is assumed if no prefix is present.

Sorting occurs before -MaxRecordCount and -Skip parameters are applied. For example, to sort by Name and then by Count (largest first) use:

```
-SortBy 'Name,-Count'
```

By default, sorting by an enumeration property uses the numeric value of the elements. You can specify a different sort order by qualifying the name with an ordered list of elements or their numeric values, or <null> to indicate the placement of null values.

Elements not mentioned are placed at the end in their numeric order. For example, to sort by two different enums and then by the object id:

```
-SortBy 'MyState(StateC,<null>,StateA,StateB),Another(0,3,2,1),Id'
```

-Filter <String>

This parameter lets you specify advanced filter expressions, and supports combination of conditions with -and and -or, and grouping with braces. For example:

```
Get-<Noun> -Filter 'Name -like "High*" -or (Priority -eq 1 -and Severity -ge 2)'
```

The syntax is close enough to PowerShell syntax that you can use script blocks in most cases. This can be easier to read as it reduces quoting:

```
Get-<Noun> -Filter { Count -ne $null }
```

The full -Filter syntax is provided below.

## EXAMPLES

Filtering by strings performs a case-insensitive wildcard match.

Separate parameters are combined with an implicit -and operator.

Normal PowerShell quoting rules apply, so you can use single or double quotes, and omit the quotes altogether for many strings. The order of parameters does not make any difference. The following are equivalent:

```
Get-<Noun> -Company Citrix -Product Xen*
```

```
Get-<Noun> -Company "citrix" -Product '[X]EN*'
```

```
Get-<Noun> -Product "Xen*" -Company "CITRIX"
```

```
Get-<Noun> -Filter { Company -eq 'Citrix' -and Product -like 'Xen*' }
```

See [about\\_Quoting\\_Rules](#) and [about\\_Wildcards](#) for details about PowerShell handling of quotes and wildcards.

To avoid wildcard matching or include quote characters, you can escape the wildcards using the normal PowerShell escape mechanisms (see [about\\_Escape\\_Characters](#)), or switch to a filter expression and the `-eq` operator:

```
Get-<Noun> -Company "Abc[*]" # Matches Abc*
Get-<Noun> -Company "Abc`*" # Matches Abc*
Get-<Noun> -Filter { Company -eq "Abc*" } # Matches Abc*
Get-<Noun> -Filter { Company -eq "A`"B`C" } # Matches A"B'C
```

Simple filtering by numbers, booleans, and TimeSpans perform direct equality comparisons, although if the value is nullable you can also search for null values. Here are some examples:

```
Get-<Noun> -Uid 123
Get-<Noun> -Enabled $true
Get-<Noun> -Duration 1:30:40
Get-<Noun> -NullableProperty $null
```

More comparisons are possible using advanced filtering with `-Filter`:

```
Get-<Noun> -Filter 'Capacity -ge 10gb'
Get-<Noun> -Filter 'Age -ge 20 -and Age -lt 40'
Get-<Noun> -Filter 'VolumeLevel -like "[123]"'
Get-<Noun> -Filter 'Enabled -ne $false'
Get-<Noun> -Filter 'NullableProperty -ne $null'
```

You can check boolean values without an explicit comparison operator, and you can also combine them with `-not`:

```
Get-<Noun> -Filter 'Enabled' # Equivalent to 'Enabled -eq $true'
Get-<Noun> -Filter '-not Enabled' # Equivalent to 'Enabled -eq $false'
```

See [about\\_Comparison\\_Operators](#) for an explanation of the operators, but note that only a subset of PowerShell operators are supported (`-eq`, `-ne`, `-gt`, `-ge`, `-lt`, `-le`, `-like`, `-notlike`, `-in`, `-notin`, `-contains`, `-notcontains`).

Enumeration values can either be specified using typed values or the string name of the enumeration value:

```
Get-<Noun> -Shape [Shapes]::Square
Get-<Noun> -Shape Circle
```

With filter expressions, typed values can be specified with simple variables or quoted strings. They also support enumerations with wildcards:

```
$s = [Shapes]::Square
Get-<Noun> -Filter { Shape -eq $s -or Shape -eq "Circle" }
Get-<Noun> -Filter { Shape -like 'C*' }
```

By their nature, floating point values, DateTime values, and TimeSpan values are best suited to relative comparisons rather than just equality. DateTime strings are converted using the locale and time zone of the user device, but you can use ISO8601 format strings (YYYY-MM-DDThh:mm:ss.sTZD) to avoid ambiguity. You can also use standard PowerShell syntax to create these values:

```
Get-<Noun> -Filter { StartTime -ge "2010-08-23T12:30:00.0Z" }
$d = [DateTime]"2010-08-23T12:30:00.0Z"
Get-<Noun> -Filter { StartTime -ge $d }
$d = (Get-Date).AddDays(-1)
Get-<Noun> -Filter { StartTime -ge $d }
```

Relative times are quite common and, when using filter expressions, you can also specify DateTime values using a relative format:

```
Get-<Noun> -Filter { StartTime -ge '-2' } # Two days ago
Get-<Noun> -Filter { StartTime -ge '-1:30' } # Hour and a half ago
Get-<Noun> -Filter { StartTime -ge '-0:0:30' } # 30 seconds ago
```

## ARRAY PROPERTIES

When filtering against list or array properties, simple parameters perform a case-insensitive wildcard match against each of the members. With filter expressions, you can use the -contains and -notcontains operators. Unlike PowerShell, these perform wildcard matching on strings.

Note that for array properties the naming convention is for the returned property to be plural, but the parameter used to search for any match is singular. The following are equivalent (assuming Users is an array property):

```
Get-<Noun> -User Fred*
Get-<Noun> -Filter { User -like "Fred*" }
Get-<Noun> -Filter { Users -contains "Fred*" }
```

You can also use the singular form with -Filter to search using other operators:

```
Match if any user in the list is called "Frederick"
Get-<Noun> -Filter { User -eq "Frederick" }
Match if any user in the list has a name alphabetically below 'F'
Get-<Noun> -Filter { User -lt 'F' }
```

## COMPLEX EXPRESSIONS

When matching against multiple values, you can use a sequence of comparisons joined with -or operators, or you can use -in and -notin:

```
Get-<Noun> -Filter { Shape -eq 'Circle' -or Shape -eq 'Square' }
$shapes = 'Circle','Square'
Get-<Noun> -Filter { Shape -in $shapes }
$sides = 1..4
Get-<Noun> -Filter { Sides -notin $sides }
```

Braces can be used to group complex expressions, and override the default left-to-right evaluation of -and and -or. You can also use -not to invert the sense of any sub-expression:

```
Get-<Noun> -Filter { Size -gt 4 -or (Color -eq 'Blue' -and Shape -eq 'Circle') }
Get-<Noun> -Filter { Sides -lt 5 -and -not (Color -eq 'Blue' -and Shape -eq 'Circle') }
```

## PAGING

The simplest way to page through data is to use the -Skip and -MaxRecordCount parameters. So, to read the first three pages of data with 10 records per page, use:

```
Get-<Noun> -Skip 0 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 10 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 20 -MaxRecordCount 10 <other filtering criteria>
```

You must include the same filtering criteria on each call, and ensure that the data is sorted consistently.

The above approach is often acceptable, but as each call performs an independent query, data changes can result in records being skipped or appearing twice. One approach to improve this is to sort by a unique id field and then start the search for the next page at the unique id after the last unique id of the previous page. For example:

```
Get the first page
Get-<Noun> -MaxRecordCount 10 -SortBy SerialNumber

SerialNumber ...
----- ---
A120004
A120007
... 7 other records ...
A120900

Get the next page
Get-<Noun> -MaxRecordCount 10 -Filter { FirstName -gt 'A120900' }

SerialNumber ...
```

----- ---  
A120901  
B220000  
...

## FILTER SYNTAX DEFINITION

<Filter> ::= <ScriptBlock> | <ComponentList>

<ScriptBlock> ::= "{" <ComponentList> "}"

<ComponentList> ::= <Component> <AndOrOperator> <ComponentList> |

<Component>

<Component> ::= <NotOperator> <Factor> |

<Factor>

<Factor> ::= "(" <ComponentList> ")" |

<PropertyName> <ComparisonOperator> <Value> |  
<PropertyName>

<AndOrOperator> ::= "-and" | "-or"

<NotOperator> ::= "-not" | "!"

<ComparisonOperator>

::= "-eq" | "-ne" | "-le" | "-ge" | "-lt" | "-gt" |  
"-like" | "-notlike" | "-contains" | "-notcontains" |  
"-in" | "-notin"

<PropertyName> ::= <simple name of property>

<Value> ::= <string literal> | <numeric literal> |

<scalar variable> | <array variable> |  
"\$null" | "\$true" | "\$false"



Numeric literals support decimal and hexadecimal literals, with optional multiplier suffixes (kb, mb, gb, tb, pb).

Dates and times can be specified as string literals. The current culture determines what formats are accepted. To avoid any ambiguity, use strings formatted to the ISO8601 standard. If not specified, the current time zone is used.

Relative date-time string literals are also supported, using a minus sign followed by a TimeSpan. For example, "-1:30" means 1 hour and 30 minutes ago.

# Add-AdminPermission

May 10, 2016

Add permissions to the set of permissions of a role.

## Syntax

```
Add-AdminPermission [-InputObject] <Permission[]> -Role <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Add-AdminPermission [-Permission] <String[]> -Role <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Add extra permissions to the set of permissions that a role maps to.

Any administrator with a right including that role immediately gains the ability to use the operations of the new permissions.

Duplicate permissions do not produce an error, and permissions are skipped if the role already contains the permission (without error).

You cannot modify the permissions of built-in roles.

## Related topics

[Remove-AdminPermission](#)

[Get-AdminPermission](#)

[Get-AdminRole](#)

[Get-AdminPermissionGroup](#)

[Test-AdminAccess](#)

## Parameters

**-InputObject**<Permission[]>

Specifies the permissions to add.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Permission**<String[]>

Specifies the list of permissions to add (by identifier).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Role**<String>

Role name or identifier of the role to update.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

Citrix.DelegatedAdmin.Sdk.Permission You can pipe a list of permissions to be added into this command.

## Return Values

None

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Add-AdminPermission -Role MyRole -Permission Global_Read,Logging_Read
```

Add a couple of specific permissions to the 'MyRole' role.

### ----- EXAMPLE 2 -----

```
C:\PS> $list = Get-AdminRole "Delivery Administrator" | Select -Expand Permissions
```

```
C:\PS> Add-AdminPermission -Role MyRole -Permission $list
```

Add all of the permissions of the Delivery Administrator role to MyRole.

# Add-AdminRight

May 10, 2016  
Grants a given right to the specified administrator.

## Syntax

Add-AdminRight -Scope <String> -Role <String> -Administrator <String> [-LoggingId <Guid>] [-AdminAddress <String>]  
[<CommonParameters>]

Add-AdminRight -Role <String> -Administrator <String> [-All] [-LoggingId <Guid>] [-AdminAddress <String>]  
[<CommonParameters>]

Add-AdminRight [-InputObject] <Right[]> -Administrator <String> [-LoggingId <Guid>] [-AdminAddress <String>]  
[<CommonParameters>]

## Detailed Description

Use the Add-AdminRight cmdlet to add rights (role and scope pairs) to an administrator.

For convenience, you can use the -All parameter to specify the 'All' scope.

Use the Get-AdminAdministrator cmdlet to determine what rights an administrator has.

## Related topics

[Get-AdminAdministrator](#)

[Get-AdminEffectiveRight](#)

[Remove-AdminRight](#)

## Parameters

**-InputObject**<Right[]>

Specifies the rights to add from Right objects.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Scope**<String>

Specifies the scope name or scope identifier.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	false

#### **-Role**<String>

Specifies the role name or role identifier.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Administrator**<String>

Specifies the name or SID of the administrator.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-All**<SwitchParameter>

Specifies the 'All' scope. This parameter avoids localization issues or having to type the identifier of the 'All' scope.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Input Type

Citrix.DelegatedAdmin.Sdk.Right You can pipe the rights to be added into this command.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Add-AdminRight -Role 'Help Desk Administrator' -Scope London -Administrator DOMAIN\Admin1
```

Assigns the 'Help Desk Administrator' role and 'London' scope to the 'Admin1' administrator.

#### ----- EXAMPLE 2 -----

```
C:\PS> Add-AdminRight -Role 'Full Administrator' -All -Administrator DOMAIN\Admin1
```

Assigns the 'Full Administrator' role and 'All' scope to the 'Admin1' administrator.

#### ----- EXAMPLE 3 -----

```
C:\PS> $admin = Get-AdminAdministrator -Name DOMAIN\ExistingAdmin
C:\PS> Add-AdminRight -InputObject $admin.Rights -Administrator DOMAIN\NewAdmin
```

Copies the administrator rights from 'ExistingAdmin' to 'NewAdmin'.

# Get-AdminAdministrator

May 10, 2016  
Gets administrators configured for this site.

## Syntax

```
Get-AdminAdministrator [[-Name] <String>] [-Sid <String>] [-Enabled <Boolean>] [-Metadata <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieves administrators matching the specified criteria. If no parameters are specified this cmdlet enumerates all administrators.

See `about_Admin_Filtering` for information about advanced filtering options.

## Related topics

- [New-AdminAdministrator](#)
- [Set-AdminAdministrator](#)
- [Remove-AdminAdministrator](#)
- [Set-AdminAdministratorMetadata](#)
- [Remove-AdminAdministratorMetadata](#)

## Parameters

**-Name**<String>

Gets administrators with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Sid**<String>

Gets administrators with the specified SID (security identifier).

Required?	false
Default Value	



Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-Enabled**<Boolean>

Gets administrators with the specified value of Enabled.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Admin\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Admin\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.DelegatedAdmin.Sdk.Administrator

Get-AdminAdministrator returns an object for each matching administrator.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> Get-AdminAdministrator -Name DOMAIN\TestUser
```

Finds the administrator object (if one exists) for user "DOMAIN\TestUser".

##### ----- EXAMPLE 2 -----

```
C:\PS> Get-AdminAdministrator -Enabled $false
```

Finds all disabled administrator objects.

# Get-AdminDBConnection

May 10, 2016

Gets the database string for the specified data store used by the DelegatedAdmin Service.

## Syntax

```
Get-AdminDBConnection [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the database connection string for the specified data store.

If the returned string is blank, no valid connection string has been specified. In this case the service is running, but is idle and awaiting specification of a valid connection string.

## Related topics

[Get-AdminServiceStatus](#)

[Set-AdminDBConnection](#)

[Test-AdminDBConnection](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

system.string

The database connection string configured for the DelegatedAdmin Service.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

NoDBConnections

The database connection string for the DelegatedAdmin Service has not been specified.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Get-AdminDBConnection

Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True  
Get the database connection string for the DelegatedAdmin Service.

# Get-AdminDBSchema

May 10, 2016

Gets a script that creates the DelegatedAdmin Service database schema for the specified data store.

## Syntax

```
Get-AdminDBSchema [-DatabaseName <String>] [-ServiceGroupName <String>] [-ScriptType <ScriptTypes>] [-LocalDatabase] [-Sid <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets SQL scripts that can be used to create a new DelegatedAdmin Service database schema, add a new DelegatedAdmin Service to an existing site, remove a DelegatedAdmin Service from a site, or create a database server logon for a DelegatedAdmin Service. If no Sid parameter is provided, the scripts obtained relate to the currently selected DelegatedAdmin Service instance, otherwise the scripts relate to DelegatedAdmin Service instance running on the machine identified by the Sid provided. When obtaining the Evict script, a Sid parameter must be supplied. The current service instance is that on the local machine, or that explicitly specified by the last usage of the -AdminAddress parameter to a DelegatedAdmin SDK cmdlet. The service instance used to obtain the scripts does not need to be a member of a site or to have had its database connection configured. The database scripts support only Microsoft SQL Server, or SQL Server Express, and require Windows integrated authentication to be used. They can be run using SQL Server's SQLCMD utility, or by copying the script into an SQL Server Management Studio (SSMS) query window and executing the query. If using SSMS, the query must be executed in 'SMDCMD mode'. The ScriptType parameter determines which script is obtained. If ScriptType is not specified, or is FullDatabase, the script contains:

- o Creation of service schema
- o Creation of database server logon
- o Creation of database user
- o Addition of database user to DelegatedAdmin Service roles

If ScriptType is Instance, the returned script contains:

- o Creation of database server logon
- o Creation of database user
- o Addition of database user to DelegatedAdmin Service roles

If ScriptType is Evict, the returned script contains:

- o Removal of DelegatedAdmin Service instance from database
- o Removal of database user

If ScriptType is Login, the returned script contains:

- o Creation of database server logon only

If the service uses two data stores they can exist in the same database. You do not need to configure a database before

using this command.

## Related topics

[Set-AdminDBConnection](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database for which the schema will be generated.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ServiceGroupName**<String>

Specifies the name of the service group to be used when creating the database schema. The service group is a collection of all the DelegatedAdmin services that share the same database instance and are considered equivalent; that is, all the services within a service group can be used interchangeably.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ScriptType**<ScriptTypes>

Specifies the type of database script returned. Available script types are:

### Database

Returns a full database script that can be used to create a database schema for the DelegatedAdmin Service in a database instance that does not already contain a schema for this service. The DatabaseName and ServiceGroupName parameters must be specified to create a script of this type.

### Instance

Returns a permissions script that can be used to add further DelegatedAdmin services to an existing database instance that already contains the full DelegatedAdmin service schema, associating the services to the Service Group. The Sid parameter can optionally be specified to create a script of this type.

### Login

Returns a database logon script that can be used to add the required logon accounts to an existing database instance that contains the DelegatedAdmin Service schema. This is used primarily when creating a mirrored database environment. The DatabaseName parameter must be specified to create a script of this type.

#### Evict

Returns a script that can be used to remove the specified DelegatedAdmin Service from the database entirely. The DatabaseName and Sid parameters must be specified to create a script of this type.

Required?	false
Default Value	Database
Accept Pipeline Input?	false

#### **-LocalDatabase**<SwitchParameter>

Specifies whether the database script is to be used in a database instance run on the same controller as other services in the service group. Including this parameter ensures the script creates only the required permissions for local services to access the database schema for DelegatedAdmin services.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Sid**<String>

Specifies the SID of the controller on which the DelegatedAdmin Service instance to remove from the database is running.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
-----------	-------



Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

### Systemstring

A string containing the required SQL script for application to a database.

## Notes

The scripts returned support Microsoft SQL Server Express Edition, Microsoft SQL Server Standard Edition, and Microsoft SQL Server Enterprise Edition databases only, and are generated on the assumption that integrated authentication will be used.

If the ScriptType parameter is not included or set to 'FullDatabase', the full database script is returned, which will:

Create the database schema.

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist).

If the ScriptType parameter is set to 'Instance', the script will:

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist) and associate it with a user.

If the ScriptType parameter is set to 'Login', the script will:

Create the logon (providing the schema does not already exist) and associate it with a pre-existing user of the same name.

If the LocalDatabase parameter is included, the NetworkService account will be added to the list of accounts permitted to access the database. This is required only if the database is run on a controller.

If the command fails, the following errors can be returned.

### Error Codes

-----

#### GetSchemasFailed

The database schema could not be found.

#### ActiveDirectoryAccountResolutionFailed

The specified Active Directory account or Group could not be found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-AdminDBSchema -DatabaseName MyDB -ServiceGroupName MyServiceGroup > c:\AdminSchema.sql
Get the full database schema for site data store of the DelegatedAdmin Service and copy it to a file called
'c:\AdminSchema.sql'.
```

This script can then be used to create the schema in a pre-existing database named 'MyDB' that does not already contain a DelegatedAdmin Service site schema.

----- **EXAMPLE 2** -----

```
c:\PS>Get-AdminDBSchema -DatabaseName MyDB -scriptType Login > c:\DelegatedAdminLogins.sql
Get the logon scripts for the DelegatedAdmin Service.
```

# Get-AdminDBVersionChangeScript

May 10, 2016

Gets a script that updates the DelegatedAdmin Service database schema.

## Syntax

```
Get-AdminDBVersionChangeScript -DatabaseName <String> -TargetVersion <Version> [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Returns a database script that can be used to upgrade or downgrade the site or secondary schema for the DelegatedAdmin Service from the current schema version to a different version.

## Related topics

[Get-AdminInstalledDBVersion](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database instance to which the update applies.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-TargetVersion**<Version>

Specifies the version of the database you want to update to.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

--	--

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.Management.Automation.PSObject

A PSObject containing the required SQL script for application to a database.

## Notes

The PSObject returned by this cmdlet contains the following properties:

-- Script The raw text of the SQL script to apply the update, or null in the case when no upgrade path to the specified target version exists.

-- NeedExclusiveAccess Indicates whether all services in the service group must be shut down during the update or not.

-- CanUndo Indicates whether the generated script allows the updated schema to be reverted to the state prior to the update.

Scripts to update the schema version are stored in the database so any service in the service group can obtain these scripts. Extreme caution should be exercised when using update scripts. Citrix recommends backing up the database before attempting to upgrade the schema. Database update scripts may require exclusive use of the schema and so may not be able to execute while any DelegatedAdmin services are running. However, this depends on the specific update being carried out.

After a schema update has been carried out, services that require the previous version of the schema may cease to operate. The ServiceState parameter reported by the Get-AdminServiceStatus command provides information about service compatibility. For example, if the schema has been upgraded to a more recent version that a service cannot use, the service reports "DBNewerVersionThanService".

If the command fails, the following errors can be returned.

## Error Codes

-----

### NoOp

The operation was successful but had no effect.

### NoDBConnections

The database connection string for the DelegatedAdmin Service has not been specified.

### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> $update = Get-AdminDBVersionChangeScript -DatabaseName MyDb -TargetVersion 1.0.75.0
```

```
C:\PS> $update.Script > update_75.sql
```

Gets an SQL update script to update the current schema to version 1.0.75.0. The resulting update\_75.sql script is suitable for direct use with the SQL Server SQLCMD utility.

# Get-AdminEffectiveAdministrator

May 10, 2016

Retrieve the effective administrator objects for a user.

## Syntax

```
Get-AdminEffectiveAdministrator [-Name] <String> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This command determines what groups the specified user belongs to and retrieves the matching administrator records. It includes the set of rights that would be granted to the user if he or she used the system.

As this command uses Active Directory to determine what groups the user has, the caller must have the ability to read this information from Active Directory.

Only enabled administrator records are returned.

## Related topics

[Get-AdminAdministrator](#)

## Parameters

**-Name**<String>

User name or SID of user to query

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

string User name or SID of user to query

## Return Values

Citrix.DelegatedAdmin.Sdk.Administrator

Administrator records matching the specified user

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Get-AdminEffectiveAdministrator MYDOMAIN\testuser
```

Retrieve the administrator records matching user 'testuser'.

# Get-AdminEffectiveRight

May 10, 2016  
Gets the set of Right objects associated with the current user.

## Syntax

Get-AdminEffectiveRight [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The Get-AdminEffectiveRight cmdlet returns the effective rights of the current user. This is the union of all rights of the enabled administrators that the current user matches, taking into account Active Directory group membership.

## Related topics

[Get-AdminAdministrator](#)

[Add-AdminRight](#)

[Remove-AdminRight](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

Citrix.DelegatedAdmin.Sdk.Right

The Rights associated with the current user

## Examples

----- **EXAMPLE 1** -----

C:\PS> Get-AdminEffectiveRight  
Return the effective rights for the current user.





# Get-AdminInstalledDBVersion

May 10, 2016

Gets a list of all available database schema versions for the DelegatedAdmin Service.

## Syntax

```
Get-AdminInstalledDBVersion [-Upgrade] [-Downgrade] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the current version of the DelegatedAdmin Service database schema, if no flags are set, otherwise returns versions for which upgrade or downgrade scripts are available and have been stored in the database.

## Related topics

## Parameters

**-Upgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be updated should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Downgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be reverted should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

### System.Version

The Get-AdminInstalledDbVersion command returns objects containing the new definition of the DelegatedAdmin Service database schema version.

Major <Integer>

Minor <Integer>

Build <Integer>

Revision <Integer>

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

Both the Upgrade and Downgrade flags were specified.

#### NoOp

The operation was successful but had no effect.

#### NoDBConnections

The database connection string for the DelegatedAdmin Service has not been specified.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-AdminInstalledDBVersion
```

Major	Minor	Build	Revision
-------	-------	-------	----------

-----	-----	-----	-----
-------	-------	-------	-------

5	6	0	0
---	---	---	---

Get the currently installed version of the DelegatedAdmin Service database schema.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-AdminInstalledDBVersion -Upgrade
```

Major	Minor	Build	Revision
-------	-------	-------	----------

-----	-----	-----	-----
-------	-------	-------	-------

6	0	0	0
---	---	---	---

Get the versions of the DelegatedAdmin Service database schema for which upgrade scripts are supplied.

# Get-AdminPermission

May 10, 2016  
Gets permissions configured for the site.

## Syntax

```
Get-AdminPermission [[-Name] <String>] [-Id <String>] [-Description <String>] [-GroupId <String>] [-GroupName <String>] [-Metadata <String>] [-Operation <String>] [-ReadOnly <Boolean>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieves permission matching the specified criteria. If no parameters are specified this cmdlet enumerates all permissions.

Permissions are configured using the Import-AdminRoleConfiguration command, and are used to represent collections of operations that are needed to perform a particular task. These permissions are presented in Citrix Studio when configuring roles.

Permissions can also have metadata associated with them.

See about\_Admin\_Filtering for information about advanced filtering options.

## Related topics

- [Add-AdminPermission](#)
- [Remove-AdminPermission](#)
- [Get-AdminRole](#)
- [Get-AdminPermissionGroup](#)
- [Test-AdminAccess](#)

## Parameters

**-Name**<String>

Gets permissions with the specified name (localized)

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Id**<String>

Gets permissions with the specified identifier.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Description**<String>

Gets permissions with the specified description.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-GroupId**<String>

Gets permissions that are a member of the specified permission group (by group id).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-GroupName**<String>

Gets permissions that are a member of the specified permission group (by group name).

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Operation**<String>

Gets permissions that contain a specific operation.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReadOnly**<Boolean>

Gets permissions with the specified value for the ReadOnly flag.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Admin\\_Filtering](#) for details.

--	--

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Admin\_Filtering for details.

Required?	false
-----------	-------



Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.DelegatedAdmin.Sdk.Permission

Get-AdminPermission returns an object for each matching permission.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Get-AdminPermission -Name *Edit*
```

Finds all permissions with 'Edit' in their names.

##### ----- **EXAMPLE 2** -----

```
C:\PS> Get-AdminPermission -Operation "Broker:SetCatalog"
C:\PS> Get-AdminPermission -Filter { Operations -contains "Broker:SetCatalog" -or Operations -contains "Broker:NewCatalog" }
```

Finds permissions that contain specific operations, with the -Filter form needed to match multiple values.

# Get-AdminPermissionGroup

May 10, 2016

Gets permission groups configured for the site.

## Syntax

```
Get-AdminPermissionGroup [[-Name] <String>] [-Id <String>] [-Metadata <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieves permission groups matching the specified criteria. If no parameters are specified this cmdlet enumerates all permission groups.

Permission groups are configured using the Import-AdminRoleConfiguration command, and are primarily used to store the localized name for a group of permissions. Permission groups can also have metadata associated with them.

See about\_Admin\_Filtering for information about advanced filtering options.

## Related topics

[Import-AdminRoleConfiguration](#)

[Get-AdminPermission](#)

## Parameters

**-Name**<String>

Gets permission groups matching the given name. This is the localized name of the group.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Id**<String>

Gets permission groups with the given identifier. This is the non-localized identifier used to associate permissions with permission groups.

Required?	false
Default Value	
Accept Pipeline Input?	

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Admin\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
-----------	-------

Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Admin\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

Citrix.DelegatedAdmin.Sdk.PermissionGroup

Get-AdminPermissionGroup returns an object for each matching permission group.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-AdminPermissionGroup -Name *s
```

Finds all permission groups that end with the letter 's'.

### ----- EXAMPLE 2 -----

```
C:\PS> $list = @("Hosts","Other")
```

```
C:\PS> Get-AdminPermissionGroup -Filter { Id -in $list } -SortBy Name
```

Retrieve two specific permission group objects with the matching identifiers.

# Get-AdminRevision

May 10, 2016

Gets the current revision of the delegated administration configuration data.

## Syntax

```
Get-AdminRevision [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Delegated Administration Service maintains a revision number that is incremented whenever its configuration is changed. This command retrieves the current revision number.

## Related topics

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

System.Int32

The configuration revision number

## Notes

If Int32.MaxValue is reached the value of the revision number will wrap around to Int32.MinValue.

In some cases the value may be incremented by more than one.

## Examples

----- **EXAMPLE 1** -----

```
C:\PS> Get-AdminRevision
```

Retrieve the current revision number.



# Get-AdminRole

May 10, 2016  
Gets roles configured for this site.

## Syntax

```
Get-AdminRole [-Name] <String> [-Id <Guid>] [-BuiltIn <Boolean>] [-Description <String>] [-Metadata <String>] [-Permission <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieves roles matching the specified criteria. If no parameters are specified, this cmdlet enumerates all roles.

See [about\\_Admin\\_Filtering](#) for information about advanced filtering options.

## Related topics

- [New-AdminRole](#)
- [Set-AdminRole](#)
- [Rename-AdminRole](#)
- [Remove-AdminRole](#)
- [Set-AdminRoleMetadata](#)
- [Remove-AdminRoleMetadata](#)

## Parameters

**-Name**<String>

Gets roles with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Id**<Guid>

Gets the role with the specified identifier.

Required?	false



Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-BuiltIn**<Boolean>

Gets roles with the specified value of the BuiltIn flag.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Description**<String>

Gets roles with the specified description.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Permission**<String>

Gets roles that contain a specific permission.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount<SwitchParameter>**

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Admin\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount<Int32>**

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip<Int32>**

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Admin\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

## Return Values

Citrix.DelegatedAdmin.Sdk.Role

Get-AdminRole returns an object for each matching role.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Get-AdminRole -Id 20852cdf-f527-4953-ba6e-e7545217122d
```

Gets the details of the role with the specific id.

### ----- **EXAMPLE 2** -----

```
C:\PS> Get-AdminRole -BuiltIn $false
```

List all custom roles.

# Get-AdminRoleConfiguration

May 10, 2016

Gets role configurations for this site.

## Syntax

```
Get-AdminRoleConfiguration [[-Name] <String>] [-Id <Guid>] [-Locale <String>] [-Priority <Int32>] [-Version <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieves role configurations matching the specified criteria. If no parameters are specified, this cmdlet enumerates and returns all role configurations.

Role configurations are part of the product configuration and define what permissions, permission groups, and built-in roles the product has. This cmdlet also provides the mapping of permissions to operations.

## Related topics

[Import-AdminRoleConfiguration](#)

## Parameters

**-Name**<String>

Gets role configurations matching the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Id**<Guid>

Gets role configurations with the specified id.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Locale**<String>

Gets role configurations with the specified locale. Role configurations usually have a consistent locale.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Priority**<Int32>

Gets role configurations with the specified priority.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Version**<String>

Gets role configurations with the matching version number.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Admin\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Admin\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Input Type

None You cannot pipe input into this cmdlet.

### Return Values

Citrix.DelegatedAdmin.Sdk.RoleConfiguration

Get-AdminRoleConfiguration returns an object for each matching role configuration.

### Notes

This command is supplied for infrastructure purposes only and is not intended for public use.

### Examples

#### ----- **EXAMPLE 1** -----

```
C:\PS> Get-AdminRoleConfiguration -Name Director
```

Retrieve the role configuration for the Citrix Director component.



# Get-AdminScope

May 10, 2016

Gets scopes configured for this site.

## Syntax

```
Get-AdminScope [[-Name] <String>] [-Id <Guid>] [-BuiltIn <Boolean>] [-Description <String>] [-Metadata <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieves scopes matching the specified criteria. If no parameters are specified this cmdlet enumerates all scopes.

There is one special built-in scope, the 'All' scope.

To determine what objects are currently in a scope, use the Get-<Prefix>ScopedObject from each of the relevant PowerShell snap-ins.

See about\_Admin\_Filtering for information about advanced filtering options.

## Related topics

[New-AdminScope](#)

[Set-AdminScope](#)

[Rename-AdminScope](#)

[Remove-AdminScope](#)

[Set-AdminScopeMetadata](#)

[Remove-AdminScopeMetadata](#)

## Parameters

**-Name**<String>

Gets scopes with the specified name.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-Id**<Guid>

Gets the scope with the specified identifier.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-BuiltIn**<Boolean>

Gets scopes with the specified value of the BuiltIn flag.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Description**<String>

Gets scopes with the specified description.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Admin\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0

Accept Pipeline Input?	false
------------------------	-------

### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Admin\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Input Type

None You cannot pipe input into this cmdlet.

### Return Values

Citrix.DelegatedAdmin.Sdk.Scope

Get-AdminScope returns an object for each matching scope.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Get-AdminScope -Name *Sales*
```

List all scopes that contain the word 'Sales'.

### ----- EXAMPLE 2 -----

```
C:\PS> Get-AdminScope -Id 21862daf-e529-4553-ba6e-f7543217111e
```

Gets the details of the scope with the specific id.

# Get-AdminService

May 10, 2016

Gets the service record entries for the DelegatedAdmin Service.

## Syntax

```
Get-AdminService [-Metadata <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns instances of the DelegatedAdmin Service that the service publishes. The service records contain account security identifier information that can be used to remove each service from the database.

A database connection for the service is required to use this command.

## Related topics

### Parameters

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Admin\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

**-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

**-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Admin\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.DelegatedAdmin.Sdk.Service

The Get-AdminServiceInstance command returns an object containing the following properties.

Uid <Integer>

Specifies the unique identifier for the service in the group. The unique identifier is an index number.

ServiceHostId <Guid>

Specifies the unique identifier for the service instance.

DNSName <String>

Specifies the domain name of the host on which the service runs.

MachineName <String>

Specifies the short name of the host on which the service runs.

CurrentState <Citrix.Fma.Sdk.ServiceCore.ServiceState>

Specifies whether the service is running, started but inactive, stopped, or failed.



LastStartTime <DateTime>

Specifies the date and time at which the service was last restarted.

LastActivityTime <DateTime>

Specifies the date and time at which the service was last stopped or restarted.

OSType

Specifies the operating system installed on the host on which the service runs.

OSVersion

Specifies the version of the operating system installed on the host on which the service runs.

ServiceVersion

Specifies the version number of the service instance. The version number is a string that reflects the full build version of the service.

DatabaseUserName <string>

Specifies for the service instance the Active Directory account name with permissions to access the database. This will be either the machine account or, if the database is running on a controller, the NetworkService account.

Sid <string>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

ActiveSiteServices <string[]>

Specifies the names of active site services currently running in the service. Site services are components that perform long-running background processing in some services. This field is empty for services that do not contain site services.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

UnknownObject

One of the specified objects was not found.

PartialData

Only a subset of the available data was returned.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

CouldNotQueryDatabase

The query required to get the database was not defined.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-AdminService
```

```
Uid : 1
ServiceHostId : aef6f464-f1ee-4042-a523-66982e0cecd0
DNSName : MyServer.company.com
MachineName : MYSERVER
CurrentState : On
LastStartTime : 04/04/2011 15:25:38
LastActivityTime : 04/04/2011 15:33:39
OSType : Win32NT
OSVersion : 6.1.7600.0
ServiceVersion : 5.1.0.0
DatabaseUserName : NT AUTHORITY\NETWORK SERVICE
SID : S-1-5-21-2316621082-1546847349-2782505528-1165
ActiveSiteServices : {MySiteService1, MySiteService2...}
```

Get all the instances of the DelegatedAdmin Service running in the current service group.

# Get-AdminServiceAddedCapability

May 10, 2016

Gets any added capabilities for the DelegatedAdmin Service on the controller.

## Syntax

```
Get-AdminServiceAddedCapability [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables updates to the DelegatedAdmin Service on the controller to be detected.

You do not need to configure a database connection before using this command.

## Related topics

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.String

String containing added capabilities.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-AdminServiceAddedCapability
```

Get the added capabilities of the DelegatedAdmin Service.

# Get-AdminServiceInstance

May 10, 2016

Gets the service instance entries for the DelegatedAdmin Service.

## Syntax

```
Get-AdminServiceInstance [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns service interfaces published by the instance of the DelegatedAdmin Service. Each instance of a service publishes multiple interfaces with distinct interface types, and each of these interfaces is represented as a ServiceInstance object. Service instances can be used to register the service with a central configuration service so that other services can use the functionality.

You do not need to configure a database connection to use this command.

## Related topics

## Parameters

**-AdminAddress** <String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.DelegatedAdmin.Sdk.ServiceInstance

The Get-AdminServiceInstance command returns an object containing the following properties.

ServiceGroupUid <Guid>

Specifies the unique identifier for the service group of which the service is a member.

ServiceGroupName <String>

Specifies the name of the service group of which the service is a member.

ServiceInstanceUID <Guid>

Specifies the unique identifier for registered service instances, which are service instances held by and obtained from a

central configuration service. Unregistered service instances do not have unique identifiers.

ServiceType <String>

Specifies the service instance type. For this service, the service instance type is always Admin.

Address

Specifies the address of the service instance. The address can be used to access the service and, when registered in the central configuration service, can be used by other services to access the service.

Binding

Specifies the binding type that must be used to communicate with the service instance. In this release of XenDesktop, the binding type is always 'wcf\_HTTP\_kerb'. This indicates that the service provides a Windows Communication Foundation endpoint that uses HTTP binding with integrated authentication.

Version

Specifies the version of the service instance. The version number is used to ensure that the correct versions of the services are used for communications.

ServiceAccount <String>

Specifies the Active Directory account name for the machine on which the service instance is running. The account name is used to provide information about the permissions required for interservice communications.

ServiceAccountSid <String>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

InterfaceType <String>

Specifies the interface type. Each service can provide multiple service instances, each for a different purpose, and the interface defines the purpose. Available interfaces are:

SDK - for PowerShell operations

InterService - for operations between different services

Peer - for communications between services of the same type

Metadata <Citrix.DelegatedAdmin.Sdk.Metadata[]>

The collection of metadata associated with registered service instances, which are service instances held by and obtained from a central configuration service. Metadata is not stored for unregistered service instances.

Notes

If the command fails, the following errors can be returned.

Error Codes

-----

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-AdminServiceInstance
```

```
Address : http://MyServer.com:80/Citrix/DelegatedAdminContract
Binding : wcf_HTTP_kerb
InterfaceType : SDK
Metadata :
MetadataMap :
ServiceAccount : ENG\MyAccount$
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164
ServiceGroupName : MyServiceGroup
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000
ServiceType : Admin
Version : 1
```

```
Address : http://MyServer.com:80/Citrix/DelegatedAdminContract/IServiceApi
Binding : wcf_HTTP_kerb
InterfaceType : InterService
Metadata :
MetadataMap :
```

ServiceAccount : ENG\MyAccount  
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164  
ServiceGroupName : MyServiceGroup  
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d  
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000  
ServiceType : Admin  
Version : 1

Get all instances of the DelegatedAdmin Service running on the specified machine. For remote services, use the AdminAddress parameter to define the service for which the interfaces are required. If the AdminAddress parameter has not been specified for the runspace, service instances running on the local machine are returned.



# Get-AdminServiceStatus

May 10, 2016

Gets the current status of the DelegatedAdmin Service on the controller.

## Syntax

```
Get-AdminServiceStatus [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables the status of the DelegatedAdmin Service on the controller to be monitored. If the service has multiple data stores it will return the overall state as an aggregate of all the data store states. For example, if the site data store status is OK and the secondary data store status is DBUnconfigured then it will return DBUnconfigured.

## Related topics

[Set-AdminDBConnection](#)

[Test-AdminDBConnection](#)

[Get-AdminDBConnection](#)

[Get-AdminDBSchema](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Get-AdminServiceStatus command returns an object containing the status of the DelegatedAdmin Service together with extra diagnostics information.

DBUnconfigured

The DelegatedAdmin Service does not have a database connection configured.

DBRejectedConnection

The database rejected the logon attempt from the DelegatedAdmin Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

#### InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the DelegatedAdmin Service schema has not been added to the database.

#### DBNotFound

The specified database could not be located with the configured connection string.

#### DBNewerVersionThanService

The version of the DelegatedAdmin Service currently in use is incompatible with the version of the DelegatedAdmin Service schema on the database. Upgrade the DelegatedAdmin Service to a more recent version.

#### DBOlderVersionThanService

The version of the DelegatedAdmin Service schema on the database is incompatible with the version of the DelegatedAdmin Service currently in use. Upgrade the database schema to a more recent version.

#### DBVersionChangeInProgress

A database schema upgrade is currently in progress.

#### OK

The DelegatedAdmin Service is running and is connected to a database containing a valid schema.

#### Failed

The DelegatedAdmin Service has failed.

#### Unknown

(0) The service status cannot be determined.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-AdminServiceStatus
```

DBUnconfigured

Get the current status of the DelegatedAdmin Service.

# Import-AdminRoleConfiguration

May 10, 2016

Imports role configuration data into the Delegated Administration Service.

## Syntax

```
Import-AdminRoleConfiguration [-Path] <String> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Import-AdminRoleConfiguration -Content <String> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

This command is intended for use by Citrix Studio to import definitions, roles, permissions, and their mappings to operations.

The supplied configuration requires a digital signature; this is used to validate the integrity of the configuration.

## Related topics

[Get-AdminRoleConfiguration](#)

## Parameters

**-Path**<String>

The path to the file containing the role configuration data.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Content**<String>

The content of the role configuration data.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Force**<SwitchParameter>

Allows older versions of role configuration to replace newer versions.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

None

#### Notes

This command is supplied for infrastructure purposes only and is not intended for public use.

#### Examples

----- **EXAMPLE 1** -----

```
C:\PS> Import-AdminRoleConfiguration -Path 'C:\MyAdminConfig.xml'
```

Imports the contents of 'C:\MyAdminConfig.xml' to the Delegated Administration Service.

# New-AdminAdministrator

May 10, 2016

Adds a new administrator to the site.

## Syntax

```
New-AdminAdministrator [-Name] <String> [-Enabled <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
New-AdminAdministrator -Sid <String> [-Enabled <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

New-AdminAdministrator creates a new administrator object in the site. Once a new administrator has been created you can assign rights (role and scope pairs) to the administrator.

Administrator objects are used to determine what rights, and therefore what permissions a particular Active Directory user has through the various SDKs and consoles of the site.

When the Enabled flag of an administrator is set to false, any rights of the administrator are ignored by the system when performing permission checks.

## Related topics

[Get-AdminAdministrator](#)

[Set-AdminAdministrator](#)

[Remove-AdminAdministrator](#)

[Set-AdminAdministratorMetadata](#)

[Remove-AdminAdministratorMetadata](#)

## Parameters

**-Name**<String>

Specifies the user or group name in Active Directory that this administrator corresponds to.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Sid**<String>

Specifies the SID (security identifier) of the user in Active Directory that this administrator corresponds to.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Enabled**<Boolean>

Specifies whether the new administrator starts off enabled or not.

Required?	false
Default Value	True
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false



## Input Type

None You cannot pipe input into this cmdlet.

## Return Values

Citrix.DelegatedAdmin.Sdk.Administrator

The newly created administrator.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> New-AdminAdministrator -Name DOMAIN\TestUser
```

Creates a new administrator object for user "DOMAIN\TestUser". It defaults to enabled.

### ----- **EXAMPLE 2** -----

```
C:\PS> New-AdminAdministrator -Sid S-1-2-34-1234567890-1234567890-1234567890-123
```

Creates a new administrator object for user with SID "S-1-2-34-1234567890-1234567890-1234567890-123". It defaults to enabled.

# New-AdminRole

May 10, 2016  
Adds a new custom role to the site.

## Syntax

New-AdminRole [-Name] <String> [-Description <String>] [-LoggingId <Guid>] [-AdminAddress <String>]  
[<CommonParameters>]

## Detailed Description

New-AdminRole adds a new custom role object to the site. Once a new role has been created, you can add permissions to the role which define what operations the role conveys.

Roles represent a job function, such as 'help desk administrator', and contain a list of permissions that are required to perform that job function.

To assign a role to an administrator, you combine it with a scope which indicates what objects the role can operate on. This pair (also known as a 'right') can then be assigned to an administrator. See Add-AdminRight for further details.

The identifier of the new role is chosen automatically, and custom roles created with this cmdlet always have their BuiltIn flag set to false.

You cannot modify built-in roles, and only some license editions support custom roles.

## Related topics

- [Get-AdminRole](#)
- [Set-AdminRole](#)
- [Rename-AdminRole](#)
- [Remove-AdminRole](#)
- [Set-AdminRoleMetadata](#)
- [Remove-AdminRoleMetadata](#)
- [Add-AdminPermission](#)
- [Add-AdminRight](#)

## Parameters

**-Name**<String>

Specifies the name of the role. Each role in a site must have a unique name.

Required?	true

Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Description**<String>

Specifies the description of the role.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.DelegatedAdmin.Sdk.Role

The newly created role.

## Notes

Roles are created without any permissions. Use the Add-AdminPermission to add permissions.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> New-AdminRole -Name Supervisor -Description "My custom supervisor role"
```

```
C:\PS> $list = Get-AdminRole 'Help Desk Administrator' | Select -Expand Permissions
```

```
C:\PS> Add-AdminPermission -Role Supervisor -Permission $list
```

```
C:\PS> Add-AdminPermission -Role Supervisor -Permission $extras
```

```
C:\PS> Add-AdminRight -Administrator DOMAIN\TestUser -Role Supervisor -All
```

Creates a new role called 'Supervisor', and then copies the permissions from the help desk role and adds some extras. Then gives this role (with the all scope) to user 'TestUser'.

# New-AdminScope

May 10, 2016

Adds a new scope to the site.

## Syntax

```
New-AdminScope [-Name] <String> [-Description <String>] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

New-AdminScope adds a new scope object to the site.

A scope represents a collection of objects. Scopes are used to group objects in a way that is relevant to the organization; for example, the set of delivery groups used by the Sales team.

You can create objects in particular scopes by specifying the -Scope parameter of a New- cmdlet for an object that can be scoped. You can then modify the contents of a scope with Add-<Noun>Scope and Remove-<Noun>Scope cmdlets from the corresponding PowerShell snap-ins.

To assign a scope to an administrator, combine it with a role and then assign this pair (also known as a 'right') to an administrator. See Add-AdminRight for further details.

The identifier of the new scope is chosen automatically.

## Related topics

[Get-AdminScope](#)

[Set-AdminScope](#)

[Rename-AdminScope](#)

[Remove-AdminScope](#)

[Set-AdminScopeMetadata](#)

[Remove-AdminScopeMetadata](#)

[Add-AdminRight](#)

## Parameters

**-Name**<String>

Specifies the name of the scope. Each scope in a site must have a unique name.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-Description**<String>

Specifies the description of the scope.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

None You cannot pipe input into this cmdlet.

#### Return Values

Citrix.DelegatedAdmin.Sdk.Scope

The newly created scope.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> New-AdminScope -Name Sales -Description "Sales department scope"
```

```
C:\PS> Add-HypHypervisorConnectionScope -HypervisorConnectionName XenServer2 -Scope Sales
```

```
C:\PS> Add-AdminRight -Administrator DOMAIN\TestUser -Role Hosting -Scope Sales
```

Creates a new scope called 'Sales', adds a hypervisor connection object to the scope, and then assigns the right to use the hosting role on the Sales scope to the 'TestUser' administrator.

# Remove-AdminAdministrator

May 10, 2016

Removes administrators from the site.

## Syntax

```
Remove-AdminAdministrator [-InputObject] <Administrator[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AdminAdministrator -Sid <String[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-AdminAdministrator [-Name] <String[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-AdminAdministrator cmdlet deletes administrators from the site.

## Related topics

[New-AdminAdministrator](#)

[Get-AdminAdministrator](#)

[Set-AdminAdministrator](#)

[Set-AdminAdministratorMetadata](#)

[Remove-AdminAdministratorMetadata](#)

## Parameters

**-InputObject** <Administrator[]>

Specifies the administrators to delete.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name** <String[]>

Specifies the name of the administrator to delete.

Required?	true
Default Value	



Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-Sid<String[]>**

Specifies the SID of the administrator to delete.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.DelegatedAdmin.Sdk.Administrator You can pipe the administrators to be deleted into this command.

#### Return Values

None

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Remove-AdminAdministrator DOMAIN\TestUser
```

Remove the administrator called "DOMAIN\TestUser".

### ----- **EXAMPLE 2** -----

```
C:\PS> Get-AdminAdministrator -Enabled $false | Remove-AdminAdministrator
```

Remove all disabled administrators.

# Remove-AdminAdministratorMetadata

May 10, 2016

Removes metadata from the given Administrator.

## Syntax

```
Remove-AdminAdministratorMetadata -AdministratorSid <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-AdminAdministratorMetadata -AdministratorSid <String> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-AdminAdministratorMetadata [-AdministratorName] <String> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-AdminAdministratorMetadata [-AdministratorName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-AdminAdministratorMetadata [-InputObject] <Administrator[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-AdminAdministratorMetadata [-InputObject] <Administrator[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Administrator.

## Related topics

[Set-AdminAdministratorMetadata](#)

## Parameters

**-AdministratorName**<String>

Name of the Administrator

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Administrator[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-AdministratorSid<String>**

Sid of the Administrator

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name<String>**

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-

LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-AdminAdministrator | % { Remove-AdminAdministratorMetadata -Map $_.MetadataMap }
Remove all metadata from all Administrator objects.
```

# Remove-AdminPermission

May 10, 2016

Remove permissions from the set of permissions of a role.

## Syntax

```
Remove-AdminPermission [-InputObject] <Permission[]> -Role <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AdminPermission [-Permission] <String[]> -Role <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AdminPermission -All -Role <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Remove permissions from the set of permissions that a role maps to.

Any administrator with a right including that role immediately loses the ability to use the operations of the removed permissions.

Duplicate permissions do not produce an error, and permissions that the roles does not already have are skipped (without error).

You cannot modify the permissions of built-in roles.

## Related topics

[Add-AdminPermission](#)

[Get-AdminPermission](#)

[Get-AdminRole](#)

[Get-AdminPermissionGroup](#)

[Test-AdminAccess](#)

## Parameters

**-InputObject** <Permission[]>

Specifies the permissions to remove.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Permission**<String[]>

Specifies the list of permissions to remove (by identifier).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-Role**<String>

Role name or identifier of the role to update.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-All**<SwitchParameter>

Remove all permissions.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false



## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

Citrix.DelegatedAdmin.Sdk.Permission You can pipe a list of permissions to be removed into this command.

## Return Values

None

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS> Remove-AdminPermission -Role MyRole -Permission Global_Read,Logging_Read
```

Remove a couple of specific permissions from the 'MyRole' role.

### ----- **EXAMPLE 2** -----

```
C:\PS> Remove-AdminPermission -Role MyRole -All
```

Remove all permissions from the 'MyRole' role.

# Remove-AdminRight

May 10, 2016

Removes rights from an administrator.

## Syntax

```
Remove-AdminRight -Scope <String> -Role <String> -Administrator <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AdminRight -Role <String> -Administrator <String> [-All] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AdminRight [-InputObject] <Right[]> -Administrator <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

This command removes rights from the specified administrator.

For convenience, you can use the -All parameter to specify the 'All' scope.

Use the Get-AdminAdministrator cmdlet to determine what rights an administrator has.

## Related topics

[Get-AdminAdministrator](#)

[Get-AdminEffectiveRight](#)

[Add-AdminRight](#)

## Parameters

**-InputObject**<Right[]>

Specifies the rights to remove.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Scope**<String>

Specifies the scope name or scope identifier.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	false

#### **-Role**<String>

Specifies the role name or role identifier.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Administrator**<String>

Specifies the name or SID of the administrator.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-All**<SwitchParameter>

Specifies the 'All' scope. This parameter avoids localization issues or having to type the identifier of the 'All' scope.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Input Type

Citrix.DelegatedAdmin.Sdk.Right You can pipe the rights to be removed into this command.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> RemoveAdminRight -Role 'Help Desk Administrator' -Scope London -Administrator DOMAIN\Admin1
Removes the 'Help Desk Administrator' role and 'London' scope from user 'Admin1'
```

#### ----- EXAMPLE 2 -----

```
C:\PS> $admin = Get-AdminAdministrator -Name DOMAIN\Admin
C:\PS> Remove-AdminRight -InputObject $admin.Rights -Administrator DOMAIN\Admin
Removes all rights from administrator 'Admin'.
```

# Remove-AdminRole

May 10, 2016  
Removes a role from the site.

## Syntax

```
Remove-AdminRole [-InputObject] <Role[]> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
Remove-AdminRole [-Id] <Guid[]> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
Remove-AdminRole [-Name] <String[]> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-AdminRole cmdlet deletes roles from the site.

You cannot remove built-in roles.

An error will be produced if the role being removed is currently assigned to an administrator unless you specify the -Force option. When -Force is specified, any rights that reference the role are also removed.

## Related topics

- [New-AdminRole](#)
- [Get-AdminRole](#)
- [Set-AdminRole](#)
- [Rename-AdminRole](#)
- [Set-AdminRoleMetadata](#)
- [Remove-AdminRoleMetadata](#)

## Parameters

**-InputObject**<Role[]>

Specifies the roles to remove (by role object).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Id**<Guid[]>

Specifies the roles to remove (by role id).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Name**<String[]>

Specifies the roles to remove (by role name).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Force**<SwitchParameter>

Allow removal of roles that are still in use.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name

or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

Citrix.DelegatedAdmin.Sdk.Role You can pipe the roles to be deleted into this command.

## Return Values

None

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS> Remove-AdminRole -Name Supervisor
Remove the Supervisor role.
```

### ----- EXAMPLE 2 -----

```
C:\PS> Get-AdminRole -BuiltIn $false | Remove-AdminRole
Attempt to remove all custom roles. This fails if one of the roles is assigned to an administrator.
```

# Remove-AdminRoleMetadata

May 10, 2016

Removes metadata from the given Role.

## Syntax

```
Remove-AdminRoleMetadata [-RoleId] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AdminRoleMetadata [-RoleId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AdminRoleMetadata [-RoleName] <String> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AdminRoleMetadata [-RoleName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AdminRoleMetadata [-InputObject] <Role[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AdminRoleMetadata [-InputObject] <Role[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Role.

## Related topics

[Set-AdminRoleMetadata](#)

## Parameters

**-RoleId**<Guid>

Id of the Role

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-RoleName**<String>

Name of the Role

--	--



Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-InputObject**<Role[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-

LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-AdminRole | % { Remove-AdminRoleMetadata -Map $_.MetadataMap }
Remove all metadata from all Role objects.
```

# Remove-AdminScope

May 10, 2016

Removes a scope from the site.

## Syntax

```
Remove-AdminScope [-InputObject] <Scope[]> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AdminScope [-Id] <Guid[]> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-AdminScope [-Name] <String[]> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Remove-AdminScope cmdlet deletes scopes from the site.

You cannot remove the built-in 'All' scope.

An error will be produced if the scope being removed is currently assigned to an administrator unless you specify the -Force option. When -Force is specified, any rights that reference the scope are also removed.

## Related topics

[New-AdminScope](#)

[Get-AdminScope](#)

[Set-AdminScope](#)

[Rename-AdminScope](#)

[Set-AdminScopeMetadata](#)

[Remove-AdminScopeMetadata](#)

## Parameters

**-InputObject** <Scope[]>

Specifies the scopes to remove (by scope object).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Id** <Guid[]>

Specifies the scopes to remove (by scope id).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Name**<String[]>

Specifies the scopes to remove (by scope name).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Force**<SwitchParameter>

Allow removal of scopes that are still in use.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Input Type

Citrix.DelegatedAdmin.Sdk.Scope You can pipe the scopes to be deleted into this command.

### Return Values

None

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Remove-AdminScope -Name Sales
```

Remove the Sales scope.

#### ----- EXAMPLE 2 -----

```
C:\PS> Get-AdminScope -BuiltIn $false | Remove-AdminScope
```

Attempt to remove all scopes (excluding the built-in 'All' scope). This fails if one of the scopes is assigned to an administrator.

# Remove-AdminScopeMetadata

May 10, 2016  
Removes metadata from the given Scope.

## Syntax

```
Remove-AdminScopeMetadata [-ScopeId] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Remove-AdminScopeMetadata [-ScopeId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Remove-AdminScopeMetadata [-ScopeName] <String> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Remove-AdminScopeMetadata [-ScopeName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Remove-AdminScopeMetadata [-InputObject] <Scope[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Remove-AdminScopeMetadata [-InputObject] <Scope[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Scope.

## Related topics

[Set-AdminScopeMetadata](#)

## Parameters

**-ScopeId**<Guid>

Id of the Scope

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-ScopeName**<String>

Name of the Scope

--	--

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-InputObject**<Scope[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with `@{"name1" = "val1"; "name2" = "val2"}`) or a string dictionary (created with `new-object "System.Collections.Generic.Dictionary[String,String]"`). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-



LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

##### InvalidParameterCombination

The cmdlet parameters are inconsistent.

##### UnknownObject

One of the specified objects was not found.

##### DatabaseError

An error occurred in the service while attempting a database operation.

##### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

##### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

##### PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-AdminScope | % { Remove-AdminScopeMetadata -Map $_.MetadataMap }
Remove all metadata from all Scope objects.
```

# Remove-AdminServiceMetadata

May 10, 2016

Removes metadata from the given Service.

## Syntax

```
Remove-AdminServiceMetadata [-ServiceHostId] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AdminServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AdminServiceMetadata [-InputObject] <Service[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-AdminServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Service.

## Related topics

[Set-AdminServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name**<String>

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-AdminService | % { Remove-AdminServiceMetadata -Map $_.MetadataMap }
```

Remove all metadata from all Service objects.

# Rename-AdminRole

May 10, 2016

Rename a role

## Syntax

```
Rename-AdminRole [-InputObject] <Role> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Rename-AdminRole [-Id] <Guid> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Rename-AdminRole [-Name] <String> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

The Rename-AdminRole cmdlet changes the name of a role.

Role names must be unique, and you cannot modify the name of built-in roles.

## Related topics

[New-AdminRole](#)

[Get-AdminRole](#)

[Set-AdminRole](#)

[Remove-AdminRole](#)

[Set-AdminRoleMetadata](#)

[Remove-AdminRoleMetadata](#)

## Parameters

**-InputObject**<Role>

Specifies the role to rename (by object).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Id**<Guid>

Specifies the role to rename (by id).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Name**<String>

Specifies the role to rename (by name).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-NewName**<String>

Specifies the new name of the role.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-PassThru**<SwitchParameter>

Returns the affected record. By default, this cmdlet does not generate any output.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create



high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.DelegatedAdmin.Sdk.Role You can pipe the role to be renamed into this command.

#### Return Values

None or Citrix.DelegatedAdmin.Sdk.Role

When you use the PassThru parameter, Rename-AdminRole generates a Citrix.DelegatedAdmin.Sdk.Role object. Otherwise, this cmdlet does not generate any output.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Rename-AdminRole -Name Supervisor -NewName HelpDeskLead
Renames the 'Supervisor' role to 'HelpDeskLead'.
```

# Rename-AdminScope

May 10, 2016

Rename a scope

## Syntax

```
Rename-AdminScope [-InputObject] <Scope> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Rename-AdminScope [-Id] <Guid> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Rename-AdminScope [-Name] <String> [-NewName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Rename-AdminScope cmdlet changes the name of a scope.

Scope names must be unique, and you cannot modify the name of the built-in 'All' scope.

## Related topics

[New-AdminScope](#)

[Get-AdminScope](#)

[Set-AdminScope](#)

[Remove-AdminScope](#)

[Set-AdminScopeMetadata](#)

[Remove-AdminScopeMetadata](#)

## Parameters

**-InputObject**<Scope>

Specifies the scope to rename (by object).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Id**<Guid>

Specifies the scope to rename (by id).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Name**<String>

Specifies the scope to rename (by name).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-NewName**<String>

Specifies the new name of the scope.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-PassThru**<SwitchParameter>

Returns the affected record. By default, this cmdlet does not generate any output.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create

high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.DelegatedAdmin.Sdk.Scope You can pipe the scope to be renamed into this command.

#### Return Values

None or Citrix.DelegatedAdmin.Sdk.Scope

When you use the PassThru parameter, Rename-AdminScope generates a Citrix.DelegatedAdmin.Sdk.Scope object. Otherwise, this cmdlet does not generate any output.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> Rename-AdminScope -Name Sales -NewName SalesDesktops
Renames the 'Sales' scope to 'SalesDesktops'.
```

# Reset-AdminServiceGroupMembership

May 10, 2016

Reloads the access permissions and configuration service locations for the DelegatedAdmin Service.

## Syntax

```
Reset-AdminServiceGroupMembership [-ConfigServiceInstance] <ServiceInstance[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Enables you to reload DelegatedAdmin Service access permissions and configuration service locations. The Reset-AdminServiceGroupMembership command must be run on at least one instance of the service type (Admin) after installation and registration with the configuration service. Without this operation, the DelegatedAdmin services will be unable to communicate with other services in the XenDesktop deployment. When the command is run, the services are updated when additional services are added to the deployment, provided that the configuration service is not stopped. The Reset-AdminServiceGroupMembership command can be run again to refresh this information if automatic updates do not occur when new services are added to the deployment. If more than one configuration service instance is passed to the command, the first instance that meets the expected service type requirements is used.

## Related topics

### Parameters

**-ConfigServiceInstance**<ServiceInstance[]>

Specifies the configuration service instance object that represents the service instance for the type 'InterService' that references a configuration service for the deployment.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Input Type

Citrix.DelegatedAdmin.Sdk.ServiceInstance[] Service instances containing a ServiceInstance object that refers to the central configuration service interservice interface can be piped to the Reset-AdminServiceGroupMembership command.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### NoSuitableServiceInstance

None of the supplied service instance objects were suitable for resetting service group membership.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config | Reset-AdminServiceGroupMembership
```

Reset the service group membership for a service in a deployment where the configuration service is configured and running on the same machine as the service.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config -AdminAddress OtherServer.example.com | Reset-AdminServiceGroupmembership
```

Reset the service group membership for a service in a deployment where the configuration service that is configured and running on a machine

named 'OtherServer.example.com'.

# Set-AdminAdministrator

May 10, 2016

Sets the properties of an administrator.

## Syntax

```
Set-AdminAdministrator [-InputObject] <Administrator[]> [-Enabled <Boolean>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-AdminAdministrator -Sid <String[]> [-Enabled <Boolean>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-AdminAdministrator [-Name] <String[]> [-Enabled <Boolean>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The Set-AdminAdministrator cmdlet is used to enable or disable an existing administrator.

You can specify the administrators to modify in a number of ways, by piping in existing objects, by passing existing objects with the InputObject parameter, or by specifying the names or SIDs explicitly.

## Related topics

[New-AdminAdministrator](#)

[Get-AdminAdministrator](#)

[Remove-AdminAdministrator](#)

[Set-AdminAdministratorMetadata](#)

[Remove-AdminAdministratorMetadata](#)

## Parameters

**-InputObject**<Administrator[]>

Specifies the administrators to modify.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String[]>

Specifies the names of the administrators to modify.



Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Sid**<String[]>

Specifies the SIDs of the administrators to modify.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Enabled**<Boolean>

Specifies the new value for the Enabled property.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PassThru**<SwitchParameter>

Returns the affected record. By default, this cmdlet does not generate any output.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Input Type

Citrix.DelegatedAdmin.Sdk.Administrator You can pipe the administrators to be modified into this command.

### Return Values

None or Citrix.DelegatedAdmin.Sdk.Administrator

When you use the PassThru parameter, Set-AdminAdministrator generates a Citrix.DelegatedAdmin.Sdk.Administrator object. Otherwise, this cmdlet does not generate any output.

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Get-AdminAdministrator -Enabled $false | Set-AdminAdministrator -Enabled $true
```

Enable all administrators that are currently disabled.

#### ----- EXAMPLE 2 -----

```
C:\PS> Set-AdminAdministrator -Name DOMAIN\TestUser1,DOMAIN\TestUser2 -Enabled $true
```

Enable two specific users specified by name (TestUser1 and TestUser2).

# Set-AdminAdministratorMetadata

May 10, 2016  
Adds or updates metadata on the given Administrator.

## Syntax

Set-AdminAdministratorMetadata -AdministratorSid <String> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-AdminAdministratorMetadata -AdministratorSid <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-AdminAdministratorMetadata [-AdministratorName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-AdminAdministratorMetadata [-AdministratorName] <String> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-AdminAdministratorMetadata [-InputObject] <Administrator[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-AdminAdministratorMetadata [-InputObject] <Administrator[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Provides the ability for additional custom data to be stored against given Administrator objects.

## Related topics

[Remove-AdminAdministratorMetadata](#)

## Parameters

**-AdministratorName**<String>

Name of the Administrator

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Administrator[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-AdministratorSid<String>**

Sid of the Administrator

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Name<String>**

Specifies the property name of the metadata to be added. The property must be unique for the Administrator specified. The property cannot contain any of the following characters \/:#.\*?=<>|[]{}"

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Value<String>**

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

--	--

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-AdminAdministratorMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Set-AdminAdministratorMetadata -AdministratorSid S-1-5-21-1505241163-3345470479-1241728991-1000 -Name property -Value value
```

Key	Value
---	----
property	value

Add metadata with a name of 'property' and a value of 'value' to the Administrator with the identifier 'S-1-5-21-1505241163-3345470479-1241728991-1000'.

# Set-AdminDBConnection

May 10, 2016  
Configures a database connection for the DelegatedAdmin Service.

## Syntax

Set-AdminDBConnection [-DBConnection] <String> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Configures a connection to a database in which the DelegatedAdmin Service can store its state. The service will attempt to connect and start using the database immediately after the connection is configured. The database connection string is updated to the specified value regardless of whether it is valid or not. Specifying an invalid connection string prevents a service from functioning until the error is corrected.

After a connection is configured, you cannot alter it without first clearing it (by setting the connection to \$null).

You do not need to configure a database connection to use this command.

## Related topics

[Get-AdminServiceStatus](#)

[Get-AdminDBConnection](#)

[Test-AdminDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be used by the DelegatedAdmin Service. Passing in \$null will clear any existing database connection configured.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Force**<SwitchParameter>

If present, allows the local administrator to set the connection string to null when there are problems contacting the database or other services.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Set-AdminDBConnection command returns an object containing the status of the DelegatedAdmin Service together with extra diagnostics information.

DBUnconfigured

The DelegatedAdmin Service does not have a database connection configured.

DBRejectedConnection

The database rejected the logon attempt from the DelegatedAdmin Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the DelegatedAdmin Service schema has not been added to the database.

DBNotFound

The specified database could not be located with the configured connection string.

DBNewerVersionThanService

The version of the DelegatedAdmin Service currently in use is incompatible with the version of the DelegatedAdmin Service schema on the database. Upgrade the DelegatedAdmin Service to a more recent version.

DBOlderVersionThanService

The version of the DelegatedAdmin Service schema on the database is incompatible with the version of the DelegatedAdmin Service currently in use. Upgrade the database schema to a more recent version.

DBVersionChangeInProgress

A database schema upgrade is currently in progress.

OK

The DelegatedAdmin Service is running and is connected to a database containing a valid schema.



## Failed

The DelegatedAdmin Service has failed.

## Unknown

The status of the DelegatedAdmin Service cannot be determined.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidDBConnectionString

The database connection string has an invalid format.

#### DatabaseConnectionDetailsAlreadyConfigured

There was already a database connection configured. After a configuration is set, it can only be set to \$null.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Set-AdminDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

#### OK

Configures a database connection string for the DelegatedAdmin Service.

### ----- EXAMPLE 2 -----

```
c:\PS>Set-AdminDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Configures an invalid database connection string for the DelegatedAdmin Service.

# Set-AdminRole

May 10, 2016  
Set the properties of a role.

## Syntax

Set-AdminRole [-InputObject <Role[]>] [-Description <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>]  
[<CommonParameters>]

Set-AdminRole [-Id] <Guid[]> [-Description <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>]  
[<CommonParameters>]

Set-AdminRole [-Name] <String[]> [-Description <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>]  
[<CommonParameters>]

## Detailed Description

The Set-AdminRole command allows the description of custom roles to be updated. You cannot modify built-in roles.

To modify the permissions of a role, use the Add-AdminPermission and Remove-AdminPermission cmdlets.

To update the metadata associated with a role, use the Set-AdminRoleMetadata and Remove-AdminRoleMetadata cmdlets.

## Related topics

- [New-AdminRole](#)
- [Get-AdminRole](#)
- [Remove-AdminRole](#)
- [Rename-AdminRole](#)
- [Set-AdminRoleMetadata](#)
- [Remove-AdminRoleMetadata](#)
- [Add-AdminPermission](#)
- [Remove-AdminPermission](#)

## Parameters

**-InputObject**<Role[]>

Specifies the roles to update (by object).

Required?	true
Default Value	

Accept Pipeline Input?	true (ByValue)
------------------------	----------------

#### **-Id**<Guid[]>

Specifies the roles to update (by id).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Name**<String[]>

Specifies the roles to update (by name).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-Description**<String>

Supplies the new description value.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PassThru**<SwitchParameter>

Returns the affected record. By default, this cmdlet does not generate any output.

Required?	false
Default Value	False
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Input Type

Citrix.DelegatedAdmin.Sdk.Role You can pipe the roles to be updated into this command.

### Return Values

None or Citrix.DelegatedAdmin.Sdk.Role

When you use the PassThru parameter, Set-AdminRole generates a Citrix.DelegatedAdmin.Sdk.Role object. Otherwise, this cmdlet does not generate any output.

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS> Set-AdminRole -Name Supervisor -Description "Helpdesk supervisor role"
```

Change the description of the 'Supervisor' role.

# Set-AdminRoleMetadata

May 10, 2016  
Adds or updates metadata on the given Role.

## Syntax

```
Set-AdminRoleMetadata [-RoleId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Set-AdminRoleMetadata [-RoleId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Set-AdminRoleMetadata [-RoleName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Set-AdminRoleMetadata [-RoleName] <String> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Set-AdminRoleMetadata [-InputObject] <Role[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Set-AdminRoleMetadata [-InputObject] <Role[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability for additional custom data to be stored against given Role objects.

## Related topics

[Remove-AdminRoleMetadata](#)

## Parameters

**-RoleId**<Guid>

Id of the Role

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-RoleName**<String>

Name of the Role

--	--

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-InputObject**<Role[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the Role specified. The property cannot contain any of the following characters \;#.\*?=<>|[]()"

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-AdminRoleMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

##### InvalidParameterCombination

The cmdlet parameters are inconsistent.

##### UnknownObject

One of the specified objects was not found.

##### DatabaseError

An error occurred in the service while attempting a database operation.

##### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

##### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

##### PermissionDenied

You do not have permission to execute this command.

##### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

##### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

##### CommunicationError

There was a problem communicating with the remote service.

##### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Set-AdminRoleMetadata -RoleId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
---	-----
property	value



Add metadata with a name of 'property' and a value of 'value' to the Role with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Set-AdminScope

May 10, 2016  
Set the properties of a scope.

## Syntax

Set-AdminScope [-InputObject] <Scope[]> [-Description <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>]  
[<CommonParameters>]

Set-AdminScope [-Id] <Guid[]> [-Description <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>]  
[<CommonParameters>]

Set-AdminScope [-Name] <String[]> [-Description <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>]  
[<CommonParameters>]

## Detailed Description

The Set-AdminScope command allows the description of scopes to be updated. You cannot modify the built-in 'All' scope.

To change the contents of a scope, use the Add-<Noun>Scope and Remove-<Noun>Scope cmdlets from the corresponding PowerShell snap-in.

To update the metadata associated with a scope, use the Set-AdminScopeMetadata and Remove-AdminScopeMetadata cmdlets.

## Related topics

- [New-AdminScope](#)
- [Get-AdminScope](#)
- [Remove-AdminScope](#)
- [Rename-AdminScope](#)
- [Set-AdminScopeMetadata](#)
- [Remove-AdminScopeMetadata](#)

## Parameters

**-InputObject**<Scope[]>

Specifies the scopes to update (by object).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Id**<Guid[]>

Specifies the scopes to update (by id).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Name**<String[]>

Specifies the scopes to update (by name).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Description**<String>

Supplies the new description value.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-PassThru**<SwitchParameter>

Returns the affected record. By default, this cmdlet does not generate any output.

Required?	false
Default Value	False
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Input Type

Citrix.DelegatedAdmin.Sdk.Scope You can pipe the scopes to be updated into this command.

### Return Values

None or Citrix.DelegatedAdmin.Sdk.Scope

When you use the PassThru parameter, Set-AdminScope generates a Citrix.DelegatedAdmin.Sdk.Scope object. Otherwise, this cmdlet does not generate any output.

### Examples

#### ----- **EXAMPLE 1** -----

```
C:\PS> Set-AdminScope -Name Sales -Description "Sales department desktops"
```

Change the description of the 'Sales' scope.

# Set-AdminScopeMetadata

May 10, 2016

Adds or updates metadata on the given Scope.

## Syntax

```
Set-AdminScopeMetadata [-ScopeId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-AdminScopeMetadata [-ScopeId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-AdminScopeMetadata [-ScopeName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-AdminScopeMetadata [-ScopeName] <String> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Set-AdminScopeMetadata [-InputObject] <Scope[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Set-AdminScopeMetadata [-InputObject] <Scope[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability for additional custom data to be stored against given Scope objects.

## Related topics

[Remove-AdminScopeMetadata](#)

## Parameters

**-ScopeId**<Guid>

Id of the Scope

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-ScopeName**<String>

Name of the Scope

--	--

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-InputObject**<Scope[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the Scope specified. The property cannot contain any of the following characters \;#. \*?=<> | []()"

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-AdminScopeMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

##### InvalidParameterCombination

The cmdlet parameters are inconsistent.

##### UnknownObject

One of the specified objects was not found.

##### DatabaseError

An error occurred in the service while attempting a database operation.

##### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

##### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

##### PermissionDenied

You do not have permission to execute this command.

##### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

##### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

##### CommunicationError

There was a problem communicating with the remote service.

##### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Set-AdminScopeMetadata -ScopelId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
---	-----
property	value



Add metadata with a name of 'property' and a value of 'value' to the Scope with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Set-AdminServiceMetadata

May 10, 2016

Adds or updates metadata on the given Service.

## Syntax

```
Set-AdminServiceMetadata [-ServiceHostId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-AdminServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-AdminServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-AdminServiceMetadata [-InputObject] <Service[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Allows you to store additional custom data against given Service objects.

## Related topics

[Remove-AdminServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the Service specified. The property cannot contain any of the following characters \/:;#.\*?=<>|[]()''

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.Collections.Generic.Dictionary[String,String]

Set-AdminServiceMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

InvalidParameterCombination

The cmdlet parameters are inconsistent.

UnknownObject

One of the specified objects was not found.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various

reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Set-AdminServiceMetadata -ServiceHostId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
---	----
property	value

Add metadata with a name of 'property' and a value of 'value' to the Service with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Test-AdminAccess

May 10, 2016

Retrieves the scopes where the specified operation is permitted.

## Syntax

```
Test-AdminAccess [-Operation] <String[]> [-Annotate] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This cmdlet evaluates what rights the current user has, and from these determines the scopes where the specified operation is permitted.

Operations are the indivisible unit of functionality that each XenDesktop service can perform, and usually correspond to individual cmdlets.

If you specify the -Annotate option or specify multiple operations to check, the resulting object is annotated with the operation the result relates to.

## Related topics

### Parameters

**-Operation**<String[]>

The operation to query.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Annotate**<SwitchParameter>

Annotates each result with the operation it relates to.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name

or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.DelegatedAdmin.Sdk.ScopeReference

The list of permissible scopes for the specified single operation.PSObject

The list of permissible scopes for each operation. This type of object is returned when the -Annotate option or multiple operations are specified.

## Notes

If the specified operation has unrestricted access a single object is returned representing the 'All' scope with a ScopeId of Guid.Empty (00000000-0000-0000-0000-000000000000).

## Examples

----- **EXAMPLE 1** -----

```
C:\PS> Test-AdminAccess -Operation 'Broker:GetCatalog'
```

Queries the scopes where 'Broker:GetCatalog' is permitted.

----- **EXAMPLE 2** -----

```
C:\PS> Test-AdminAccess -Operation 'Broker:GetCatalog','Broker:GetMachine'
```

Queries the scopes where 'Broker:GetCatalog' or 'Broker:GetMachine' are permitted.

# Test-AdminDBConnection

May 10, 2016  
Tests a database connection for the DelegatedAdmin Service.

## Syntax

Test-AdminDBConnection [-DBConnection] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Tests a connection to the database in which the DelegatedAdmin Service can store its state. The service will attempt to connect to the database without affecting the current connection to the database.

You do not have to clear the connection to use this command.

## Related topics

[Get-AdminServiceStatus](#)

[Get-AdminDBConnection](#)

[Set-AdminDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be tested by the DelegatedAdmin Service.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.



Accept Pipeline Input?	false
------------------------	-------

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Test-AdminDBConnection command returns an object containing the status of the DelegatedAdmin Service if the connection string of the specified data store were to be set to the string being tested, together with extra diagnostics information for the specified connection string.

DBRejectedConnection

The database rejected the logon attempt from the DelegatedAdmin Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the DelegatedAdmin Service schema has not been added to the database.

DBNotFound

The specified database could not be located with the configured connection string.

DBNewerVersionThanService

The version of the DelegatedAdmin Service currently in use is incompatible with the version of the DelegatedAdmin Service schema on the database. Upgrade the DelegatedAdmin Service to a more recent version.

DBOlderVersionThanService

The version of the DelegatedAdmin Service schema on the database is incompatible with the version of the DelegatedAdmin Service currently in use. Upgrade the database schema to a more recent version.

DBVersionChangeInProgress

A database schema upgrade is currently in progress.

OK

The Set-AdminDBConnection command would succeed if it were executed with the supplied connection string.

Failed

The DelegatedAdmin Service has failed.

Unknown

The status of the DelegatedAdmin Service cannot be determined.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

InvalidDBConnectionString

The database connection string has an invalid format.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Test-AdminDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Tests a database connection string for the DelegatedAdmin Service.

----- **EXAMPLE 2** -----

```
c:\PS>Test-AdminDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Tests an invalid database connection string for the DelegatedAdmin Service.

# Citrix.EnvTest.Admin.V1

Nov 02, 2016

## Übersicht

Name	Beschreibung
<a href="#">EnvTestEnvTestSnapin</a>	Der Citrix Umgebungstestdienst bietet Tools zum Testen und Überprüfen des Zustands einer XenDesktop-Installation.
<a href="#">EnvTest Filtering</a>	Beschreibt die gebräuchlichsten Filteroptionen für XenDesktop-Cmdlets.

## Cmdlets

Name	Beschreibung
<a href="#">Get-EnvTestConfiguration</a>	Ruft die Konfigurationsoptionen für den Umgebungstestdienst ab.
<a href="#">Get-EnvTestDBConnection</a>	Ruft die Datenbankzeichenfolge für den angegebenen Datenspeicher ab, der vom Umgebungstestdienst verwendet wird.
<a href="#">Get-EnvTestDBSchema</a>	Ruft ein Skript ab, das das Umgebungstestdienst-Datenbankschema für den angegebenen Datenspeicher erstellt.
<a href="#">Get-EnvTestDBVersionChangeScript</a>	Ruft ein Skript ab, das das Umgebungstestdienst-Datenbankschema aktualisiert.
<a href="#">Get-EnvTestDefinition</a>	Ruft mindestens eine Testdefinition ab.
<a href="#">Get-EnvTestInstalledDBVersion</a>	Ruft eine Liste aller verfügbaren Versionen des Datenbankschemas für den Umgebungstestdienst ab.
<a href="#">Get-EnvTestService</a>	Ruft die Datensatzeinträge für den Umgebungstestdienst ab.
<a href="#">Get-EnvTestServiceAddedCapability</a>	Ruft zusätzliche Funktionen für den Umgebungstestdienst auf dem Controller ab.
<a href="#">Get-EnvTestServiceInstance</a>	Ruft die Dienstinstanzeinträge für den Umgebungstestdienst ab.
<a href="#">Get-EnvTestServiceStatus</a>	Ruft den aktuellen Status des Umgebungstestdiensts auf dem Controller ab.
<a href="#">Get-EnvTestSuiteDefinition</a>	Ruft mindestens eine Testsuitedefinition ab.
<a href="#">Get-EnvTestTask</a>	Ruft mindestens eine Umgebungstestaufgabe ab.

<b>Name</b> <b>New-</b>	<b>Beschreibung</b>
<a href="#">EnvTestDiscoveryTargetDefinition</a>	Erstellt ein neues Objekt "EnvTestDiscoveryTargetDefinition".
<a href="#">Remove-EnvTestServiceMetadata</a>	Entfernt Metadaten aus dem angegebenen Dienst.
<a href="#">Remove-EnvTestTask</a>	Entfernt abgeschlossene Aufgaben aus der Datenbank für den Umgebungstestdienst.
<a href="#">Remove-EnvTestTaskMetadata</a>	Entfernt Metadaten aus der angegebenen Aufgabe.
<a href="#">Reset-EnvTestServiceGroupMembership</a>	Lädt die Zugriffsberechtigungen und Konfigurationsdienstspeicherorte für den Umgebungstestdienst neu.
<a href="#">Set-EnvTestConfiguration</a>	Legt die Konfigurationsoptionen für den Umgebungstestdienst fest.
<a href="#">Set-EnvTestDBConnection</a>	Konfiguriert eine Datenbankverbindung für den Umgebungstestdienst.
<a href="#">Set-EnvTestServiceMetadata</a>	Fügt Metadaten dem angegebenen Dienst hinzu oder aktualisiert sie.
<a href="#">Set-EnvTestTaskMetadata</a>	Fügt Metadaten der angegebenen Aufgabe hinzu oder aktualisiert sie.
<a href="#">Start-EnvTestTask</a>	Startet eine neue Testaufgabe.
<a href="#">Stop-EnvTestTask</a>	Stoppt eine Aufgabe, die ausgeführt wird, bevor sie beendet ist.
<a href="#">Switch-EnvTestTask</a>	Legt die aktuelle Aufgabe fest, die durch einen Aufruf von Get-EnvTestTask ohne Parameter zurückgegeben wird.
<a href="#">Test-EnvTestDBConnection</a>	Testet eine Datenbankverbindung für den Umgebungstestdienst.

# about\_EnvTestEnvTestSnapin

May 10, 2016

## TOPIC

about\_EnvTestEnvTestSnapin

## SHORT DESCRIPTION

The Citrix Environment Test Service provides tools to test and inspect the state of a XenDesktop installation.

## COMMAND PREFIX

All commands in this snap-in have the noun prefixed with 'EnvTest'.

## LONG DESCRIPTION

The Citrix Environment Test Service provides tools to test and inspect the state of a XenDesktop installation at different points during and after configuration and install.

# about\_EnvTest\_Filtering

May 10, 2016

## TOPIC

XenDesktop - Advanced Dataset Filtering

## SHORT DESCRIPTION

Describes the common filtering options for XenDesktop cmdlets.

## LONG DESCRIPTION

Some cmdlets operate on large quantities of data and, to reduce the overhead of sending all of that data over the network, many of the Get- cmdlets support server-side filtering of the results.

The conventional way of filtering results in PowerShell is to pipeline them into Where-Object, Select-Object, and Sort-Object, for example:

```
Get-<Noun> | Where { $_.Size = 'Small' } | Sort 'Date' | Select -First 10
```

However, for most XenDesktop cmdlets the data is stored remotely and it would be slow and inefficient to retrieve large amounts of data over the network and then discard most of it. Instead, many of the Get- cmdlets provide filtering parameters that allow results to be processed on the server, returning only the required results.

You can filter results by most object properties using parameters derived from the property name. You can also sort results or limit them to a specified number of records:

```
Get-<Noun> -Size 'Small' -SortBy 'Date' -MaxRecordCount 10
```

You can express more complex filter conditions using a syntax and set of operators very similar to those used by PowerShell expressions.

Those cmdlets that support filtering have the following common parameters:

-MaxRecordCount <int>

Specifies the maximum number of results to return.  
For example, to return only the first nine results use:

```
Get-<Noun> -MaxRecordCount 9
```

If not specified, only the first 250 records are returned, and if more are available, a warning is produced:

WARNING: Only first 250 records returned. Use -MaxRecordCount to retrieve more.

You can suppress this warning by using -WarningAction or by specifying a value for -MaxRecordCount.

To retrieve all records, specify a large number for -MaxRecordCount. As the value is an integer, you can use the following:

```
Get-<Noun> -MaxRecordCount [int]::MaxValue
```

-ReturnTotalRecordCount [<SwitchParameter>]

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. For example:

```
Get-<Noun> -MaxRecordCount 9 -ReturnTotalRecordCount
```

....

```
Get-<Noun> : Returned 9 of 10 items
```

```
At line:1 char:18
```

```
+ Get-<Noun> <<<< -MaxRecordCount 9 -ReturnTotalRecordCount
```

```
+ CategoryInfo : OperationStopped: (:) [Get-<Noun>], PartialDataException
```

```
+ FullyQualifiedErrorId : PartialData,Citrix.<SDKName>.SDK.Get<Noun>
```

The count can be accessed using the TotalAvailableResultCount property:

```
$count = $error[0].TotalAvailableResultCount
```

-Skip <int>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

-SortBy <string>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order, respectively. Ascending order is assumed if no prefix is present.

Sorting occurs before -MaxRecordCount and -Skip parameters are applied. For example, to sort by Name and then by Count (largest first) use:

```
-SortBy 'Name,-Count'
```

By default, sorting by an enumeration property uses the numeric value of the elements. You can specify a different sort order by qualifying the name with an ordered list of elements or their numeric values, or <null> to indicate the placement of null values.

Elements not mentioned are placed at the end in their numeric order. For example, to sort by two different enums and then by the object id:

```
-SortBy 'MyState(StateC,<null>,StateA,StateB),Another(0,3,2,1),Id'
```

-Filter <String>

This parameter lets you specify advanced filter expressions, and supports combination of conditions with -and and -or, and grouping with braces. For example:

```
Get-<Noun> -Filter 'Name -like "High*" -or (Priority -eq 1 -and Severity -ge 2)'
```

The syntax is close enough to PowerShell syntax that you can use script blocks in most cases. This can be easier to read as it reduces quoting:

```
Get-<Noun> -Filter { Count -ne $null }
```

The full -Filter syntax is provided below.

## EXAMPLES

Filtering by strings performs a case-insensitive wildcard match.

Separate parameters are combined with an implicit -and operator.

Normal PowerShell quoting rules apply, so you can use single or double quotes, and omit the quotes altogether for many strings. The order of parameters does not make any difference. The following are equivalent:

```
Get-<Noun> -Company Citrix -Product Xen*
Get-<Noun> -Company "citrix" -Product '[X]EN*'
Get-<Noun> -Product "Xen*" -Company "CITRIX"
Get-<Noun> -Filter { Company -eq 'Citrix' -and Product -like 'Xen*' }
```



See [about\\_Quoting\\_Rules](#) and [about\\_Wildcards](#) for details about PowerShell handling of quotes and wildcards.

To avoid wildcard matching or include quote characters, you can escape the wildcards using the normal PowerShell escape mechanisms (see [about\\_Escape\\_Characters](#)), or switch to a filter expression and the `-eq` operator:

```
Get-<Noun> -Company "Abc[*]" # Matches Abc*
Get-<Noun> -Company "Abc`*" # Matches Abc*
Get-<Noun> -Filter { Company -eq "Abc*" } # Matches Abc*
Get-<Noun> -Filter { Company -eq "A`"B`C`" } # Matches A"B'C
```

Simple filtering by numbers, booleans, and TimeSpans perform direct equality comparisons, although if the value is nullable you can also search for null values. Here are some examples:

```
Get-<Noun> -Uid 123
Get-<Noun> -Enabled $true
Get-<Noun> -Duration 1:30:40
Get-<Noun> -NullableProperty $null
```

More comparisons are possible using advanced filtering with `-Filter`:

```
Get-<Noun> -Filter 'Capacity -ge 10gb'
Get-<Noun> -Filter 'Age -ge 20 -and Age -lt 40'
Get-<Noun> -Filter 'VolumeLevel -like "[123]"'
Get-<Noun> -Filter 'Enabled -ne $false'
Get-<Noun> -Filter 'NullableProperty -ne $null'
```

You can check boolean values without an explicit comparison operator, and you can also combine them with `-not`:

```
Get-<Noun> -Filter 'Enabled' # Equivalent to 'Enabled -eq $true'
Get-<Noun> -Filter '-not Enabled' # Equivalent to 'Enabled -eq $false'
```

See [about\\_Comparison\\_Operators](#) for an explanation of the operators, but note that only a subset of PowerShell operators are supported (`-eq`, `-ne`, `-gt`, `-ge`, `-lt`, `-le`, `-like`, `-notlike`, `-in`, `-notin`, `-contains`, `-notcontains`).

Enumeration values can either be specified using typed values or the string name of the enumeration value:

```
Get-<Noun> -Shape [Shapes]::Square
Get-<Noun> -Shape Circle
```

With filter expressions, typed values can be specified with simple variables or quoted strings. They also support enumerations with wildcards:

```
$s = [Shapes]::Square
Get-<Noun> -Filter { Shape -eq $s -or Shape -eq "Circle" }
Get-<Noun> -Filter { Shape -like 'C*' }
```

By their nature, floating point values, DateTime values, and TimeSpan values are best suited to relative comparisons rather than just equality. DateTime strings are converted using the locale and time zone of the user device, but you can use ISO8601 format strings (YYYY-MM-DDThh:mm:ss.sTZD) to avoid ambiguity. You can also use standard PowerShell syntax to create these values:

```
Get-<Noun> -Filter { StartTime -ge "2010-08-23T12:30:00.0Z" }
$d = [DateTime]"2010-08-23T12:30:00.0Z"
Get-<Noun> -Filter { StartTime -ge $d }
$d = (Get-Date).AddDays(-1)
Get-<Noun> -Filter { StartTime -ge $d }
```

Relative times are quite common and, when using filter expressions, you can also specify DateTime values using a relative format:

```
Get-<Noun> -Filter { StartTime -ge '-2' } # Two days ago
Get-<Noun> -Filter { StartTime -ge '-1:30' } # Hour and a half ago
Get-<Noun> -Filter { StartTime -ge '-0:0:30' } # 30 seconds ago
```

## ARRAY PROPERTIES

When filtering against list or array properties, simple parameters perform a case-insensitive wildcard match against each of the members. With filter expressions, you can use the -contains and -notcontains operators. Unlike PowerShell, these perform wildcard matching on strings.

Note that for array properties the naming convention is for the returned property to be plural, but the parameter used to search for any match is singular. The following are equivalent (assuming Users is an array property):

```
Get-<Noun> -User Fred*
Get-<Noun> -Filter { User -like "Fred*" }
Get-<Noun> -Filter { Users -contains "Fred*" }
```

You can also use the singular form with -Filter to search using other operators:

```
Match if any user in the list is called "Frederick"
Get-<Noun> -Filter { User -eq "Frederick" }
Match if any user in the list has a name alphabetically below 'F'
Get-<Noun> -Filter { User -lt 'F' }
```

## COMPLEX EXPRESSIONS

When matching against multiple values, you can use a sequence of comparisons joined with -or operators, or you can use -in and -notin:

```
Get-<Noun> -Filter { Shape -eq 'Circle' -or Shape -eq 'Square' }
$shapes = 'Circle','Square'
Get-<Noun> -Filter { Shape -in $shapes }
$sides = 1..4
Get-<Noun> -Filter { Sides -notin $sides }
```

Braces can be used to group complex expressions, and override the default left-to-right evaluation of -and and -or. You can also use -not to invert the sense of any sub-expression:

```
Get-<Noun> -Filter { Size -gt 4 -or (Color -eq 'Blue' -and Shape -eq 'Circle') }
Get-<Noun> -Filter { Sides -lt 5 -and -not (Color -eq 'Blue' -and Shape -eq 'Circle') }
```

## PAGING

The simplest way to page through data is to use the -Skip and -MaxRecordCount parameters. So, to read the first three pages of data with 10 records per page, use:

```
Get-<Noun> -Skip 0 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 10 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 20 -MaxRecordCount 10 <other filtering criteria>
```

You must include the same filtering criteria on each call, and ensure that the data is sorted consistently.

The above approach is often acceptable, but as each call performs an independent query, data changes can result in records being skipped or appearing twice. One approach to improve this is to sort by a unique id field and then start the search for the next page at the unique id after the last unique id of the previous page. For example:

```
Get the first page
Get-<Noun> -MaxRecordCount 10 -SortBy SerialNumber

SerialNumber ...

A120004
A120007
... 7 other records ...
A120900

Get the next page
Get-<Noun> -MaxRecordCount 10 -Filter { FirstName -gt 'A120900' }

SerialNumber ...
```

-----  
A120901  
B220000  
...

## FILTER SYNTAX DEFINITION

<Filter> ::= <ScriptBlock> | <ComponentList>

<ScriptBlock> ::= "{" <ComponentList> "}"

<ComponentList> ::= <Component> <AndOrOperator> <ComponentList> |

<Component>

<Component> ::= <NotOperator> <Factor> |

<Factor>

<Factor> ::= "(" <ComponentList> ")" |

<PropertyName> <ComparisonOperator> <Value> |  
<PropertyName>

<AndOrOperator> ::= "-and" | "-or"

<NotOperator> ::= "-not" | "!"

<ComparisonOperator>

::= "-eq" | "-ne" | "-le" | "-ge" | "-lt" | "-gt" |  
"-like" | "-notlike" | "-contains" | "-notcontains" |  
"-in" | "-notin"

<PropertyName> ::= <simple name of property>

<Value> ::= <string literal> | <numeric literal> |

<scalar variable> | <array variable> |  
"\$null" | "\$true" | "\$false"

Numeric literals support decimal and hexadecimal literals, with optional multiplier suffixes (kb, mb, gb, tb, pb).

Dates and times can be specified as string literals. The current culture determines what formats are accepted. To avoid any ambiguity, use strings formatted to the ISO8601 standard. If not specified, the current time zone is used.

Relative date-time string literals are also supported, using a minus sign followed by a TimeSpan. For example, "-1:30" means 1 hour and 30 minutes ago.

# Get-EnvTestConfiguration

May 10, 2016

Gets the Environment Test Service's configuration options

## Syntax

```
Get-EnvTestConfiguration [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets the Environment Test Service's configuration options and returns them as key/value pairs.

## Related topics

### Parameters

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Dictionary<string, object>

All configuration settings

## Examples

----- **EXAMPLE 1** -----

Get-EnvTestConfiguration  
Gets all configuration options

# Get-EnvTestDBConnection

May 10, 2016

Gets the database string for the specified data store used by the EnvTest Service.

## Syntax

```
Get-EnvTestDBConnection [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the database connection string for the specified data store.

If the returned string is blank, no valid connection string has been specified. In this case the service is running, but is idle and awaiting specification of a valid connection string.

## Related topics

[Get-EnvTestServiceStatus](#)

[Set-EnvTestDBConnection](#)

[Test-EnvTestDBConnection](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

system.string

The database connection string configured for the EnvTest Service.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----



NoDBConnections

The database connection string for the EnvTest Service has not been specified.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Get-EnvTestDBConnection

Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True

Get the database connection string for the EnvTest Service.

# Get-EnvTestDBSchema

May 10, 2016

Gets a script that creates the EnvTest Service database schema for the specified data store.

## Syntax

```
Get-EnvTestDBSchema [-DatabaseName <String>] [-ServiceGroupName <String>] [-ScriptType <ScriptTypes>] [-LocalDatabase] [-Sid <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets SQL scripts that can be used to create a new EnvTest Service database schema, add a new EnvTest Service to an existing site, remove a EnvTest Service from a site, or create a database server logon for a EnvTest Service. If no Sid parameter is provided, the scripts obtained relate to the currently selected EnvTest Service instance, otherwise the scripts relate to EnvTest Service instance running on the machine identified by the Sid provided. When obtaining the Evict script, a Sid parameter must be supplied. The current service instance is that on the local machine, or that explicitly specified by the last usage of the -AdminAddress parameter to a EnvTest SDK cmdlet. The service instance used to obtain the scripts does not need to be a member of a site or to have had its database connection configured. The database scripts support only Microsoft SQL Server, or SQL Server Express, and require Windows integrated authentication to be used. They can be run using SQL Server's SQLCMD utility, or by copying the script into an SQL Server Management Studio (SSMS) query window and executing the query. If using SSMS, the query must be executed in 'SMDCMD mode'. The ScriptType parameter determines which script is obtained. If ScriptType is not specified, or is FullDatabase, the script contains:

- o Creation of service schema
- o Creation of database server logon
- o Creation of database user
- o Addition of database user to EnvTest Service roles

If ScriptType is Instance, the returned script contains:

- o Creation of database server logon
- o Creation of database user
- o Addition of database user to EnvTest Service roles

If ScriptType is Evict, the returned script contains:

- o Removal of EnvTest Service instance from database
- o Removal of database user

If ScriptType is Login, the returned script contains:

- o Creation of database server logon only

If the service uses two data stores they can exist in the same database. You do not need to configure a database before using this command.

## Related topics

[Set-EnvTestDBConnection](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database for which the schema will be generated.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ServiceGroupName**<String>

Specifies the name of the service group to be used when creating the database schema. The service group is a collection of all the EnvTest services that share the same database instance and are considered equivalent; that is, all the services within a service group can be used interchangeably.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ScriptType**<ScriptTypes>

Specifies the type of database script returned. Available script types are:

### Database

Returns a full database script that can be used to create a database schema for the EnvTest Service in a database instance that does not already contain a schema for this service. The DatabaseName and ServiceGroupName parameters must be specified to create a script of this type.

### Instance

Returns a permissions script that can be used to add further EnvTest services to an existing database instance that already contains the full EnvTest service schema, associating the services to the Service Group. The Sid parameter can optionally be specified to create a script of this type.

### Login

Returns a database logon script that can be used to add the required logon accounts to an existing database instance that contains the EnvTest Service schema. This is used primarily when creating a mirrored database environment. The

DatabaseName parameter must be specified to create a script of this type.

#### Evict

Returns a script that can be used to remove the specified EnvTest Service from the database entirely. The DatabaseName and Sid parameters must be specified to create a script of this type.

Required?	false
Default Value	Database
Accept Pipeline Input?	false

#### -LocalDatabase<SwitchParameter>

Specifies whether the database script is to be used in a database instance run on the same controller as other services in the service group. Including this parameter ensures the script creates only the required permissions for local services to access the database schema for EnvTest services.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -Sid<String>

Specifies the SID of the controller on which the EnvTest Service instance to remove from the database is running.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

#### -AdminAddress<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

### System.string

A string containing the required SQL script for application to a database.

## Notes

The scripts returned support Microsoft SQL Server Express Edition, Microsoft SQL Server Standard Edition, and Microsoft SQL Server Enterprise Edition databases only, and are generated on the assumption that integrated authentication will be used.

If the ScriptType parameter is not included or set to 'FullDatabase', the full database script is returned, which will:

Create the database schema.

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist).

If the ScriptType parameter is set to 'Instance', the script will:

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist) and associate it with a user.

If the ScriptType parameter is set to 'Login', the script will:

Create the logon (providing the schema does not already exist) and associate it with a pre-existing user of the same name.

If the LocalDatabase parameter is included, the NetworkService account will be added to the list of accounts permitted to access the database. This is required only if the database is run on a controller.

If the command fails, the following errors can be returned.

## Error Codes

-----

### GetSchemasFailed

The database schema could not be found.

### ActiveDirectoryAccountResolutionFailed

The specified Active Directory account or Group could not be found.

### DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-EnvTestDBSchema -DatabaseName MyDB -ServiceGroupName MyServiceGroup > c:\EnvTestSchema.sql
```

Get the full database schema for site data store of the EnvTest Service and copy it to a file called 'c:\EnvTestSchema.sql'.

This script can then be used to create the schema in a pre-existing database named 'MyDB' that does not already contain a EnvTest Service site schema.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-EnvTestDBSchema -DatabaseName MyDB -scriptType Login > c:\EnvTestLogins.sql
```

Get the logon scripts for the EnvTest Service.

# Get-EnvTestDBVersionChangeScript

May 10, 2016

Gets a script that updates the EnvTest Service database schema.

## Syntax

```
Get-EnvTestDBVersionChangeScript -DatabaseName <String> -TargetVersion <Version> [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Returns a database script that can be used to upgrade or downgrade the site or secondary schema for the EnvTest Service from the current schema version to a different version.

## Related topics

[Get-EnvTestInstalledDBVersion](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database instance to which the update applies.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-TargetVersion**<Version>

Specifies the version of the database you want to update to.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

--	--

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.Management.Automation.PSObject

A PSObject containing the required SQL script for application to a database.

## Notes

The PSObject returned by this cmdlet contains the following properties:

-- Script The raw text of the SQL script to apply the update, or null in the case when no upgrade path to the specified target version exists.

-- NeedExclusiveAccess Indicates whether all services in the service group must be shut down during the update or not.

-- CanUndo Indicates whether the generated script allows the updated schema to be reverted to the state prior to the update.

Scripts to update the schema version are stored in the database so any service in the service group can obtain these scripts. Extreme caution should be exercised when using update scripts. Citrix recommends backing up the database before attempting to upgrade the schema. Database update scripts may require exclusive use of the schema and so may not be able to execute while any EnvTest services are running. However, this depends on the specific update being carried out.

After a schema update has been carried out, services that require the previous version of the schema may cease to operate. The ServiceState parameter reported by the Get-EnvTestServiceStatus command provides information about service compatibility. For example, if the schema has been upgraded to a more recent version that a service cannot use, the service reports "DBNewerVersionThanService".

If the command fails, the following errors can be returned.

## Error Codes

-----

### NoOp

The operation was successful but had no effect.

### NoDBConnections

The database connection string for the EnvTest Service has not been specified.

### DatabaseError

An error occurred in the service while attempting a database operation.



#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> $update = Get-EnvTestDBVersionChangeScript -DatabaseName MyDb -TargetVersion 1.0.75.0
```

```
C:\PS> $update.Script > update_75.sql
```

Gets an SQL update script to update the current schema to version 1.0.75.0. The resulting update\_75.sql script is suitable for direct use with the SQL Server SQLCMD utility.

# Get-EnvTestDefinition

May 10, 2016

Gets the one or more test definitions

## Syntax

```
Get-EnvTestDefinition [-TestId <String[]>] [-CultureName <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns a list of test definitions that are available from currently running components.

## Related topics

[Get-EnvTestSuiteDefinition](#)

[Get-EnvTestTask](#)

[Start-EnvTestTask](#)

[Switch-EnvTestTask](#)

[Stop-EnvTestTask](#)

[Remove-EnvTestTask](#)

[Add-EnvTestTaskMetadata](#)

[Remove-EnvTestTaskMetadata](#)

## Parameters

**-TestId**<String[]>

The id of one or more tests.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

**-CultureName**<String>

The culture name in which to produce results. The culture name is in standard language/region-code format; for example "en-US".

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Input Type

System.String A test id. System.String[] An array of test ids.

### Return Values

Citrix.EnvTest.Sdk.EnvTestDefinition

One or more test definitions.

### Examples

#### ----- **EXAMPLE 1** -----

```
$allTestDefinitions = Get-EnvTestDefinition
```

Retrieve all tests.

#### ----- **EXAMPLE 2** -----

```
$allTestDefinitionsTranslatedIntoSpanish = Get-EnvTestDefinition -CultureName es-ES
```

Retrieve all tests with localized properties returned in Spanish.

#### ----- **EXAMPLE 3** -----

```
$monitorConfigServiceRegistrationDefinition = Get-EnvTestDefinition -TestId Monitor_RegisteredWithConfigurationService
```

Retrieve the definition of the 'Monitor\_RegisteredWithConfigurationService' test.

# Get-EnvTestInstalledDBVersion

May 10, 2016

Gets a list of all available database schema versions for the EnvTest Service.

## Syntax

Get-EnvTestInstalledDBVersion [-Upgrade] [-Downgrade] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Returns the current version of the EnvTest Service database schema, if no flags are set, otherwise returns versions for which upgrade or downgrade scripts are available and have been stored in the database.

## Related topics

## Parameters

**-Upgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be updated should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Downgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be reverted should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

### System.Version

The Get-EnvTestInstalledDbVersion command returns objects containing the new definition of the EnvTest Service database schema version.

Major <Integer>

Minor <Integer>

Build <Integer>

Revision <Integer>

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

Both the Upgrade and Downgrade flags were specified.

#### NoOp

The operation was successful but had no effect.

#### NoDBConnections

The database connection string for the EnvTest Service has not been specified.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-EnvTestInstalledDBVersion
```

Major	Minor	Build	Revision
-------	-------	-------	----------

-----	-----	-----	-----
-------	-------	-------	-------

5	6	0	0
---	---	---	---

Get the currently installed version of the EnvTest Service database schema.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-EnvTestInstalledDBVersion -Upgrade
```

Major	Minor	Build	Revision
-------	-------	-------	----------

-----	-----	-----	-----
-------	-------	-------	-------

6	0	0	0
---	---	---	---

Get the versions of the EnvTest Service database schema for which upgrade scripts are supplied.

# Get-EnvTestService

May 10, 2016

Gets the service record entries for the EnvTest Service.

## Syntax

```
Get-EnvTestService [-Metadata <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns instances of the EnvTest Service that the service publishes. The service records contain account security identifier information that can be used to remove each service from the database.

A database connection for the service is required to use this command.

## Related topics

### Parameters

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_EnvTest\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false



### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_EnvTest\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.EnvTest.Sdk.Service

The Get-EnvTestServiceInstance command returns an object containing the following properties.

Uid <Integer>

Specifies the unique identifier for the service in the group. The unique identifier is an index number.

ServiceHostId <Guid>

Specifies the unique identifier for the service instance.

DNSName <String>

Specifies the domain name of the host on which the service runs.

MachineName <String>

Specifies the short name of the host on which the service runs.

CurrentState <Citrix.Fma.Sdk.ServiceCore.ServiceState>

Specifies whether the service is running, started but inactive, stopped, or failed.

LastStartTime <DateTime>

Specifies the date and time at which the service was last restarted.

LastActivityTime <DateTime>

Specifies the date and time at which the service was last stopped or restarted.

OSType

Specifies the operating system installed on the host on which the service runs.

OSVersion

Specifies the version of the operating system installed on the host on which the service runs.

ServiceVersion

Specifies the version number of the service instance. The version number is a string that reflects the full build version of the service.

DatabaseUserName <string>

Specifies for the service instance the Active Directory account name with permissions to access the database. This will be either the machine account or, if the database is running on a controller, the NetworkService account.

Sid <string>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

ActiveSiteServices <string[]>

Specifies the names of active site services currently running in the service. Site services are components that perform long-running background processing in some services. This field is empty for services that do not contain site services.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

UnknownObject

One of the specified objects was not found.

PartialData

Only a subset of the available data was returned.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

CouldNotQueryDatabase

The query required to get the database was not defined.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Get-EnvTestService

Uid : 1  
ServiceHostId : aef6f464-f1ee-4042-a523-66982e0cecd0  
DNSName : MyServer.company.com  
MachineName : MYSERVER  
CurrentState : On  
LastStartTime : 04/04/2011 15:25:38  
LastActivityTime : 04/04/2011 15:33:39  
OSType : Win32NT  
OSVersion : 6.1.7600.0  
ServiceVersion : 5.1.0.0  
DatabaseUserName : NT AUTHORITY\NETWORK SERVICE  
SID : S-1-5-21-2316621082-1546847349-2782505528-1165  
ActiveSiteServices : {MySiteService1, MySiteService2...}  
Get all the instances of the EnvTest Service running in the current service group.

# Get-EnvTestServiceAddedCapability

May 10, 2016

Gets any added capabilities for the EnvTest Service on the controller.

## Syntax

```
Get-EnvTestServiceAddedCapability [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables updates to the EnvTest Service on the controller to be detected.

You do not need to configure a database connection before using this command.

## Related topics

### Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.String

String containing added capabilities.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-EnvTestServiceAddedCapability
```

Get the added capabilities of the EnvTest Service.

# Get-EnvTestServiceInstance

May 10, 2016

Gets the service instance entries for the EnvTest Service.

## Syntax

```
Get-EnvTestServiceInstance [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns service interfaces published by the instance of the EnvTest Service. Each instance of a service publishes multiple interfaces with distinct interface types, and each of these interfaces is represented as a ServiceInstance object. Service instances can be used to register the service with a central configuration service so that other services can use the functionality.

You do not need to configure a database connection to use this command.

## Related topics

### Parameters

**-AdminAddress** <String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.EnvTest.Sdk.ServiceInstance

The Get-EnvTestServiceInstance command returns an object containing the following properties.

ServiceGroupUid <Guid>

Specifies the unique identifier for the service group of which the service is a member.

ServiceGroupName <String>

Specifies the name of the service group of which the service is a member.

ServiceInstanceUID <Guid>

Specifies the unique identifier for registered service instances, which are service instances held by and obtained from a

central configuration service. Unregistered service instances do not have unique identifiers.

ServiceType <String>

Specifies the service instance type. For this service, the service instance type is always EnvTest.

Address

Specifies the address of the service instance. The address can be used to access the service and, when registered in the central configuration service, can be used by other services to access the service.

Binding

Specifies the binding type that must be used to communicate with the service instance. In this release of XenDesktop, the binding type is always 'wcf\_HTTP\_kerb'. This indicates that the service provides a Windows Communication Foundation endpoint that uses HTTP binding with integrated authentication.

Version

Specifies the version of the service instance. The version number is used to ensure that the correct versions of the services are used for communications.

ServiceAccount <String>

Specifies the Active Directory account name for the machine on which the service instance is running. The account name is used to provide information about the permissions required for interservice communications.

ServiceAccountSid <String>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

InterfaceType <String>

Specifies the interface type. Each service can provide multiple service instances, each for a different purpose, and the interface defines the purpose. Available interfaces are:

SDK - for PowerShell operations

InterService - for operations between different services

Peer - for communications between services of the same type

Metadata <Citrix.EnvTest.Sdk.Metadata[]>

The collection of metadata associated with registered service instances, which are service instances held by and obtained from a central configuration service. Metadata is not stored for unregistered service instances.

Notes

If the command fails, the following errors can be returned.

Error Codes

-----

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-EnvTestServiceInstance
```

```
Address : http://MyServer.com:80/Citrix/EnvTestContract
Binding : wcf_HTTP_kerb
InterfaceType : SDK
Metadata :
MetadataMap :
ServiceAccount : ENG\MyAccount$
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164
ServiceGroupName : MyServiceGroup
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000
ServiceType : EnvTest
Version : 1
```

```
Address : http://MyServer.com:80/Citrix/EnvTestContract/IServiceApi
Binding : wcf_HTTP_kerb
InterfaceType : InterService
Metadata :
MetadataMap :
```



ServiceAccount : ENG\MyAccount  
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164  
ServiceGroupName : MyServiceGroup  
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d  
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000  
ServiceType : EnvTest  
Version : 1

Get all instances of the EnvTest Service running on the specified machine. For remote services, use the AdminAddress parameter to define the service for which the interfaces are required. If the AdminAddress parameter has not been specified for the runspace, service instances running on the local machine are returned.

# Get-EnvTestServiceStatus

May 10, 2016

Gets the current status of the EnvTest Service on the controller.

## Syntax

```
Get-EnvTestServiceStatus [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables the status of the EnvTest Service on the controller to be monitored. If the service has multiple data stores it will return the overall state as an aggregate of all the data store states. For example, if the site data store status is OK and the secondary data store status is DBUnconfigured then it will return DBUnconfigured.

## Related topics

[Set-EnvTestDBConnection](#)

[Test-EnvTestDBConnection](#)

[Get-EnvTestDBConnection](#)

[Get-EnvTestDBSchema](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Get-EnvTestServiceStatus command returns an object containing the status of the EnvTest Service together with extra diagnostics information.

DBUnconfigured

The EnvTest Service does not have a database connection configured.

DBRejectedConnection

The database rejected the logon attempt from the EnvTest Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

#### InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the EnvTest Service schema has not been added to the database.

#### DBNotFound

The specified database could not be located with the configured connection string.

#### DBNewerVersionThanService

The version of the EnvTest Service currently in use is incompatible with the version of the EnvTest Service schema on the database. Upgrade the EnvTest Service to a more recent version.

#### DBOlderVersionThanService

The version of the EnvTest Service schema on the database is incompatible with the version of the EnvTest Service currently in use. Upgrade the database schema to a more recent version.

#### DBVersionChangeInProgress

A database schema upgrade is currently in progress.

#### OK

The EnvTest Service is running and is connected to a database containing a valid schema.

#### Failed

The EnvTest Service has failed.

#### Unknown

(0) The service status cannot be determined.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- **EXAMPLE 1** -----

```
c:\PS>Get-EnvTestServiceStatus
```

DBUnconfigured

Get the current status of the EnvTest Service.

# Get-EnvTestSuiteDefinition

May 10, 2016

Gets one or more test suite definitions.

## Syntax

```
Get-EnvTestSuiteDefinition [-TestSuiteId <String[]>] [-CultureName <String>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Returns a list of test suite definitions that are available from currently running components.

## Related topics

[Get-EnvTestDefinition](#)

[Get-EnvTestTask](#)

[Start-EnvTestTask](#)

[Switch-EnvTestTask](#)

[Stop-EnvTestTask](#)

[Remove-EnvTestTask](#)

[Add-EnvTestTaskMetadata](#)

[Remove-EnvTestTaskMetadata](#)

## Parameters

**-TestSuiteId**<String[]>

The id of one or more test suites.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

**-CultureName**<String>

The culture name in which to produce results. The culture name is in standard language/region-code format; for example "en-US".

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

System.String A test suite id. System.String[] An array of test suite ids.

#### Return Values

Citrix.EnvTest.Sdk.EnvTestSuiteDefinition

The definition of a test suite

#### Examples

##### ----- **EXAMPLE 1** -----

```
$allTestSuiteDefinitions = Get-EnvTestSuiteDefinition
```

Retrieve all test suites.

##### ----- **EXAMPLE 2** -----

```
$allTestSuiteDefinitionsTranslatedIntoSpanish = Get-EnvTestSuiteDefinition -CultureName es-ES
```

Retrieve all test suites with localized properties returned in Spanish.

##### ----- **EXAMPLE 3** -----

```
$infrastructureSuiteDefinition = Get-EnvTestSuiteDefinition -TestSuiteId Infrastructure
```

Retrieve the definition of the 'Infrastructure' test suite.

# Get-EnvTestTask

May 10, 2016

Gets one or more EnvTestTask(s)

## Syntax

```
Get-EnvTestTask [-TaskId <Guid>] [-List] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns either the current task, a specified task, or list of tasks that are currently known to the EnvTest Service.

## Related topics

[Get-EnvTestDefinition](#)

[Get-EnvTestSuiteDefinition](#)

[Start-EnvTestTask](#)

[Switch-EnvTestTask](#)

[Stop-EnvTestTask](#)

[Remove-EnvTestTask](#)

[Add-EnvTestTaskMetadata](#)

[Remove-EnvTestTaskMetadata](#)

## Parameters

**-TaskId**<Guid>

Specifies the task identifier to be returned. This value can be retrieved from an existing task's \$task.TaskId property.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-List**<SwitchParameter>

List all running tasks, including the current one.

Required?	false
Default Value	false

Accept Pipeline Input?	false
------------------------	-------

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

Citrix.EnvTest.Sdk.EnvTestTask

A description of a previously started task.

#### Examples

##### ----- **EXAMPLE 1** -----

```
$currentTask = Get-EnvTestTask
```

Retrieve the current task. The current task is the most recently created task unless Switch-EnvTestTask explicitly changes it.

##### ----- **EXAMPLE 2** -----

```
$taskOfSpecifidId = Get-EnvTestTask -TaskId 36C0EC52-2039-4D6E-B690-9F02F8CEFFCC
```

Retrieve a fresh copy of a task object based on a known task id, which is always a Guid. This id can be retrieved from an existing task object via its \$task.TaskId property.

##### ----- **EXAMPLE 3** -----

```
$allKnownTasks = Get-EnvTestTask -List
```

Retrieve the list of current tasks. This list includes any task started by the Start-EnvTestTask cmdlet since the service started that has not later been removed via Remove-EnvTestTask.



# New-EnvTestDiscoveryTargetDefinition

May 10, 2016  
Creates a new EnvTestDiscoveryTargetDefinition object

## Syntax

```
New-EnvTestDiscoveryTargetDefinition -TestId <String> [-TargetIdType <String>] [-TargetId <String>] [-AdminAddress <String>] [<CommonParameters>]

New-EnvTestDiscoveryTargetDefinition -TestSuiteId <String> [-TargetIdType <String>] [-TargetId <String>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Creates a new EnvTestDiscoveryTargetDefinition object that can be piped into Start-EnvTestTask to define one or more targets of execution, optionally including root objects for discovery.

## Related topics

- [Get-EnvTestDefinition](#)
- [Get-EnvTestSuiteDefinition](#)
- [Get-EnvTestTask](#)
- [Start-EnvTestTask](#)
- [Switch-EnvTestTask](#)
- [Stop-EnvTestTask](#)
- [Remove-EnvTestTask](#)
- [Add-EnvTestTaskMetadata](#)
- [Remove-EnvTestTaskMetadata](#)

## Parameters

**-TestId**<String>  
Test identifiers. If specified, do not specify -TestSuiteId.

Required?	true
Default Value	Empty
Accept Pipeline Input?	false

**-TestSuiteId**<String>  
Test suite identifiers. If specified, do not specify -TestId.

Required?	true
Default Value	Empty
Accept Pipeline Input?	false

### **-TargetIdType<String>**

Describes the type of corresponding object passed with -TargetId

Required?	false
Default Value	Empty
Accept Pipeline Input?	false

### **-TargetId<String>**

The Ids that object tests or test suites will target. By default, other components are queried for objects related to these.

Required?	false
Default Value	Empty
Accept Pipeline Input?	false

### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.EnvTest.Sdk.EnvTestDiscoveryTargetDefinition

Defines a target of a task

### Examples

#### ----- EXAMPLE 1 -----

```
$singleSimpleTestTaskTarget = New-EnvTestDiscoveryTargetDefinition -TestId Monitor_RegisteredWithConfigurationService
```

```
$singleSimpleTestTaskTarget | Start-EnvTestTask
```

Create a discovery target definition with a single test and no target object, then start a task based on it.

#### ----- EXAMPLE 2 -----

```
$singleSimpleTestSuiteTaskTarget = New-EnvTestDiscoveryTargetDefinition -TestSuiteId Infrastructure
```

```
$singleSimpleTestSuiteTaskTarget | Start-EnvTestTask
```

Create a discovery target definition with a single test suite and no target object, then start a task based on it.

#### ----- EXAMPLE 3 -----

```
$singleTestSuiteTaskTarget = New-EnvTestDiscoveryTargetDefinition -TestSuiteId Catalog -TargetIdType Catalog -TargetId $(Get-BrokerCatalog).Uuid
```

```
$singleTestSuiteTaskTarget | Start-EnvTestTask
```

Create a discovery target definition with a single test suite and a catalog target object, then start a task based on it.

#### ----- EXAMPLE 4 -----

```
$singleSimpleTestSuiteTaskTarget = New-EnvTestDiscoveryTargetDefinition -TestSuiteId Infrastructure
$singleTestSuiteTaskTarget = New-EnvTestDiscoveryTargetDefinition -TestSuiteId Catalog -TargetIdType Catalog -TargetId $(Get-BrokerCatalog).Uuid
@($singleSimpleTestSuiteTaskTarget, $singleTestSuiteTaskTarget) | Start-EnvTestTask
```

Create two different discovery target definitions, put them in an array, then start a task based on both.

# Remove-EnvTestServiceMetadata

May 10, 2016

Removes metadata from the given Service.

## Syntax

```
Remove-EnvTestServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-EnvTestServiceMetadata [-ServiceHostId] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-EnvTestServiceMetadata [-InputObject] <Service[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-EnvTestServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Service.

## Related topics

[Set-EnvTestServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name<String>**

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-EnvTestService | %{ Remove-EnvTestServiceMetadata -Map $_.MetadataMap }
```

Remove all metadata from all Service objects.

# Remove-EnvTestTask

May 10, 2016

Removes from the database completed tasks for the EnvTest Service.

## Syntax

```
Remove-EnvTestTask [-TaskId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-EnvTestTask [-Task <EnvTestTask>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables completed tasks that have run within the EnvTest Service to be removed from the database.

## Related topics

[Get-EnvTestDefinition](#)

[Get-EnvTestSuiteDefinition](#)

[Get-EnvTestTask](#)

[Start-EnvTestTask](#)

[Switch-EnvTestTask](#)

[Stop-EnvTestTask](#)

[Add-EnvTestTaskMetadata](#)

[Remove-EnvTestTaskMetadata](#)

## Parameters

**-TaskId**<Guid>

Specifies the identifier of the task to be removed, retrieveable from the \$task.TaskId property.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Task**<EnvTestTask>

Specifies the task to be removed.

Required?	false
-----------	-------



Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Examples

##### ----- **EXAMPLE 1** -----

**Remove-EnvTestTask**

Removes the current task from the EnvTest service.

##### ----- **EXAMPLE 2** -----

**Remove-EnvTestTask -TaskId 50A4139F-2B55-4A97-A1BE-20EE4E124AA3**

Removes a task from the EnvTest service via its id, which is available from an existing task's \$task.TaskId property.

##### ----- **EXAMPLE 3** -----

**\$secondTask = \$(Get-EnvTestTask -List)[1]**

**Remove-EnvTestTask -Task \$secondTask**

Removes the second task in the list returned by Get-EnvTestTask -List.

# Remove-EnvTestTaskMetadata

May 10, 2016

Removes metadata from the given Task.

## Syntax

```
Remove-EnvTestTaskMetadata [-TaskTaskId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-EnvTestTaskMetadata [-TaskTaskId] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-EnvTestTaskMetadata [-InputObject] <EnvTestTask[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Remove-EnvTestTaskMetadata [-InputObject] <EnvTestTask[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Task.

## Related topics

[Set-EnvTestTaskMetadata](#)

## Parameters

**-TaskTaskId**<Guid>

Id of the Task

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<EnvTestTask[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name<String>**

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-EnvTestTask | % { Remove-EnvTestTaskMetadata -Map $_.MetadataMap }
```

Remove all metadata from all Task objects.

# Reset-EnvTestServiceGroupMembership

May 10, 2016

Reloads the access permissions and configuration service locations for the EnvTest Service.

## Syntax

```
Reset-EnvTestServiceGroupMembership [-ConfigServiceInstance] <ServiceInstance[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Enables you to reload EnvTest Service access permissions and configuration service locations. The Reset-EnvTestServiceGroupMembership command must be run on at least one instance of the service type (EnvTest) after installation and registration with the configuration service. Without this operation, the EnvTest services will be unable to communicate with other services in the XenDesktop deployment. When the command is run, the services are updated when additional services are added to the deployment, provided that the configuration service is not stopped. The Reset-EnvTestServiceGroupMembership command can be run again to refresh this information if automatic updates do not occur when new services are added to the deployment. If more than one configuration service instance is passed to the command, the first instance that meets the expected service type requirements is used.

## Related topics

### Parameters

**-ConfigServiceInstance**<ServiceInstance[]>

Specifies the configuration service instance object that represents the service instance for the type 'InterService' that references a configuration service for the deployment.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Input Type

Citrix.EnvTest.Sdk.ServiceInstance[] Service instances containing a ServiceInstance object that refers to the central configuration service interservice interface can be piped to the Reset-EnvTestServiceGroupMembership command.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### NoSuitableServiceInstance

None of the supplied service instance objects were suitable for resetting service group membership.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config | Reset-EnvTestServiceGroupMembership
```

Reset the service group membership for a service in a deployment where the configuration service is configured and running on the same machine as the service.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config -AdminAddress OtherServer.example.com | Reset-EnvTestServiceGroupmembership
```

Reset the service group membership for a service in a deployment where the configuration service that is configured and running on a machine named 'OtherServer.example.com'.

# Set-EnvTestConfiguration

May 10, 2016

Sets the Environment Test Service's configuration options

## Syntax

```
Set-EnvTestConfiguration [-PerTypeDiscoveredObjectLimit <Int32>] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Sets the Environment Test Service's configuration options and broadcasts the changes to other machines in the Site.

## Related topics

## Parameters

### **-PerTypeDiscoveredObjectLimit**<Int32>

Sets the maximum number of objects of a particular type to be explored when discovering objects during a test run.  
Default: 50

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.



Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Input Type

Integer Must be greater than zero.

### Return Values

System.String

### Examples

#### ----- **EXAMPLE 1** -----

Set-EnvTestConfiguration -PerTypeDiscoveredObjectLimit 100

Set the maximum to 100

# Set-EnvTestDBConnection

May 10, 2016  
Configures a database connection for the EnvTest Service.

## Syntax

Set-EnvTestDBConnection [-DBConnection] <String> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Configures a connection to a database in which the EnvTest Service can store its state. The service will attempt to connect and start using the database immediately after the connection is configured. The database connection string is updated to the specified value regardless of whether it is valid or not. Specifying an invalid connection string prevents a service from functioning until the error is corrected.

After a connection is configured, you cannot alter it without first clearing it (by setting the connection to \$null).

You do not need to configure a database connection to use this command.

## Related topics

[Get-EnvTestServiceStatus](#)

[Get-EnvTestDBConnection](#)

[Test-EnvTestDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be used by the EnvTest Service. Passing in \$null will clear any existing database connection configured.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Force**<SwitchParameter>

If present, allows the local administrator to set the connection string to null when there are problems contacting the database or other services.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Set-EnvTestDBConnection command returns an object containing the status of the EnvTest Service together with extra diagnostics information.

#### DBUnconfigured

The EnvTest Service does not have a database connection configured.

#### DBRejectedConnection

The database rejected the logon attempt from the EnvTest Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

#### InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the EnvTest Service schema has not been added to the database.

#### DBNotFound

The specified database could not be located with the configured connection string.

#### DBNewerVersionThanService

The version of the EnvTest Service currently in use is incompatible with the version of the EnvTest Service schema on the database. Upgrade the EnvTest Service to a more recent version.

#### DBOlderVersionThanService

The version of the EnvTest Service schema on the database is incompatible with the version of the EnvTest Service currently in use. Upgrade the database schema to a more recent version.

#### DBVersionChangeInProgress

A database schema upgrade is currently in progress.

#### OK

The EnvTest Service is running and is connected to a database containing a valid schema.

#### Failed

The EnvTest Service has failed.

Unknown

The status of the EnvTest Service cannot be determined.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidDBConnectionString

The database connection string has an invalid format.

#### DatabaseConnectionDetailsAlreadyConfigured

There was already a database connection configured. After a configuration is set, it can only be set to \$null.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Set-EnvTestDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Configures a database connection string for the EnvTest Service.

### ----- EXAMPLE 2 -----

```
c:\PS>Set-EnvTestDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Configures an invalid database connection string for the EnvTest Service.

# Set-EnvTestServiceMetadata

May 10, 2016

Adds or updates metadata on the given Service.

## Syntax

```
Set-EnvTestServiceMetadata [-ServiceHostId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-EnvTestServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-EnvTestServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-EnvTestServiceMetadata [-InputObject] <Service[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Allows you to store additional custom data against given Service objects.

## Related topics

[Remove-EnvTestServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the Service specified. The property cannot contain any of the following characters \/:;#.\*?=<>|[]()''

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.Collections.Generic.Dictionary[String,String]

Set-EnvTestServiceMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

InvalidParameterCombination

The cmdlet parameters are inconsistent.

UnknownObject

One of the specified objects was not found.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Set-EnvTestServiceMetadata -ServiceHostId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value

Key	Value
---	----
property	value

Add metadata with a name of 'property' and a value of 'value' to the Service with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.



# Set-EnvTestTaskMetadata

May 10, 2016  
Adds or updates metadata on the given Task.

## Syntax

Set-EnvTestTaskMetadata [-TaskTaskId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-EnvTestTaskMetadata [-TaskTaskId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-EnvTestTaskMetadata [-InputObject] <EnvTestTask[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-EnvTestTaskMetadata [-InputObject] <EnvTestTask[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Provides the ability for additional custom data to be stored against given Task objects.

## Related topics

[Remove-EnvTestTaskMetadata](#)

## Parameters

**-TaskTaskId**<Guid>

Id of the Task

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<EnvTestTask[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the Task specified. The property cannot contain any of the following characters \/:;#.\*?=<>|[]()"

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	

Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-EnvTestTaskMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

InvalidParameterCombination

The cmdlet parameters are inconsistent.

UnknownObject

One of the specified objects was not found.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Set-EnvTestTaskMetadata -TaskTaskId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
---	----
property	value

Add metadata with a name of 'property' and a value of 'value' to the Task with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Start-EnvTestTask

May 10, 2016  
Starts a new test task.

### Syntax

Start-EnvTestTask -TestId <String> [-TargetIdType <String>] [-TargetId <String>] [-CultureName <String>] [-IgnoreRelatedObjects] [-RunAsynchronously] [-ExcludeNotRunTests] [-AdminAddress <String>] [<CommonParameters>]

Start-EnvTestTask -TestSuiteId <String> [-TargetIdType <String>] [-TargetId <String>] [-CultureName <String>] [-IgnoreRelatedObjects] [-RunAsynchronously] [-ExcludeNotRunTests] [-AdminAddress <String>] [<CommonParameters>]

Start-EnvTestTask -InputObject <PSObject[]> [-CultureName <String>] [-IgnoreRelatedObjects] [-RunAsynchronously] [-ExcludeNotRunTests] [-AdminAddress <String>] [<CommonParameters>]

### Detailed Description

Starts a new test task based on a set of criteria provided via parameters or piped input and either waits for the tests to run or returns immediately depending on how it is called. When running a test suite and providing a target object for that suite, the service will discover related objects by default, but this behavior may be disabled if desired.

### Related topics

- [Get-EnvTestDefinition](#)
- [Get-EnvTestSuiteDefinition](#)
- [Get-EnvTestTask](#)
- [New-EnvTestDiscoveryTargetDefinition](#)
- [Switch-EnvTestTask](#)
- [Stop-EnvTestTask](#)
- [Remove-EnvTestTask](#)
- [Add-EnvTestTaskMetadata](#)
- [Remove-EnvTestTaskMetadata](#)

### Parameters

**-TestId**<String>

Test identifiers. If specified, do not specify -TestSuiteId.

Required?	true
Default Value	Empty
Accept Pipeline Input?	false

**-TestSuiteId**<String>

Test suite identifiers. If specified, do not specify -TestId.

Required?	true
Default Value	Empty
Accept Pipeline Input?	false

**-InputObject**<PSObject[]>

One or more test targets defining the task, see Input Types for details about what kind of objects are permissible. Any valid object passed to this parameter may also be piped into this cmdlet.

Required?	true
Default Value	Empty
Accept Pipeline Input?	true (ByValue)

**-TargetIdType**<String>

Describes the type of corresponding object passed with -TargetId

Required?	false
-----------	-------

Default Value	Empty
Accept Pipeline Input?	false

#### -TargetId<String>

The Ids that object tests or tests suites will target. By default, other components are queried for objects related to these.

Required?	false
Default Value	Empty
Accept Pipeline Input?	false

#### -CultureName<String>

The culture name in which to produce results. The culture name is in standard language/region-code format; for example "en-US".

Required?	false
Default Value	The current user interface culture
Accept Pipeline Input?	false

#### -IgnoreRelatedObjects<SwitchParameter>

Do not ask other components for objects related to a specified target.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### -RunAsynchronously<SwitchParameter>

Do not wait for the test run to complete, return immediately.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### -ExcludeNotRunTests<SwitchParameter>

If set, tests that are not run because no object matching their requirements is found are NOT included in test counts and results.

Required?	false
Default Value	False (include Not Run tests)
Accept Pipeline Input?	false

#### -AdminAddress<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

Input Type

Citrix.EnvTest.Sdk.EnvTestDiscoveryTargetDefinition A single EnvTestDiscoveryTargetDefinition can be specified to target one test or test suite. Citrix.EnvTest.Sdk.EnvTestDiscoveryTargetDefinition[] An array of EnvTestDiscoveryTargetDefinition(s) can be specified to target any combination of tests and/or test suites. PSCustomObject A single PSCustomObject with properties matching the required EnvTestDiscoveryTargetDefinition properties can be specified to target one test or test suite. PSCustomObject[] An array of PSCustomObject(s) with properties matching the required EnvTestDiscoveryTargetDefinition properties can be specified to target any combination of tests and/or test suites.

#### Return Values

Citrix.EnvTest.Sdk.EnvTestTask

The newly started task.

#### Examples

##### ----- EXAMPLE 1 -----

```
$singleSimpleTestTask = Start-EnvTestTask -TestId Monitor_RegisteredWithConfigurationService
```

Create and start a task with a single test and no target object.

##### ----- EXAMPLE 2 -----

```
$singleSimpleTestTaskInSpanish = Start-EnvTestTask -TestId Monitor_RegisteredWithConfigurationService -CultureName es-ES
```

Create and start a task with a single test and no target object, with localized properties translated into Spanish.

##### ----- EXAMPLE 3 -----

```
$singleSimpleTestSuiteTask = Start-EnvTestTask -TestSuiteId Infrastructure
```

Create and start a task with a single test suite and no target object.

##### ----- EXAMPLE 4 -----

```
$singleTestSuiteTask = Start-EnvTestTask -TestSuiteId Catalog -TargetIdType Catalog -TargetId $(Get-BrokerCatalog).Uuid
```

Create and start a task with a single test suite and a catalog target object.

##### ----- EXAMPLE 5 -----

```
$singleTestSuiteTask = Start-EnvTestTask -TestSuiteId Catalog -TargetIdType Catalog -TargetId $(Get-BrokerCatalog).Uuid -RunAsynchronously
```

Create and start a task with a single test suite and a catalog target object, and return immediately not waiting for the tests to complete.

##### ----- EXAMPLE 6 -----

```
$adAccountPool = Get-AcctIdentityPool
```

```
$singleTestTaskWithNoObjectDiscovery = StartEnvTestTask -IgnoreRelatedObjects -TestId ADIdentity_IdentityPool_ValidatePoolsUnlocked -TargetIdType IdentityPool -TargetId $adAccountPool.
```

Create and start a task with a single test, a target object for that test, and no object discovery based on that target.

##### ----- EXAMPLE 7 -----

```
$singleSimpleTestSuiteTaskTarget = New-EnvTestDiscoveryTargetDefinition -TestSuiteId Infrastructure
```

```
$singleTestSuiteTaskTarget = New-EnvTestDiscoveryTargetDefinition -TestSuiteId Catalog -TargetIdType Catalog -TargetId $(Get-BrokerCatalog).Uuid
```

```
$inputObjects = @($singleSimpleTestSuiteTaskTarget, $singleTestSuiteTaskTarget)
```

```
Start-EnvTestTask -InputObject $inputObjects
```

Create two different discovery target definitions, put them in an array, then start a task based on both via -InputObject.

# Stop-EnvTestTask

May 10, 2016  
Stops a still running task from completing.

## Syntax

```
Stop-EnvTestTask [-TaskId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
Stop-EnvTestTask [-Task <EnvTestTask>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Stops a still running task from completing. A task may still be retrieved via Get-EnvTestTask until it Remove-EnvTestTask is called with its task id.

## Related topics

- [Get-EnvTestDefinition](#)
- [Get-EnvTestSuiteDefinition](#)
- [Get-EnvTestTask](#)
- [New-EnvTestTask](#)
- [Start-EnvTestTask](#)
- [Switch-EnvTestTask](#)
- [Remove-EnvTestTask](#)
- [Add-EnvTestTaskMetadata](#)
- [Remove-EnvTestTaskMetadata](#)

## Parameters

**-TaskId**<Guid>  
The id of the task to stop, retrieveable from the \$task.TaskId property.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Task**<EnvTestTask>  
An EnvTestTask representing the task to stop



Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Examples

#### ----- **EXAMPLE 1** -----

Stop-EnvTestTask

Stops the current task from completing if it is still running.

#### ----- **EXAMPLE 2** -----

Stop-EnvTestTask -TestId 50A4139F-2B55-4A97-A1BE-20EE4E124AA3

Stops a task from completing via its id, which is available from an existing task's \$task.TaskId property.

#### ----- **EXAMPLE 3** -----

\$secondTask = \$(Get-EnvTestTask -List)[1]

Stop-EnvTestTask -Task \$secondTask

Stops the second task in the list returned by Get-EnvTestTask -List.

# Switch-EnvTestTask

May 10, 2016

Sets the current task that will be returned by a call to Get-EnvTestTask with no parameters.

## Syntax

Switch-EnvTestTask [-TaskId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Switch-EnvTestTask [-Task <EnvTestTask>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Sets the current task that will be returned by a call to Get-EnvTestTask with no parameters.

## Related topics

[Get-EnvTestDefinition](#)

[Get-EnvTestSuiteDefinition](#)

[Get-EnvTestTask](#)

[New-EnvTestTask](#)

[Start-EnvTestTask](#)

[Stop-EnvTestTask](#)

[Remove-EnvTestTask](#)

[Add-EnvTestTaskMetadata](#)

[Remove-EnvTestTaskMetadata](#)

## Parameters

**-TaskId**<Guid>

Specifies the identifier of the task to be made current.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Task**<EnvTestTask>

The task object to be made current, retrieveable from the \$task.TaskId property.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Input Type

System.Management.Automation.PSObject Objects containing the TaskId parameter can be piped to the Remove-EnvTestTask command.

### Examples

#### ----- EXAMPLE 1 -----

```
Switch-EnvTestTask -TaskId 50A4139F-2B55-4A97-A1BE-20EE4E124AA3
```

Switches the current task to another via its id, which is available from an existing task's Stask.TaskId property.

#### ----- EXAMPLE 2 -----

```
$secondTask = $(Get-EnvTestTask -List)[1]
```

```
Switch-EnvTestTask -Task $switchTask
```

Switches the current task to the second in the list returned by Get-EnvTestTask -List.

# Test-EnvTestDBConnection

May 10, 2016

Tests a database connection for the EnvTest Service.

## Syntax

```
Test-EnvTestDBConnection [-DBConnection] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Tests a connection to the database in which the EnvTest Service can store its state. The service will attempt to connect to the database without affecting the current connection to the database.

You do not have to clear the connection to use this command.

## Related topics

[Get-EnvTestServiceStatus](#)

[Get-EnvTestDBConnection](#)

[Set-EnvTestDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be tested by the EnvTest Service.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Test-EnvTestDBConnection command returns an object containing the status of the EnvTest Service if the connection string of the specified data store were to be set to the string being tested, together with extra diagnostics information for the specified connection string.

DBRejectedConnection

The database rejected the logon attempt from the EnvTest Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the EnvTest Service schema has not been added to the database.

DBNotFound

The specified database could not be located with the configured connection string.

DBNewerVersionThanService

The version of the EnvTest Service currently in use is incompatible with the version of the EnvTest Service schema on the database. Upgrade the EnvTest Service to a more recent version.

DBOlderVersionThanService

The version of the EnvTest Service schema on the database is incompatible with the version of the EnvTest Service currently in use. Upgrade the database schema to a more recent version.

DBVersionChangeInProgress

A database schema upgrade is currently in progress.

OK

The Set-EnvTestDBConnection command would succeed if it were executed with the supplied connection string.

Failed

The EnvTest Service has failed.

Unknown

The status of the EnvTest Service cannot be determined.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

InvalidDBConnectionString

The database connection string has an invalid format.

PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Test-EnvTestDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Tests a database connection string for the EnvTest Service.

##### ----- EXAMPLE 2 -----

```
c:\PS>Test-EnvTestDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Tests an invalid database connection string for the EnvTest Service.

# Citrix.Host.Admin.V2

May 10, 2016

## Overview

Name	Description
<a href="#">HypHostSnapin</a>	The Host Service PowerShell snap-in provides administrative functions for
<a href="#">Hyp Filtering</a>	Describes the common filtering options for XenDesktop cmdlets.

## Cmdlets

Name	Description
<a href="#">Add-HypHostingUnitMetadata</a>	Adds metadata on the given HostingUnit.
<a href="#">Add-HypHostingUnitNetwork</a>	Makes additional hypervisor networks available for use in a HostingUnit.
<a href="#">Add-HypHostingUnitStorage</a>	Adds storage locations to a hosting unit.
<a href="#">Add-HypHypervisorConnectionAddress</a>	Add a connection address to a hypervisor connection.
<a href="#">Add-HypHypervisorConnectionMetadata</a>	Adds metadata on the given HypervisorConnection.
<a href="#">Add-HypHypervisorConnectionScope</a>	Add the specified HypervisorConnection(s) to the given scope(s).
<a href="#">Add-HypMetadata</a>	Adds metadata to a hypervisor connection or a hosting unit.
<a href="#">Get-HypConfigurationDataForItem</a>	Retrieves the configuration data for an item in the Host Service provider path. Note: For this release, only VM items are supported for this operation.
<a href="#">Get-HypConfigurationObjectForItem</a>	Retrieves the configuration data for an item in the Host Service provider path. Note: For this release, only VM items are supported for this operation.
<a href="#">Get-HypConnectionRegion</a>	Enumerates the regions of a hypervisor connection that are based on cloud technology.
<a href="#">Get-HypDBConnection</a>	Gets the database string for the specified data store used by the Host Service.

<b>Get-HypDBSchema</b> Name	Description
	Gets a script that creates the Host Service database schema for the specified data store.
Get-HypDBVersionChangeScript	Gets a script that updates the Host Service database schema.
Get-HypHypervisorPlugin	Gets the available hypervisor types.
Get-HypInstalledDBVersion	Gets a list of all available database schema versions for the Host Service.
Get-HypScopedObject	Gets the details of the scoped objects for the Host Service.
Get-HypService	Gets the service record entries for the Host Service.
Get-HypServiceAddedCapability	Gets any added capabilities for the Host Service on the controller.
Get-HypServiceInstance	Gets the service instance entries for the Host Service.
Get-HypServiceStatus	Gets the current status of the Host Service on the controller.
Get-HypVMMacAddress	Retrieves a list the MAC addresses for the VMs in the specified connection.
Get-HypVolumeServiceConfiguration	Gets instances of the VolumeServiceConfiguration that are configured for this site.
Get-HypXenServerAddress	Gets all the available addresses for a XenServer hypervisor connection.
Grant-HypSecurityGroupEgress	Adds an egress rule to a security group.
Grant-HypSecurityGroupIngress	Adds an ingress rule to a security group.
New-HypVMSnapshot	Creates a new snapshot for the specified VM item path.
Remove-HypHostingUnitMetadata	Removes metadata from the given HostingUnit.
Remove-HypHostingUnitNetwork	Removes networks from a hosting unit.
Remove-HypHostingUnitStorage	Removes storage from a hosting unit.
Remove-HypHypervisorConnectionAddress	Removes addresses from the list of available connection addresses.



<b>Remove-Name</b>	<b>Description</b>
<a href="#">Remove-HypHypervisorConnectionMetadata</a>	Removes metadata from the given HypervisorConnection.
<a href="#">Remove-HypHypervisorConnectionScope</a>	Remove the specified HypervisorConnection(s) from the given scope(s).
<a href="#">Remove-HypMetadata</a>	Removes metadata from a hypervisor connection or hosting unit.
<a href="#">Remove-HypServiceMetadata</a>	Removes metadata from the given Service.
<a href="#">Reset-HypServiceGroupMembership</a>	Reloads the access permissions and configuration service locations for the Host Service.
<a href="#">Revoke-HypSecurityGroupEgress</a>	Removes an egress rule from a security group.
<a href="#">Revoke-HypSecurityGroupIngress</a>	Removes an ingress rule from a security group.
<a href="#">Set-HypAdminConnection</a>	Set the controller to be used by the cmdlets that form the Host service PowerShell snap-in.
<a href="#">Set-HypDBConnection</a>	Configures a database connection for the Host Service.
<a href="#">Set-HypHostingUnitMetadata</a>	Adds or updates metadata on the given HostingUnit.
<a href="#">Set-HypHostingUnitStorage</a>	Sets options for a storage location on a hosting unit.
<a href="#">Set-HypHypervisorConnectionMetadata</a>	Adds or updates metadata on the given HypervisorConnection.
<a href="#">Set-HypServiceMetadata</a>	Adds or updates metadata on the given Service.
<a href="#">Set-HypVolumeServiceConfiguration</a>	Applies a change to one of the VolumeServiceConfiguration instances in the site.
<a href="#">Start-HypVM</a>	Starts a VM.
<a href="#">Stop-HypVM</a>	Stops a VM by issuing a Shutdown request
<a href="#">Test-HypDBConnection</a>	Tests a database connection for the Host Service.
<a href="#">Test-HypHostingUnitNameAvailable</a>	Checks to ensure that the proposed name for a hosting unit is unused.
<a href="#">Test-</a>	Checks to ensure that the proposed name for a hypervisor connection is

<a href="#">HypervisorConnectionNameAvailable</a> <b>Name</b>	unused. <b>Description</b>
<a href="#">Update-HypervisorConnection</a>	Requests the host service to update the connection properties that depend on the version of hypervisor in use.

# about\_HypHostSnapin

May 10, 2016

## TOPIC

about\_HypHostSnapin

## SHORT DESCRIPTION

The Host Service PowerShell snap-in provides administrative functions for the Host Service.

## COMMAND PREFIX

All commands in this snap-in have 'Hyp' in their name.

## LONG DESCRIPTION

The Host Service PowerShell snap-in enables both local and remote administration of the Host Service. It lets you configure XenDesktop deployments to make use of hypervisors, networks, and storage, and enables browsing of their contents using the Host PowerShell provider (Citrix.HypervisorProvider).

The provider creates a default PSDrive with a drive identifier of 'XDHyp'. This drive provides two root folders:

### HostingUnits Folder

Contains a list of all the hosting units that have been defined using the new-Item provider command.

Hosting units define not only a hypervisor connection, but also reference networks and storage. Hosting units are used by the Machine Creation Service to provide the information required to create and manage virtual machines that can be used by other services. A hosting unit references a root path. This is a specific point in the provider connection tree. The hosting unit is constrained to provide only items below this point in the tree. This restricts the locations that the Machine Creation Service can use to create virtual machines and the networks and storage that can be used.

### Connections Folder

Contains a list of all the hosting unit connections that are defined using the new-Item provider command.

Change directory to a specific connection and use the dir/Get-ChildItem command to list all of the infrastructure (such as folders, hypervisors, networks, storage, and virtual machines) that is available in the hosting unit to which the connection refers. The paths to these items

are used as the input to commands in the Host Service and Machine Creation Service snap-ins.

The contents of the Hosting Unit and Connection folders reflect the content and structure of the hypervisor to which they refer. The item extensions reflect the object type for each item. Not all item types are appropriate for all hypervisor types. Possible item types are:

- Virtual Machine (.vm)
- Snapshot (.snapshot)
- Cluster (.cluster)
- Host (.host)
- HostGroup (.hostgroup)
- DataCenter (.datacenter)
- Folder (.folder)
- ResourcePool (.resourcepool)
- ComputeResource (.computeresource)

When used with a path that refers to the Host provider (with a default drive of 'XDHyp:'), the Host provider extends the standard New-Item, Get-Item, Get-ChildItem, Remove-Item, Rename-Item, and Set-Item commands as described below. For more information about the basic behavior of these commands, see the help for the command. The Move-Item and Copy-Item commands are not supported for the Host provider.

#### New-Item

The following parameters are available when using New-Item in the Connections directory (or a path that refers to the directory). The Credential parameter is not supported.

##### -Name

Specifies the name of the connection, which must not contain any of the following characters: \;:#.\*?=<>|[](){}". The Name parameter is optional but, if not included, the connection name must be specified as part of the Path parameter.

##### -HypervisorAddress <String[]>

Specifies an array of addresses that can be used to contact the required hypervisor. All the addresses are considered equivalent, that is, all of the addresses provide access to the same virtual machines, snapshots, network, and storage.

##### -ConnectionType <ConnectionType>

Specifies the type of hypervisor that the connection is for. Supported hypervisor types are:

XenServer

SCVMM (Microsoft Hyper-V)  
vCenter (VMware vSphere/ESX)  
Custom

**-PluginId <String>**

Specifies the class name for the Citrix Managed Machine library that is used to access the hypervisor. You can obtain this list using the `Get-HypHypervisorPlugin` command. The `PluginId` parameter must be specified if the `ConnectionType` is set to 'Custom'.

**-UserName <String>, -Password <String>, -SecurePassword <SecureString>**

Specifies the credentials for the connection to the hypervisor. You can specify the password as either `Password` or `SecurePassword`, but not both. The `UserName` parameter, and either `Password` or `SecurePassword`, specify the same information as the `HypervisorCredential` parameter, so use of one precludes use of the other.

**-HypervisorCredential <PSCredential>**

Specifies credentials for the connection to the hypervisor. The `HypervisorCredential` parameter specifies the same information as `UserName` and either `Password` or `SecurePassword`, so use of one precludes use of the other.

**-Persist**

Specifies that the connection details are persistent. If this parameter is not included, the connection is held only for the duration of the current runspace and `PSDrive` combination. Only persistent connection items are visible to administrators in other runspace and `PSDrives`. Hosting units cannot be created from connections that are not persistent.

**-AdminAddress <String>**

Specifies the address of the Host Service that the command communicates with. After it is set, this address is used for all commands in the Host Service PowerShell snap-in. If this parameter is not included or set by another command, or if `Set-HypAdminConnection` has not been used, the command attempts to use a local Host Service.

**-LoggingId <String>**

Specifies the identifier of the high-level operation that this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the `Start-LogHighLevelOperation` and `Stop-LogHighLevelOperation` cmdlets.

**-Scopes <String[]>**

Specifies the list of administrative scopes that the new connection will be a part of. The scopes control which administrators are able to work with the connection.

The following parameters are available when using New-Item in the HostingUnits directory (or a path that refers to the directory).

**-Name**

Specifies the name of the hosting unit, which must not contain any of the following characters: \;#.\*?=<>|[](){}'. The Name parameter is optional but, if not included, the hosting unit name must be provided as part of the Path parameter.

**-HypervisorConnectionName <String>**

Specifies the name of the connection that the hosting unit uses. To create a hosting unit, the referenced connection must be persistent and the ConnectionType must not be set to 'Custom'.

**-HypervisorConnectionUid <Guid>**

Specifies the unique identifier of the connection that the hosting unit uses. To create a hosting unit, the referenced connection must be persistent and the ConnectionType must not be set to 'Custom'.

**-RootPath <String>**

Specifies the location in a connection that is used as the starting reference for the hosting unit. The path must point to an item in the connection that is marked as a SymLink. The root of a XenServer connection is a special case that is also considered a SymLink. If this parameter is not included, the current location in the provider is used.

**-NetworkPath <String>**

Specifies the path in a connection to the network item that is used when the Machine Creation Service creates new virtual machines.

**-StoragePath <String>**

Specifies one or more paths in a connection to storage items that are used when the Machine Creation Service creates new virtual machines. After they are set, you can modify storage paths using the Add-HypHostingUnitStorage and Remove-HypHostingUnitStorage commands. If the connection is based on cloud infrastructure, storage items are typically not available, in which case this parameter can be omitted.

**-PersonalvDiskStoragePath <String>**

Specifies one or more paths in a connection to storage items

that are used when the Machine Creation Service creates disks for the virtual machines. After they are set, you can modify storage paths using the Add-HypHostingUnitStorage and Remove-HypHostingUnitStorage commands.

**-NoVmTagging**

Specifies that new virtual machines are not tagged with metadata from the hypervisor. By default, all virtual machines created by the Machine Creation Service are tagged to show they are created by XenDesktop. These tags are used by the provider to restrict the list of virtual machines displayed when viewing the content of the Connections or HostingUnit paths. If this parameter is not included, all virtual machines are displayed at all times.

**-AdminAddress <String>**

Specifies the address of the Host Service that the command will communicate with. After it is set, this address is used for all commands in the Host Service PowerShell snap-in. If this parameter is not included or set by another command, or if Set-HypAdminConnection has not been used, the command attempts to use a local Host Service.

**-UseLocalStorageCaching**

When Get-HypServiceAddedCapability indicates that the LocalStorageCaching feature is available, use this parameter to specify that the virtual machines created for this hosting unit will use local storage caching for their disk images.

**-GpuGroup**

Specifies a path to a GPU Group in a connection that will be used when

Machine Creation Services creates VMs. Only a single GPU Group is supported and the value is immutable.

**-LoggingId <String>**

Specifies the identifier of the high-level operation that this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

The following parameters are available when using New-Item relative to a connection or hosting unit.

**-ItemType <string>**

Specifies the type of item to create when invoked relative to a connection or hosting unit. Supported ItemType values are:

SecurityGroup (cloud only)

-Description

Specifies a description for the security group.

-VpcId

Specifies a VPC ID for the security group.

Get-Item

Get-ChildItem (alias dir)

Rename-Item

Remove-Item

The following additional parameters are available.

AdminAddress

Specifies the address of the Host Service that the command communicates with. After it is set, this address is used for all commands in the Host Service PowerShell snap-in. If this parameter is not included or set by another command, or if Set-HypAdminConnection has not been used, the command attempts to use a local Host Service.

-LoggingId <String>

Specifies the identifier of the high-level operation that this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

You can specify the Path parameter by name or by using the unique identifier for the connection or hosting unit enclosed in braces.

For example:

-Path XDHyp:\{1233-3213ACDF-12323}

The Filter parameter is not supported. Virtual machines created by the Machine Creation Service are returned only if the -Force parameter is used or if virtual machine tagging was disabled on the hosting unit with the NoVmTagging parameter when the VMs were created.



## Set-Item

The following parameters are available when using Set-Item in the Connections directory.

**-UserName <String>, -Password <String>, -SecurePassword <SecureString>**

Specifies the credentials for the connection to the hypervisor. The password can be given as either Password or SecurePassword, but not both. The UserName parameter and either Password or SecurePassword specify the same information as the HypervisorCredential parameter, so use of one precludes use of the other.

**-HypervisorCredential <PSCredential>**

Specifies credentials for the connection to the hypervisor. The HypervisorCredential parameter specifies the same information as the UserName parameter and either Password or SecurePassword, so use of one precludes use of the other.

**-HypervisorAddress <string[]>**

Specifies the addresses of the hypervisor that this connection represents. This replaces all existing addresses.

**-LoggingId <String>**

Specifies the identifier of the high-level operation that this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

**-MaintenanceMode <Boolean>**

Places the connection into maintenance mode which disables all communication between XenDesktop and the Hypervisor. Use this mode when making changes to the hypervisor; for instance, if the password for the access account needs to be changed.

**-AdminAddress <String>**

Specifies the address of the Host Service that the command communicates with. After it is set, this address is used for all commands in the Host Service PowerShell snap-in. If this parameter is not included or set by another command, or if Set-HypAdminConnection has not been used, the command tries to use a local Host Service.

The Credential parameter is not supported. Addresses can be modified with the Add-HypHypervisorConnectionAddress and Remove-HypHypervisorConnectionAddress commands. Metadata can be modified with the Add-HypMetadata and Remove-HypMetadata commands.

The following parameters are available when using New-Item in the

HostingUnits directory.

**-Name**

Specifies the name of the hosting unit, which must not contain any of the following characters: \;#.\*?=<>|[](){}'. The Name parameter is optional but, if not included, the hosting unit name must be provided as part of the Path parameter.

**-NetworkPath <String>**

Specifies the path in a connection to the network item that is used when the Machine Creation Service creates new virtual machines.

**-NoVmTagging**

Specifies that new virtual machines are not tagged with metadata from the hypervisor. By default, all virtual machines created by the Machine Creation Service are tagged to show they are created by XenDesktop. These tags are used by the provider to restrict the list of virtual machines displayed when viewing the content of the Connections or HostingUnit paths. If this parameter is not included, all virtual machines are displayed at all times.

**-AdminAddress <String>**

Specifies the address of the Host Service that the command communicates with. After it is set, this address is used for all commands in the Host Service PowerShell snap-in. If this parameter is not included or set by another command, or if Set-HypAdminConnection has not been used, the command attempts to use a local Host Service.

**-UseLocalStorageCaching <bool>**

When Get-HypServiceAddedCapability indicates that the LocalStorageCaching feature is available, use this parameter to specify that the virtual machines created for this hosting unit will use local storage caching for their disk images.

**-LoggingId <String>**

Specifies the identifier of the high-level operation that this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Storage can be modified with the Add-HypHostingUnitStorage and Remove-HypHostingUnitStorage commands. Metadata can be modified with the Add-HypMetadata and Remove-HypMetadata commands.

The item types returned when these commands are used in the Host Service provider are defined in the Object Definitions section below.

## EXAMPLES

### To Create a Persistent Hypervisor Connection

```
new-item -path "xdhyp:\Connections" -Name MyConn -ConnectionType
XenServer -HypervisorAddress "http:\address" -UserName user
-Passwd password -Persist
```

```
PSPPath: Citrix.HostingUnitService.Admin.V1.0\
 Citrix.Hypervisor::XDHyp:\Connections\MyConn
PSParentPath: Citrix.HostingUnitService.Admin.V1.0\
 Citrix.Hypervisor::XDHyp:\Connections
PSChildName: MyConn
PSDrive: XDHyp
PSProvider: Citrix.HostingUnitService.Admin.V1.0\
 Citrix.Hypervisor
PSIsContainer: True
HypervisorConnectionUid: 04e6daa2-5cbd-4491-b70c-6daf733ee82a
HypervisorConnectionName: MyConn
ConnectionType: XenServer
HypervisorAddress: {http:\address}
UserName: user
Persistent: True
PluginId: XenFactory
SupportsPvsVMs: True
Revision: 1b0b0d02-bc1b-49d8-b2b0-be6fb7f150ad
MaintenanceMode: False
Metadata: {MaxAbsoluteActiveActions = 100,
 MaxAbsoluteNewActionsPerMinute = 100,
 MaxPowerActionsPercentageOfDesktops = 20}
```

### To Create a Hosting Unit

```
new-item -Path "xdhyp:\HostingUnits"
-Name MyHU
-HypervisorConnectionName MyConn
-RootPath XDHYP:\Connections\MyConn
-NetworkPath XDHYP:\Connections\MyConn\Network 0.network
-StoragePath XDHYP:\Connections\MyConn\Local storage on
myXenServer.storage
```

```
PSPPath: Citrix.HostingUnitService.Admin.V1.0\
 Citrix.Hypervisor::XDHyp:\HostingUnits\MyHU
PSParentPath: Citrix.HostingUnitService.Admin.V1.0\
 Citrix.Hypervisor::XDHyp:\HostingUnits
```

```
PSChildName: MyHU
PSDrive: XDHyp
PSProvider: Citrix.HostingUnitService.Admin.V1.0\
 Citrix.Hypervisor
PSIsContainer: True
HostingUnitUid: df91f886-1141-4280-bd59-2ee260a4df79
HostingUnitName: MyHU
HypervisorConnection: MyConn
RootPath: /
RootId:
NetworkPath: /Network 0.network
NetworkId: ab47080b-ca15-771a-c8dc-6ad9650158f1
Storage: {/Local storage on myXenServer.storage}
VMTaggingEnabled: True
Metadata: {}
```

Relative paths can be used for all parameters.

## OBJECT DEFINITIONS

Citrix.XDInterServiceTypes.HypervisorConnection

The hypervisor connection object is returned when a connection item is located. This item has the following parameters.

HypervisorConnectionUid <Guid>

Specifies the unique identifier for the connection item.

HypervisorConnectionName <String>

Specifies the name of the connection item.

ConnectionType <Citrix.XDInterServiceTypes.ConnectionType>

Specifies the type of hypervisor that the connection is for.

Supported hypervisor types are:

- XenServer
- SCVMM (Microsoft Hyper-V)
- vCenter (VMware vSphere/ESX)
- Custom

HypervisorAddress <String[]>

Specifies the addresses that can be used to contact the required hypervisor.

Username <String>

Specifies the administrator user name for the connection to the hypervisor (from the user name and password given as administrator credentials when

setting up this connection)

Persistent <Boolean>

Specifies whether or not the connection is persistent.

PluginId <String>

Specifies the Citrix Managed Machine class identifier for the hypervisor.

SupportsPvsVM <Boolean>

Specifies whether or not the connection can be used as part of a hosting unit. If this parameter is set to 'True', a hosting unit referencing this connection can be created.

Revision <Guid>

A unique identifier that is changed every time any properties of the connection are changed.

MaintenanceMode <Boolean>

Specifies whether or not the connection is currently in maintenance mode.

Metadata <Citrix.XDInterServiceTypes.Metadata[]>

The collection of metadata associated with the connection.

Citrix.XDInterServiceTypes.HostingUnit

The hosting unit object is returned when a hosting unit item is located. This item has the following parameters.

HostingUnitName <String>

Specifies the name of the hosting unit.

HostingUnitUid <Guid>

Specifies the unique identifier for the hosting unit.

HypervisorConnection <Citrix.XDInterServiceTypes.HypervisorConnection>

Specifies the hypervisor connection item that the hosting unit references.

NetworkId <String>

Specifies the unique identifier for the network in the hypervisor context that the hosting unit references.

NetworkPath <String>

Specifies the path to the network in the Host Service provider.

RootId <String>

Specifies the unique identifier for the root connection item in the hypervisor context that the hosting unit references.

RootPath <String>

Specifies the path to the root of the hosting unit from within the Host Service provider.

Storage <Citrix.XDInterServiceTypes.Storage[]>

Specifies the collection of storage objects that are defined for use as part of the hosting unit. This object contains the following parameters.

StorageID <String>

Specifies the identifier for the storage in the hypervisor context.

StoragePath <String>

Specifies the path to the storage in the Host Service provider.

VMTaggingEnabled <Boolean>

Specifies whether or not virtual machine metadata is used to tag virtual machines created by XenDesktop.

Metadata <Citrix.XDInterServiceTypes.Metadata[]>

Specifies the collection of metadata associated with the hosting unit.

Citrix.HostingUnitService.Sdk.HypervisorObject

The hypervisor object is returned when any item is located from within a Connection or HostingUnit folder. This item has the following parameters.

AdditionalData <Dictionary<String,String>

Stores extra untyped data for the item. This dictionary contains different information for each item type.

For Storage Items

Key = StorageType

Values = Shared or Local

For VM Items

Key = PowerState Values = PoweredOn,

PoweredOff,

Suspended,  
TransitioningToOn,  
TransitioningToOff,  
Suspending,  
Resuming,  
Unknown,  
Error

Name <String>  
Specifies the name of the item.

FullName <String>  
Specifies the name with the appropriate file extension.

ObjectPath <String>  
Specifies the relative path to the item from the root of the connection of which the item is part.

FullPath <String>  
Specifies the absolute path to the item in the Citrix.Hypervisor provider.

Id <String>  
Specifies the identity of the item in the hypervisor context.

IsContainer <Boolean>  
Specifies whether or not the item can contain other items.

IsSymLink <Boolean>  
Specifies whether or not the item can be used as a RootPath of a hosting unit.

ObjectType <Citrix.HostingUnitService.Sdk.NodeType>  
Specifies the item type.

## ERROR CODES

The provider commands can return the following error codes.

### New-Item

ConnectionNameOrUidInvalid  
The name or unique identifier of the hypervisor connection

specified for the hosting unit is invalid.

#### HostingUnitRootPathInvalid

The root path specified for the hosting unit is invalid. The root path must be a valid item in the hypervisor tree and a SymLink.

#### HostingUnitNetworkPathInvalid

The network path specified for the hosting unit is invalid. The network path must be a valid item in the hypervisor tree and relative to the root path.

#### HostingUnitStoragePathInvalid

The storage path specified for the hosting unit is invalid. The storage path must be a valid item in the hypervisor tree and relative to the root path.

#### HostingUnitDuplicateObjectExists

A hosting unit object with the same name already exists. Hosting unit names must be unique.

#### HostingUnitStorageDuplicateObjectExists

A hosting unit storage object with the same storage path already exists for this hosting unit. There can be only one object for each combination of hosting unit and storage path.

#### HypervisorNotContactable

The hypervisor could not be contacted at the supplied address, which is either invalid or unreachable.

#### HypervisorAddressInvalidFormat

The address is not in a valid form for the specified hypervisor type.

#### ConnectionAddressInvalid

The specified address is invalid and does not belong to the same pool.

#### HypervisorConnectionDuplicateObjectExists

A hypervisor connection object with the same name already exists. Hypervisor connection names must be unique.

#### HypervisorConnectionAddressDuplicateObjectExists

A hypervisor connection address object with the same address already exists for this hypervisor connection. There can be only one object for each combination of hypervisor connection and address.



#### HypervisorConnectionForHostingUnitsVirtual

The specified hypervisor connection for the hosting unit is a virtual

non-persistent connection. A persistent connection is required when creating a hosting unit.

#### InputNameInvalid

The name specified for the hosting unit or hypervisor connection is invalid because it contains one or more of the following characters:

V;:#.\*?=<>|[](){}.

#### InputPathInvalid

The specified path is invalid because it is not in one of the following formats:

XDHyp:\Connections\<Name>

XDHyp:\Connections\{Guid}

#### ExceptionThrown

An unexpected error occurred.

#### DatabaseError

There was a problem communicating with the database.

#### CommunicationError

There was a problem communicating with the remote service.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ScopeNotFound

One or more of the scopes nominated for the new connection do not exist.

#### CannotCreateSecurityGroupHere

Security groups can only be created in a virtual private cloud (VPC).

#### Get-Item and Get-ChildItem

#### HypervisorConnectionObjectNotFound

The hypervisor connection object specified in the path could not be found.

#### HostingUnitObjectNotFound

The specified hosting unit object could not be found.

#### HypervisorInMaintenanceMode

The hypervisor for the specified connection is currently in maintenance mode and cannot be accessed.

#### FailureToRetrieveConnectionPassword

The password for the HypervisorConnection cannot be retrieved or decrypted.

#### ExceptionThrown

An unexpected error occurred.

#### DatabaseError

There was a problem communicating with the database.

#### CommunicationError

There was a problem communicating with the remote service.

### Remove-Item

#### HostingUnitObjectToDeleteDoesNotExist

The specified hosting unit object does not exist.

#### HypervisorConnectionObjectToDeleteDoesNotExist

The specified hypervisor connection object does not exist.

#### InputPathInvalid

The specified path is invalid because it is not in one of the following formats:

XDHyp:\Connections\<Name>

XDHyp:\Connections\{Guid}

XDHyp:\HostingUnits\<Name>

XDHyp:\HostingUnits\{Guid}

#### HypervisorConnectionObjectToDeleteIsInUse

The specified hypervisor connection object cannot be deleted because it is being used by one or more hosting units.

#### ExceptionThrown

An unexpected error occurred.

#### DatabaseError

There was a problem communicating with the database.

#### CommunicationError

There was a problem communicating with the remote service.

#### ObjectCannotBeRemoved

The specified object cannot be removed.

## Rename-Item

### InputPathInvalid

The specified path is invalid because it is not in one of the following formats:

XDHyp:\Connections\<Name>  
XDHyp:\Connections\{Guid}  
XDHyp:\HostingUnits\<Name>  
XDHyp:\HostingUnits\{Guid}

### HostingUnitDuplicateNameExists

A hosting unit object with the same name already exists. Hosting unit names must be unique.

### HypervisorConnectionDuplicateNameExists

A hypervisor connection object with the same name already exists. Hypervisor connection names must be unique.

### InputNameInvalid

The new name specified for the hosting unit or hypervisor connection is invalid because it contains one or more of the following characters: \ ; # . \* ? = < > | [ ] ( ) ' " { } .

### ExceptionThrown

An unexpected error occurred.

### DatabaseError

There was a problem communicating with the database.

### CommunicationError

There was a problem communicating with the remote service.

## Set-Item

### InputPathInvalid

The specified path is invalid because it is not in one of the following formats:

XDHyp:\Connections\<Name>  
XDHyp:\Connections\{Guid}  
XDHyp:\HostingUnits\<Name>  
XDHyp:\HostingUnits\{Guid}

### HostingUnitObjectToUpdateDoesNotExist

The specified hosting unit object does not exist.

#### HostingUnitNetworkPathInvalid

The new network path specified for the hosting unit is invalid.  
Either the network path does not exist or it is not relative to the root path of the hosting unit.

#### HypervisorConnectionObjectToUpdateDoesNotExist

The specified hypervisor connection object does not exist.

#### ExceptionThrown

An unexpected error occurred.

#### DatabaseError

There was a problem communicating with the database.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation.  
Communication with the database failed for various reasons.

#### CommunicationError

There was a problem communicating with the remote service.

# about\_Hyp\_Filtering

May 10, 2016

## TOPIC

XenDesktop - Advanced Dataset Filtering

## SHORT DESCRIPTION

Describes the common filtering options for XenDesktop cmdlets.

## LONG DESCRIPTION

Some cmdlets operate on large quantities of data and, to reduce the overhead of sending all of that data over the network, many of the Get- cmdlets support server-side filtering of the results.

The conventional way of filtering results in PowerShell is to pipeline them into Where-Object, Select-Object, and Sort-Object, for example:

```
Get-<Noun> | Where { $_.Size = 'Small' } | Sort 'Date' | Select -First 10
```

However, for most XenDesktop cmdlets the data is stored remotely and it would be slow and inefficient to retrieve large amounts of data over the network and then discard most of it. Instead, many of the Get- cmdlets provide filtering parameters that allow results to be processed on the server, returning only the required results.

You can filter results by most object properties using parameters derived from the property name. You can also sort results or limit them to a specified number of records:

```
Get-<Noun> -Size 'Small' -SortBy 'Date' -MaxRecordCount 10
```

You can express more complex filter conditions using a syntax and set of operators very similar to those used by PowerShell expressions.

Those cmdlets that support filtering have the following common parameters:

-MaxRecordCount <int>

Specifies the maximum number of results to return.  
For example, to return only the first nine results use:

```
Get-<Noun> -MaxRecordCount 9
```

If not specified, only the first 250 records are returned, and if more are available, a warning is produced:

WARNING: Only first 250 records returned. Use -MaxRecordCount to retrieve more.

You can suppress this warning by using -WarningAction or by specifying a value for -MaxRecordCount.

To retrieve all records, specify a large number for -MaxRecordCount. As the value is an integer, you can use the following:

```
Get-<Noun> -MaxRecordCount [int]::MaxValue
```

-ReturnTotalRecordCount [<SwitchParameter>]

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. For example:

```
Get-<Noun> -MaxRecordCount 9 -ReturnTotalRecordCount
```

....

```
Get-<Noun> : Returned 9 of 10 items
```

```
At line:1 char:18
```

```
+ Get-<Noun> <<<< -MaxRecordCount 9 -ReturnTotalRecordCount
```

```
+ CategoryInfo : OperationStopped: (:) [Get-<Noun>], PartialDataException
```

```
+ FullyQualifiedErrorId : PartialData,Citrix.<SDKName>.SDK.Get<Noun>
```

The count can be accessed using the TotalAvailableResultCount property:

```
$count = $error[0].TotalAvailableResultCount
```

-Skip <int>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

-SortBy <string>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order, respectively. Ascending order is assumed if no prefix is present.

Sorting occurs before -MaxRecordCount and -Skip parameters are applied. For example, to sort by Name and then by Count (largest first) use:

```
-SortBy 'Name,-Count'
```

By default, sorting by an enumeration property uses the numeric value of the elements. You can specify a different sort order by qualifying the name with an ordered list of elements or their numeric values, or <null> to indicate the placement of null values.

Elements not mentioned are placed at the end in their numeric order. For example, to sort by two different enums and then by the object id:

```
-SortBy 'MyState(StateC,<null>,StateA,StateB),Another(0,3,2,1),Id'
```

-Filter <String>

This parameter lets you specify advanced filter expressions, and supports combination of conditions with -and and -or, and grouping with braces. For example:

```
Get-<Noun> -Filter 'Name -like "High*" -or (Priority -eq 1 -and Severity -ge 2)'
```

The syntax is close enough to PowerShell syntax that you can use script blocks in most cases. This can be easier to read as it reduces quoting:

```
Get-<Noun> -Filter { Count -ne $null }
```

The full -Filter syntax is provided below.

## EXAMPLES

Filtering by strings performs a case-insensitive wildcard match.

Separate parameters are combined with an implicit -and operator.

Normal PowerShell quoting rules apply, so you can use single or double quotes, and omit the quotes altogether for many strings. The order of parameters does not make any difference. The following are equivalent:

```
Get-<Noun> -Company Citrix -Product Xen*
Get-<Noun> -Company "citrix" -Product '[X]EN*'
Get-<Noun> -Product "Xen*" -Company "CITRIX"
Get-<Noun> -Filter { Company -eq 'Citrix' -and Product -like 'Xen*' }
```

See [about\\_Quoting\\_Rules](#) and [about\\_Wildcards](#) for details about PowerShell handling of quotes and wildcards.

To avoid wildcard matching or include quote characters, you can escape the wildcards using the normal PowerShell escape mechanisms (see [about\\_Escape\\_Characters](#)), or switch to a filter expression and the `-eq` operator:

```
Get-<Noun> -Company "Abc[*]" # Matches Abc*
Get-<Noun> -Company "Abc`*" # Matches Abc*
Get-<Noun> -Filter { Company -eq "Abc*" } # Matches Abc*
Get-<Noun> -Filter { Company -eq "A`"B`C`" } # Matches A"B'C
```

Simple filtering by numbers, booleans, and TimeSpans perform direct equality comparisons, although if the value is nullable you can also search for null values. Here are some examples:

```
Get-<Noun> -Uid 123
Get-<Noun> -Enabled $true
Get-<Noun> -Duration 1:30:40
Get-<Noun> -NullableProperty $null
```

More comparisons are possible using advanced filtering with `-Filter`:

```
Get-<Noun> -Filter 'Capacity -ge 10gb'
Get-<Noun> -Filter 'Age -ge 20 -and Age -lt 40'
Get-<Noun> -Filter 'VolumeLevel -like "[123]"'
Get-<Noun> -Filter 'Enabled -ne $false'
Get-<Noun> -Filter 'NullableProperty -ne $null'
```

You can check boolean values without an explicit comparison operator, and you can also combine them with `-not`:

```
Get-<Noun> -Filter 'Enabled' # Equivalent to 'Enabled -eq $true'
Get-<Noun> -Filter '-not Enabled' # Equivalent to 'Enabled -eq $false'
```

See [about\\_Comparison\\_Operators](#) for an explanation of the operators, but note that only a subset of PowerShell operators are supported (`-eq`, `-ne`, `-gt`, `-ge`, `-lt`, `-le`, `-like`, `-notlike`, `-in`, `-notin`, `-contains`, `-notcontains`).

Enumeration values can either be specified using typed values or the string name of the enumeration value:

```
Get-<Noun> -Shape [Shapes]::Square
Get-<Noun> -Shape Circle
```

With filter expressions, typed values can be specified with simple variables or quoted strings. They also support enumerations with wildcards:



```
$s = [Shapes]::Square
Get-<Noun> -Filter { Shape -eq $s -or Shape -eq "Circle" }
Get-<Noun> -Filter { Shape -like 'C*' }
```

By their nature, floating point values, DateTime values, and TimeSpan values are best suited to relative comparisons rather than just equality. DateTime strings are converted using the locale and time zone of the user device, but you can use ISO8601 format strings (YYYY-MM-DDThh:mm:ss.sTZD) to avoid ambiguity. You can also use standard PowerShell syntax to create these values:

```
Get-<Noun> -Filter { StartTime -ge "2010-08-23T12:30:00.0Z" }
$d = [DateTime]"2010-08-23T12:30:00.0Z"
Get-<Noun> -Filter { StartTime -ge $d }
$d = (Get-Date).AddDays(-1)
Get-<Noun> -Filter { StartTime -ge $d }
```

Relative times are quite common and, when using filter expressions, you can also specify DateTime values using a relative format:

```
Get-<Noun> -Filter { StartTime -ge '-2' } # Two days ago
Get-<Noun> -Filter { StartTime -ge '-1:30' } # Hour and a half ago
Get-<Noun> -Filter { StartTime -ge '-0:0:30' } # 30 seconds ago
```

## ARRAY PROPERTIES

When filtering against list or array properties, simple parameters perform a case-insensitive wildcard match against each of the members. With filter expressions, you can use the -contains and -notcontains operators. Unlike PowerShell, these perform wildcard matching on strings.

Note that for array properties the naming convention is for the returned property to be plural, but the parameter used to search for any match is singular. The following are equivalent (assuming Users is an array property):

```
Get-<Noun> -User Fred*
Get-<Noun> -Filter { User -like "Fred*" }
Get-<Noun> -Filter { Users -contains "Fred*" }
```

You can also use the singular form with -Filter to search using other operators:

```
Match if any user in the list is called "Frederick"
Get-<Noun> -Filter { User -eq "Frederick" }
Match if any user in the list has a name alphabetically below 'F'
Get-<Noun> -Filter { User -lt 'F' }
```

## COMPLEX EXPRESSIONS

When matching against multiple values, you can use a sequence of comparisons joined with -or operators, or you can use -in and -notin:

```
Get-<Noun> -Filter { Shape -eq 'Circle' -or Shape -eq 'Square' }
$shapes = 'Circle','Square'
Get-<Noun> -Filter { Shape -in $shapes }
$sides = 1..4
Get-<Noun> -Filter { Sides -notin $sides }
```

Braces can be used to group complex expressions, and override the default left-to-right evaluation of -and and -or. You can also use -not to invert the sense of any sub-expression:

```
Get-<Noun> -Filter { Size -gt 4 -or (Color -eq 'Blue' -and Shape -eq 'Circle') }
Get-<Noun> -Filter { Sides -lt 5 -and -not (Color -eq 'Blue' -and Shape -eq 'Circle') }
```

## PAGING

The simplest way to page through data is to use the -Skip and -MaxRecordCount parameters. So, to read the first three pages of data with 10 records per page, use:

```
Get-<Noun> -Skip 0 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 10 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 20 -MaxRecordCount 10 <other filtering criteria>
```

You must include the same filtering criteria on each call, and ensure that the data is sorted consistently.

The above approach is often acceptable, but as each call performs an independent query, data changes can result in records being skipped or appearing twice. One approach to improve this is to sort by a unique id field and then start the search for the next page at the unique id after the last unique id of the previous page. For example:

```
Get the first page
Get-<Noun> -MaxRecordCount 10 -SortBy SerialNumber

SerialNumber ...

A120004
A120007
... 7 other records ...
A120900

Get the next page
Get-<Noun> -MaxRecordCount 10 -Filter { FirstName -gt 'A120900' }

SerialNumber ...
```

----- ---  
A120901  
B220000  
...

## FILTER SYNTAX DEFINITION

<Filter> ::= <ScriptBlock> | <ComponentList>

<ScriptBlock> ::= "{" <ComponentList> "}"

<ComponentList> ::= <Component> <AndOrOperator> <ComponentList> |

<Component>

<Component> ::= <NotOperator> <Factor> |

<Factor>

<Factor> ::= "(" <ComponentList> ")" |

<PropertyName> <ComparisonOperator> <Value> |  
<PropertyName>

<AndOrOperator> ::= "-and" | "-or"

<NotOperator> ::= "-not" | "!"

<ComparisonOperator>

::= "-eq" | "-ne" | "-le" | "-ge" | "-lt" | "-gt" |  
"-like" | "-notlike" | "-contains" | "-notcontains" |  
"-in" | "-notin"

<PropertyName> ::= <simple name of property>

<Value> ::= <string literal> | <numeric literal> |

<scalar variable> | <array variable> |  
"\$null" | "\$true" | "\$false"

Numeric literals support decimal and hexadecimal literals, with optional multiplier suffixes (kb, mb, gb, tb, pb).

Dates and times can be specified as string literals. The current culture determines what formats are accepted. To avoid any ambiguity, use strings formatted to the ISO8601 standard. If not specified, the current time zone is used.

Relative date-time string literals are also supported, using a minus sign followed by a TimeSpan. For example, "-1:30" means 1 hour and 30 minutes ago.

# Add-HypHostingUnitMetadata

May 10, 2016  
Adds metadata on the given HostingUnit.

## Syntax

```
Add-HypHostingUnitMetadata [-HostingUnitUid] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Add-HypHostingUnitMetadata [-HostingUnitUid] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Add-HypHostingUnitMetadata [-HostingUnitName] <String> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Add-HypHostingUnitMetadata [-HostingUnitName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Add-HypHostingUnitMetadata [-InputObject] <HostingUnit[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Add-HypHostingUnitMetadata [-InputObject] <HostingUnit[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability for additional custom data to be stored against given HostingUnit objects. This cmdlet will not overwrite existing metadata on an object - use the Set-HypHostingUnitMetadata cmdlet instead.

## Related topics

- [Set-HypHostingUnitMetadata](#)
- [Remove-HypHostingUnitMetadata](#)

## Parameters

**-HostingUnitUid**<Guid>

Id of the HostingUnit

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-HostingUnitName**<String>

Name of the HostingUnit

--	--

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-InputObject**<HostingUnit[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the HostingUnit specified. The property cannot contain any of the following characters \/:;#.\*?=<>|[]()"

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Value**<String>

Specifies the value for the property.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

Citrix.Host.Sdk.Metadata

Add-HypHostingUnitMetadata returns an array of objects containing the new definition of the metadata.

\n Property <string>

\n Specifies the name of the property.

\n Value <string>

\n Specifies the value for the property.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

InvalidParameterCombination

The cmdlet parameters are inconsistent.

UnknownObject

One of the specified objects was not found.

DuplicateObject

One of the specified metadata already exists.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Add-HypHostingUnitMetadata -HostingUnitUid 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Property	Value
property	value

Add metadata with a name of 'property' and a value of 'value' to the HostingUnit with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.



# Add-HypHostingUnitNetwork

May 10, 2016

Makes additional hypervisor networks available for use in a HostingUnit.

## Syntax

Add-HypHostingUnitNetwork [-LiteralPath <String>] [-NetworkPath <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Use this command to extend the set of hypervisor networks that are made available through the HostingUnit to the Citrix Machine Creation Service. When new machines are created, their virtual NICs can be associated only with networks that are in this set. This command cannot be used if the connection for the hosting unit is in maintenance mode.

## Related topics

New-Item

[Add-HypMetadata](#)

[remove-HypHostingUnitNetwork](#)

## Parameters

**-LiteralPath**<String>

Specifies the path to the hosting unit to which the network will be added. The path must be in one of the following formats: <drive>:\HostingUnits\<HostingUnitName> or <drive>:\HostingUnits\{<HostingUnit Uid>}

Required?	true
Default Value	
Accept Pipeline Input?	false

**-NetworkPath**<String>

Specifies the path to the network that will be added. The path must be in one of the following formats: <drive>:\Connections\<ConnectionName>\MyNetwork.network or <drive>:\Connections\{<Connection Uid>\MyNetwork.network or <drive>:\HostingUnits\<HostingUnitName>\MyNetwork.network or <drive>:\HostingUnits\{<hostingUnit Uid>\MyNetwork.network

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. This can be a host name or an IP address.

Required?	false
Default Value	LocalHost. When a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

System.string You can pipe a string that contains a path to Add-HypHostingUnitNetwork (NetworkPath parameter).

#### Return Values

Citrix.Host.Sdk.HostingUnit

Add-HypHostingUnitNetwork returns an object containing the new definition of the hosting unit.

HostingUnitUid <Guid>

Specifies the unique identifier for the hosting unit.

HostingUnitName <string>

Specifies the name of the hosting unit.

HypervisorConnection <Citrix.Host.Sdk.HypervisorConnection>

Specifies the connection that the hosting unit uses to access a hypervisor.

RootId <string>

Identifies, to the hypervisor, the root of the hosting unit.

RootPath <string>

The hosting unit provider path that represents the root of the hosting unit.

Storage <Citrix.Host.Sdk.Storage[]>

The list of storage items that the hosting unit can use.

PersonalDiskStorage <Citrix.XDPowerShell.Storage[]>

The list of storage items that the hosting unit can use for storing personal data.

VMTaggingEnabled <Boolean>

Specifies whether or not the metadata in the hypervisor can be used to store information about the XenDesktop Machine Creation Service.

NetworkId <string>

The hypervisor's internal identifier that represents the default network specified for the hosting unit.

NetworkPath <string>

The hosting unit provider path to the default network specified for the hosting unit.

Metadata <Citrix.Host.Sdk.Metadata[]>

A list of key value pairs that can store additional information about the hosting unit.

PermittedNetworks <Citrix.Host.Sdk.Network[]>

A full list of the hypervisor networks that are exposed for use in the hosting unit.

#### Notes

The network path must be valid for the hosting unit. The rules that are applied are as follows: XenServer (HypervisorConnection Type = XenServer)

NA

VMWare vSphere/ESX (HypervisorConnection Type = vCenter)

The network path must be directly contained in the root path item of the hosting unit.

Microsoft SCVMM/Hyper-v (HypervisorConnection Type = SCVMM)

NA

In the case of failure, the following errors can result.

Error Codes

-----

HostingUnitsPathInvalid

The path provided is not to an item in a subdirectory of a hosting unit item.

HostingUnitNetworkPathInvalid

The specified path is invalid.

HostingUnitNetworkPathInvalid

The network path cannot be found or is invalid. See notes above about validity.

HostingUnitNetworkDuplicateObjectExists

The specified network path is already part of the hosting unit.

HypervisorInMaintenanceMode

The hypervisor for the connection is in maintenance mode.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation. Communication with the database failed for various reasons.

CommunicationError

An error occurred while communicating with the service.

PermissionDenied

The user does not have administrative rights to perform this operation.

ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used, or examine the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Add-HypHostingUnitNetwork -LiteralPath XDHyp:\HostingUnits\MyHostingUnit -NetworkPath 'XDHyp:\HostingUnits\MyHostingUnits\newNetwork.network'
```

```
HostingUnitUid : bcd3d649-86d1-4aa8-8ec2-d322b6a2c457
```

```
HostingUnitName : MyHostingUnit
```

```

HypervisorConnection : MyConnection
RootPath : /
RootId :
NetworkPath : /Network 0.network
NetworkId : ab47080b-ca15-771a-c8dc-6ad9650158f1
Storage : {/Local storage.storage}
PersonalvDiskStorage : {}
VMTaggingEnabled : True
Metadata : {}
PermittedNetworks : {/Network 0.network, /newNetwork.network}

```

The command adds a new network called "newNetwork.network" to the hosting unit called "MyHostingUnit".

#### ----- EXAMPLE 2 -----

```

XDHyp:\HostingUnits\MyHostingUnit>Add-HypHostingUnitNetwork -LiteralPath . -NetworkPath 'XDHyp:\HostingUnits\MyHostingUnits\newNetwork.network'

```

```

HostingUnitUid : bcd3d649-86d1-4aa8-8ec2-d322b6a2c457
HostingUnitName : MyHostingUnit
HypervisorConnection : MyConnection
RootPath : /
RootId :
NetworkPath : /Network 0.network
NetworkId : ab47080b-ca15-771a-c8dc-6ad9650158f1
Storage : {/Local storage.storage}
PersonalvDiskStorage : {}
VMTaggingEnabled : True
Metadata : {}
PermittedNetworks : {/Network 0.network, /newNetwork.network}

```

The command adds a new network called "newNetwork.network" to the current directory. The dot (.) represents the current location (not its contents).

#### ----- EXAMPLE 3 -----

```

XDHyp:\HostingUnits\MyHostingUnit>dir *.network | Add-HypHostingUnitNetwork -LiteralPath XDHyp:\HostingUnits\MyHostingUnit

```

The command adds all of the networks that are available in the hosting unit to the specified hosting unit.

#### ----- EXAMPLE 4 -----

```

c:\PS>Add-HypHostingUnitNetwork -LiteralPath XDHyp:\HostingUnits\MyHostingUnit -NetworkPath 'XDHyp:\HostingUnits\MyHostingUnits\newNetwork.network'

```

```

HostingUnitUid : bcd3d649-86d1-4aa8-8ec2-d322b6a2c457
HostingUnitName : MyHostingUnit
HypervisorConnection : MyConnection
RootPath : /
RootId :
NetworkPath : /Network 0.network
NetworkId : ab47080b-ca15-771a-c8dc-6ad9650158f1
Storage : {/Local storage.storage}
PersonalvDiskStorage : {}
VMTaggingEnabled : True
Metadata : {}
PermittedNetworks : {/Network 0.network, /newNetwork.network}

```

The command adds a new network location called "newNetwork.network" to the hosting unit called "MyHostingUnit".

# Add-HypHostingUnitStorage

May 10, 2016  
Adds storage locations to a hosting unit.

## Syntax

Add-HypHostingUnitStorage [-LiteralPath <String>] [-StoragePath <String>] [-StorageType <StorageType>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Use this command to add storage locations for storing the hard disks required by the virtual machines created by the Citrix Machine Creation Service. You cannot use this command if the connection for the hosting unit is in maintenance mode.

## Related topics

- New-Item
- Add-HypMetadata
- remove-HypHostingUnitStorage

## Parameters

**-LiteralPath**<String>

Specifies the path to the hosting unit to which storage will be added. The path must be in one of the following formats: <drive>:\HostingUnits\<HostingUnitName> or <drive>:\HostingUnits\{<HostingUnit Uid>}

Required?	true
Default Value	
Accept Pipeline Input?	false

**-StoragePath**<String>

Specifies the path to the storage that will be added. The path must be in one of the following formats: <drive>:\Connections\<ConnectionName>\MyStorage.storage or <drive>:\Connections\{<Connection Uid>}\MyStorage.storage or <drive>:\HostingUnits\<HostingUnitName>\MyStorage.storage or <drive>:\HostingUnits\{<hostingUnit Uid>}\MyStorage.storage

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-StorageType**<StorageType>

Specifies the type of storage in StoragePath. Supported storage types are: OSStorage PersonalDiskStorage

Required?	false
Default Value	OSStorage
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. This can be a host name or an IP address.

Required?	false
Default Value	LocalHost. When a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Input Type

Systemstring You can pipe a string that contains a path to Add-HypHostingUnitStorage (StoragePath parameter).

### Return Values

Citrix.Host.Sdk.HostingUnit

Add-HypHostingUnitStorage returns an object containing the new definition of the hosting unit.

HostingUnitUid <Guid>

Specifies the unique identifier for the hosting unit.

HostingUnitName <string>

Specifies the name of the hosting unit.

HypervisorConnection <Citrix.Host.Sdk.HypervisorConnection>

Specifies the connection that the hosting unit uses to access a hypervisor.

RootId <string>

Identifies, to the hypervisor, the root of the hosting unit.

RootPath <string>

The hosting unit provider path that represents the root of the hosting unit.

Storage <Citrix.Host.Sdk.Storage[]>

The list of storage items that the hosting unit can use.

PersonalDiskStorage <Citrix.XDPowerShell.Storage[]>

The list of storage items that the hosting unit can use for storing personal data.

VMTaggingEnabled <Boolean>

Specifies whether or not the metadata in the hypervisor can be used to store information about the XenDesktop Machine Creation Service.

NetworkId <string>

The hypervisor's internal identifier that represents the network specified for the hosting unit.

NetworkPath <string>

The hosting unit provider path to the network specified for the hosting unit.

Metadata <Citrix.Host.Sdk.Metadata[]>

A list of key value pairs that can store additional information about the hosting unit.

### Notes

The storage path must be valid for the hosting unit. The rules that are applied are as follows: XenServer (HypervisorConnection Type = XenServer)

NA

VMWare vSphere/ESX (HypervisorConnection Type = vCenter)

The storage path must be directly contained in the root path item of the hosting unit.

Microsoft SCVMM/Hyper-v (HypervisorConnection Type = SCVMM)

Only one storage entry for these connection types is valid, and it must reference an SMB share. Additionally, if a Hyper-V failover cluster is used the SMB share must be the top-level mount point of the cluster shared volume on one of the servers in the cluster (i.e. C:\ClusterStorage).

In the case of failure, the following errors can result.

### Error Codes

-----

HostingUnitsPathInvalid

The path provided is not to an item in a subdirectory of a hosting unit item.

HostingUnitStoragePathInvalid

The specified path is invalid.

HostingUnitStoragePathInvalid

The storage path cannot be found or is invalid. See notes above about validity.

HostingUnitStorageDuplicateObjectExists

The specified storage path is already part of the hosting unit.

HypervisorInMaintenanceMode

The hypervisor for the connection is in maintenance mode.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation. Communication with the database failed for various reasons.

CommunicationError

An error occurred while communicating with the service.

PermissionDenied

The user does not have administrative rights to perform this operation.

ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used, or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Add-HypHostingUnitStorage -LiteralPath XDHyp:\HostingUnits\MyHostingUnit -StoragePath 'XDHyp:\HostingUnits\MyHostingUnits\newStorage.storage'
```

```
HostingUnitUid : bcd3d649-86d1-4aa8-8ec2-d322b6a2c457
HostingUnitName : MyHostingUnit
HypervisorConnection : MyConnection
RootPath : /
RootId :
NetworkPath : /Network 0.network
NetworkId : ab47080b-ca15-771a-c8dc-6ad9650158f1
Storage : {/Local storage.storage, /newStorage.storage}
PersonalvDiskStorage : {/newStorage.storage}
VMTaggingEnabled : True
Metadata : {}
```

The command adds a new storage location called "newStorage.storage" to the hosting unit called "MyHostingUnit".

### ----- EXAMPLE 2 -----

```
XDHyp:\HostingUnits\MyHostingUnit>Add-HypHostingUnitStorage -LiteralPath . -StoragePath 'XDHyp:\HostingUnits\MyHostingUnits\newStorage.storage' -StorageType OSSStorage
```

```
HostingUnitUid : bcd3d649-86d1-4aa8-8ec2-d322b6a2c457
HostingUnitName : MyHostingUnit
HypervisorConnection : MyConnection
RootPath : /
RootId :
NetworkPath : /Network 0.network
NetworkId : ab47080b-ca15-771a-c8dc-6ad9650158f1
Storage : {/Local storage.storage, /newStorage.storage}
PersonalvDiskStorage : {/Local storage.storage}
VMTaggingEnabled : True
Metadata : {}
```

The command adds a new storage location called "newStorage.storage" to the current directory. The dot (.) represents the current location (not its contents).

### ----- EXAMPLE 3 -----

```
XDHyp:\HostingUnits\MyHostingUnit>dir *.storage | Add-HypHostingUnitStorage -LiteralPath XDHyp:\HostingUnits\MyHostingUnit
```

The command adds all of the storage that is available in the hosting unit to the specified hosting unit.

### ----- EXAMPLE 4 -----

```
c:\PS>Add-HypHostingUnitStorage -LiteralPath XDHyp:\HostingUnits\MyHostingUnit -StoragePath 'XDHyp:\HostingUnits\MyHostingUnits\newStorage.storage' -StorageType PersonalvDiskStorage
```

```
HostingUnitUid : bcd3d649-86d1-4aa8-8ec2-d322b6a2c457
HostingUnitName : MyHostingUnit
HypervisorConnection : MyConnection
RootPath : /
RootId :
NetworkPath : /Network 0.network
NetworkId : ab47080b-ca15-771a-c8dc-6ad9650158f1
Storage : {/Local storage.storage}
PersonalvDiskStorage : {/Local storage.storage, /newStorage.storage}
VMTaggingEnabled : True
Metadata : {}
```

The command adds a new storage location called "newStorage.storage" to the hosting unit called "MyHostingUnit".



# Add-HypHypervisorConnectionAddress

May 10, 2016  
Add a connection address to a hypervisor connection.

## Syntax

Add-HypHypervisorConnectionAddress [-LiteralPath] <String> [-HypervisorAddress] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Use this command to add addresses by which a hypervisor can be contacted. All addresses added to a single hypervisor connection are assumed to be equivalent (i.e. they all result in the ability to communicate with the same hypervisors). The hypervisor that the addresses reference is stored at the point of creation of the hypervisor connection. Once this is done, to be valid, all addresses must resolve to this hypervisor.

The addresses required are; XenServer - The address of the XenServer machines (must all reference the same XenServer pool). VMWare vSphere/ESX - The address of a vCenter Server. Microsoft SCVMM/Hyper-V - the address of an SCVMM server.

## Related topics

[Remove-HypHypervisorConnectionAddress](#)

[Get-HypXenServerAddress](#)

## Parameters

**-LiteralPath**<String>

Specifies the path within a Host Service provider to the hypervisor connection item to which to add the address. The path specified must be in one of the following formats; <drive>:\Connections\<HypervisorConnectionName> or <drive>:\Connections\{HypervisorConnection Uid>}

Required?	true
Default Value	
Accept Pipeline Input?	false

**-HypervisorAddress**<String>

Specifies the address to be used. The address will be validated and the hypervisor must be contactable at the address supplied.

XenServer (ConnectionType = XenServer) The address being added must reference the same XenServer pool referenced by any existing addresses for the same connection. VMware vSphere\ESX (ConnectionType = vCenter) The address being added must be to the same VMware vCenter as the existing addresses. SCVMM\Hyper-v (ConnectionType = SCVMM) The address being added must be to the same SCVMM server as the existing addresses. Custom Connection Types ?Validation here I assume none need to check?  
#PUBS NOTE: ASSUME THIS TO BE UPDATED?#

Required?	true

Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

Citrix.Host.Sdk.HypervisorConnection

Add-HypHypervisorConnectionAddress returns an object containing the new definition of the hypervisor connection.

HypervisorConnectionUid <Guid>

Specifies the unique identifier for the hypervisor connection.

HypervisorConnectionName <string>

Specifies the name of the hypervisor connection.

ConnectionType <Citrix.XDInterServiceTypes.ConnectionType>

Specifies the connection type of the connection.

XenServer - A Citrix XenServer hypervisor

SCVMM - A Microsoft SCVMM/Hyper-V hypervisor

vCenter - A VMWare vSphere/ESX hypervisor

Custom - A 3rd party hypervisor

HypervisorAddress <string[]>

A list of addresses that can be used to communicate with the hypervisor.

UserName <string>

The user name that is used when connecting to the hypervisor.

Persistent <Boolean>

Indicates whether the connection is stored in the database or is a temporary connection only with the same lifetime as the current Powershell session.

PlugInId <string>

The Citrix identifier for the Citrix Machine Management plug-in.

Revision <Guid>

Identifier for the current version of the hypervisor connection. This value changes every time a property of the hypervisor connection is updated.

SupportsPvsVMs <Boolean>

Indicates whether or not the connection can be used as part of a hosting unit and therefore used by the Citrix XenDesktop Machine Creation Service to create virtual machines. Only the built-in supported hypervisor connection types can be used for this (i.e. XenServer, SCVMM and vCenter).

Metadata <Citrix.Host.Sdk.Metadata[]>

A list of key value pairs that can store additional information relating to the hosting unit.

## Notes

The address format must be valid for the hypervisor connection. The rules that are applied are as below: XenServer (HypervisorConnection Type = XenServer)

http:\\<IP Address>

or http:\\<server Name>

or https:\\<server Name>

Note: To use multiple addresses to the same XenServer pool to provide failover functionality, all XenServers in the pool must have a shared block storage device. If the use of https connections and failover is required, the certificates on the servers must be trusted by all of the controllers (typically this means having a root certificate installed).

VMWare vSphere/ESX (HypervisorConnection Type = vCenter)

http:\\<IP Address>\\<sdk path>

http:\\<server Name>\\<sdk path>

or https:\\<Server Name>\\<sdk path>

Notes:

\* Http requires manual vCenter configuration; https is the only supported default format.

\* Default sdk path is 'sdk'

Microsoft SCVMM/Hyper-v (HypervisorConnection Type = SCVMM)

IP Address

server Name (short or FQDN)

Note: If XenServers are moved to new XenServer pools after being added to a hypervisor connection, unpredictable results can occur. Once a XenServer is assigned to a hypervisor connection, it must not be moved to a new XenServer pool.

In the case of failure the following errors can result.

Error Codes

-----

InputConnectionsPathInvalid

The path provided is not to an item in a sub directory of a hosting unit item.

HypervisorConnectionAddressForeignKeyObjectDoesNotExist

The hypervisor connection to which the address is to be added could not be located.

ConnectionAddressInvalid

The address could not be used to contact a hypervisor or the validation rules have not been met.

HypervisorConnectionAddressDuplicateObjectExists

The address specified is already part of the hypervisor connection.

HypervisorInMaintenanceMode

The hypervisor for the connection is in maintenance mode.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

CommunicationError

An error occurred while communicating with the service.

PermissionDenied

The user does not have administrative rights to perform this operation.

ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Add-HypHypervisorConnectionAddress -LiteralPath XDHyp:\Connections\MyConnection -HypervisorAddress http:\\myserver.com
```

```
PSPath : Citrix.HostingUnitService.Admin.V1.0\Citrix.Hypervisor::XDHyp:\Connections\MyConnection
PSParentPath : Citrix.HostingUnitService.Admin.V1.0\Citrix.Hypervisor::XDHyp:\Connections
PSChildName : MyConnection
PSDrive : XDHyp
PSProvider : Citrix.HostingUnitService.Admin.V1.0\Citrix.Hypervisor
PSIsContainer : True
HypervisorConnectionUid : 85581f42-c5da-4976-970c-ebc3448ea1e3
HypervisorConnectionName : MyConnection
ConnectionType : XenServer
HypervisorAddress : {http:\\myserver2.com,http:\\myserver.com}
UserName : root
Persistent : False
PluginId : XenFactory
SupportsPvsVMs : True
Revision : 4c95c857-c54d-4f92-abef-0cce32c02502
Metadata :
Add the address 'http:\\myserver.com' to the hypervisor connection called "MyConnection".
```

# Add-HypHypervisorConnectionMetadata

May 10, 2016  
Adds metadata on the given HypervisorConnection.

## Syntax

Add-HypHypervisorConnectionMetadata [-HypervisorConnectionUid] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Add-HypHypervisorConnectionMetadata [-HypervisorConnectionUid] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Add-HypHypervisorConnectionMetadata [-HypervisorConnectionName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Add-HypHypervisorConnectionMetadata [-HypervisorConnectionName] <String> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Add-HypHypervisorConnectionMetadata [-InputObject] <HypervisorConnection[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Add-HypHypervisorConnectionMetadata [-InputObject] <HypervisorConnection[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Provides the ability for additional custom data to be stored against given HypervisorConnection objects. This cmdlet will not overwrite existing metadata on an object - use the Set-HypHypervisorConnectionMetadata cmdlet instead.

## Related topics

[Set-HypHypervisorConnectionMetadata](#)

[Remove-HypHypervisorConnectionMetadata](#)

## Parameters

**-HypervisorConnectionUid**<Guid>

Id of the HypervisorConnection

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-HypervisorConnectionName**<String>

Name of the HypervisorConnection

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject<HypervisorConnection[]>**

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name<String>**

Specifies the property name of the metadata to be added. The property must be unique for the HypervisorConnection specified. The property cannot contain any of the following characters \/:;#.\*?=<>|[]0""

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Value<String>**

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

Citrix.Host.Sdk.Metadata

Add-HypHypervisorConnectionMetadata returns an array of objects containing the new definition of the metadata.

\n Property <string>

\n Specifies the name of the property.

\n Value <string>

\n Specifies the value for the property.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DuplicateObject

One of the specified metadata already exists.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError



There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Add-HypHypervisorConnectionMetadata -HypervisorConnectionUid 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value

Property	Value
-----	-----
property	value

Add metadata with a name of 'property' and a value of 'value' to the HypervisorConnection with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Add-HypHypervisorConnectionScope

May 10, 2016

Add the specified HypervisorConnection(s) to the given scope(s).

## Syntax

Add-HypHypervisorConnectionScope [-Scope] <String[]> -InputObject <HypervisorConnection[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Add-HypHypervisorConnectionScope [-Scope] <String[]> -HypervisorConnectionUid <Guid[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Add-HypHypervisorConnectionScope [-Scope] <String[]> -HypervisorConnectionName <String[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

The AddHypHypervisorConnectionScope cmdlet is used to associate one or more HypervisorConnection objects with given scope(s).

There are multiple parameter sets for this cmdlet, allowing you to identify the HypervisorConnection objects in different ways:

- HypervisorConnection objects can be piped in or specified by the InputObject parameter
- The HypervisorConnectionUid parameter specifies objects by HypervisorConnectionUid
- The HypervisorConnectionName parameter specifies objects by HypervisorConnectionName (supports wildcards)

To add a HypervisorConnection to a scope you need permission to change the scopes of the HypervisorConnection and permission to add objects to all of the scopes you have specified.

If the HypervisorConnection is already in a scope, that scope will be silently ignored.

## Related topics

[Remove-HypHypervisorConnectionScope](#)

[Get-HypScopedObject](#)

## Parameters

**-Scope**<String[]>

Specifies the scopes to add the objects to.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-InputObject**<HypervisorConnection[]>

Specifies the HypervisorConnection objects to be added.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-HypervisorConnectionUid**<Guid[]>

Specifies the HypervisorConnection objects to be added by HypervisorConnectionUid.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-HypervisorConnectionName**<String[]>

Specifies the HypervisorConnection objects to be added by HypervisorConnectionName.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

None

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### UnknownObject

One of the specified objects was not found.

#### ScopeNotFound

One of the specified scopes was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command with the specified objects or scopes.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Add-HypHypervisorConnectionScope Finance -HypervisorConnectionUid 6702C5D0-C073-4080-A0EE-EC74CB537C52
Adds a single HypervisorConnection to the 'Finance' scope.
```

### ----- EXAMPLE 2 -----

```
c:\PS>Add-HypHypervisorConnectionScope Finance,Marketing -HypervisorConnectionUid 6702C5D0-C073-4080-A0EE-EC74CB537C52
```

Adds a single HypervisorConnection to the multiple scopes.

----- **EXAMPLE 3** -----

```
c:\PS>Get-HypHypervisorConnection | Add-HypHypervisorConnectionScope Finance
```

Adds all visible HypervisorConnection objects to the 'Finance' scope.

----- **EXAMPLE 4** -----

```
c:\PS>Add-HypHypervisorConnectionScope Finance -HypervisorConnectionName A*
```

Adds HypervisorConnection objects with a name starting with an 'A' to the 'Finance' scope.

# Add-HypMetadata

May 10, 2016

Adds metadata to a hypervisor connection or a hosting unit.

## Syntax

```
Add-HypMetadata [-LiteralPath] <String> [-Property] <String> [-Value] <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Use this command to store additional custom data against a hosting unit or hypervisor connection. This data is not used by the Machine Creation Service, and is provided only for consumers of the services to store any data that may be required for their operations. The metadata is returned along with the hypervisor connection or hosting unit that it is assigned to.

## Related topics

[Remove-HypMetadata](#)

## Parameters

**-LiteralPath**<String>

Specifies the path within a hosting unit provider to the hosting unit or hypervisor connection item to which to add the metadata. The path specified must be in one of the following formats: <drive>:\HostingUnits\<HostingUnitName> or <drive>:\HostingUnits\{<HostingUnit Uid> or <drive>:\Connections\<Connection Name> or <drive>:\Connections\{<Connection Uid>}

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Property**<String>

Specifies the property name of the metadata to be added. The property must be unique for the item specified by the path.

The property cannot contain any of the following characters \/:;#.\*?=<>|[]()""

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Input Type

System.string You can pipe a string the contains a path to Add-HypMetadata (Path parameter).

### Return Values

Citrix.Host.Sdk.Metadata

Add-HypMetadata returns an object containing the new definition of the metadata.

Property <string>

Specifies the property of the metadata.

Value <string>

Specifies the value of the metadata.

## Notes

In the case of failure, the following errors can result.

### Error Codes

-----

#### InvalidPath

The path provided is not in the required format.

#### HostingUnitMetadataForeignKeyObjectDoesNotExist

The hosting unit supplied in the path does not exist.

#### HypervisorConnectionMetadataForeignKeyObjectDoesNotExist

The hypervisor connection supplied in the path does not exist.

#### HostingUnitMetadataDuplicateObjectExists

Metadata for the specified hosting unit item already exists with the same property name.

#### HypervisorConnectionMetadataDuplicateObjectExists

Metadata for the specified hypervisor connection item already exists with the same property name.

#### MetadataContainerUndefined

The specified path does not reference a hosting unit or a hypervisor connection.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.



PermissionDenied

The user does not have administrative rights to perform this operation.

ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Add-HypMetadata -LiteralPath XDHyp:\Connections\MyConnection -Property MyProperty -Value MyValue
```

Property	Value
----------	-------

-----	-----
-------	-------

MyProperty	MyValue
------------	---------

The command adds the metadata with the property name of "MyProperty" and value of "MyValue" to the hypervisor connection item called "MyConnection".

### ----- EXAMPLE 2 -----

```
c:\PS>dir xdhyp\connections\Citrix* | Add-HypMetadata -Property MyProperty -Value MyValue
```

Property	Value
----------	-------

-----	-----
-------	-------

MyProperty	MyValue
------------	---------

MyProperty	MyValue
------------	---------

MyProperty	MyValue
------------	---------

The command adds the metadata with the property name of "MyProperty" and value of "MyValue" to all the hypervisor connection items that begin with the string "Citrix".

# Get-HypConfigurationDataForItem

May 10, 2016

Retrieves the configuration data for an item in the Host Service provider path. Note: For this release, only VM items are supported for this operation.

## Syntax

```
Get-HypConfigurationDataForItem [-LiteralPath] <String> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This command provides a mechanism for retrieving extra data about an entry in the hosting unit service provider. The referenced item must be contained within the connections directory in the provider (i.e. XDHyp:\Connections).

This mechanism is used for obtaining data that is not required frequently and/or has a high overhead associated with its retrieval (so as to maintain the responsiveness of the provider). For this release of the PowerShell snap-in, the only provider items that can be used with this operation are VM items. For a VM, this provides a mechanism to obtain the number of CPUs, the amount of Memory (in MB) and hard disk drive capacity (in GB).

## Related topics

Get-Item

## Parameters

**-LiteralPath**<String>

Specifies the path within a hosting unit provider to the item for which configuration data is to be retrieved. The path specified must be in one of the following formats; <drive>:\Connections\<Connection Name>\<Item Path of VM object> or <drive>:\Connections\{<connection Uid>\<Item Path of VM object>} or <drive>:\HostingUnits\<HostingUnit Name>\<Item Path of VM object> or <drive>:\HostingUnits\{<hostingUnit Uid>\<Item Path of VM object>}

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Input Type

System.string You can pipe a string that contains a path to Get-HypConfigurationDataForItem

## Return Values

PSObject

Get-HypConfigurationDataForItem returns a PSObject containing the properties that are appropriate for the item specified by the path.

## Properties for VM Item

-----

CPUCount <int>

Specifies the number of CPUs assigned to the VM.

MemoryMB <int>

The amount of memory allocated to the VM.

HardDiskSizeGB <int>

The capacity of the primary hard drive assigned to the VM.

## Notes

For this release, this cmdlet provides only configuration data for VM objects in the provider. Using a path to an item that is not a VM results in a error.

In the case of failure the following errors can result.

## Error Codes

-----

InputHypervisorItemPathInvalid

The path provided is not to an item in a sub-directory of a connection item or a hosting unit item.

InvalidHypervisorItemPath

No item exists with the specified path.

InvalidHypervisorItem

The item specified by the path exists, but is not a VM Item.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

CommunicationError

An error occurred while communicating with the service.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

HypervisorPermissionDenied

The hypervisor login used does not provide authorization to access the data for this item.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
C:\PS>Get-HypConfigurationDataForItem -LiteralPath XDHyp:\Connections\MyConnection\MyVm.vm
```

CpuCount	MemoryMB	HardDiskSizeGB
1	1024	24

This command gets the configuration properties for a VM called 'MyVm.vm' within a hypervisor connection called 'MyConnection'.

----- **EXAMPLE 2** -----

```
XDHyp:\HostingUnits\PS>Get-HypConfigurationDataForItem -LiteralPath .\MyVm.vm
```

CpuCount	MemoryMB	HardDiskSizeGB
1	1024	24

This command gets the configuration properties for a VM called 'MyVm.vm' within the current directory. The dot (.) represents the current location (not its contents).

----- **EXAMPLE 3** -----

```
C:\PS>(Get-HypConfigurationDataForItem -LiteralPath XDHyp:\Connections\MyConnection\MyVm.vm).CPUCount
```

This command gets the CPU count for a VM called 'MyVm.vm'. The CPUCount is just one property of the VM items. To see all properties of an item, type "(Get-HypConfigurationDataForItem <ItemPath>) | Get-Member".

# Get-HypConfigurationObjectForItem

May 10, 2016

Retrieves the configuration data for an item in the Host Service provider path. Note: For this release, only VM items are supported for this operation.

## Syntax

```
Get-HypConfigurationObjectForItem [-LiteralPath] <String> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

This command provides a mechanism for retrieving extra data about an entry in the hosting unit service provider. The referenced item must be contained within the connections directory in the provider (i.e. XDHyp:\Connections).

This mechanism is used for obtaining data that is not required frequently and/or has a high overhead associated with its retrieval (so as to maintain the responsiveness of the provider). For this release of the PowerShell snap-in the only provider items that can be used with this operation are VM items. For a VM this provides a mechanism to obtain the number of CPUs, the amount of Memory (in MB) and hard disk drive capacity (GB).

## Related topics

Get-Item

## Parameters

**-LiteralPath**<String>

Specifies the path within a hosting unit provider to the item for which configuration data is to be retrieved. The path specified must be in one of the following formats; <drive>:\Connections\<Connection Name>\<Item Path of VM object> or <drive>:\Connections\{<connection Uid>\<Item Path of VM object>} or <drive>:\HostingUnits\<HostingUnit Name>\<Item Path of VM object> or <drive>:\HostingUnits\{<hostingUnit Uid>\<Item Path of VM object>}

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value will become the default.

Accept Pipeline Input?	false
------------------------	-------

## Input Type

System.string You can pipe a string the contains a path to Get-HypConfigurationDataForItem

## Return Values

### PSObject

Get-HypConfigurationDataForItem returns a PSObject containing the properties that are appropriate for the item specified by the path.

### Properties for VM Item

-----

#### CPUCount <int>

Specifies the number of CPUs assigned to the VM.

#### MemoryMB <int>

The amount of memory allocated to the VM.

#### HardDiskSizeGB <int>

The capacity of the primary hard drive assigned to the VM.

### Network Map

The networks that this Vm or Snapshot is connected to

## Notes

For this release this cmdlet only provides configuration data for VM objects in the provider. Using a path to an item that is not a VM will result in a error.

In the case of failure the following errors can be produced.

### Error Codes

-----

#### InputHypervisorItemPathInvalid

The path provided is not to an item in a sub directory of a connection item or a hosting unit item.

#### InvalidHypervisorItemPath

No item exists with the specified path.

#### InvalidHypervisorItem

The item specified by the path exists, but is not a VM Item.

#### DatabaseError

An error occurred in the service whilst attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed as the database for the service is not configured.

#### DataStoreException

An error occurred in the service whilst attempting a database operation - communication to database failed for various reasons.

#### CommunicationError

An error occurred whilst communicating with the service.

#### InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

#### HypervisorPermissionDenied

The hypervisor login used does not provide authorization to access the data for this item.

#### ExceptionThrown

An unexpected error occurred. To locate more details see the Windows event logs on the controller being used, or examine the XenDesktop logs.

### Examples

#### ----- EXAMPLE 1 -----

```
C:\PS>Get-HypConfigurationDataForItem -LiteralPath XDHyp:\Connections\MyConnection\MyVm.vm
```

CpuCount	MemoryMB	HardDiskSizeGB
1	1024	24

This command gets the configuration properties for a VM called 'MyVm.vm' within a hypervisor connection called 'MyConnection'.

#### ----- EXAMPLE 2 -----

```
XDHyp\HostingUnits\PS>Get-HypConfigurationDataForItem -LiteralPath .\MyVm.vm
```

CpuCount	MemoryMB	HardDiskSizeGB
1	1024	24

This command gets the configuration properties for a VM called 'MyVm.vm' within the current directory. The dot (.) represents the current location (not its contents).



### ----- EXAMPLE 3 -----

```
C:\PS>(Get-HypConfigurationDataForItem -LiteralPath XDHyp:\Connections\MyConnection\MyVm.vm).CPUCount
```

This command gets the CPU count for a VM called 'MyVm.vm'. The CPUCount is just one property of the VM items. To see all properties of an item, type "(Get-HypConfigurationDataForItem <ItemPath>) | Get-Member".

# Get-HypConnectionRegion

May 10, 2016

Enumerates the regions of a hypervisor connection that are based on cloud technology.

## Syntax

```
Get-HypConnectionRegion [-LiteralPath] <String> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Use this command to enumerate the available regions within a public or private cloud, when making hypervisor connections to cloud services. Sometimes, regions need to be selected and applied before the cloud connection can be used in a meaningful way. This cmdlet allows the supported regions to be listed so that one may be selected.

Once a region has been chosen, use the standard Set-Item provider cmdlet to select it. See the examples for further details.

This cmdlet may return no output, in which case the cloud connection can be considered "regionless" (or, implicitly, all within a single region). In such cases, there is no need to select a region, and the hypervisor connection can be used as is.

## Related topics

Set-Item

## Parameters

**-LiteralPath**<String>

Specifies the path to the hypervisor connection whose regions are being examined. This cmdlet is valid only for hypervisor connections that have the UsesCloudInfrastructure flag set to true. The path must be in one of the following formats: <drive>:\Connections\<ConnectionName> or <drive>:\Connections\{<Connection Uid>}

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide a host name or an IP address.

Required?	false
Default Value	LocalHost. When a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Input Type

System.string You can pipe a string that contains a path to Get-HypConnectionRegion (LiteralPath parameter).

## Return Values

Citrix.Host.Sdk.HypervisorRegion

Get-HypConnectionRegion returns zero or more instances of the HypervisorConnectionRegion object, each of which contain the following properties:

Name <string> Specifies the unique name of the region. Endpoint <string> Specifies the URL endpoint that is specific to the region, if relevant. This may be an empty string, and is returned only for information purposes.

A full list of the hypervisor networks that are exposed for use in the hosting unit.

## Notes

In the case of failure, the following errors can be produced.

### Error Codes

-----

#### ConnectionsPathInvalid

The path provided is not to an item in the Connections subdirectory of the host service provider.

#### HypervisorConnectionObjectNotFound

The path provided could not be resolved to an existing hypervisor connection. The name or GUID is invalid.

#### HypervisorInMaintenanceMode

The hypervisor for the connection is in maintenance mode.

#### ConnectionIsNotCloud

The hypervisor connection is not associated with cloud infrastructure, making it invalid to enumerate regions.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation. Communication with the database failed for various reasons.

## CommunicationError

An error occurred while communicating with the service.

## PermissionDenied

The user does not have administrative rights to perform this operation.

## ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used, or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-HypConnectionRegions -LiteralPath XDHyp:\Connections\AWS
```

```
RegionName : us-east-1
Endpoint : ec2.us-east-1.amazonaws.com
```

```
RegionName : us-west-1
Endpoint : ec2.us-west-1.amazonaws.com
```

```
RegionName : eu-west-1
Endpoint : ec2.eu-west-1.amazonaws.com
```

```
(...)
```

```
c:\PS>Set-Item -Path XDHyp:\Connections\AWS -Region "us-east-1"
```

This sequence of commands enumerates the available regions of an Amazon AWS cloud connection, and then selects one of them for use in the connection.

# Get-HypDBConnection

May 10, 2016

Gets the database string for the specified data store used by the Host Service.

## Syntax

```
Get-HypDBConnection [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the database connection string for the specified data store.

If the returned string is blank, no valid connection string has been specified. In this case the service is running, but is idle and awaiting specification of a valid connection string.

## Related topics

[Get-HypServiceStatus](#)

[Set-HypDBConnection](#)

[Test-HypDBConnection](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

system.string

The database connection string configured for the Host Service.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

NoDBConnections

The database connection string for the Host Service has not been specified.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Get-HypDBConnection

Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True  
Get the database connection string for the Host Service.

# Get-HypDBSchema

May 10, 2016

Gets a script that creates the Host Service database schema for the specified data store.

## Syntax

```
Get-HypDBSchema [-DatabaseName <String>] [-ServiceGroupName <String>] [-ScriptType <ScriptTypes>] [-LocalDatabase] [-Sid <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets SQL scripts that can be used to create a new Host Service database schema, add a new Host Service to an existing site, remove a Host Service from a site, or create a database server logon for a Host Service. If no Sid parameter is provided, the scripts obtained relate to the currently selected Host Service instance, otherwise the scripts relate to Host Service instance running on the machine identified by the Sid provided. When obtaining the Evict script, a Sid parameter must be supplied. The current service instance is that on the local machine, or that explicitly specified by the last usage of the -AdminAddress parameter to a Host SDK cmdlet. The service instance used to obtain the scripts does not need to be a member of a site or to have had its database connection configured. The database scripts support only Microsoft SQL Server, or SQL Server Express, and require Windows integrated authentication to be used. They can be run using SQL Server's SQLCMD utility, or by copying the script into an SQL Server Management Studio (SSMS) query window and executing the query. If using SSMS, the query must be executed in 'SMDCMD mode'. The ScriptType parameter determines which script is obtained. If ScriptType is not specified, or is FullDatabase, the script contains:

- o Creation of service schema
- o Creation of database server logon
- o Creation of database user
- o Addition of database user to Host Service roles

If ScriptType is Instance, the returned script contains:

- o Creation of database server logon
- o Creation of database user
- o Addition of database user to Host Service roles

If ScriptType is Evict, the returned script contains:

- o Removal of Host Service instance from database
- o Removal of database user

If ScriptType is Login, the returned script contains:

- o Creation of database server logon only

If the service uses two data stores they can exist in the same database. You do not need to configure a database before using this command.

## Related topics

[Set-HypDBConnection](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database for which the schema will be generated.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ServiceGroupName**<String>

Specifies the name of the service group to be used when creating the database schema. The service group is a collection of all the Host services that share the same database instance and are considered equivalent; that is, all the services within a service group can be used interchangeably.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ScriptType**<ScriptTypes>

Specifies the type of database script returned. Available script types are:

### Database

Returns a full database script that can be used to create a database schema for the Host Service in a database instance that does not already contain a schema for this service. The DatabaseName and ServiceGroupName parameters must be specified to create a script of this type.

### Instance

Returns a permissions script that can be used to add further Host services to an existing database instance that already contains the full Host service schema, associating the services to the Service Group. The Sid parameter can optionally be specified to create a script of this type.

### Login

Returns a database logon script that can be used to add the required logon accounts to an existing database instance that contains the Host Service schema. This is used primarily when creating a mirrored database environment. The



DatabaseName parameter must be specified to create a script of this type.

#### Evict

Returns a script that can be used to remove the specified Host Service from the database entirely. The DatabaseName and Sid parameters must be specified to create a script of this type.

Required?	false
Default Value	Database
Accept Pipeline Input?	false

#### -LocalDatabase<SwitchParameter>

Specifies whether the database script is to be used in a database instance run on the same controller as other services in the service group. Including this parameter ensures the script creates only the required permissions for local services to access the database schema for Host services.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -Sid<String>

Specifies the SID of the controller on which the Host Service instance to remove from the database is running.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

#### -AdminAddress<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

System.string

A string containing the required SQL script for application to a database.

## Notes

The scripts returned support Microsoft SQL Server Express Edition, Microsoft SQL Server Standard Edition, and Microsoft SQL Server Enterprise Edition databases only, and are generated on the assumption that integrated authentication will be used.

If the ScriptType parameter is not included or set to 'FullDatabase', the full database script is returned, which will:

Create the database schema.

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist).

If the ScriptType parameter is set to 'Instance', the script will:

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist) and associate it with a user.

If the ScriptType parameter is set to 'Login', the script will:

Create the logon (providing the schema does not already exist) and associate it with a pre-existing user of the same name.

If the LocalDatabase parameter is included, the NetworkService account will be added to the list of accounts permitted to access the database. This is required only if the database is run on a controller.

If the command fails, the following errors can be returned.

## Error Codes

-----

### GetSchemasFailed

The database schema could not be found.

### ActiveDirectoryAccountResolutionFailed

The specified Active Directory account or Group could not be found.

### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Get-HypDBSchema -DatabaseName MyDB -ServiceGroupName MyServiceGroup > c:\HypSchema.sql
```

Get the full database schema for site data store of the Host Service and copy it to a file called 'c:\HypSchema.sql'.

This script can then be used to create the schema in a pre-existing database named 'MyDB' that does not already contain a Host Service site schema.

##### ----- EXAMPLE 2 -----

```
c:\PS>Get-HypDBSchema -DatabaseName MyDB -scriptType Login > c:\HostLogins.sql
```

Get the logon scripts for the Host Service.

# Get-HypDBVersionChangeScript

May 10, 2016

Gets a script that updates the Host Service database schema.

## Syntax

```
Get-HypDBVersionChangeScript -DatabaseName <String> -TargetVersion <Version> [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Returns a database script that can be used to upgrade or downgrade the site or secondary schema for the Host Service from the current schema version to a different version.

## Related topics

[Get-HypInstalledDBVersion](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database instance to which the update applies.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-TargetVersion**<Version>

Specifies the version of the database you want to update to.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.Management.Automation.PSObject

A PSObject containing the required SQL script for application to a database.

## Notes

The PSObject returned by this cmdlet contains the following properties:

- Script The raw text of the SQL script to apply the update, or null in the case when no upgrade path to the specified target version exists.
- NeedExclusiveAccess Indicates whether all services in the service group must be shut down during the update or not.
- CanUndo Indicates whether the generated script allows the updated schema to be reverted to the state prior to the update.

Scripts to update the schema version are stored in the database so any service in the service group can obtain these scripts. Extreme caution should be exercised when using update scripts. Citrix recommends backing up the database before attempting to upgrade the schema. Database update scripts may require exclusive use of the schema and so may not be able to execute while any Host services are running. However, this depends on the specific update being carried out.

After a schema update has been carried out, services that require the previous version of the schema may cease to operate. The ServiceState parameter reported by the Get-HypServiceStatus command provides information about service compatibility. For example, if the schema has been upgraded to a more recent version that a service cannot use, the service reports "DBNewerVersionThanService".

If the command fails, the following errors can be returned.

## Error Codes

-----

### NoOp

The operation was successful but had no effect.

### NoDBConnections

The database connection string for the Host Service has not been specified.

### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> $update = Get-HypDBVersionChangeScript -DatabaseName MyDb -TargetVersion 1.0.75.0
```

```
C:\PS> $update.Script > update_75.sql
```

Gets an SQL update script to update the current schema to version 1.0.75.0. The resulting update\_75.sql script is suitable for direct use with the SQL Server SQLCMD utility.

# Get-HypHypervisorPlugin

May 10, 2016

Gets the available hypervisor types.

## Syntax

```
Get-HypHypervisorPlugin [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Use this command to retrieve a list of all the available hypervisor types, and their localized names.

## Related topics

New-Item

## Parameters

**-AdminAddress** <String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Host.Sdk.HypervisorPlugin

Get-HypHypervisorPlugin returns a list of objects containing the definition of the hypervisor plug-ins.

ConnectionType <Citrix.XDInterServiceTypes.ConnectionType>

The hypervisor connection type. This can be one of the following:

XenServer - XenServer hypervisor

SCVMM - Microsoft SCVMM/Hyper-V

vCenter - VMWare vSphere/ESX

Custom - a third-party hypervisor

DisplayName <string>

The localized display name (localized using the locale of the Powershell snap-in session)

PluginFactoryName <string>

The name of the hypervisor plug-in factory used to manage the hypervisor connections.

## Notes

To use third-party plug-ins, the plug-in assemblies must be installed into the appropriate location on each controller machine that forms part of the Citrix controller site. Failure to do this can result in unpredictable behavior, especially during service failover conditions.

In the case of failure the following errors can result.

## Error Codes

-----

### DatabaseError

An error occurred in the service while attempting a database operation.

### CommunicationError

An error occurred while communicating with the service.

### InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS> Get-HypHypervisorPlugin | Format-Table -AutoSize
```

ConnectionType	DisplayName	PluginFactoryName
SCVMM	Microsoft virtualization	MicrosoftPSFactory
VCenter	VMware virtualization	VmwareFactory
XenServer	Citrix XenServer	XenFactory

Get the available hypervisor management plug-ins.



# Get-HypInstalledDBVersion

May 10, 2016

Gets a list of all available database schema versions for the Host Service.

## Syntax

```
Get-HypInstalledDBVersion [-Upgrade] [-Downgrade] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the current version of the Host Service database schema, if no flags are set, otherwise returns versions for which upgrade or downgrade scripts are available and have been stored in the database.

## Related topics

## Parameters

**-Upgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be updated should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Downgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be reverted should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

### System.Version

The Get-HypInstalledDbVersion command returns objects containing the new definition of the Host Service database schema version.

Major <Integer>

Minor <Integer>

Build <Integer>

Revision <Integer>

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

Both the Upgrade and Downgrade flags were specified.

#### NoOp

The operation was successful but had no effect.

#### NoDBConnections

The database connection string for the Host Service has not been specified.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-HypInstalledDBVersion
```

Major	Minor	Build	Revision
-------	-------	-------	----------

----	----	----	-----
------	------	------	-------

5	6	0	0
---	---	---	---

Get the currently installed version of the Host Service database schema.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-HypInstalledDBVersion -Upgrade
```

Major	Minor	Build	Revision
-------	-------	-------	----------

----	----	----	-----
------	------	------	-------

6	0	0	0
---	---	---	---

Get the versions of the Host Service database schema for which upgrade scripts are supplied.

# Get-HypScopedObject

May 10, 2016

Gets the details of the scoped objects for the Host Service.

## Syntax

```
Get-HypScopedObject [-ScopeId <Guid>] [-ScopeName <String>] [-ObjectType <ScopedObjectType>] [-ObjectId <String>] [-ObjectName <String>] [-Description <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns a list of directly scoped objects including the names and identifiers of both the scope and object as well as the object description for display purposes.

There will be at least one result for every directly scoped object. When an object is associated with multiple scopes the output contains one result per scope duplicating the object details.

No records are returned for the All scope, though if an object is not in any scope a result with a null ScopeId and ScopeName will be returned.

## Related topics

### Parameters

**-ScopeId**<Guid>

Gets scoped object entries for the given scope identifier.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ScopeName**<String>

Gets scoped object entries with the given scope name.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ObjectType<ScopedObjectType>**

Gets scoped object entries for objects of the given type.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ObjectId<String>**

Gets scoped object entries for objects with the specified object identifier.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ObjectName<String>**

Gets scoped object entries for objects with the specified object identifier.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-Description<String>**

Gets scoped object entries for objects with the specified description.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Hyp\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

**-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Hyp\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.Host.Sdk.ScopedObject

The Get-HypScopedObject command returns an object containing the following properties:

ScopeId <Guid?>

Specifies the unique identifier of the scope.

ScopeName <String>

Specifies the display name of the scope.

ObjectType <ScopedObjectType>

Type of the object this entry relates to.

ObjectId <String>

Unique identifier of the object.

ObjectName <String>

Display name of the object

Description <String>

Description of the object (possibly \$null if the object type does not have a description).

Notes

If the command fails, the following errors can be returned.

Error Codes

-----

UnknownObject

One of the specified objects was not found.

PartialData

Only a subset of the available data was returned.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

CouldNotQueryDatabase

The query required to get the database was not defined.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied



You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-HypScopedObject -ObjectType Scheme
```

ScopeId : eff6f464-f1ee-4442-add3-99982e0cec01

ScopeName : Sales

ObjectType : Scheme

ObjectId : cd4174ee-9e4b-4e57-b126-9dbf757fe493

ObjectName : MyExampleScheme

Description : Test scheme

ScopeId : 304e0fa7-d390-47f0-a94f-7e956a324c41

ScopeName : Finance

ObjectType : Scheme

ObjectId : cd4174ee-9e4b-4e57-b126-9dbf757fe493

ObjectName : MyExampleScheme

Description : Test scheme

ScopeId :

ScopeName :

ObjectType : Scheme

ObjectId : 5062e46b-71bc-4ac9-901a-30fe6797e2f6

ObjectName : AnotherScheme

Description : Another scheme in no scopes

Gets all of the scoped objects with type Scheme. The example output shows a scheme object (MyExampleScheme) in two scopes Sales and Finance, and another scheme (AnotherScheme) that is not in any scope. The ScopeId and ScopeName values returned are null in the final record.

# Get-HypService

May 10, 2016

Gets the service record entries for the Host Service.

## Syntax

```
Get-HypService [-Metadata <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns instances of the Host Service that the service publishes. The service records contain account security identifier information that can be used to remove each service from the database.

A database connection for the service is required to use this command.

## Related topics

### Parameters

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Hyp\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-Sort By**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See [about\\_Hyp\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.Host.Sdk.Service

The Get-HypServiceInstance command returns an object containing the following properties.

Uid <Integer>

Specifies the unique identifier for the service in the group. The unique identifier is an index number.

ServiceHostId <Guid>

Specifies the unique identifier for the service instance.

DNSName <String>

Specifies the domain name of the host on which the service runs.

MachineName <String>

Specifies the short name of the host on which the service runs.

CurrentState <Citrix.Fma.Sdk.ServiceCore.ServiceState>

Specifies whether the service is running, started but inactive, stopped, or failed.

LastStartTime <DateTime>

Specifies the date and time at which the service was last restarted.

LastActivityTime <DateTime>

Specifies the date and time at which the service was last stopped or restarted.

OSType

Specifies the operating system installed on the host on which the service runs.

OSVersion

Specifies the version of the operating system installed on the host on which the service runs.

ServiceVersion

Specifies the version number of the service instance. The version number is a string that reflects the full build version of the service.

DatabaseUserName <string>

Specifies for the service instance the Active Directory account name with permissions to access the database. This will be either the machine account or, if the database is running on a controller, the NetworkService account.

Sid <string>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

ActiveSiteServices <string[]>

Specifies the names of active site services currently running in the service. Site services are components that perform long-running background processing in some services. This field is empty for services that do not contain site services.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

UnknownObject

One of the specified objects was not found.

PartialData

Only a subset of the available data was returned.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

CouldNotQueryDatabase

The query required to get the database was not defined.

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-HypService
```

```
UId : 1
ServiceHostId : aef6f464-f1ee-4042-a523-66982e0cecd0
DNSName : MyServer.company.com
MachineName : MYSERVER
CurrentState : On
LastStartTime : 04/04/2011 15:25:38
LastActivityTime : 04/04/2011 15:33:39
OSType : Win32NT
OSVersion : 6.1.7600.0
ServiceVersion : 5.1.0.0
DatabaseUserName : NT AUTHORITY\NETWORK SERVICE
SID : S-1-5-21-2316621082-1546847349-2782505528-1165
ActiveSiteServices : {MySiteService1, MySiteService2...}
Get all the instances of the Host Service running in the current service group.
```

# Get-HypServiceAddedCapability

May 10, 2016

Gets any added capabilities for the Host Service on the controller.

## Syntax

```
Get-HypServiceAddedCapability [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables updates to the Host Service on the controller to be detected.

You do not need to configure a database connection before using this command.

## Related topics

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.String

List containing added capabilities.

String containing added capabilities.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- **EXAMPLE 1** -----

```
c:\PS>Get-HypServiceAddedCapability
Get the added capabilities of the Host Service.
```



# Get-HypServiceInstance

May 10, 2016

Gets the service instance entries for the Host Service.

## Syntax

```
Get-HypServiceInstance [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns service interfaces published by the instance of the Host Service. Each instance of a service publishes multiple interfaces with distinct interface types, and each of these interfaces is represented as a ServiceInstance object. Service instances can be used to register the service with a central configuration service so that other services can use the functionality.

You do not need to configure a database connection to use this command.

## Related topics

## Parameters

**-AdminAddress** <String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Host.Sdk.ServiceInstance

The Get-HypServiceInstance command returns an object containing the following properties.

ServiceGroupUid <Guid>

\n Specifies the unique identifier for the service group of which the service is a member.

ServiceGroupName <String>

\n Specifies the name of the service group of which the service is a member.

ServiceInstanceUID <Guid>

\n Specifies the unique identifier for registered service instances, which are service instances held by and obtained from a

central configuration service. Unregistered service instances do not have unique identifiers.

ServiceType <String>

\n Specifies the service instance type. For this service, the service instance type is always Hyp.

Address

\n Specifies the address of the service instance. The address can be used to access the service and, when registered in the central configuration service, can be used by other services to access the service.

Binding

\n Specifies the binding type that must be used to communicate with the service instance. In this release of XenDesktop, the binding type is always 'wcf\_HTTP\_kerb'. This indicates that the service provides a Windows Communication Foundation endpoint that uses HTTP binding with integrated authentication.

Version

\n Specifies the version of the service instance. The version number is used to ensure that the correct versions of the services are used for communications.

ServiceAccount <String>

\n Specifies the Active Directory account name for the machine on which the service instance is running. The account name is used to provide information about the permissions required for interservice communications.

ServiceAccountSid <String>

\n Specifies the Active Directory account security identifier for the machine on which the service instance is running.

InterfaceType <String>

\n Specifies the interface type. Each service can provide multiple service instances, each for a different purpose, and the interface defines the purpose. Available interfaces are:

\n SDK - for PowerShell operations

\n InterService - for operations between different services

\n Peer - for communications between services of the same type

Metadata <Citrix.Host.Sdk.Metadata[]>

The collection of metadata associated with registered service instances, which are service instances held by and obtained from a central configuration service. Metadata is not stored for unregistered service instances.

Notes

If the command fails, the following errors can be returned.

Error Codes

-----

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-HypServiceInstance
```

```
Address : http://MyServer.com:80/Citrix/SdkHostingUnitService
Binding : wcf_HTTP_kerb
InterfaceType : SDK
Metadata :
MetadataMap :
ServiceAccount : ENG\MyAccount$
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164
ServiceGroupName : MyServiceGroup
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000
ServiceType : Hyp
Version : 1
```

```
Address : http://MyServer.com:80/Citrix/SdkHostingUnitService/IServiceApi
Binding : wcf_HTTP_kerb
InterfaceType : InterService
Metadata :
MetadataMap :
```

ServiceAccount : ENG\MyAccount  
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164  
ServiceGroupName : MyServiceGroup  
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d  
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000  
ServiceType : Hyp  
Version : 1

Get all instances of the Host Service running on the specified machine. For remote services, use the AdminAddress parameter to define the service for which the interfaces are required. If the AdminAddress parameter has not been specified for the runspace, service instances running on the local machine are returned.

# Get-HypServiceStatus

May 10, 2016

Gets the current status of the Host Service on the controller.

## Syntax

```
Get-HypServiceStatus [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables the status of the Host Service on the controller to be monitored. If the service has multiple data stores it will return the overall state as an aggregate of all the data store states. For example, if the site data store status is OK and the secondary data store status is DBUnconfigured then it will return DBUnconfigured.

## Related topics

[Set-HypDBConnection](#)

[Test-HypDBConnection](#)

[Get-HypDBConnection](#)

[Get-HypDBSchema](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Get-HypServiceStatus command returns an object containing the status of the Host Service together with extra diagnostics information.

DBUnconfigured

The Host Service does not have a database connection configured.

DBRejectedConnection

The database rejected the logon attempt from the Host Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

#### InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the Host Service schema has not been added to the database.

#### DBNotFound

The specified database could not be located with the configured connection string.

#### DBNewerVersionThanService

The version of the Host Service currently in use is incompatible with the version of the Host Service schema on the database. Upgrade the Host Service to a more recent version.

#### DBOlderVersionThanService

The version of the Host Service schema on the database is incompatible with the version of the Host Service currently in use. Upgrade the database schema to a more recent version.

#### DBVersionChangeInProgress

A database schema upgrade is currently in progress.

#### OK

The Host Service is running and is connected to a database containing a valid schema.

#### Failed

The Host Service has failed.

#### Unknown

(0) The service status cannot be determined.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-HypServiceStatus
```

DBUnconfigured

Get the current status of the Host Service.

# Get-HypVMMacAddress

May 10, 2016

Retrieves a list the MAC addresses for the VMs in the specified connection.

## Syntax

```
Get-HypVMMacAddress [-LiteralPath] <String> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Use this command to obtain a list of MAC addresses of all the virtual machines in the specified connection.

## Related topics

Get-Item

## Parameters

**-LiteralPath**<String>

The path to a connection item in the hosting provider. Paths to anything other than a connection item will result in an error being returned. The path can be provided as either <drive>:\connections\<Connection Name> or <drive>:\connections\{<Connection Uid>}

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

Systemstring You can pipe a string that contains a path to Get-HypConfigurationDataForItem

## Return Values



Citrix.Host.Sdk.HypervisorVMObject

Get-HypVMMMacAddress returns an object containing the following properties.

MacAddress <string> - specifies the MAC address of the VM.

VMId <string> - specifies the identifier for the VM as defined in the hypervisor hosting it.

## Notes

The path must refer to a connection item. Hosting unit items are not valid.

In the case of failure, the following errors can result.

## Error Codes

-----

### InputConnectionsPathInvalid

If the path is not provided in an expected format, an InputConnectionsPathInvalid error results.

### HypervisorConnectionObjectNotFound

The hypervisor connection object specified cannot be found.

### HypervisorInMaintenanceMode

The hypervisor for the connection is in maintenance mode.

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

### CommunicationError

An error occurred while communicating with the service.

### InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS>Get-HypVMMacAddress -LiteralPath XDHyp:\Connections\MyConnection
```

MacAddress	VMId
52:b0:1c:ed:60:fa	f3395d2a-a196-41c2-e37d-764acf871599
62:6f:f0:40:d5:af	5f4457b0-cc3c-f806-8ca7-5f57e4bdf2d1
4e:a5:9f:00:b2:0c	3115177b-85a9-d8ee-d0f9-0c7437483c09

This command gets the MAC addresses for the connection called "MyConnection".

### ----- EXAMPLE 2 -----

```
XDHyp:\Connections\MyConnection>Get-HypVMMacAddress -Path .
```

MacAddress	VMId
52:b0:1c:ed:60:fa	f3395d2a-a196-41c2-e37d-764acf871599
62:6f:f0:40:d5:af	5f4457b0-cc3c-f806-8ca7-5f57e4bdf2d1
4e:a5:9f:00:b2:0c	3115177b-85a9-d8ee-d0f9-0c7437483c09

This command gets the MAC addresses for the connection at the current directory. The dot (.) represents the current location (not its contents).

### ----- EXAMPLE 3 -----

```
C:\PS>Get-HypVMMacAddress -LiteralPath Xdhyp:\connections\"{268c66db-9b8c-47f6-9265-42326dbff006}"
```

This command gets the MAC addresses for the connection that has a ConnectionUid of 268c66db-9b8c-47f6-9265-42326dbff006.

# Get-HypVolumeServiceConfiguration

May 10, 2016

Gets instances of the VolumeServiceConfiguration that are configured for this site.

## Syntax

```
Get-HypVolumeServiceConfiguration [[-VolumeServiceConfigurationName] <String>] [-VolumeServiceConfigurationUid <Guid>] [-ConnectionType <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Retrieve VolumeServiceConfigurations whose properties match the given filter criteria. If no parameters are specified, this cmdlet retrieves all VolumeServiceConfiguration objects.

## Related topics

[Set-HypVolumeServiceConfiguration](#)

## Parameters

**-VolumeServiceConfigurationName**<String>

Specifies a filter for the configuration name.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-VolumeServiceConfigurationUid**<Guid>

Specifies a filter for the configuration ID.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ConnectionType**<String>

Specifies a filter for the cloud connection type, such as "AWS" or "CloudPlatform".

--	--

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property<String[]>**

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount<SwitchParameter>**

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Hyp\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount<Int32>**

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip<Int32>**

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Hyp\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

Input Type

string The name of a configuration (or set of configurations) to retrieve.

## Return Values

Citrix.Host.Sdk.VolumeServiceConfiguration

This cmdlet returns matching VolumeServiceConfiguration objects.

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS>Get-HypVolumeServiceConfiguration -VolumeServiceConfigurationName SiteDefault -ConnectionType CloudPlatform
```

This command lists the site default volume service configuration for connections that use Citrix Cloud Platform.

# Get-HypXenServerAddress

May 10, 2016  
Gets all the available addresses for a XenServer hypervisor connection.

## Syntax

Get-HypXenServerAddress [-LiteralPath] <String> [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Use this cmdlet to retrieve all the available hypervisor connection addresses that can be used to connect to the same XenServer pool. When used in conjunction with Add-HypHypervisorAddress, you can easily populate a connection with all the addresses that can be used to provide full failover support for a XenServer connection.

If the addresses are https addresses, the command uses the certificates installed on the XenServers to provide suitable https addresses where possible. Only servers that can be resolved are returned.

## Related topics

[Add-HypHypervisorConnectionAddress](#)

## Parameters

**-LiteralPath**<String>

Specifies the path within a Host Service provider to the hypervisor connection item to which to add the address. The path specified must be in one of the following formats: <drive>:\Connections\<HypervisorConnectionName> or <drive>:\Connections\{HHypervisorConnection Uid>}

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Systemstring

Get-HypXenServerAddress returns a list of strings containing the available address.

## Notes

For this to work as required with https connections, the certificates installed on the XenServers must be trusted by all controllers. Typically this means having the root certificate for the certificate trust chain installed on all controllers. Wildcard certificates are not supported for this operation.

In the case of failure, the following errors can result.

## Error Codes

-----

InputConnectionsPathInvalid

The path provided is not to an item in a sub-directory of a hosting unit item.

HypervisorConnectionNotXenServer

The hypervisor connection to which the path refers is not for a Citrix XenServer hypervisor.

#### HypervisorConnectionObjectNotFound

The hypervisor connection at the path specified cannot be located.

#### HypervisorInMaintenanceMode

The hypervisor for the connection is in maintenance mode.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Get-HypXenServerAddress -Path XDHyp:\Connections\MyConnection
```

```
https:\myserver.com
```

```
https:\myServer1.com
```

Gets the available addresses for the connection "MyConnection".

##### ----- EXAMPLE 2 -----

```
c:\PS>Get-HypXenServerAddress -LiteralPath XDHyp:\Connections\MyConnection | Add-HypHypervisorConnectionAddress -Path XDHyp:\Connections\MyConnectionPath
```

Adds all the available addresses for the XenServer pool in "MyConnection" to the hypervisor connection.



# Grant-HypSecurityGroupEgress

May 10, 2016

Adds an egress rule to a security group.

## Syntax

```
Grant-HypSecurityGroupEgress [-LiteralPath] <String> -GroupId <String[]> -Protocol <String> [-FromPort <Decimal>] [-ToPort <Decimal>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Grant-HypSecurityGroupEgress [-LiteralPath] <String> -IPRange <String[]> -Protocol <String> [-FromPort <Decimal>] [-ToPort <Decimal>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Adding an egress rule permits network traffic from instances within the security group to pass to one or more destination CIDR IP address ranges or security groups.

## Related topics

Amazon AuthorizeSecurityGroupEgress: <http://docs.aws.amazon.com/AWSEC2/latest/APIReference/ApiReference-query-AuthorizeSecurityGroupEgress.html>

IANA protocol numbers: <http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml>

[Grant-HypSecurityGroupIngress](#)

[Revoke-HypSecurityGroupIngress](#)

[Revoke-HypSecurityGroupEgress](#)

## Parameters

### **-LiteralPath**<String>

Specifies the full XD Hyp provider path to the security group, equivalent to the FullPath property of the security group object. The path can specify a security group relative to a hypervisor connection or hosting unit.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Protocol**<String>

Specifies the protocol name or number. Protocol numbers can be found at: <http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml>

Use -1 to specify all protocols.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-GroupId<String[]>**

Specifies one or more destination security groups to which traffic will be permitted by this rule. This parameter cannot be specified in conjunction with IPRange.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-IPRange<String[]>**

Specifies one or more destination CIDR IP address ranges to which traffic will be permitted by this rule. This parameter cannot be specified in conjunction with IPRange.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-FromPort<Decimal>**

The start of the port range for port based protocols. For ICMP this specifies the type number.

Use -1 to specify all ICMP types.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-ToPort<Decimal>**

The end of the port range for port based protocols. For ICMP this specifies the type number, where -1 can be used to specify all ICMP types.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and

Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

**Input Type**

System.string The LiteralPath can be piped in.

**Return Values**

None

**Notes**

Security groups can be added and removed using the New-Item and Remove-Item cmdlets.

**Examples**

----- **EXAMPLE 1** -----

```
c:\PS> $Group = New-Item -ItemType SecurityGroup -Path XDHyp:\Connections\AWS -Name MySecurityGroup -Description 'Example group'
c:\PS> Grant-HypSecurityGroupEgress $Group.FullPath -Protocol '-1' -IPRange '0.0.0.0/16'
```

Create a security group and grant full egress to anywhere.

----- **EXAMPLE 2** -----

```
c:\PS> $Group1 = New-Item -ItemType SecurityGroup -Path XDHyp:\Connections\AWS -Name MySecurityGroup1 -Description 'Example group 1'
c:\PS> $Group2 = New-Item -ItemType SecurityGroup -Path XDHyp:\Connections\AWS\MySecurityGroup2 -Description 'Example group 2'
c:\PS> Grant-HypSecurityGroupEgress $Group1.FullPath -FromPort 0 -ToPort 0 -Protocol icmp -GroupId $Group2.Id
c:\PS> Grant-HypSecurityGroupIngress $Group2.FullPath -FromPort 0 -ToPort 0 -Protocol icmp -GroupId $Group1.Id
```

Make 2 security groups and permit group 1 to ping group 2.

# Grant-HypSecurityGroupIngress

May 10, 2016

Adds an ingress rule to a security group.

## Syntax

```
Grant-HypSecurityGroupIngress [-LiteralPath <String> -GroupId <String[]> -Protocol <String> [-FromPort <Decimal>] [-ToPort <Decimal>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Grant-HypSecurityGroupIngress [-LiteralPath <String> -IPRange <String[]> -Protocol <String> [-FromPort <Decimal>] [-ToPort <Decimal>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Adding an egress rule permits network traffic from source CIDR IP address ranges or security groups to pass to instances within a security group.

## Related topics

Amazon AuthorizeSecurityGroupEgress: <http://docs.aws.amazon.com/AWSEC2/latest/APIReference/ApiReference-query-AuthorizeSecurityGroupEgress.html>

IANA protocol numbers: <http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml>

[Grant-HypSecurityGroupEgress](#)

[Revoke-HypSecurityGroupIngress](#)

[Revoke-HypSecurityGroupEgress](#)

## Parameters

**-LiteralPath**<String>

Specifies the full XD Hyp provider path to the security group, equivalent to the FullPath property of the security group object. The path can specify a security group relative to a hypervisor connection or hosting unit.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Protocol**<String>

Specifies the protocol name or number. Protocol numbers can be found at: <http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml>

Use -1 to specify all protocols.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-GroupId<String[]>**

Specifies one or more source security groups from which traffic will be permitted by this rule. This parameter cannot be specified in conjunction with IPRange.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-IPRange<String[]>**

Specifies one or more source CIDR IP address ranges from which traffic will be permitted by this rule. This parameter cannot be specified in conjunction with IPRange.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-FromPort<Decimal>**

The start of the port range for port based protocols. For ICMP this specifies the type number.

Use -1 to specify all ICMP types.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-ToPort<Decimal>**

The end of the port range for port based protocols. For ICMP this specifies the type number, where -1 can be used to specify all ICMP types.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and

Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

**Input Type**

System.string The LiteralPath can be piped in.

**Return Values**

None

**Notes**

Security groups can be added and removed using the New-Item and Remove-Item cmdlets.

**Examples**

----- **EXAMPLE 1** -----

```
c:\PS> $Group = New-Item -ItemType SecurityGroup -Path XDHyp:\Connections\AWS -Name MySecurityGroup -Description 'Example group'
c:\PS> Grant-HypSecurityGroupIngress $Group.FullPath -FromPort 80 -ToPort 80 -Protocol tcp -IPRange '0.0.0.0/0'
Create a security group and grant ingress on port 80 from anywhere.
```

----- **EXAMPLE 2** -----

```
c:\PS> $Group1 = New-Item -ItemType SecurityGroup -Path XDHyp:\Connections\AWS -Name MySecurityGroup1 -Description 'Example group 1'
c:\PS> $Group2 = New-Item -ItemType SecurityGroup -Path XDHyp:\Connections\AWS\MySecurityGroup2 -Description 'Example group 2'
c:\PS> Grant-HypSecurityGroupEgress $Group1.FullPath -FromPort 8080 -ToPort 8080 -Protocol tcp -GroupId $Group2.Id
c:\PS> Grant-HypSecurityGroupIngress $Group2.FullPath -FromPort 8080 -ToPort 8080 -Protocol tcp -GroupId $Group1.Id
c:\PS> Grant-HypSecurityGroupEgress $Group2.FullPath -Protocol '-1' -GroupId $Group1.Id
c:\PS> Grant-HypSecurityGroupIngress $Group1.FullPath -Protocol '-1' -GroupId $Group2.Id
Make 2 security groups and permit group 1 access to group 2 only on port 8080 while granting full access to group 1 from group 2.
```

# New-HypVMSnapshot

May 10, 2016  
Creates a new snapshot for the specified VM item path.

## Syntax

New-HypVMSnapshot [-LiteralPath] <String> [-SnapshotName] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [[-SnapshotDescription] <String>] [-<CommonParameters>]

## Detailed Description

Use this command to create a new snapshot of a virtual machine, for a given Host Service provider path to a VM. The resulting snapshot can then be used for operations that require a snapshot to work.

## Related topics

## Parameters

**-LiteralPath**<String>

Specifies the path within a hosting unit provider to the virtual machine item for which to create a new snapshot. The path specified must be in one of the following formats: <drive>:\Connections\<Connection Name>\<Item Path of VM object> or <drive>:\Connections\{<connection Uid>\<Item Path of VM object>} or <drive>:\HostingUnits\<HostingUnit Name>\<Item Path of VM object> or <drive>:\HostingUnits\{<hostingUnit Uid>\<Item Path of VM object>}

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-SnapshotName**<String>

The name of the new snapshot. This is visible in the hypervisor management console.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

#### **-SnapshotDescription**<String>

The description to add to the snapshot. This is visible in the hypervisor management console.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### Input Type

Systemstring You can pipe a string that contains a path to Get-HypConfigurationDataForItem

#### Return Values

Systemstring.

The provider path to the newly created snapshot.

#### Notes

In the case of failure, the following errors can result.

#### Error Codes

-----

##### InputHypervisorItemPathInvalid

The path provided is not to an item in a sub-directory of a connection item or a hosting unit item.

##### InvalidHypervisorItemPath

No item exists with the specified path.

##### InvalidHypervisorItem

The item specified by the path exists, but is not a VM Item.

##### SnapshotNameAlreadyInUse

The specified name is already in use and will cause a name resolution clash.

##### FailedToCreateSnapshot

The snapshot creation process failed.

##### HypervisorInMaintenanceMode

The hypervisor is in maintenance mode.

##### DatabaseError

An error occurred in the service while attempting a database operation.

##### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

##### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

##### CommunicationError



An error occurred while communicating with the service.

PermissionDenied

The user does not have administrative rights to perform this operation.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

SnapshotChainTooLong

Snapshot creation failed. Snapshot chain is too long.

SnapshotCreationNotAuthorized

Snapshot creation failed. User not authorized to create snapshots.

Examples

----- **EXAMPLE 1** -----

```
C:\PS>New-HypVMSnapshot -LiteralPath XDHyp:\Connections\MyConnection\MyVm.vm -SnapshotName "New snapshot" -SnapshotDescription "Example snapshot"
```

XDHyp:\Connections\MyConnection\MyVm.vm\New snapshot.snapshot

This command creates a snapshot of a VM called 'MyVm.vm' within a hypervisor connection called 'MyConnection'.

# Remove-HypHostingUnitMetadata

May 10, 2016

Removes metadata from the given HostingUnit.

## Syntax

```
Remove-HypHostingUnitMetadata [-HostingUnitUid] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-HypHostingUnitMetadata [-HostingUnitUid] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-HypHostingUnitMetadata [-HostingUnitName] <String> -Name <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Remove-HypHostingUnitMetadata [-HostingUnitName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Remove-HypHostingUnitMetadata [-InputObject] <HostingUnit[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Remove-HypHostingUnitMetadata [-InputObject] <HostingUnit[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given HostingUnit.

## Related topics

[Add-HypHostingUnitMetadata](#)

[Set-HypHostingUnitMetadata](#)

## Parameters

**-HostingUnitUid**<Guid>

Id of the HostingUnit

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-HostingUnitName**<String>

Name of the HostingUnit

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-InputObject**<HostingUnit[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create

high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- **EXAMPLE 1** -----

```
c:\PS>Get-HypHostingUnit | %{ Remove-HypHostingUnitMetadata -Map $_.MetadataMap }
Remove all metadata from all HostingUnit objects.
```

# Remove-HypHostingUnitNetwork

May 10, 2016  
Removes networks from a hosting unit.

## Syntax

Remove-HypHostingUnitNetwork [-LiteralPath] <String> [-NetworkPath] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Use this command to remove networks from a hosting unit. This does not remove the network from the hypervisor, only the reference to the network for the Host Service. After it is removed, the network is no longer available for associating with virtual NICs (when creating new virtual machines with the Machine Creation Service). Any virtual machines that have been created already continue to use the network until they are removed from the deployment. This command cannot be used if the connection for the hosting unit is in maintenance mode. If the network to be removed no longer exists on the hypervisor for the hosting unit, you must supply a fully qualified path to the network location.

## Related topics

[Add-HypHostingUnitNetwork](#)

## Parameters

**-LiteralPath**<String>

Required?	true
Default Value	
Accept Pipeline Input?	false

**-NetworkPath**<String>

Specifies the path in the hosting unit provider of the network to remove. The path specified must be in one of the following formats: <drive>:\Connections\  
<HostingUnitName>\MyNetwork.network or <drive>:\Connections\{<hostingUnit Uid>\MyNetwork.network

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. This can be a host name or an IP address.

Required?	false

Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

System.String You can pipe a string that contains a path to Remove-HypHostingUnitNetwork (Path parameter).

## Notes

In the case of failure, the following errors can result.

## Error Codes

-----

### HostingUnitsPathInvalid

The path provided is not to an item in a subdirectory of a hosting unit item.

### HostingUnitNetworkObjectToDeleteDoesNotExist

The network path specified is not part of the hosting unit.

### HypervisorInMaintenanceMode

The hypervisor for the connection is in maintenance mode.

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

### DataStoreException

An error occurred in the service while attempting a database operation. Communication with the database failed for various reasons.

### CommunicationError

An error occurred while communicating with the service.

### PermissionDenied

The user does not have administrative rights to perform this operation.

### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used, or examine the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

c:\PS>Remove-HypHostingUnitNetwork -LiteralPath XDHyp:\HostingUnits\MyHostingUnit -NetworkPath 'XDHyp:\HostingUnits\MyHostingUnits\newNetwork.network'  
The command removes the network called "newNetwork.network" from the hosting unit called "MyHostingUnit"

# Remove-HypHostingUnitStorage

May 10, 2016  
Removes storage from a hosting unit.

## Syntax

Remove-HypHostingUnitStorage [-LiteralPath] <String> [-StoragePath <String>] [-StorageType <StorageType>] [-LoggingId <Guid>] [-AdminAddress <String>] [-CommonParameters]

## Detailed Description

Use this command to remove storage locations from a hosting unit. This does not remove the storage from the hypervisor, only the reference to the storage for the Host Service. After removal, the storage is no longer used to store hard disks (when creating new virtual machines with the Machine Creation Service). The hard disks located already on the storage remain in place and virtual machines that have been created already continue to use the storage until they are removed from the deployment. Do not use this command if the connection for the hosting unit is in maintenance mode. If the storage location to be removed no longer exists on the hypervisor for the hosting unit, you must supply a fully qualified path to the storage location.

## Related topics

[Add-HypHostingUnitStorage](#)

## Parameters

**-LiteralPath**<String>

Required?	true
Default Value	
Accept Pipeline Input?	false

**-StoragePath**<String>

Specifies the path in the hosting unit provider of the storage to remove. If StoragePath is not specified, all storage is removed from the hosting unit. The path specified must be in one of the following formats: <drive>:\Connections\<HostingUnitName>\MyStorage.storage or <drive>:\Connections\{<hostingUnitUid>}\MyStorage.storage

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

**-StorageType**<StorageType>

Specifies the type of storage in StoragePath. Supported storage types are: OSStorage PersonalDiskStorage

Required?	false
Default Value	OSStorage
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.



Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. This can be a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

System.string You can pipe a string that contains a path to Remove-HypHostingUnitStorage (Path parameter).

#### Notes

After storage is removed, it is the administrator's responsibility to maintain its contents. The Citrix XenDesktop Machine Creation Service does not attempt to clean up any data that is stored in the storage location.

If all storage is removed from the hosting unit, other features of the Machine Creation Service stops functioning until some storage is added again.

In the case of failure, the following errors can result.

#### Error Codes

-----

##### HostingUnitsPathInvalid

The path provided is not to an item in a subdirectory of a hosting unit item.

##### HostingUnitStorageObjectToDeleteDoesNotExist

The storage path specified is not part of the hosting unit.

##### HypervisorInMaintenanceMode

The hypervisor for the connection is in maintenance mode.

##### DatabaseError

An error occurred in the service while attempting a database operation.

##### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

##### DataStoreException

An error occurred in the service while attempting a database operation. Communication with the database failed for various reasons.

##### CommunicationError

An error occurred while communicating with the service.

##### PermissionDenied

The user does not have administrative rights to perform this operation.

## ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used, or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Remove-HypHostingUnitStorage -LiteralPath XDHyp:\HostingUnits\MyHostingUnit -StoragePath 'XDHyp:\HostingUnits\MyHostingUnits\newStorage.storage'
```

The command removes the OS storage location called "newStorage.storage" from the hosting unit called "MyHostingUnit"

### ----- EXAMPLE 2 -----

```
c:\PS>Get-ChildItem XDHYP:\HostingUnits\Host*.storage | Remove-HypHostingUnitStorage XDHYP:\HostingUnits\Host1
```

The command removes all OS storage from all hosting units called "Host1".

### ----- EXAMPLE 3 -----

```
c:\PS>Get-ChildItem XDHYP:\HostingUnits\Host*.storage | Remove-HypHostingUnitStorage -StorageType PersonalVdiskStorage
```

The command removes all PersonalVdisk storage from all hosting units called "Host1".

# Remove-HypHypervisorConnectionAddress

May 10, 2016

Removes addresses from the list of available connection addresses.

## Syntax

```
Remove-HypHypervisorConnectionAddress [-LiteralPath] <String> [-HypervisorAddress] <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Use this command to remove addresses that can be used to connect to the hypervisor specified by the hypervisor connection. If all addresses are removed, the connection cannot be used until a valid address is added to the hypervisor connection.

## Related topics

[Add-HypHypervisorConnectionAddress](#)

## Parameters

### **-LiteralPath**<String>

Specifies the path within a Host Service provider to the hosting unit item to which to add the address. The path specified must be in one of the following formats: <drive>:\HostingUnits\<HostingUnitName> or <drive>:\HostingUnits\{HostingUnit Uid>

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-HypervisorAddress**<String>

Specifies the address to be removed. If this parameter is not provided, all addresses are removed from the hypervisor connection.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

Systemstring You can pipe a string that contains a path to Remove-HypHypervisorConnectionAddress (Path parameter).

## Notes

Changes to a hypervisor connection affect all entities that reference the connection. If all addresses are removed from the connection, any other entities that reference the hypervisor connection (e.g. hosting units) cannot make new connections to the hypervisor.

In the case of failure, the following errors can result.

## Error Codes

-----

### InputConnectionsPathInvalid

The path provided is not to an item in a subdirectory of a hosting unit item.

### HypervisorConnectionAddressForeignKeyObjectDoesNotExist

The hypervisor connection to which the address is to be added could not be located.

### HypervisorInMaintenanceMode

The hypervisor for the connection is in maintenance mode.

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

### CommunicationError

An error occurred while communicating with the service.

### PermissionDenied

The user does not have administrative rights to perform this operation.

### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Remove-HypHypervisorConnectionAddress -LiteralPath XDHyp:\HostingUnits\MyHypervisorConnection -HypervisorAddress 'http:\\myserver.com'
```

The command removes the address "http:\\myserver.com" from the hypervisor connection called "MyHypervisorConnection".

----- **EXAMPLE 2** -----

```
c:\PS>Get-Item -Path XDHYP:\Connections* | Remove-HypHypervisorConnectionAddress
```

The command removes all addresses from all the hypervisor connections currently defined.

# Remove-HypHypervisorConnectionMetadata

May 10, 2016

Removes metadata from the given HypervisorConnection.

## Syntax

```
Remove-HypHypervisorConnectionMetadata [-HypervisorConnectionUid] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-HypHypervisorConnectionMetadata [-HypervisorConnectionUid] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-HypHypervisorConnectionMetadata [-HypervisorConnectionName] <String> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-HypHypervisorConnectionMetadata [-HypervisorConnectionName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-HypHypervisorConnectionMetadata [-InputObject] <HypervisorConnection[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-HypHypervisorConnectionMetadata [-InputObject] <HypervisorConnection[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given HypervisorConnection.

## Related topics

[Add-HypHypervisorConnectionMetadata](#)

[Set-HypHypervisorConnectionMetadata](#)

## Parameters

**-HypervisorConnectionUid**<Guid>

Id of the HypervisorConnection

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-HypervisorConnectionName**<String>

Name of the HypervisorConnection

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-InputObject**<HypervisorConnection[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create

high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.



PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-HypHypervisorConnection | % { Remove-HypHypervisorConnectionMetadata -Map $_.MetadataMap }
Remove all metadata from all HypervisorConnection objects.
```

# Remove-HypHypervisorConnectionScope

May 10, 2016  
Remove the specified HypervisorConnection(s) from the given scope(s).

## Syntax

```
Remove-HypHypervisorConnectionScope [-Scope] <String[]> -InputObject <HypervisorConnection[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Remove-HypHypervisorConnectionScope [-Scope] <String[]> -HypervisorConnectionUid <Guid[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Remove-HypHypervisorConnectionScope [-Scope] <String[]> -HypervisorConnectionName <String[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The RemoveHypHypervisorConnectionScope cmdlet is used to remove one or more HypervisorConnection objects from the given scope(s).

There are multiple parameter sets for this cmdlet, allowing you to identify the HypervisorConnection objects in different ways:

- HypervisorConnection objects can be piped in or specified by the InputObject parameter
- The HypervisorConnectionUid parameter specifies objects by HypervisorConnectionUid
- The HypervisorConnectionName parameter specifies objects by HypervisorConnectionName (supports wildcards)

To remove a HypervisorConnection from a scope you need permission to change the scopes of the HypervisorConnection.

If the HypervisorConnection is not in a specified scope, that scope will be silently ignored.

## Related topics

- [Add-HypHypervisorConnectionScope](#)
- [Get-HypScopedObject](#)

## Parameters

**-Scope**<String[]>

Specifies the scopes to remove the objects from.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-InputObject**<HypervisorConnection[]>

Specifies the HypervisorConnection objects to be removed.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByValue, ByPropertyName)
------------------------	--------------------------------

#### **-HypervisorConnectionUid**<Guid[]>

Specifies the HypervisorConnection objects to be removed by HypervisorConnectionUid.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-HypervisorConnectionName**<String[]>

Specifies the HypervisorConnection objects to be removed by HypervisorConnectionName.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

Return Values

None

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### UnknownObject

One of the specified objects was not found.

#### ScopeNotFound

One of the specified scopes was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command with the specified objects or scopes.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Remove-HypHypervisorConnectionScope Finance -HypervisorConnectionUid 6702C5D0-C073-4080-A0EE-EC74CB537C52
```

Removes a single HypervisorConnection from the 'Finance' scope.

### ----- EXAMPLE 2 -----

```
c:\PS>Remove-HypHypervisorConnectionScope Finance,Marketing -HypervisorConnectionUid 6702C5D0-C073-4080-A0EE-EC74CB537C52
```

Removes a single HypervisorConnection from multiple scopes.

### ----- EXAMPLE 3 -----

```
c:\PS>Get-HypHypervisorConnection | Remove-HypHypervisorConnectionScope Finance
```

Removes all visible HypervisorConnection objects from the 'Finance' scope.

----- **EXAMPLE 4** -----

```
c:\PS>Remove-HypHypervisorConnectionScope Finance -HypervisorConnectionName A*
```

Removes HypervisorConnection objects with a name starting with an 'A' from the 'Finance' scope.

# Remove-HypMetadata

May 10, 2016

Removes metadata from a hypervisor connection or hosting unit.

## Syntax

```
Remove-HypMetadata [-LiteralPath] <String> [-Property <String>] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Use this command to remove custom data from a specific hosting unit or hypervisor connection. If the Property parameter is not provided, all metadata is removed from the specified item.

## Related topics

[Remove-HypMetadata](#)

## Parameters

### **-LiteralPath**<String>

Specifies the path within a Host Service provider to the hosting unit or hypervisor connection item from which to remove the metadata. The path specified must be in one of the following formats: <drive>:\HostingUnits\<HostingUnitName> or <drive>:\HostingUnits\{<HostingUnit Uid> or <drive>:\Connections\<Connection Name> or <drive>:\Connections\{<Connection Uid>}

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Property**<String>

Specifies the property name of the metadata to be removed.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

#### Input Type

System.string You can pipe a string that contains a path to Add-HypMetadata (Path parameter).

#### Notes

In the case of failure, the following errors can result.

#### Error Codes

-----

##### InvalidPath

The path provided is not in the required format.

##### HostingUnitMetadataObjectToDeleteDoesNotExist

The hosting unit supplied in the path does not exist.

##### HypervisorConnectionObjectToDeleteDoesNotExist

The hypervisor connection supplied in the path does not exist.

##### MetadataContainerUndefined

The specified path does not reference a hosting unit or a hypervisor connection.

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## CommunicationError

An error occurred while communicating with the service.

## PermissionDenied

The user does not have administrative rights to perform this operation.

## ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS>Remove-HypMetadata -LiteralPath XDHyp:\Connections\MyConnection -Property MyProperty
```

The command removes the metadata with the property "MyProperty" from the hypervisor connection called "MyConnection".

### ----- EXAMPLE 2 -----

```
C:\PS>Remove-HypMetadata -LiteralPath XDHyp:\Connections\MyConnection
```

The command removes all the metadata from the hypervisor connection called "MyConnection".

### ----- EXAMPLE 3 -----

```
C:\PS>dir XDHyp:\connections | Remove-HypMetadata
```

The command removes all the metadata from all the hypervisor connections.



# Remove-HypServiceMetadata

May 10, 2016

Removes metadata from the given Service.

## Syntax

```
Remove-HypServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-HypServiceMetadata [-ServiceHostId] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-HypServiceMetadata [-InputObject] <Service[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-HypServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Service.

## Related topics

[Set-HypServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name<String>**

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-HypService | %{ Remove-HypServiceMetadata -Map $_.MetadataMap }
Remove all metadata from all Service objects.
```

# Reset-HypServiceGroupMembership

May 10, 2016

Reloads the access permissions and configuration service locations for the Host Service.

## Syntax

```
Reset-HypServiceGroupMembership [-ConfigServiceInstance] <ServiceInstance[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Enables you to reload Host Service access permissions and configuration service locations. The Reset-HypServiceGroupMembership command must be run on at least one instance of the service type (Hyp) after installation and registration with the configuration service. Without this operation, the Host services will be unable to communicate with other services in the XenDesktop deployment. When the command is run, the services are updated when additional services are added to the deployment, provided that the configuration service is not stopped. The Reset-HypServiceGroupMembership command can be run again to refresh this information if automatic updates do not occur when new services are added to the deployment. If more than one configuration service instance is passed to the command, the first instance that meets the expected service type requirements is used.

## Related topics

### Parameters

**-ConfigServiceInstance**<ServiceInstance[]>

Specifies the configuration service instance object that represents the service instance for the type 'InterService' that references a configuration service for the deployment.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
-----------	-------

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Host.Sdk.ServiceInstance[] Service instances containing a ServiceInstance object that refers to the central configuration service interservice interface can be piped to the Reset-HypServiceGroupMembership command.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### NoSuitableServiceInstance

None of the supplied service instance objects were suitable for resetting service group membership.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config | Reset-HypServiceGroupMembership

Reset the service group membership for a service in a deployment where the configuration service is configured and running on the same machine as the service.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config -AdminAddress OtherServer.example.com | Reset-HypServiceGroupmembership
```

Reset the service group membership for a service in a deployment where the configuration service that is configured and running on a machine named 'OtherServer.example.com'.

# Revoke-HypSecurityGroupEgress

May 10, 2016  
Removes an egress rule from a security group.

## Syntax

Revoke-HypSecurityGroupEgress [-LiteralPath <String> -GroupId <String[]> -Protocol <String> [-FromPort <Decimal>] [-ToPort <Decimal>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Revoke-HypSecurityGroupEgress [-LiteralPath <String> -IPRange <String[]> -Protocol <String> [-FromPort <Decimal>] [-ToPort <Decimal>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

To remove a rule, specify parameters matching an existing rule's values.

## Related topics

Amazon AuthorizeSecurityGroupEgress: <http://docs.aws.amazon.com/AWSEC2/latest/APIReference/ApiReference-query-AuthorizeSecurityGroupEgress.html>

IANA protocol numbers: <http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml>

[Grant-HypSecurityGroupIngress](#)

[Grant-HypSecurityGroupEgress](#)

[Revoke-HypSecurityGroupIngress](#)

## Parameters

**-LiteralPath**<String>

Specifies the full XDHy provider path to the security group, equivalent to the FullPath property of the security group object. The path can specify a security group relative to a hypervisor connection or hosting unit.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Protocol**<String>

Specifies the protocol name or number. Protocol numbers can be found at: <http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml>

Use -1 to specify all protocols.

Required?	true
Default Value	
Accept Pipeline Input?	false



--	--

#### **-GroupId<String[]>**

Specifies one or more destination security groups to which traffic will be permitted by this rule. This parameter cannot be specified in conjunction with IPRange.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-IPRange<String[]>**

Specifies one or more destination CIDR IP address ranges to which traffic will be permitted by this rule. This parameter cannot be specified in conjunction with IPRange.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-FromPort<Decimal>**

The start of the port range for port based protocols. For ICMP this specifies the type number.

Use -1 to specify all ICMP types.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-ToPort<Decimal>**

The end of the port range for port based protocols. For ICMP this specifies the type number, where -1 can be used to specify all ICMP types.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

System.string The LiteralPath can be piped in.

#### Return Values

None

#### Notes

Security groups cannot be removed in AWS if they are referenced by rules from other security groups.

Security groups can be added and removed using the New-Item and Remove-Item cmdlets.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS> $Group = New-Item -ItemType SecurityGroup -Path XDHyp:\Connections\AWS -Name MySecurityGroup -Description 'Example group'
c:\PS> Grant-HypSecurityGroupEgress $Group.FullPath -Protocol '-1' -IPRange '0.0.0.0/0'
c:\PS> Revoke-HypSecurityGroupEgress $Group.FullPath -Protocol '-1' -IPRange '0.0.0.0/0'
c:\PS> Remove-Item $Group.FullPath
```

Create a security group, grant full egress to anywhere, then revoke access and delete the security group.

# Revoke-HypSecurityGroupIngress

May 10, 2016

Removes an ingress rule from a security group.

## Syntax

Revoke-HypSecurityGroupIngress [-LiteralPath <String> -GroupId <String[]> -Protocol <String> [-FromPort <Decimal>] [-ToPort <Decimal>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Revoke-HypSecurityGroupIngress [-LiteralPath <String> -IPRange <String[]> -Protocol <String> [-FromPort <Decimal>] [-ToPort <Decimal>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

To remove a rule, specify parameters matching an existing rule's values.

## Related topics

Amazon AuthorizeSecurityGroupEgress: <http://docs.aws.amazon.com/AWSEC2/latest/APIReference/ApiReference-query-AuthorizeSecurityGroupEgress.html>

IANA protocol numbers: <http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml>

[Grant-HypSecurityGroupIngress](#)

[Grant-HypSecurityGroupEgress](#)

[Revoke-HypSecurityGroupIngress](#)

## Parameters

**-LiteralPath**<String>

Specifies the full XD Hyp provider path to the security group, equivalent to the FullPath property of the security group object. The path can specify a security group relative to a hypervisor connection or hosting unit.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Protocol**<String>

Specifies the protocol name or number. Protocol numbers can be found at: <http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml>

Use -1 to specify all protocols.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-GroupId**<String[]>

Specifies one or more source security groups from which traffic will be permitted by this rule. This parameter cannot be specified in conjunction with IPRange.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-IPRange**<String[]>

Specifies one or more source CIDR IP address ranges from which traffic will be permitted by this rule. This parameter cannot be specified in conjunction with IPRange.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-FromPort**<Decimal>

The start of the port range for port based protocols. For ICMP this specifies the type number.

Use -1 to specify all ICMP types.

Required?	false
Default Value	0
Accept Pipeline Input?	false

**-ToPort**<Decimal>

The end of the port range for port based protocols. For ICMP this specifies the type number, where -1 can be used to specify all ICMP types.

Required?	false
Default Value	0
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

Systemstring The LiteralPath can be piped in.

#### Return Values

None

#### Notes

Security groups cannot be removed in AWS if they are referened by rules from other security groups.

Security groups can be added and removed using the New-Item and Remove-Item cmdlets.

#### Examples

##### ----- **EXAMPLE 1** -----

```
c:\PS> $Group1 = New-Item -ItemType SecurityGroup -Path XDHyp:\Connections\AWS -Name MySecurityGroup1 -Description 'Example group 1'
c:\PS> $Group2 = New-Item -ItemType SecurityGroup -Path XDHyp:\Connections\AWS -Name MySecurityGroup2 -Description 'Example group 2'
c:\PS> Grant-HypSecurityGroupEgress $Group1.FullPath -FromPort 8080 -ToPort 8085 -Protocol tcp -GroupId $Group2.Id
c:\PS> Grant-HypSecurityGroupIngress $Group2.FullPath -FromPort 8080 -ToPort 8085 -Protocol tcp -GroupId $Group1.Id
c:\PS> Revoke-HypSecurityGroupEgress $Group1.FullPath -FromPort 8080 -ToPort 8085 -Protocol tcp -GroupId $Group2.Id
c:\PS> Revoke-HypSecurityGroupIngress $Group2.FullPath -FromPort 8080 -ToPort 8085 -Protocol tcp -GroupId $Group1.Id
```

Create 2 security groups, grant access from group 1 to group 2, then revoke access.

# Set-HypAdminConnection

May 10, 2016

Set the controller to be used by the cmdlets that form the Host service PowerShell snap-in.

## Syntax

Set-HypAdminConnection [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Use this command to set the default controller address to be used by the cmdlets to communicate with the controller. Most Host service cmdlets take an 'AdminAddress' parameter that can be used to define the controller (when used, this overrides this setting). However, the Set-Location cmdlet in the Host service provider does not offer this option. Therefore, the controller address must be set using this cmdlet, if no other cmdlets have set the address previously in the current runspace.

## Related topics

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Examples

----- **EXAMPLE 1** -----

c:\PS>Set-HypAdminConnection -AdminAddress myserver.com

This command sets the controller address for the commands to be "myserver.com". All commands run use this address until it is altered, either by another call to this command or by the use of the 'AdminAddress' parameter in another command in the Host service snap-in.

# Set-HypDBConnection

May 10, 2016

Configures a database connection for the Host Service.

## Syntax

```
Set-HypDBConnection [-DBConnection] <String> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Configures a connection to a database in which the Host Service can store its state. The service will attempt to connect and start using the database immediately after the connection is configured. The database connection string is updated to the specified value regardless of whether it is valid or not. Specifying an invalid connection string prevents a service from functioning until the error is corrected.

After a connection is configured, you cannot alter it without first clearing it (by setting the connection to \$null).

You do not need to configure a database connection to use this command.

## Related topics

[Get-HypServiceStatus](#)

[Get-HypDBConnection](#)

[Test-HypDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be used by the Host Service. Passing in \$null will clear any existing database connection configured.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Force**<SwitchParameter>

If present, allows the local administrator to set the connection string to null when there are problems contacting the database or other services.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and

Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

**Return Values**

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Set-HypDBConnection command returns an object containing the status of the Host Service together with extra diagnostics information.

**DBUnconfigured**

The Host Service does not have a database connection configured.

**DBRejectedConnection**

The database rejected the logon attempt from the Host Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

**InvalidDBConfigured**

The expected stored procedures are missing from the database. This may be because the Host Service schema has not been added to the database.

**DBNotFound**

The specified database could not be located with the configured connection string.

**DBNewerVersionThanService**

The version of the Host Service currently in use is incompatible with the version of the Host Service schema on the database. Upgrade the Host Service to a more recent version.

**DBOlderVersionThanService**

The version of the Host Service schema on the database is incompatible with the version of the Host Service currently in use. Upgrade the database schema to a more recent version.

**DBVersionChangeInProgress**



A database schema upgrade is currently in progress.

OK

The Host Service is running and is connected to a database containing a valid schema.

Failed

The Host Service has failed.

Unknown

The status of the Host Service cannot be determined.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

InvalidDBConnectionString

The database connection string has an invalid format.

DatabaseConnectionDetailsAlreadyConfigured

There was already a database connection configured. After a configuration is set, it can only be set to \$null.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Set-HypDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Configures a database connection string for the Host Service.

----- **EXAMPLE 2** -----

```
c:\PS>Set-HypDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Configures an invalid database connection string for the Host Service.

# Set-HypHostingUnitMetadata

May 10, 2016  
Adds or updates metadata on the given HostingUnit.

## Syntax

```
Set-HypHostingUnitMetadata [-HostingUnitUid] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Set-HypHostingUnitMetadata [-HostingUnitUid] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Set-HypHostingUnitMetadata [-HostingUnitName] <String> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Set-HypHostingUnitMetadata [-HostingUnitName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Set-HypHostingUnitMetadata [-InputObject] <HostingUnit[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Set-HypHostingUnitMetadata [-InputObject] <HostingUnit[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability for additional custom data to be stored against given HostingUnit objects.

## Related topics

- [Add-HypHostingUnitMetadata](#)
- [Remove-HypHostingUnitMetadata](#)

## Parameters

**-HostingUnitUid**<Guid>  
Id of the HostingUnit

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-HostingUnit Name**<String>  
Name of the HostingUnit

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-InputObject**<HostingUnit[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the HostingUnit specified. The property cannot contain any of the following characters \;#.\*?=<>|[]{}"

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Value**<String>

Specifies the value for the property.

Required?	true

Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-HypHostingUnitMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

InvalidParameterCombination

The cmdlet parameters are inconsistent.

UnknownObject

One of the specified objects was not found.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Set-HypHostingUnitMetadata -HostingUnitUid 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value

Key	Value
---	-----
property	value

Add metadata with a name of 'property' and a value of 'value' to the HostingUnit with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Set-HypHostingUnitStorage

May 10, 2016  
Sets options for a storage location on a hosting unit.

## Syntax

Set-HypHostingUnitStorage [-LiteralPath] <String> [-StoragePath] <String> [-StorageType <StorageType>] [-Superseded <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Use this command to set options for storage locations that are defined for a hosting unit. Do not use this command if the connection for the hosting unit is in maintenance mode.

## Related topics

## Parameters

**-LiteralPath**<String>

Specifies the path to the hosting unit which defines the storage. The path must be in one of the following formats: <drive>:\HostingUnits\<HostingUnitName> or <drive>:\HostingUnits\{<HostingUnit Uid>}

Required?	true
Default Value	
Accept Pipeline Input?	false

**-StoragePath**<String>

Specifies the path to the storage that will be modified. The path must be in one of the following formats: <drive>:\Connections\<ConnectionName>\MyStorage.storage or <drive>:\Connections\{<Connection Uid>\MyStorage.storage or <drive>:\HostingUnits\<HostingUnitName>\MyStorage.storage or <drive>:\HostingUnits\{<hostingUnit Uid>\MyStorage.storage

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-StorageType**<StorageType>

Specifies the type of storage in StoragePath. Supported storage types are: OSStorage PersonalDiskStorage

Required?	false
Default Value	OSStorage
Accept Pipeline Input?	false

**-Superseded**<Boolean>

Specifies that this storage has been superseded and must not be used for future provisioning operations. Existing virtual machines using this storage will continue to function.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects. This can be a host name or an IP address.

Required?	false
Default Value	LocalHost. When a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

Input Type

System.string You can pipe a string that contains a path to Add-HypHostingUnitStorage (StoragePath parameter).

Return Values

Citrix.XDPowerShell.HostingUnit

Add-HypHostingUnitStorage returns an object containing the new definition of the hosting unit.

HostingUnitUid <Guid>

Specifies the unique identifier for the hosting unit.

HostingUnitName <string>

Specifies the name of the hosting unit.

HypervisorConnection <Citrix.XDPowerShell.HypervisorConnection>

Specifies the connection that the hosting unit uses to access a hypervisor.

RootId <string>

Identifies, to the hypervisor, the root of the hosting unit.

RootPath <string>

The hosting unit provider path that represents the root of the hosting unit.

Storage <Citrix.XDPowerShell.Storage[]>

The list of storage items that the hosting unit can use.

PersonalVDiskStorage <Citrix.XDPowerShell.Storage[]>

The list of storage items that the hosting unit can use for storing personal data.

VMTaggingEnabled <Boolean>

Specifies whether or not the metadata in the hypervisor can be used to store information about the XenDesktop Machine Creation Service.

NetworkId <string>

The hypervisor's internal identifier that represents the network specified for the hosting unit.

NetworkPath <string>

The hosting unit provider path to the network specified for the hosting unit.

Metadata <Citrix.XDPowerShell.Metadata[]>

A list of key value pairs that can store additional information about the hosting unit.



## Notes

The storage path must be valid for the hosting unit. The rules that are applied are as follows: XenServer (HypervisorConnection Type = XenServer)

NA

VMWare vSphere/ESX (HypervisorConnection Type = vCenter)

The storage path must be directly contained in the root path item of the hosting unit.

Microsoft SCVMM/Hyper-v (HypervisorConnection Type = SCVMM)

Only one storage entry for these connection types is valid, and it must reference an SMB share. Additionally, if a Hyper-V failover cluster is used the SMB share must be the top-level mount point of the cluster shared volume on one of the servers in the cluster (i.e. C:\ClusterStorage).

In the case of failure, the following errors can be produced.

### Error Codes

-----

#### HostingUnitsPathInvalid

The path provided is not to an item in a subdirectory of a hosting unit item.

#### HostingUnitStoragePathInvalid

The specified path is invalid.

#### HostingUnitStoragePathInvalid

The storage path cannot be found or is invalid. See notes above about validity.

#### HostingUnitStorageDuplicateObjectExists

The specified storage path is already part of the hosting unit.

#### HypervisorInMaintenanceMode

The hypervisor for the connection is in maintenance mode.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation. Communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used, or examine the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Set-HypHostingUnitStorage -LiteralPath XDHyp:\HostingUnits\MyHostingUnit -StoragePath 'XDHyp:\HostingUnits\MyHostingUnits\newStorage.storage' -Superseded $true
```

```
HostingUnitUid : bcd3d649-86d1-4aa8-8ec2-d322b6a2c457
HostingUnitName : MyHostingUnit
HypervisorConnection : MyConnection
RootPath : /
RootId :
NetworkPath : /Network 0.network
```

```
NetworkId : ab47080b-ca15-771a-c8dc-6ad9650158f1
Storage : {/Local storage.storage, /newStorage.storage}
PersonalvDiskStorage : {/newStorage.storage}
VMTaggingEnabled : True
Metadata : {}
```

The command updates a storage location called "newStorage.storage" associated with the hosting unit called "MyHostingUnit".

# Set-HypHypervisorConnectionMetadata

May 10, 2016  
Adds or updates metadata on the given HypervisorConnection.

## Syntax

Set-HypHypervisorConnectionMetadata [-HypervisorConnectionUid] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-HypHypervisorConnectionMetadata [-HypervisorConnectionUid] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-HypHypervisorConnectionMetadata [-HypervisorConnectionName] <String> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-HypHypervisorConnectionMetadata [-HypervisorConnectionName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-HypHypervisorConnectionMetadata [-InputObject] <HypervisorConnection[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-HypHypervisorConnectionMetadata [-InputObject] <HypervisorConnection[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Provides the ability for additional custom data to be stored against given HypervisorConnection objects.

## Related topics

- [Add-HypHypervisorConnectionMetadata](#)
- [Remove-HypHypervisorConnectionMetadata](#)

## Parameters

**-HypervisorConnectionUid**<Guid>

Id of the HypervisorConnection

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-HypervisorConnectionName**<String>

Name of the HypervisorConnection

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<HypervisorConnection[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name<String>**

Specifies the property name of the metadata to be added. The property must be unique for the HypervisorConnection specified. The property cannot contain any of the following characters \/:#. \*?=<> | [] {} ""

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Value<String>**

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress<String>**

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-HypHypervisorConnectionMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

##### InvalidParameterCombination

The cmdlet parameters are inconsistent.

##### UnknownObject

One of the specified objects was not found.

##### DatabaseError

An error occurred in the service while attempting a database operation.

##### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

##### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

##### PermissionDenied

You do not have permission to execute this command.

##### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

##### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Set-HypHypervisorConnectionMetadata -HypervisorConnectionUid 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value

Key	Value
---	----
property	value

Add metadata with a name of 'property' and a value of 'value' to the HypervisorConnection with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Set-HypServiceMetadata

May 10, 2016

Adds or updates metadata on the given Service.

## Syntax

```
Set-HypServiceMetadata [-ServiceHostId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-HypServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-HypServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-HypServiceMetadata [-InputObject] <Service[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Allows you to store additional custom data against given Service objects.

## Related topics

[Remove-HypServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the Service specified. The property cannot contain any of the following characters \/:#.\*?=<>|[]()''

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false



Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-HypServiceMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

InvalidParameterCombination

The cmdlet parameters are inconsistent.

UnknownObject

One of the specified objects was not found.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Set-HypServiceMetadata -ServiceHostId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
---	----
property	value

Add metadata with a name of 'property' and a value of 'value' to the Service with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Set-HypVolumeServiceConfiguration

May 10, 2016  
Applies a change to one of the VolumeServiceConfiguration instances in the site.

### Syntax

Set-HypVolumeServiceConfiguration -VolumeWorkerPackageUri <String> -ConnectionType <String> -VolumeServiceConfigurationName <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

Set-HypVolumeServiceConfiguration -RegionName <String> -BaseLinuxTemplateId <String> -ConnectionType <String> -VolumeServiceConfigurationName <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

### Detailed Description

Volume service configurations are used to control how cloud-based host connections behave when provisioning machines. They contain two pieces of information. The first is a per-region specification of the cloud template that provides the standard Linux operating system for the cloud. The second is the specification of a URL from which the Citrix Volume Worker software can be installed (not all cloud connections make use of this URL, but they all make use of the template map).  
  
Each cmdlet invocation can be used to either change the volume worker URL, or to modify (or add) an entry in the Linux template map. These two operations are supported by parameter sets. To change both properties, you must invoke the cmdlet twice.

### Related topics

[Get-HypVolumeServiceConfiguration](#)

### Parameters

**-ConnectionType**<String>

Specifies the cloud connection type, such as "AWS" or "CloudPlatform".

Required?	true
Default Value	
Accept Pipeline Input?	false

**-VolumeServiceConfigurationName**<String>

Specifies the name of the configuration you want to modify. This parameter is used alongside the ConnectionType to specify a single configuration set unambiguously. There can be only one named configuration per connection type. In a newly-configured site, there will be exactly one configuration set called "SiteDefault" for each of CloudPlatform and AWS.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-VolumeWorkerPackageUri**<String>

Specifies a (new) URI for the volume worker package.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-RegionName**<String>

Specifies the cloud region in which the Linux template resides. This parameter is used only when passing the BaseLinuxTemplateId parameter. The format of the string is cloud-specific. For example, for an AWS-based cloud, it would be a region identifier such as "us-east-1".

Required?	true
Default Value	
Accept Pipeline Input?	false

**-BaseLinuxTemplateId**<String>

Specifies a change to the standard Linux template that should be used in the cloud. When passing this parameter, you must also specify the RegionName parameter to indicate the region in which

this template resides. The format of the string is cloud-specific. For example, for an AWS-based cloud, it would be an AMI identifier such as "ami-1234abcd".

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

Return Values

Citrix.Host.Sdk.VolumeServiceConfiguration

This cmdlet returns the updated VolumeServiceConfiguration object.

Examples

----- **EXAMPLE 1** -----

C:\PS>Set-HypVolumeServiceConfiguration -VolumeServiceConfigurationName SiteDefault -ConnectionType CloudPlatform -RegionName Region1 -BaseLinuxTemplateId 9883ba7a-74bf-4002-9b12-1  
This command specifies a Linux template that should be used by default for CloudPlatform connections.

----- **EXAMPLE 2** -----

C:\PS>Set-HypVolumeServiceConfiguration -VolumeServiceConfigurationName SiteDefault -ConnectionType CloudPlatform -VolumeWorkerPackageUri "http://cloudadmin.net/citrix/volumeworker/versic  
This command specifies a worker package download URL that should be used by default for CloudPlatform connections.

# Start-HypVM

May 10, 2016  
Starts a VM.

## Syntax

Start-HypVM [-LiteralPath <String>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

This command provides a mechanism to start a VM.

## Related topics

## Parameters

### **-LiteralPath**<String>

Specifies the path within a hosting unit provider to the virtual machine item to start. The path specified must be in one of the following formats: <drive>:\Connections\<Connection Name>\<Item Path of VM object> or <drive>:\Connections\{<connection Uid>\<Item Path of VM object>} or <drive>:\HostingUnits\<HostingUnit Name>\<Item Path of VM object> or <drive>:\HostingUnits\{<hostingUnit Uid>\<Item Path of VM object>}

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

System.string You can pipe a string that contains a path.

## Return Values

System.void.

## Notes

In the case of failure, the following errors can result.

### Error Codes

-----

#### InputHypervisorItemPathInvalid

The path provided is not to an item in a sub-directory of a connection item or a hosting unit item.

#### InvalidHypervisorItemPath

No item exists with the specified path.

#### InvalidHypervisorItem

The item specified by the path exists, but is not a VM Item.

#### HypervisorInMaintenanceMode

The hypervisor is in maintenance mode.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### CommunicationError

An error occurred while communicating with the service.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

C:\PS>Start-HypVM -LiteralPath XDHyp:\Connections\MyConnection\MyVm.vm

This command starts a VM called 'MyVm.vm' within a hypervisor connection called 'MyConnection'.

# Stop-HypVM

May 10, 2016

Stops a VM by issuing a Shutdown request

## Syntax

Stop-HypVM [-LiteralPath <String>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Use this command to change the power state of a VM from running to stopped.

## Related topics

## Parameters

### **-LiteralPath**<String>

Specifies the path within a hosting unit provider to the virtual machine item to stop. The path specified must be in one of the following formats: <drive>:\Connections\<Connection Name>\<Item Path of VM object> or <drive>:\Connections\{<connection Uid>\<Item Path of VM object>} or <drive>:\HostingUnits\<HostingUnit Name>\<Item Path of VM object> or <drive>:\HostingUnits\{<hostingUnit Uid>\<Item Path of VM object>}

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in will connect to. This can be provided as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Input Type

System.string You can pipe a string that contains a path.

## Return Values

System.void.

## Notes

In the case of failure, the following errors can result.

### Error Codes

-----

#### InputHypervisorItemPathInvalid

The path provided is not to an item in a sub-directory of a connection item or a hosting unit item.

#### InvalidHypervisorItemPath

No item exists with the specified path.

#### InvalidHypervisorItem

The item specified by the path exists, but is not a VM Item.

#### HypervisorInMaintenanceMode

The hypervisor is in maintenance mode.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### CommunicationError

An error occurred while communicating with the service.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

C:\PS>Stop-HypVM -LiteralPath XDHyp:\Connections\MyConnection\MyVm.vm

This command stops a VM called 'MyVm.vm' within a hypervisor connection called 'MyConnection'.



# Test-HypDBConnection

May 10, 2016

Tests a database connection for the Host Service.

## Syntax

```
Test-HypDBConnection [-DBConnection] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Tests a connection to the database in which the Host Service can store its state. The service will attempt to connect to the database without affecting the current connection to the database.

You do not have to clear the connection to use this command.

## Related topics

[Get-HypServiceStatus](#)

[Get-HypDBConnection](#)

[Set-HypDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be tested by the Host Service.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
-----------	-------

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Test-HypDBConnection command returns an object containing the status of the Host Service if the connection string of the specified data store were to be set to the string being tested, together with extra diagnostics information for the specified connection string.

DBRejectedConnection

The database rejected the logon attempt from the Host Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the Host Service schema has not been added to the database.

DBNotFound

The specified database could not be located with the configured connection string.

DBNewerVersionThanService

The version of the Host Service currently in use is incompatible with the version of the Host Service schema on the database. Upgrade the Host Service to a more recent version.

DBOlderVersionThanService

The version of the Host Service schema on the database is incompatible with the version of the Host Service currently in use. Upgrade the database schema to a more recent version.

DBVersionChangeInProgress

A database schema upgrade is currently in progress.

OK

The Set-HypDBConnection command would succeed if it were executed with the supplied connection string.

Failed

The Host Service has failed.

Unknown

The status of the Host Service cannot be determined.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

InvalidDBConnectionString

The database connection string has an invalid format.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Test-HypDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Tests a database connection string for the Host Service.

##### ----- EXAMPLE 2 -----

```
c:\PS>Test-HypDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Tests an invalid database connection string for the Host Service.

# Test-HypHostingUnitNameAvailable

May 10, 2016

Checks to ensure that the proposed name for a hosting unit is unused.

## Syntax

Test-HypHostingUnitNameAvailable -HostingUnitName <String[]> [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Use this command to check that the proposed name for a hosting unit is unused. This check is done without regard for scoping of the existing hosting unit, so the names of inaccessible hosting units are also checked.

## Related topics

New-Item

Rename-Item

## Parameters

**-HostingUnitName**<String[]>

The name or names of the hosting units(s) to be tested.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

object[]

An array of PSObjects which pair the name and availability of the name

## Notes

In the case of failure, the following errors can result.

### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

Test-HypHostingUnitNameAvailable -HostingUnitName \$NewHostingUnitName

This tests whether the value of \$NewHostingUnitName is unique or not, and can be used to create a new hosting unit or rename an existing one without failing. True is returned if the name is unique.

# Test-HypHypervisorConnectionNameAvailable

May 10, 2016

Checks to ensure that the proposed name for a hypervisor connection is unused.

## Syntax

```
Test-HypHypervisorConnectionNameAvailable -HypervisorConnectionName <String[]> [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Use this command to check that the proposed name for a hypervisor connection is unused. This check is done without regard for scoping of the existing hypervisor connection, so the names of inaccessible hypervisor connection are also checked.

## Related topics

New-Item

Rename-Item

## Parameters

**-HypervisorConnectionName**<String[]>

The name or names of the hypervisor connection(s) to be tested.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

object[]

An array of PSObjects which pair the name and availability of the name

## Notes

In the case of failure, the following errors can result.

### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

Test-HypHypervisorConnectionNameAvailable -HypervisorConnectionName \$NewConnectionName

This tests whether the value of \$NewConnectionName is unique or not, and can be used to create a new hypervisor connection or rename an existing one. True is returned if the name is unique.

# Update-HypHypervisorConnection

May 10, 2016

Requests the host service to update the connection properties that depend on the version of hypervisor in use.

## Syntax

```
Update-HypHypervisorConnection [-LiteralPath] <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Use this command to update the version-specific properties of a hypervisor connection, after an upgrade to the hypervisor system which may provide new capabilities.

## Related topics

## Parameters

### **-LiteralPath**<String>

Specifies the path within a Host Service provider to the hypervisor connection item to be updated. The path specified must be in one of the following formats; <drive>:\Connections\<HypervisorConnectionName> or <drive>:\Connections\{HypervisorConnection Uid}

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in will connect to. This can be provided as a



host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value will become the default.
Accept Pipeline Input?	false

## Notes

In the case of failure, the following errors can result.

### Error Codes

-----

#### InvalidPath

The path provided is not in the required format.

#### HypervisorInMaintenanceMode

The hypervisor is in maintenance mode and cannot be updated.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Update-HypHypervisorConnection -LiteralPath xdhyp:\connections\Connection1
```

This command requests that the properties of Connection1 be updated to match the current capabilities of the hypervisor.

# Citrix.MachineCreation.Admin.V2

May 10, 2016

## Overview

Name	Description
<a href="#">ProvMachineCreationSnapin</a>	The Machine Creation Service PowerShell snap-in provides administrative
<a href="#">Prov Filtering</a>	Describes the common filtering options for XenDesktop cmdlets.
<a href="#">providers</a>	Describes how Windows PowerShell providers provide access to data and

## Cmdlets

Name	Description
<a href="#">Add-ProvSchemeControllerAddress</a>	Adds a list of host names (as DNS addresses) to a provisioning scheme.
<a href="#">Add-ProvSchemeMetadata</a>	Adds metadata on the given ProvisioningScheme.
<a href="#">Add-ProvSchemeScope</a>	Add the specified ProvisioningScheme(s) to the given scope(s).
<a href="#">Add-ProvTaskMetadata</a>	Adds metadata on the given Task.
<a href="#">Get-ProvDBConnection</a>	Gets the database string for the specified data store used by the MachineCreation Service.
<a href="#">Get-ProvDBSchema</a>	Gets a script that creates the MachineCreation Service database schema for the specified data store.
<a href="#">Get-ProvDBVersionChangeScript</a>	Gets a script that updates the MachineCreation Service database schema.
<a href="#">Get-ProvInstalledDBVersion</a>	Gets a list of all available database schema versions for the MachineCreation Service.
<a href="#">Get-ProvObjectReference</a>	Returns the number of local objects holding references to objects from other services.
<a href="#">Get-ProvScheme</a>	Gets the list of provisioning schemes.
<a href="#">Get-ProvSchemeMasterVMImageHistory</a>	Gets the list of master VM snapshots that have been used to provide hard disks to provisioning schemes.

Name	Description
<a href="#">Get-ProvScopedObject</a>	Gets details of the scoped objects for the MachineCreation Service.
<a href="#">Get-ProvService</a>	Gets the service record entries for the MachineCreation Service.
<a href="#">Get-ProvServiceAddedCapability</a>	Gets any added capabilities for the MachineCreation Service on the controller.
<a href="#">Get-ProvServiceConfigurationData</a>	Gets configuration data for the service.
<a href="#">Get-ProvServiceInstance</a>	Gets the service instance entries for the MachineCreation Service.
<a href="#">Get-ProvServiceStatus</a>	Gets the current status of the MachineCreation Service on the controller.
<a href="#">Get-ProvTask</a>	Gets the task history for the MachineCreation Service.
<a href="#">Get-ProvVM</a>	Gets the VMs that were created using Citrix XenDesktop Machine Creation Services.
<a href="#">Lock-ProvVM</a>	Locks a VM.
<a href="#">New-ProvScheme</a>	Creates a new provisioning scheme.
<a href="#">New-ProvVM</a>	Creates a new virtual machine.
<a href="#">Publish-ProvMasterVmlImage</a>	Update the master image associated with the provisioning scheme.
<a href="#">Remove-ProvScheme</a>	Removes a provisioning scheme
<a href="#">Remove-ProvSchemeControllerAddress</a>	Removes metadata from a provisioning scheme.
<a href="#">Remove-ProvSchemeMasterVmlImageHistory</a>	Removes the history of provisioning scheme master image VMs.
<a href="#">Remove-ProvSchemeMetadata</a>	Removes metadata from the given ProvisioningScheme.
<a href="#">Remove-ProvSchemeScope</a>	Remove the specified ProvisioningScheme(s) from the given scope(s).
<a href="#">Remove-ProvServiceConfigurationData</a>	Removes configuration data from the service.
<a href="#">Remove-ProvServiceMetadata</a>	Removes metadata from the given Service.

<b>Name</b>	<b>Description</b>
<a href="#">Remove-ProvTask</a>	Removes from the database completed tasks for the MachineCreation Service.
<a href="#">Remove-ProvTaskMetadata</a>	Removes metadata from the given Task.
<a href="#">Remove-ProvVM</a>	Removes virtual machines.
<a href="#">Rename-ProvScheme</a>	Renames a provisioning scheme.
<a href="#">Reset-ProvServiceGroupMembership</a>	Reloads the access permissions and configuration service locations for the MachineCreation Service.
<a href="#">Set-ProvDBConnection</a>	Configures a database connection for the MachineCreation Service.
<a href="#">Set-ProvScheme</a>	Changes the parameter values for a provisioning scheme.
<a href="#">Set-ProvSchemeMetadata</a>	Adds or updates metadata on the given ProvisioningScheme.
<a href="#">Set-ProvServiceConfigurationData</a>	Sets the value for the given key in the service configuration data.
<a href="#">Set-ProvServiceMetadata</a>	Adds or updates metadata on the given Service.
<a href="#">Set-ProvTaskMetadata</a>	Adds or updates metadata on the given Task.
<a href="#">Stop-ProvTask</a>	Stops currently running MachineCreation Service tasks.
<a href="#">Switch-ProvTask</a>	Moves all MachineCreation Service tasks from the current execution host to another.
<a href="#">Test-ProvDBConnection</a>	Tests a database connection for the MachineCreation Service.
<a href="#">Test-ProvSchemeNameAvailable</a>	Checks to ensure that the proposed name for a provisioning scheme is unused.
<a href="#">Unlock-ProvScheme</a>	Unlocks a Provisioning Scheme.
<a href="#">Unlock-ProvVM</a>	Unlocks a VM.

# about\_ProvMachineCreationSnapin

May 10, 2016

## TOPIC

about\_ProvMachineCreationSnapin

## SHORT DESCRIPTION

The Machine Creation Service PowerShell snap-in provides administrative functions for the Machine Creation Service.

## COMMAND PREFIX

All commands in this snap-in have 'Prov' in their name.

## LONG DESCRIPTION

The Machine Creation Service PowerShell snap-in enables both local and remote administration of the Machine Creation Service. It provides facilities to create virtual machines and manage the associated disk images.

The snap-in provides two main entities:

### Provisioning Scheme

Specifies details of new virtual machines created by the Machine Creation Service. Provisioning schemes define the following information.

#### Hosting Unit

Provides details of the hypervisor and storage on which new virtual machines will be created. Stored and maintained by the Host Service and PowerShell snap-in.

#### Identity Pool

Lists the Active Directory computer accounts available for use by new virtual machines. Stored and maintained by the Active Directory Identity Service and PowerShell snap-in.

#### Master Image

Specifies the disk image that will be used for new virtual machines. Accessed through the hosting provider in the Host Service snap-in.

### Provisioned VM

Defines the virtual machines created by the Machine Creation Services. These virtual machines are associated with the provisioning scheme from

which they were created.

The processes of creating provisioning schemes and new virtual machines can take a significant amount of time to complete. For this reason, these long-running tasks can be run asynchronously so that other commands are accessible while the processes are running. Note, however, that only one long-running task can operate on a provisioning scheme at any one time. The processes are monitored using the `Get-ProvTask` command. For more information, see the help for `Get-ProvTask`.

# about\_Prov\_Filtering

May 10, 2016

## TOPIC

XenDesktop - Advanced Dataset Filtering

## SHORT DESCRIPTION

Describes the common filtering options for XenDesktop cmdlets.

## LONG DESCRIPTION

Some cmdlets operate on large quantities of data and, to reduce the overhead of sending all of that data over the network, many of the Get- cmdlets support server-side filtering of the results.

The conventional way of filtering results in PowerShell is to pipeline them into Where-Object, Select-Object, and Sort-Object, for example:

```
Get-<Noun> | Where { $_.Size = 'Small' } | Sort 'Date' | Select -First 10
```

However, for most XenDesktop cmdlets the data is stored remotely and it would be slow and inefficient to retrieve large amounts of data over the network and then discard most of it. Instead, many of the Get- cmdlets provide filtering parameters that allow results to be processed on the server, returning only the required results.

You can filter results by most object properties using parameters derived from the property name. You can also sort results or limit them to a specified number of records:

```
Get-<Noun> -Size 'Small' -SortBy 'Date' -MaxRecordCount 10
```

You can express more complex filter conditions using a syntax and set of operators very similar to those used by PowerShell expressions.

Those cmdlets that support filtering have the following common parameters:

-MaxRecordCount <int>

Specifies the maximum number of results to return.  
For example, to return only the first nine results use:

```
Get-<Noun> -MaxRecordCount 9
```

If not specified, only the first 250 records are returned, and if more are available, a warning is produced:



WARNING: Only first 250 records returned. Use -MaxRecordCount to retrieve more.

You can suppress this warning by using -WarningAction or by specifying a value for -MaxRecordCount.

To retrieve all records, specify a large number for -MaxRecordCount. As the value is an integer, you can use the following:

```
Get-<Noun> -MaxRecordCount [int]::MaxValue
```

-ReturnTotalRecordCount [<SwitchParameter>]

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. For example:

```
Get-<Noun> -MaxRecordCount 9 -ReturnTotalRecordCount
....
```

```
Get-<Noun> : Returned 9 of 10 items
At line:1 char:18
+ Get-<Noun> <<<< -MaxRecordCount 9 -ReturnTotalRecordCount
+ CategoryInfo : OperationStopped: (:) [Get-<Noun>], PartialDataException
+ FullyQualifiedErrorId : PartialData,Citrix.<SDKName>.SDK.Get<Noun>
```

The count can be accessed using the TotalAvailableResultCount property:

```
$count = $error[0].TotalAvailableResultCount
```

-Skip <int>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

-SortBy <string>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order, respectively. Ascending order is assumed if no prefix is present.

Sorting occurs before -MaxRecordCount and -Skip parameters are applied. For example, to sort by Name and then by Count (largest first) use:

```
-SortBy 'Name,-Count'
```

By default, sorting by an enumeration property uses the numeric value of the elements. You can specify a different sort order by qualifying the name with an ordered list of elements or their numeric values, or <null> to indicate the placement of null values.

Elements not mentioned are placed at the end in their numeric order. For example, to sort by two different enums and then by the object id:

```
-SortBy 'MyState(StateC,<null>,StateA,StateB),Another(0,3,2,1),Id'
```

-Filter <String>

This parameter lets you specify advanced filter expressions, and supports combination of conditions with -and and -or, and grouping with braces. For example:

```
Get-<Noun> -Filter 'Name -like "High*" -or (Priority -eq 1 -and Severity -ge 2)'
```

The syntax is close enough to PowerShell syntax that you can use script blocks in most cases. This can be easier to read as it reduces quoting:

```
Get-<Noun> -Filter { Count -ne $null }
```

The full -Filter syntax is provided below.

## EXAMPLES

Filtering by strings performs a case-insensitive wildcard match.

Separate parameters are combined with an implicit -and operator.

Normal PowerShell quoting rules apply, so you can use single or double quotes, and omit the quotes altogether for many strings. The order of parameters does not make any difference. The following are equivalent:

```
Get-<Noun> -Company Citrix -Product Xen*
```

```
Get-<Noun> -Company "citrix" -Product '[X]EN*'
```

```
Get-<Noun> -Product "Xen*" -Company "CITRIX"
```

```
Get-<Noun> -Filter { Company -eq 'Citrix' -and Product -like 'Xen*' }
```

See [about\\_Quoting\\_Rules](#) and [about\\_Wildcards](#) for details about PowerShell handling of quotes and wildcards.

To avoid wildcard matching or include quote characters, you can escape the wildcards using the normal PowerShell escape mechanisms (see [about\\_Escape\\_Characters](#)), or switch to a filter expression and the `-eq` operator:

```
Get-<Noun> -Company "Abc[*]" # Matches Abc*
Get-<Noun> -Company "Abc`*" # Matches Abc*
Get-<Noun> -Filter { Company -eq "Abc*" } # Matches Abc*
Get-<Noun> -Filter { Company -eq "A`"B`C" } # Matches A"B'C
```

Simple filtering by numbers, booleans, and TimeSpans perform direct equality comparisons, although if the value is nullable you can also search for null values. Here are some examples:

```
Get-<Noun> -Uid 123
Get-<Noun> -Enabled $true
Get-<Noun> -Duration 1:30:40
Get-<Noun> -NullableProperty $null
```

More comparisons are possible using advanced filtering with `-Filter`:

```
Get-<Noun> -Filter 'Capacity -ge 10gb'
Get-<Noun> -Filter 'Age -ge 20 -and Age -lt 40'
Get-<Noun> -Filter 'VolumeLevel -like "[123]"'
Get-<Noun> -Filter 'Enabled -ne $false'
Get-<Noun> -Filter 'NullableProperty -ne $null'
```

You can check boolean values without an explicit comparison operator, and you can also combine them with `-not`:

```
Get-<Noun> -Filter 'Enabled' # Equivalent to 'Enabled -eq $true'
Get-<Noun> -Filter '-not Enabled' # Equivalent to 'Enabled -eq $false'
```

See [about\\_Comparison\\_Operators](#) for an explanation of the operators, but note that only a subset of PowerShell operators are supported (`-eq`, `-ne`, `-gt`, `-ge`, `-lt`, `-le`, `-like`, `-notlike`, `-in`, `-notin`, `-contains`, `-notcontains`).

Enumeration values can either be specified using typed values or the string name of the enumeration value:

```
Get-<Noun> -Shape [Shapes]::Square
Get-<Noun> -Shape Circle
```

With filter expressions, typed values can be specified with simple variables or quoted strings. They also support enumerations with wildcards:

```
$s = [Shapes]::Square
Get-<Noun> -Filter { Shape -eq $s -or Shape -eq "Circle" }
Get-<Noun> -Filter { Shape -like 'C*' }
```

By their nature, floating point values, DateTime values, and TimeSpan values are best suited to relative comparisons rather than just equality. DateTime strings are converted using the locale and time zone of the user device, but you can use ISO8601 format strings (YYYY-MM-DDThh:mm:ss.sTZD) to avoid ambiguity. You can also use standard PowerShell syntax to create these values:

```
Get-<Noun> -Filter { StartTime -ge "2010-08-23T12:30:00.0Z" }
$d = [DateTime]"2010-08-23T12:30:00.0Z"
Get-<Noun> -Filter { StartTime -ge $d }
$d = (Get-Date).AddDays(-1)
Get-<Noun> -Filter { StartTime -ge $d }
```

Relative times are quite common and, when using filter expressions, you can also specify DateTime values using a relative format:

```
Get-<Noun> -Filter { StartTime -ge '-2' } # Two days ago
Get-<Noun> -Filter { StartTime -ge '-1:30' } # Hour and a half ago
Get-<Noun> -Filter { StartTime -ge '-0:0:30' } # 30 seconds ago
```

## ARRAY PROPERTIES

When filtering against list or array properties, simple parameters perform a case-insensitive wildcard match against each of the members. With filter expressions, you can use the -contains and -notcontains operators. Unlike PowerShell, these perform wildcard matching on strings.

Note that for array properties the naming convention is for the returned property to be plural, but the parameter used to search for any match is singular. The following are equivalent (assuming Users is an array property):

```
Get-<Noun> -User Fred*
Get-<Noun> -Filter { User -like "Fred*" }
Get-<Noun> -Filter { Users -contains "Fred*" }
```

You can also use the singular form with -Filter to search using other operators:

```
Match if any user in the list is called "Frederick"
Get-<Noun> -Filter { User -eq "Frederick" }
Match if any user in the list has a name alphabetically below 'F'
Get-<Noun> -Filter { User -lt 'F' }
```

## COMPLEX EXPRESSIONS

When matching against multiple values, you can use a sequence of comparisons joined with -or operators, or you can use -in and -notin:

```
Get-<Noun> -Filter { Shape -eq 'Circle' -or Shape -eq 'Square' }
$shapes = 'Circle','Square'
Get-<Noun> -Filter { Shape -in $shapes }
$sides = 1..4
Get-<Noun> -Filter { Sides -notin $sides }
```

Braces can be used to group complex expressions, and override the default left-to-right evaluation of -and and -or. You can also use -not to invert the sense of any sub-expression:

```
Get-<Noun> -Filter { Size -gt 4 -or (Color -eq 'Blue' -and Shape -eq 'Circle') }
Get-<Noun> -Filter { Sides -lt 5 -and -not (Color -eq 'Blue' -and Shape -eq 'Circle') }
```

## PAGING

The simplest way to page through data is to use the -Skip and -MaxRecordCount parameters. So, to read the first three pages of data with 10 records per page, use:

```
Get-<Noun> -Skip 0 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 10 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 20 -MaxRecordCount 10 <other filtering criteria>
```

You must include the same filtering criteria on each call, and ensure that the data is sorted consistently.

The above approach is often acceptable, but as each call performs an independent query, data changes can result in records being skipped or appearing twice. One approach to improve this is to sort by a unique id field and then start the search for the next page at the unique id after the last unique id of the previous page. For example:

```
Get the first page
Get-<Noun> -MaxRecordCount 10 -SortBy SerialNumber

SerialNumber ...
----- ---
A120004
A120007
... 7 other records ...
A120900

Get the next page
Get-<Noun> -MaxRecordCount 10 -Filter { FirstName -gt 'A120900' }

SerialNumber ...
```

----- ---  
A120901  
B220000  
...

## FILTER SYNTAX DEFINITION

<Filter> ::= <ScriptBlock> | <ComponentList>

<ScriptBlock> ::= "{" <ComponentList> "}"

<ComponentList> ::= <Component> <AndOrOperator> <ComponentList> |

<Component>

<Component> ::= <NotOperator> <Factor> |

<Factor>

<Factor> ::= "(" <ComponentList> ")" |

<PropertyName> <ComparisonOperator> <Value> |  
<PropertyName>

<AndOrOperator> ::= "-and" | "-or"

<NotOperator> ::= "-not" | "!"

<ComparisonOperator>

::= "-eq" | "-ne" | "-le" | "-ge" | "-lt" | "-gt" |  
"-like" | "-notlike" | "-contains" | "-notcontains" |  
"-in" | "-notin"

<PropertyName> ::= <simple name of property>

<Value> ::= <string literal> | <numeric literal> |

<scalar variable> | <array variable> |  
"\$null" | "\$true" | "\$false"

Numeric literals support decimal and hexadecimal literals, with optional multiplier suffixes (kb, mb, gb, tb, pb).

Dates and times can be specified as string literals. The current culture determines what formats are accepted. To avoid any ambiguity, use strings formatted to the ISO8601 standard. If not specified, the current time zone is used.

Relative date-time string literals are also supported, using a minus sign followed by a TimeSpan. For example, "-1:30" means 1 hour and 30 minutes ago.

# about\_providers

May 10, 2016

## TOPIC

about\_Providers

## SHORT DESCRIPTION

Describes how Windows PowerShell providers provide access to data and components that would not otherwise be easily accessible at the command line. The data is presented in a consistent format that resembles a file system drive.

## LONG DESCRIPTION

Windows PowerShell providers are Microsoft .NET Framework-based programs that make the data in a specialized data store available in Windows PowerShell so that you can view and manage it.

The data that a provider exposes appears in a drive, and you access the data in a path like you would on a hard disk drive. You can use any of the built-in cmdlets that the provider supports to manage the data in the provider drive. And, you can use custom cmdlets that are designed especially for the data.

The providers can also add dynamic parameters to the built-in cmdlets. These are parameters that are available only when you use the cmdlet with the provider data.

## BUILT-IN PROVIDERS

Windows PowerShell includes a set of built-in providers that you can use to access the different types of data stores.

Provider Drive Data store ----- Alias Alias: Windows PowerShell aliases

Certificate Cert: x509 certificates for digital signatures

Environment Env: Windows environment variables

FileSystem \* File system drives, directories, and files

Function Function: Windows PowerShell functions

Registry HKLM:, HKCU Windows registry

Variable Variable: Windows PowerShell variables

WS-Management WSMAN WS-Management configuration information

\* The FileSystem drives vary on each system.

You can also create your own Windows PowerShell providers, and you can install providers that others develop. To list the providers that are available in your session, type:

```
get-psprovider
```



## INSTALLING AND REMOVING PROVIDERS

Windows PowerShell providers are delivered to you in Windows PowerShell snap-ins, which are .NET Framework-based programs that are compiled into .dll files. The snap-ins can include providers and cmdlets.

Before you use the provider features, you have to install the snap-in and then add it to your Windows PowerShell session. For more information, see `about_PsSnapins`.

You cannot uninstall a provider, although you can remove the Windows PowerShell snap-in for the provider from the current session. If you do, you will remove all the contents of the snap-in, including its cmdlets.

To remove a provider from the current session, use the `Remove-PsSnapin` cmdlet. This cmdlet does not uninstall the provider, but it makes the provider unavailable in the session.

You can also use the `Remove-PsDrive` cmdlet to remove any drive from the current session. This data on the drive is not affected, but the drive is no longer available in that session.

## VIEWING PROVIDERS

To view the Windows PowerShell providers on your computer, type:

```
get-psprovider
```

The output lists the built-in providers and the providers that you added to the session.

## THE PROVIDER CMDLETS

The following cmdlets are designed to work with the data exposed by any provider. You can use the same cmdlets in the same way to manage the different types of data that providers expose. After you learn to manage the data of one provider, you can use the same procedures with the data from any provider.

For example, the `New-Item` cmdlet creates a new item. In the C: drive that is supported by the `FileSystem` provider, you can use `New-Item` to create a new file or folder. In the drives that are supported by the `Registry` provider, you can use `New-Item` to create a new registry key. In the `Alias:` drive, you can use `New-Item` to create a new alias.

For detailed information about any of the following cmdlets, type:

```
get-help <cmdlet-name> -detailed
```

## CHILDITEM CMDLETS

```
Get-ChildItem
```

## CONTENT CMDLETS

```
Add-Content
```

Clear-Content  
Get-Content  
Set-Content

## ITEM CMDLETS

Clear-Item  
Copy-Item  
Get-Item  
Invoke-Item  
Move-Item  
New-Item  
Remove-Item  
Rename-Item  
Set-Item

## ITEMPROPERTY CMDLETS

Clear-ItemProperty  
Copy-ItemProperty  
Get-ItemProperty  
Move-ItemProperty  
New-ItemProperty  
Remove-ItemProperty  
Rename-ItemProperty  
Set-ItemProperty

## LOCATION CMDLETS

Get-Location  
Pop-Location  
Push-Location  
Set-Location

## PATH CMDLETS

Join-Path  
Convert-Path  
Split-Path  
Resolve-Path  
Test-Path

## PSDRIVE CMDLETS

- Get-PSDrive
- New-PSDrive
- Remove-PSDrive

## PSPROVIDER CMDLETS

- Get-PSPProvider

## VIEWING PROVIDER DATA

The primary benefit of a provider is that it exposes its data in a familiar and consistent way. The model for data presentation is a file system drive.

To use data that the provider exposes, you view it, move through it, and change it as though it were data on a hard drive. Therefore, the most important information about a provider is the name of the drive that it supports.

The drive is listed in the default display of the Get-PsProvider cmdlet, but you can get information about the provider drive by using the Get-PsDrive cmdlet. For example, to get all the properties of the Function: drive, type:

```
get-psdrive Function | format-list *
```

You can view and move through the data in a provider drive just as you would on a file system drive.

To view the contents of a provider drive, use the Get-Item or Get-ChildItem cmdlets. Type the drive name followed by a colon (:). For example, to view the contents of the Alias: drive, type:

```
get-item alias:
```

You can view and manage the data in any drive from another drive by including the drive name in the path. For example, to view the HKLM\Software registry key in the HKLM: drive from another drive, type:

```
get-childitem hklm:\software
```

To open the drive, use the Set-Location cmdlet. Remember the colon when you specify the drive path. For example, to change your location to the root directory of the Cert: drive, type:

```
set-location cert:
```

Then, to view the contents of the Cert: drive, type:

```
get-childitem
```

## MOVING THROUGH HIERARCHICAL DATA

You can move through a provider drive just as you would a hard disk drive. If the data is arranged in a hierarchy of items within items, use a backslash (\) to indicate a child item. Use the following format:

```
drive:\location\child-location\...
```

For example, to change your location to the HKLM\Software registry key, type a Set-Location command, such as:

```
set-location hklm:\software
```

You can also use relative references to locations. A dot (.) represents the current location. For example, if you are in the HKLM:\Software\Microsoft registry key, and you want to list the registry subkeys in the HKLM:\Software\Microsoft\PowerShell key, type the following command:

```
get-childitem .powershell
```

## FINDING DYNAMIC PARAMETERS

Dynamic parameters are cmdlet parameters that are added to a cmdlet by a provider. These parameters are available only when the cmdlet is used with the provider that added them.

For example, the Cert: drive adds the CodeSigningCert parameter to the Get-Item and Get-ChildItem cmdlets. You can use this parameter only when you use Get-Item or Get-ChildItem in the Cert: drive.

For a list of the dynamic parameters that a provider supports, see the Help file for the provider. Type:

```
get-help <provider-name>
```

For example:

```
get-help certificate
```

## LEARNING ABOUT PROVIDERS

Although all provider data appears in drives, and you use the same methods to move through them, the similarity stops there. The data stores that the provider exposes can be as varied as Active Directory locations and Microsoft Exchange Server mailboxes.

For information about individual Windows PowerShell providers, type:

```
get-help <ProviderName>
```

For example:

```
get-help registry
```

For a list of Help topics about the providers, type:

```
get-help * -category provider
```

SEE ALSO

[about\\_Locations](#)

[about\\_Path\\_Syntax](#)

# Add-ProvSchemeControllerAddress

May 10, 2016  
Adds a list of host names (as DNS addresses) to a provisioning scheme.

Syntax

```
Add-ProvSchemeControllerAddress [-ProvisioningSchemeName] <String> [-ControllerAddress] <String[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Add-ProvSchemeControllerAddress -ProvisioningSchemeUid <Guid> [-ControllerAddress] <String[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

Detailed Description

Provides the ability to associate controller hosts (and hence implicitly a set of brokers) with a specific provisioning scheme. This optional data is passed to the machines created by the Machine Creation Services, where it is used to associate the newly created machine with a broker. The list is returned along with the provisioning scheme that it is assigned to.

- Related topics
- [Get-ProvScheme](#)
  - [Remove-ProvSchemeControllerAddress](#)

Parameters

**-ProvisioningSchemeName**<String>

The name for the provisioning scheme that the list of addresses is to be added to.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ProvisioningSchemeUid**<Guid>

The unique identifier for the provisioning scheme that the list of addresses is to be added to.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-ControllerAddress**<String[]>

Specifies the array of DNS names to be added to the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.MachineCreation.Sdk.ProvisioningScheme You can pipe an object containing a parameter called 'ProvisioningSchemeName' to Add-ProvSchemeControllerAddress.

#### Return Values

Citrix.MachineCreation.Sdk.ProvisioningScheme

Add-ProvSchemeControllerAddress returns the updated ProvisioningScheme object containing the union of the old and new controller address lists.

ProvisioningSchemeUid <Guid>

The unique identifier for the provisioning scheme.

ProvisioningSchemeName <string>

The name of the provisioning scheme.

CpuCount <int>

The number of processors that VMs will be created with when using this scheme.

MemoryMB <int>

The maximum amount of memory that VMs will be created with when using this scheme.

MasterImageVM <string>

The path within the hosting unit provider to the VM or snapshot of which the scheme is currently using a copy.

MasterImageVMDate <DateTime>

The date and time that the copy of the VM image was made for the scheme.

IdentityPoolUid <Guid>

The unique identifier of the identity pool (from the ADIdentity PowerShell snap-in) that the scheme uses.

IdentityPoolName <string>

The name of the identity pool (from the ADIdentity PowerShell snap-in) that the scheme uses.

HostingUnitUid <Guid>

The unique identifier of the hosting unit (from the Hosting Unit PowerShell snap-in) that the scheme will use.

HostingUnitName <string>

The name of the hosting unit (from the Hosting Unit PowerShell snap-in) that the scheme will use.

CleanOnBoot <Boolean>

Indicates whether or not the VMs created are to be reset to a clean state on each boot.

TaskId <Guid>

The identifier of any current task that is running for the provisioning scheme.

Metadata <Citrix.MachineCreation.Sdk.Metadata[]>

The metadata associated with this provisioning scheme.

ControllerAddress <string[]>

The DNS names of the controllers associated with this provisioning scheme for Quick Deploy purposes.

#### Notes

In the case of failure, the following errors can result.

#### Error Codes

-----

#### ProvisioningSchemeNotFound

The specified provisioning scheme could not be located.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS>Add-ProvSchemeControllerAddress -ProvisioningSchemeUid "01a4a008-8ce8-4165-ba9c-cdf15a6b0501" -ControllerAddress (ddcA.citrix.com,ddcB.citrix.com,ddcC.citrix2.com)
```

ProvisioningSchemeUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501

ProvisioningSchemeName : Scheme2

CpuCount : 1

MemoryMB : 1024

MasterImageVM : Base.vm/Base.snapshot

MasterImageVMDate : 17/05/2010 09:53:40

IdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16

IdentityPoolName : idPool1

HostingUnitUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501

HostingUnitName : HostUnit1

CleanOnBoot : True

TaskId : 00000000-0000-0000-0000-000000000000

Metadata : {}

ControllerAddress : {ddcA.citrix.com,ddcB.citrix.com,ddcC.citrix2.com}

Add a set of controllers to the provisioning scheme with the identifier "01a4a008-8ce8-4165-ba9c-cdf15a6b0501".

##### ----- EXAMPLE 2 -----

```
C:\PS>Get-ProvScheme -ProvisioningSchemeName scheme1 | Add-ProvSchemeControllerAddress -ControllerAddress (ddcA.citrix.com,ddcB.citrix.com,ddcC.citrix2.com)
```

ProvisioningSchemeUid : 7585f0de-192e-4847-a6d8-22713c3a2f42

ProvisioningSchemeName : Scheme1

CpuCount : 1

MemoryMB : 1024

MasterImageVM : Base.vm/Base.snapshot

MasterImageVMDate : 17/05/2010 09:53:40

IdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16

IdentityPoolName : idPool1

HostingUnitUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501

HostingUnitName : HostUnit1

CleanOnBoot : True

TaskId : 00000000-0000-0000-0000-000000000000



Metadata : {}

ControllerAddress : {ddcA.citrix.com,ddcB.citrix.com,ddcC.citrix2.com}

Add controller addresses to a provisioning scheme using a ProvisioningScheme object.

# Add-ProvSchemeMetadata

May 10, 2016  
Adds metadata on the given ProvisioningScheme.

## Syntax

```
Add-ProvSchemeMetadata [-ProvisioningSchemeUid] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Add-ProvSchemeMetadata [-ProvisioningSchemeUid] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Add-ProvSchemeMetadata [-ProvisioningSchemeName] <String> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]

Add-ProvSchemeMetadata [-ProvisioningSchemeName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Add-ProvSchemeMetadata [-InputObject] <ProvisioningScheme[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]

Add-ProvSchemeMetadata [-InputObject] <ProvisioningScheme[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability for additional custom data to be stored against given ProvisioningScheme objects. This cmdlet will not overwrite existing metadata on an object - use the Set-ProvSchemeMetadata cmdlet instead.

## Related topics

- [Set-ProvSchemeMetadata](#)
- [Remove-ProvSchemeMetadata](#)

## Parameters

**-ProvisioningSchemeUid**<Guid>

Id of the ProvisioningScheme

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-ProvisioningSchemeName**<String>

Name of the ProvisioningScheme

Required?	true
Default Value	
Accept Pipeline Input?	

Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-InputObject**<ProvisioningScheme[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the ProvisioningScheme specified. The property cannot contain any of the following characters \/:#.\*?=<> | []()"

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.MachineCreation.Sdk.Metadata

Add-ProvSchemeMetadata returns an array of objects containing the new definition of the metadata.

\n Property <string>

\n Specifies the name of the property.

\n Value <string>

\n Specifies the value for the property.

### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

InvalidParameterCombination

The cmdlet parameters are inconsistent.

UnknownObject

One of the specified objects was not found.

DuplicateObject

One of the specified metadata already exists.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Add-ProvSchemeMetadata -ProvisioningSchemeUid 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Property	Value
property	value

Add metadata with a name of 'property' and a value of 'value' to the ProvisioningScheme with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Add-ProvSchemeScope

May 10, 2016

Add the specified ProvisioningScheme(s) to the given scope(s).

## Syntax

```
Add-ProvSchemeScope [-Scope] <String[]> -InputObject <ProvisioningScheme[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Add-ProvSchemeScope [-Scope] <String[]> -ProvisioningSchemeUid <Guid[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Add-ProvSchemeScope [-Scope] <String[]> -ProvisioningSchemeName <String[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The AddProvSchemeScope cmdlet is used to associate one or more ProvisioningScheme objects with given scope(s).

There are multiple parameter sets for this cmdlet, allowing you to identify the ProvisioningScheme objects in different ways:

- ProvisioningScheme objects can be piped in or specified by the InputObject parameter
- The ProvisioningSchemeUid parameter specifies objects by ProvisioningSchemeUid
- The ProvisioningSchemeName parameter specifies objects by ProvisioningSchemeName (supports wildcards)

To add a ProvisioningScheme to a scope you need permission to change the scopes of the ProvisioningScheme and permission to add objects to all of the scopes you have specified.

If the ProvisioningScheme is already in a scope, that scope will be silently ignored.

## Related topics

[Remove-ProvSchemeScope](#)

[Get-ProvScopedObject](#)

## Parameters

**-Scope**<String[]>

Specifies the scopes to add the objects to.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-InputObject**<ProvisioningScheme[]>

Specifies the ProvisioningScheme objects to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

### **-ProvisioningSchemeUid**<Guid[]>

Specifies the ProvisioningScheme objects to be added by ProvisioningSchemeUid.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

### **-ProvisioningSchemeName**<String[]>

Specifies the ProvisioningScheme objects to be added by ProvisioningSchemeName.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

None

### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### UnknownObject

One of the specified objects was not found.

#### ScopeNotFound

One of the specified scopes was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command with the specified objects or scopes.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.



ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Add-ProvSchemeScope Finance -ProvisioningSchemeUid 6702C5D0-C073-4080-A0EE-EC74CB537C52
```

Adds a single ProvisioningScheme to the 'Finance' scope.

### ----- EXAMPLE 2 -----

```
c:\PS>Add-ProvSchemeScope Finance,Marketing -ProvisioningSchemeUid 6702C5D0-C073-4080-A0EE-EC74CB537C52
```

Adds a single ProvisioningScheme to the multiple scopes.

### ----- EXAMPLE 3 -----

```
c:\PS>Get-ProvScheme | Add-ProvSchemeScope Finance
```

Adds all visible ProvisioningScheme objects to the 'Finance' scope.

### ----- EXAMPLE 4 -----

```
c:\PS>Add-ProvSchemeScope Finance -ProvisioningSchemeName A*
```

Adds ProvisioningScheme objects with a name starting with an 'A' to the 'Finance' scope.

# Add-ProvTaskMetadata

May 10, 2016

Adds metadata on the given Task.

## Syntax

```
Add-ProvTaskMetadata [-TaskId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Add-ProvTaskMetadata [-TaskId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Add-ProvTaskMetadata [-InputObject] <Task[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Add-ProvTaskMetadata [-InputObject] <Task[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

## Detailed Description

Use this cmdlet to store additional custom data against given Task objects. This cmdlet does not overwrite existing metadata on an object - use the Set-ProvTaskMetadata cmdlet instead.

## Related topics

[Set-ProvTaskMetadata](#)

[Remove-ProvTaskMetadata](#)

[Get-ProvTask](#)

[Stop-ProvTask](#)

[Remove-ProvTask](#)

[Switch-ProvTask](#)

## Parameters

**-TaskId**<Guid>

Id of the Task

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

### **-InputObject**<Task[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the Task specified. The property cannot contain any of the following characters \;#. \*?=<> | []()''

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.MachineCreation.Sdk.Metadata

Add-ProvTaskMetadata returns an array of objects containing the new definition of the metadata.

Property <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

### Notes

If the command fails, the following errors can result.

#### Error Codes

-----

InvalidParameterCombination

The cmdlet parameters are inconsistent.

UnknownObject

One of the specified objects was not found.

DuplicateObject

One of the specified metadata already exists.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Add-ProvTaskMetadata -TaskId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Property	Value
-----	-----
property	value

Add metadata with a name of 'property' and a value of 'value' to the Task with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Get-ProvDBConnection

May 10, 2016

Gets the database string for the specified data store used by the MachineCreation Service.

## Syntax

```
Get-ProvDBConnection [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the database connection string for the specified data store.

If the returned string is blank, no valid connection string has been specified. In this case the service is running, but is idle and awaiting specification of a valid connection string.

## Related topics

[Get-ProvServiceStatus](#)

[Set-ProvDBConnection](#)

[Test-ProvDBConnection](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

system.string

The database connection string configured for the MachineCreation Service.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

NoDBConnections

The database connection string for the MachineCreation Service has not been specified.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Get-ProvDBConnection

Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True  
Get the database connection string for the MachineCreation Service.

# Get-ProvDBSchema

May 10, 2016

Gets a script that creates the MachineCreation Service database schema for the specified data store.

## Syntax

```
Get-ProvDBSchema [-DatabaseName <String>] [-ServiceGroupName <String>] [-ScriptType <ScriptTypes>] [-LocalDatabase] [-Sid <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets SQL scripts that can be used to create a new MachineCreation Service database schema, add a new MachineCreation Service to an existing site, remove a MachineCreation Service from a site, or create a database server logon for a MachineCreation Service. If no Sid parameter is provided, the scripts obtained relate to the currently selected MachineCreation Service instance, otherwise the scripts relate to MachineCreation Service instance running on the machine identified by the Sid provided. When obtaining the Evict script, a Sid parameter must be supplied. The current service instance is that on the local machine, or that explicitly specified by the last usage of the -AdminAddress parameter to a MachineCreation SDK cmdlet. The service instance used to obtain the scripts does not need to be a member of a site or to have had its database connection configured. The database scripts support only Microsoft SQL Server, or SQL Server Express, and require Windows integrated authentication to be used. They can be run using SQL Server's SQLCMD utility, or by copying the script into an SQL Server Management Studio (SSMS) query window and executing the query. If using SSMS, the query must be executed in 'SMDCMD mode'. The ScriptType parameter determines which script is obtained. If ScriptType is not specified, or is FullDatabase, the script contains:

- o Creation of service schema
- o Creation of database server logon
- o Creation of database user
- o Addition of database user to MachineCreation Service roles

If ScriptType is Instance, the returned script contains:

- o Creation of database server logon
- o Creation of database user
- o Addition of database user to MachineCreation Service roles

If ScriptType is Evict, the returned script contains:

- o Removal of MachineCreation Service instance from database
- o Removal of database user

If ScriptType is Login, the returned script contains:

- o Creation of database server logon only

If the service uses two data stores they can exist in the same database. You do not need to configure a database before



using this command.

## Related topics

[Set-ProvDBConnection](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database for which the schema will be generated.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ServiceGroupName**<String>

Specifies the name of the service group to be used when creating the database schema. The service group is a collection of all the MachineCreation services that share the same database instance and are considered equivalent; that is, all the services within a service group can be used interchangeably.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ScriptType**<ScriptTypes>

Specifies the type of database script returned. Available script types are:

### Database

Returns a full database script that can be used to create a database schema for the MachineCreation Service in a database instance that does not already contain a schema for this service. The DatabaseName and ServiceGroupName parameters must be specified to create a script of this type.

### Instance

Returns a permissions script that can be used to add further MachineCreation services to an existing database instance that already contains the full MachineCreation service schema, associating the services to the Service Group. The Sid parameter can optionally be specified to create a script of this type.

### Login

Returns a database logon script that can be used to add the required logon accounts to an existing database instance that contains the MachineCreation Service schema. This is used primarily when creating a mirrored database environment. The DatabaseName parameter must be specified to create a script of this type.

#### Evict

Returns a script that can be used to remove the specified MachineCreation Service from the database entirely. The DatabaseName and Sid parameters must be specified to create a script of this type.

Required?	false
Default Value	Database
Accept Pipeline Input?	false

#### **-LocalDatabase**<SwitchParameter>

Specifies whether the database script is to be used in a database instance run on the same controller as other services in the service group. Including this parameter ensures the script creates only the required permissions for local services to access the database schema for MachineCreation services.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Sid**<String>

Specifies the SID of the controller on which the MachineCreation Service instance to remove from the database is running.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
-----------	-------

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

### Systemstring

A string containing the required SQL script for application to a database.

## Notes

The scripts returned support Microsoft SQL Server Express Edition, Microsoft SQL Server Standard Edition, and Microsoft SQL Server Enterprise Edition databases only, and are generated on the assumption that integrated authentication will be used.

If the ScriptType parameter is not included or set to 'FullDatabase', the full database script is returned, which will:

Create the database schema.

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist).

If the ScriptType parameter is set to 'Instance', the script will:

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist) and associate it with a user.

If the ScriptType parameter is set to 'Login', the script will:

Create the logon (providing the schema does not already exist) and associate it with a pre-existing user of the same name.

If the LocalDatabase parameter is included, the NetworkService account will be added to the list of accounts permitted to access the database. This is required only if the database is run on a controller.

If the command fails, the following errors can be returned.

### Error Codes

-----

#### GetSchemasFailed

The database schema could not be found.

#### ActiveDirectoryAccountResolutionFailed

The specified Active Directory account or Group could not be found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-ProvDBSchema -DatabaseName MyDB -ServiceGroupName MyServiceGroup > c:\ProvSchema.sql
Get the full database schema for site data store of the MachineCreation Service and copy it to a file called
'c:\ProvSchema.sql'.
```

This script can then be used to create the schema in a pre-existing database named 'MyDB' that does not already contain a MachineCreation Service site schema.

----- **EXAMPLE 2** -----

```
c:\PS>Get-ProvDBSchema -DatabaseName MyDB -scriptType Login > c:\MachineCreationLogins.sql
Get the logon scripts for the MachineCreation Service.
```

# Get-ProvDBVersionChangeScript

May 10, 2016

Gets a script that updates the MachineCreation Service database schema.

## Syntax

```
Get-ProvDBVersionChangeScript -DatabaseName <String> -TargetVersion <Version> [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Returns a database script that can be used to upgrade or downgrade the site or secondary schema for the MachineCreation Service from the current schema version to a different version.

## Related topics

[Get-ProvInstalledDBVersion](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database instance to which the update applies.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-TargetVersion**<Version>

Specifies the version of the database you want to update to.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.Management.Automation.PSObject

A PSObject containing the required SQL script for application to a database.

## Notes

The PSObject returned by this cmdlet contains the following properties:

-- Script The raw text of the SQL script to apply the update, or null in the case when no upgrade path to the specified target version exists.

-- NeedExclusiveAccess Indicates whether all services in the service group must be shut down during the update or not.

-- CanUndo Indicates whether the generated script allows the updated schema to be reverted to the state prior to the update.

Scripts to update the schema version are stored in the database so any service in the service group can obtain these scripts. Extreme caution should be exercised when using update scripts. Citrix recommends backing up the database before attempting to upgrade the schema. Database update scripts may require exclusive use of the schema and so may not be able to execute while any MachineCreation services are running. However, this depends on the specific update being carried out.

After a schema update has been carried out, services that require the previous version of the schema may cease to operate. The ServiceState parameter reported by the Get-ProvServiceStatus command provides information about service compatibility. For example, if the schema has been upgraded to a more recent version that a service cannot use, the service reports "DBNewerVersionThanService".

If the command fails, the following errors can be returned.

### Error Codes

-----

#### NoOp

The operation was successful but had no effect.

#### NoDBConnections

The database connection string for the MachineCreation Service has not been specified.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> $update = Get-ProvDBVersionChangeScript -DatabaseName MyDb -TargetVersion 1.0.75.0
```

```
C:\PS> $update.Script > update_75.sql
```

Gets an SQL update script to update the current schema to version 1.0.75.0. The resulting update\_75.sql script is suitable for direct use with the SQL Server SQLCMD utility.

# Get-ProvInstalledDBVersion

May 10, 2016

Gets a list of all available database schema versions for the MachineCreation Service.

## Syntax

```
Get-ProvInstalledDBVersion [-Upgrade] [-Downgrade] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the current version of the MachineCreation Service database schema, if no flags are set, otherwise returns versions for which upgrade or downgrade scripts are available and have been stored in the database.

## Related topics

## Parameters

**-Upgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be updated should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Downgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be reverted should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.



Accept Pipeline Input?	false
------------------------	-------

## Return Values

### System.Version

The Get-ProvInstalledDbVersion command returns objects containing the new definition of the MachineCreation Service database schema version.

Major <Integer>

Minor <Integer>

Build <Integer>

Revision <Integer>

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

Both the Upgrade and Downgrade flags were specified.

#### NoOp

The operation was successful but had no effect.

#### NoDBConnections

The database connection string for the MachineCreation Service has not been specified.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ProvInstalledDBVersion
```

Major	Minor	Build	Revision
-------	-------	-------	----------

-----	-----	-----	-----
-------	-------	-------	-------

5	6	0	0
---	---	---	---

Get the currently installed version of the MachineCreation Service database schema.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-ProvInstalledDBVersion -Upgrade
```

Major	Minor	Build	Revision
-------	-------	-------	----------

-----	-----	-----	-----
-------	-------	-------	-------

6	0	0	0
---	---	---	---

Get the versions of the MachineCreation Service database schema for which upgrade scripts are supplied.

# Get-ProvObjectReference

May 10, 2016

Returns the number of local objects holding references to objects from other services.

## Syntax

```
Get-ProvObjectReference [-HostingUnitUid <Guid[]>] [-IdentityPoolUid <Guid[]>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Returns for each hosting unit or identity pool GUID the number of references by provisioning schemes or by long running task. This check is done without regard for scoping of existing provisioning schemes, references by inaccessible schemes are also checked.

## Related topics

## Parameters

**-HostingUnitUid**<Guid[]>

The identifiers of the hosting units(s) to be tested.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-IdentityPoolUid**<Guid[]>

The identifiers of the identity pool(s) to be tested.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
-----------	-------

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

### ObjectReferenceCount

An object which contain the input object identifier, its type, the type of referencing object and the number of references.

## Notes

In the case of failure, the following errors can result.

### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
Get-ProvObjectReference -HostingUnitUid 5B66A060-85E1-4DBD-9D1B-BF79881D3BB1
```

```
Count : 1
```

ObjectId : 5b66a060-85e1-4dbd-9d1b-bf79881d3bb1  
Source : ProvisioningScheme  
Target : HostingUnit

Count : 0  
ObjectId : 5b66a060-85e1-4dbd-9d1b-bf79881d3bb1  
Source : Task  
Target : HostingUnit

This checks a single hosting unit for objects that have references to them; in this case we only have a single provisioning scheme that relates to it.

# Get-ProvScheme

May 10, 2016

Gets the list of provisioning schemes.

## Syntax

```
Get-ProvScheme [[-ProvisioningSchemeName] <String>] [-ProvisioningSchemeUid <Guid>] [-Scopeld <Guid>] [-ScopeName <String>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Lets you retrieve the list of defined provisioning schemes.

## Related topics

[New-ProvScheme](#)

[Remove-ProvScheme](#)

[Add-ProvSchemeMetadata](#)

[Remove-ProvSchemeMetadata](#)

[Add-ProvSchemeControllerAddress](#)

[Remove-ProvSchemeControllerAddress](#)

## Parameters

**-ProvisioningSchemeName**<String>

The name of the provisioning scheme.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ProvisioningSchemeUid**<Guid>

The unique identifier of the provisioning scheme.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-ScopeId**<Guid>

Gets only results with a scope matching the specified scope identifier.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ScopeName**<String>

Gets only results with a scope matching the specified scope name.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

See about\_Prov\_Filtering for details.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

See about\_Prov\_Filtering for details.

Required?	false
Default Value	false
Accept Pipeline Input?	false

### **-Skip**<Int32>

See about\_Prov\_Filtering for details.

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy**<String>

See about\_Prov\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Filter**<String>

See about\_Prov\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. When a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false



## Return Values

Citrix.MachineCreation.Sdk.ProvisioningScheme

This object provides details of the provisioning scheme and contains the following information:

ProvisioningSchemeUid <Guid>

The unique identifier for the provisioning scheme.

ProvisioningSchemeName <string>

The name of the provisioning scheme.

CpuCount <int>

The number of processors that VMs will be created with when using this scheme.

MemoryMB <int>

The maximum amount of memory that VMs will be created with when using this scheme.

MasterImageVM <string>

The path within the hosting unit provider to the copy of the VM snapshot that the scheme uses.

MasterImageVMDate <DateTime>

The date and time that the copy was made of the VM snapshot used by the scheme.

IdentityPoolUid <Guid>

The unique identifier of the identity pool (from the ADIdentity PowerShell snap-in) that the scheme uses.

IdentityPoolName <string>

The name of the identity pool (from the ADIdentity PowerShell snap-in) that the scheme uses.

HostingUnitUid <Guid>

The unique identifier of the hosting unit (from the Hosting Unit PowerShell snap-in) that the scheme uses.

HostingUnitName <string>

The name of the hosting unit (from the Hosting Unit PowerShell snap-in) that the scheme uses.

CleanOnBoot <Boolean>

Indicates whether the VMs that are created will be reset to a clean state on each boot.

TaskId <Guid>

The identifier of any current task that is running for the provisioning scheme.

Metadata <Citrix.MachineCreation.Sdk.Metadata[]>

The metadata associated with this provisioning scheme.

ControllerAddress <string[]>

The DNS names of the controllers associated with this provisioning scheme for Quick Deploy purposes.

VMMetadata <char[]>

The opaque VM metadata block

UsePersonalVDiskStorage <bool>

True if the scheme will use personal vDisk storage.

PersonalVDiskDriveLetter <char>

The drive letter for the personal vDisk

PersonalVDiskDriveSize <int>

The size of the personal vDisk in GB

ProfileUsagePercentage <double>

The percentage of the personal vDisk to be used for profile data

DedicatedTenancy <bool>

Whether to use dedicated tenancy when creating machines in Cloud Hypervisors.

## Notes

In the case of failure, the following errors can result.

### Error Codes

-----

#### PartialData

Only a subset of the available data was returned.

#### CouldNotQueryDatabase

The query to get the database was not defined.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

#### CommunicationError

An error occurred while communicating with the service.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used, or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

C:\PS>Get-ProvScheme

```
ProvisioningSchemeUid : 7585f0de-192e-4847-a6d8-22713c3a2f42
ProvisioningSchemeName : Scheme1
CpuCount : 1
MemoryMB : 1024
MasterImageVM : /Base.vm/base.snapshot
MasterImageVMDate : 17/05/2010 09:27:50
IdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16
IdentityPoolName : idPool1
HostingUnitUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501
HostingUnitName : HostUnit1
CleanOnBoot : True
TaskId : 00000000-0000-0000-0000-000000000000
Metadata : {Department = Sales}
ControllerAddress : {}
VMMetadata : {0, 1, 0, 0...}
```

```
ProvisioningSchemeUid : 43d82099-1fd7-4617-93f0-25b160813905
ProvisioningSchemeName : Scheme2
CpuCount : 1
MemoryMB : 1024
MasterImageVM : /Base.vm/base.snapshot
MasterImageVMDate : 17/05/2010 09:53:40
IdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16
IdentityPoolName : idPool1
HostingUnitUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501
HostingUnitName : HostUnit1
CleanOnBoot : True
TaskId : 00000000-0000-0000-0000-000000000000
Metadata : {}
```

ControllerAddress : {}  
VMMetadata : {0, 1, 0, 0...}  
Returns all of the available provisioning schemes.

----- **EXAMPLE 2** -----

C:\PS>Get-ProvScheme -ProvisioningSchemeName Scheme[0-1]

ProvisioningSchemeUid : 7585f0de-192e-4847-a6d8-22713c3a2f42  
ProvisioningSchemeName : Scheme1  
CpuCount : 1  
MemoryMB : 1024  
MasterImageVM : /Base.vm/base.snapshot  
MasterImageVMDate : 17/05/2010 09:27:50  
IdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16  
IdentityPoolName : idPool1  
HostingUnitUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501  
HostingUnitName : HostUnit1  
CleanOnBoot : True  
TaskId : 00000000-0000-0000-0000-000000000000  
Metadata : {}  
ControllerAddress : {}  
VMMetadata : {0, 1, 0, 0...}  
Returns all of the provisioning schemes that have the name 'Scheme0' or 'Scheme1'.

# Get-ProvSchemeMasterVMImageHistory

May 10, 2016

Gets the list of master VM snapshots that have been used to provide hard disks to provisioning schemes.

## Syntax

```
Get-ProvSchemeMasterVMImageHistory [-ProvisioningSchemeName <String>] [-ProvisioningSchemeUid <Guid>] [-MasterImageVM <String>] [-VMImageHistoryUid <Guid>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [-CommonParameters]
```

## Detailed Description

Provides the ability to discover the master VM snapshots that have been used previously to provide the hard disk image for provisioning schemes. This information includes the date and time when the image was introduced. This information can be used to roll back a provisioning scheme to a previous image.

## Related topics

[Publish-ProvMasterVMImage](#)

[Remove-ProvSchemeMasterVMImageHistory](#)

## Parameters

**-ProvisioningSchemeName**<String>

The name of the provisioning scheme.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ProvisioningSchemeUid**<Guid>

The unique identifier of the provisioning scheme.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-MasterImageVM**<String>

The path to the snapshot item in the hosting unit PowerShell provider.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-VMImageHistoryUid**<Guid>

The unique identifier for the Image History item.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

See about\_Prov\_Filtering for details.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

See about\_Prov\_Filtering for details.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### **-Skip**<Int32>

See about\_Prov\_Filtering for details.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-Sort By**<String>

See about\_Prov\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Filter**<String>

See about\_Prov\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.MachineCreation.Sdk.VMImage

The object that represents a master VM history. This contains the following parameters:

VMImageHistoryUid <Guid>

The unique identifier for the History item.

ProvisioningSchemeUid <Guid>

The unique identifier for the provisioning scheme that the VM was used for.

ProvisioningSchemeName <string>

The name of the provisioning scheme that the VM was used for.

MasterImageVM <string>

The path to the Snapshot item that was used as the master VM image.

Date <DateTime>

The date and time that the VM or snapshot was used in the provisioning scheme.

## Notes

In the case of failure, the following errors can result.

### Error Codes

-----

#### PartialData

Only a subset of the available data was returned.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

#### CouldNotQueryDatabase

The query required to get the database was not defined.

#### CommunicationError

An error occurred while communicating with the service.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

C:\PS>Get-ProvSchemeMasterVMImageHistory

VMImageHistoryUid : 3cba3a75-89cd-4868-989b-27feb378fec5  
ProvisioningSchemeUid : 7585f0de-192e-4847-a6d8-22713c3a2f42  
ProvisioningSchemeName : MyScheme  
MasterImageVM : /Base.vm/base.snapshot  
Date : 17/05/2010 09:27:50

Gets all the hard disk images that have been used across all provisioning schemes.

----- **EXAMPLE 2** -----

C:\PS>Get-ProvSchemeMasterVMImageHistory -ProvisioningScheme MyScheme -masterImageVM "/BaseXp.vm/update1.snapshot" | Publish-ProvMasterVmImage  
Roll back the provisioning scheme to use the hard disk from the update1.snapshot for the provisioning scheme called "MyScheme".



# Get-ProvScopedObject

May 10, 2016

Gets the details of the scoped objects for the MachineCreation Service.

## Syntax

```
Get-ProvScopedObject [-ScopelId <Guid>] [-ScopeName <String>] [-ObjectType <ScopedObjectType>] [-ObjectId <String>] [-ObjectName <String>] [-Description <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns a list of directly scoped objects including the names and identifiers of both the scope and object as well as the object description for display purposes.

There will be at least one result for every directly scoped object. When an object is associated with multiple scopes the output contains one result per scope duplicating the object details.

No records are returned for the All scope, though if an object is not in any scope a result with a null ScopelId and ScopeName will be returned.

## Related topics

### Parameters

**-ScopelId**<Guid>

Gets scoped object entries for the given scope identifier.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ScopeName**<String>

Gets scoped object entries with the given scope name.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-ObjectType**<ScopedObjectType>

Gets scoped object entries for objects of the given type.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-ObjectId**<String>

Gets scoped object entries for objects with the specified object identifier.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-ObjectName**<String>

Gets scoped object entries for objects with the specified object identifier.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-Description**<String>

Gets scoped object entries for objects with the specified description.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Prov\_Filtering for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

**-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See [about\\_Prov\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.MachineCreation.Sdk.ScopedObject

The Get-ProvScopedObject command returns an object containing the following properties:

ScopeId <Guid>

Specifies the unique identifier of the scope.

ScopeName <String>

Specifies the display name of the scope.

ObjectType <ScopedObjectType>

Type of the object this entry relates to.

ObjectId <String>

Unique identifier of the object.

ObjectName <String>

Display name of the object

Description <String>

Description of the object (possibly \$null if the object type does not have a description).

Notes

If the command fails, the following errors can be returned.

Error Codes

-----

UnknownObject

One of the specified objects was not found.

PartialData

Only a subset of the available data was returned.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

CouldNotQueryDatabase

The query required to get the database was not defined.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-ProvScopedObject -ObjectType Scheme
```

ScopelId : eff6f464-f1ee-4442-add3-99982e0cec01

ScopeName : Sales

ObjectType : Scheme

Objectld : cd4174ee-9e4b-4e57-b126-9dbf757fe493

ObjectName : MyExampleScheme

Description : Test scheme

ScopelId : 304e0fa7-d390-47f0-a94f-7e956a324c41

ScopeName : Finance

ObjectType : Scheme

Objectld : cd4174ee-9e4b-4e57-b126-9dbf757fe493

ObjectName : MyExampleScheme

Description : Test scheme

ScopelId :

ScopeName :

ObjectType : Scheme

Objectld : 5062e46b-71bc-4ac9-901a-30fe6797e2f6

ObjectName : AnotherScheme

Description : Another scheme in no scopes

Gets all of the scoped objects with type Scheme. The example output shows a scheme object (MyExampleScheme) in two scopes Sales and Finance, and another scheme (AnotherScheme) that is not in any scope. The ScopelId and ScopeName values returned are null in the final record.

# Get-ProvService

May 10, 2016

Gets the service record entries for the MachineCreation Service.

## Syntax

```
Get-ProvService [-Metadata <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns instances of the MachineCreation Service that the service publishes. The service records contain account security identifier information that can be used to remove each service from the database.

A database connection for the service is required to use this command.

## Related topics

### Parameters

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Prov\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-Sort By**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false



### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See [about\\_Prov\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.MachineCreation.Sdk.Service

The Get-ProvServiceInstance command returns an object containing the following properties.

Uid <Integer>

Specifies the unique identifier for the service in the group. The unique identifier is an index number.

ServiceHostId <Guid>

Specifies the unique identifier for the service instance.

DNSName <String>

Specifies the domain name of the host on which the service runs.

MachineName <String>

Specifies the short name of the host on which the service runs.

CurrentState <Citrix.Fma.Sdk.ServiceCore.ServiceState>

Specifies whether the service is running, started but inactive, stopped, or failed.

LastStartTime <DateTime>

Specifies the date and time at which the service was last restarted.

LastActivityTime <DateTime>

Specifies the date and time at which the service was last stopped or restarted.

OSType

Specifies the operating system installed on the host on which the service runs.

OSVersion

Specifies the version of the operating system installed on the host on which the service runs.

ServiceVersion

Specifies the version number of the service instance. The version number is a string that reflects the full build version of the service.

DatabaseUserName <string>

Specifies for the service instance the Active Directory account name with permissions to access the database. This will be either the machine account or, if the database is running on a controller, the NetworkService account.

Sid <string>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

ActiveSiteServices <string[]>

Specifies the names of active site services currently running in the service. Site services are components that perform long-running background processing in some services. This field is empty for services that do not contain site services.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

UnknownObject

One of the specified objects was not found.

PartialData

Only a subset of the available data was returned.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

CouldNotQueryDatabase

The query required to get the database was not defined.

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ProvService
```

```
Uid : 1
ServiceHostId : aef6f464-f1ee-4042-a523-66982e0cecd0
DNSName : MyServer.company.com
MachineName : MYSERVER
CurrentState : On
LastStartTime : 04/04/2011 15:25:38
LastActivityTime : 04/04/2011 15:33:39
OSType : Win32NT
OSVersion : 6.1.7600.0
ServiceVersion : 5.1.0.0
DatabaseUserName : NT AUTHORITY\NETWORK SERVICE
SID : S-1-5-21-2316621082-1546847349-2782505528-1165
ActiveSiteServices : {MySiteService1, MySiteService2...}
Get all the instances of the MachineCreation Service running in the current service group.
```

# Get-ProvServiceAddedCapability

May 10, 2016

Gets any added capabilities for the MachineCreation Service on the controller.

## Syntax

```
Get-ProvServiceAddedCapability [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables updates to the MachineCreation Service on the controller to be detected.

You do not need to configure a database connection before using this command.

## Related topics

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.String

String containing added capabilities.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ProvServiceAddedCapability
```

Get the added capabilities of the MachineCreation Service.

# Get-ProvServiceConfigurationData

May 10, 2016

Gets configuration data for the service.

## Syntax

```
Get-ProvServiceConfigurationData [-Name <String[]>] [-Value <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to determine the configuration parameters for the service

## Related topics

[Set-ProvServiceConfigurationData](#)

[Remove-ProvServiceConfigurationData](#)

## Parameters

**-Name**<String[]>

The Configuration data item Name .

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Value**<String[]>

The Configuration data item value.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Prov\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Prov\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

Citrix.MachineCreation.Sdk.ServiceConfigurationData

This object provides details of the configuration data and contains the following information:

Name <string>

The Name for the configuration item.

Value <string>

The value for the configuration item.

#### Notes

In the case of failure the following errors can be produced.

Error Codes

-----

PartialData

Only a subset of the available data was returned.

CouldNotQueryDatabase

The query required to get the database was not defined.

DatabaseError



An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

An error occurred while communicating with the service.

ExceptionThrown

An unexpected error occurred. To locate more details see the Windows event logs on the controller being used, or examine the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

C:\PS>Get-ProvConfiguratuionData

Name : DeltaDiskDelete.timeDelay

Value : 10

Get the Configuration data for the service.

# Get-ProvServiceInstance

May 10, 2016

Gets the service instance entries for the MachineCreation Service.

## Syntax

```
Get-ProvServiceInstance [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns service interfaces published by the instance of the MachineCreation Service. Each instance of a service publishes multiple interfaces with distinct interface types, and each of these interfaces is represented as a ServiceInstance object. Service instances can be used to register the service with a central configuration service so that other services can use the functionality.

You do not need to configure a database connection to use this command.

## Related topics

## Parameters

**-AdminAddress** <String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.MachineCreation.Sdk.ServiceInstance

The Get-ProvServiceInstance command returns an object containing the following properties.

ServiceGroupUid <Guid>

Specifies the unique identifier for the service group of which the service is a member.

ServiceGroupName <String>

Specifies the name of the service group of which the service is a member.

ServiceInstanceUID <Guid>

Specifies the unique identifier for registered service instances, which are service instances held by and obtained from a

central configuration service. Unregistered service instances do not have unique identifiers.

ServiceType <String>

Specifies the service instance type. For this service, the service instance type is always Prov.

Address

Specifies the address of the service instance. The address can be used to access the service and, when registered in the central configuration service, can be used by other services to access the service.

Binding

Specifies the binding type that must be used to communicate with the service instance. In this release of XenDesktop, the binding type is always 'wcf\_HTTP\_kerb'. This indicates that the service provides a Windows Communication Foundation endpoint that uses HTTP binding with integrated authentication.

Version

Specifies the version of the service instance. The version number is used to ensure that the correct versions of the services are used for communications.

ServiceAccount <String>

Specifies the Active Directory account name for the machine on which the service instance is running. The account name is used to provide information about the permissions required for interservice communications.

ServiceAccountSid <String>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

InterfaceType <String>

Specifies the interface type. Each service can provide multiple service instances, each for a different purpose, and the interface defines the purpose. Available interfaces are:

SDK - for PowerShell operations

InterService - for operations between different services

Peer - for communications between services of the same type

Metadata <Citrix.MachineCreation.Sdk.Metadata[]>

The collection of metadata associated with registered service instances, which are service instances held by and obtained from a central configuration service. Metadata is not stored for unregistered service instances.

Notes

If the command fails, the following errors can be returned.

Error Codes

-----

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ProvServiceInstance
```

```
Address : http://MyServer.com:80/Citrix/MachineCreationService
Binding : wcf_HTTP_kerb
InterfaceType : SDK
Metadata :
MetadataMap :
ServiceAccount : ENG\MyAccount$
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164
ServiceGroupName : MyServiceGroup
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000
ServiceType : Prov
Version : 1
```

```
Address : http://MyServer.com:80/Citrix/MachineCreationService/IServiceApi
Binding : wcf_HTTP_kerb
InterfaceType : InterService
Metadata :
MetadataMap :
```

ServiceAccount : ENG\MyAccount  
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164  
ServiceGroupName : MyServiceGroup  
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d  
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000  
ServiceType : Prov  
Version : 1

Get all instances of the MachineCreation Service running on the specified machine. For remote services, use the AdminAddress parameter to define the service for which the interfaces are required. If the AdminAddress parameter has not been specified for the runspace, service instances running on the local machine are returned.

# Get-ProvServiceStatus

May 10, 2016

Gets the current status of the MachineCreation Service on the controller.

## Syntax

```
Get-ProvServiceStatus [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables the status of the MachineCreation Service on the controller to be monitored. If the service has multiple data stores it will return the overall state as an aggregate of all the data store states. For example, if the site data store status is OK and the secondary data store status is DBUnconfigured then it will return DBUnconfigured.

## Related topics

[Set-ProvDBConnection](#)

[Test-ProvDBConnection](#)

[Get-ProvDBConnection](#)

[Get-ProvDBSchema](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Get-ProvServiceStatus command returns an object containing the status of the MachineCreation Service together with extra diagnostics information.

DBUnconfigured

The MachineCreation Service does not have a database connection configured.

DBRejectedConnection

The database rejected the logon attempt from the MachineCreation Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

#### InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the MachineCreation Service schema has not been added to the database.

#### DBNotFound

The specified database could not be located with the configured connection string.

#### DBNewerVersionThanService

The version of the MachineCreation Service currently in use is incompatible with the version of the MachineCreation Service schema on the database. Upgrade the MachineCreation Service to a more recent version.

#### DBOlderVersionThanService

The version of the MachineCreation Service schema on the database is incompatible with the version of the MachineCreation Service currently in use. Upgrade the database schema to a more recent version.

#### DBVersionChangeInProgress

A database schema upgrade is currently in progress.

#### OK

The MachineCreation Service is running and is connected to a database containing a valid schema.

#### Failed

The MachineCreation Service has failed.

#### Unknown

(0) The service status cannot be determined.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ProvServiceStatus
```

DBUnconfigured

Get the current status of the MachineCreation Service.



# Get-ProvTask

May 10, 2016

Gets the task history for the MachineCreation Service.

## Syntax

```
Get-ProvTask [-TaskId] <Guid> [-Type <JobType>] [-Active <Boolean>] [-Metadata <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns a list of tasks that have run or are currently running within the MachineCreation Service.

## Related topics

[Remove-ProvTask](#)

[Stop-ProvTask](#)

[Switch-ProvTask](#)

[Add-ProvTaskMetadata](#)

[Remove-ProvTaskMetadata](#)

## Parameters

**-TaskId**<Guid>

Specifies the task identifier to be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Type**<JobType>

Specifies the type of task to be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Active**<Boolean>

Specifies whether currently running tasks only or completed tasks only are returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See about\_Prov\_Filtering for details.

Required?	false
Default Value	False

Accept Pipeline Input?	false
------------------------	-------

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

#### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Prov\_Filtering for details.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Notes

If the command fails, the following errors can result.

#### Error Codes

-----

#### UnknownObject

One of the specified objects was not found.

#### PartialData

Only a subset of the available data was returned.

#### InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

#### CouldNotQueryDatabase

The query required to get the database was not defined.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for

various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-ProvTask -Active $false
```

```
TaskId : cfe5b3a2-c471-443e-b05e-8658e672d10f
Type : MyTask
Host : NYSERVER
Status : Finished
CurrentOperation :
TaskProgress : 100
TaskExpectedCompletion : 10/10/2012 15:28:12
LastUpdateTime : 10/10/2012 15:28:12
ActiveElapsedTime : 56
DateFinished : 10/10/2012 15:28:12
TerminatingError :
```

...

Get all completed tasks for the MachineCreation Service.

All tasks will publish at least the fields listed above, plus more related to the particular task being performed.

# Get-ProvVM

May 10, 2016

Gets the VMs that were created using Citrix XenDesktop Machine Creation Services.

## Syntax

```
Get-ProvVM [-ProvisioningSchemeName] <String> [-ProvisioningSchemeUid <Guid>] [-VMName <String>] [-Locked <Boolean>] [-Tag <String>] [-OutOfDate] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to obtain a list of the VMs that were created using Machine Creation Services.

## Related topics

[New-ProvVM](#)

[Remove-ProvVM](#)

[Lock-ProvVM](#)

[Unlock-ProvVM](#)

## Parameters

**-ProvisioningSchemeName**<String>

The name of the provisioning scheme.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ProvisioningSchemeUid**<Guid>

The unique identifier of the provisioning scheme.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-VMName<String>**

The name of the VM in the hypervisor context.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Locked<Boolean>**

Indicates whether only VMs that are marked as locked are returned or not (see Lock-ProvVM and Unlock-ProvVM for details).

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Tag<String>**

The tag string that was associated with the VM when it was locked.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-OutOfDate<SwitchParameter>**

Indicates that the image currently assigned to the VM is out of date. The image will be updated the next time the VM is restarted.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount**<SwitchParameter>

See about\_Prov\_Filtering for details.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-MaxRecordCount**<Int32>

See about\_Prov\_Filtering for details.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-Skip**<Int32>

See about\_Prov\_Filtering for details.

Required?	false
Default Value	0
Accept Pipeline Input?	false

**-SortBy**<String>

See about\_Prov\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Filter**<String>



See about\_Prov\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

Citrix.MachineCreation.Sdk.ProvisionedVirtualMachine

The object has the following properties:

ADAccountSid <string>

The SID of the AD computer account that the VM is using.

ADAccountName <string>

The name of the AD computer account that the VM is using.

CpuCount <int>

The number of processors that the VM has been allocated.

CreationDate <DateTime>

The date and time that the VM was created.

Domain <string>

The Domain of the AD computer account that the VM is using.

ImageOutOfDate <Boolean>

Indicates if the image will be updated next time the VM is started.

Lock <Boolean>

Indicates whether the VM is locked or not.

MemoryMB <int>

The maximum amount of memory that VMs will be created with when using this scheme.

ProvisioningSchemeName <string>

The name of the provisioning scheme that the VM is part of.

ProvisioningSchemeUid <Guid>

The unique identifier for the provisioning scheme that the VM is part of.

Tag <string>

Provides the string associated with a locked VM.

VMId <string>

The identifier for the VM (hypervisor context).

VMName <string>

The name of the VM (hypervisor context).

AssignedImage <string>

The identifier (in the hypervisor) for the hard disk image that the VM is currently assigned.

BootedImage <string>

The identifier (in the hypervisor) for the hard disk image with which the VM is currently started.

HostingUnitUid <Guid>

The unique identifier for the hosting unit that was used to create the VM.

HypervisorConnectionUid <Guid>

The unique identifier of the hypervisor connection that was used to create the VM.

LastBootTime <DateTime>

The date and time of the last start of the VM.

OSDiskIndex <int>

The disk index at which the hard disk image, from which the VM is currently started, is attached (or [int]::MinValue for VMs inherited from versions of XenDesktop before 5.6)

PersonalVDiskIndex <int>

The disk index at which the personal vdisk is attached (defaults to [int]::MinValue for VMs without a personal vdisk)

PersonalVDiskStorage <string>

The identifier (in the hypervisor) for the storage on which the personal virtual disk image from which the VM is currently started is located. This is set only if the VM has a personal vDisk attached

StorageId <string>

The identifier (in the hypervisor) for the storage on which the hard disk image, from which the VM is currently started, is located.

## Notes

In the case of failure, the following errors can result.

### Error Codes

-----

#### PartialData

Only a subset of the available data was returned.

#### CouldNotQueryDatabase

The query required to get the database was not defined.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

#### CommunicationError

An error occurred while communicating with the service.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

C:\PS>Get-ProvVM -provisioningSchemeName MyScheme

```

ADAccountName : MYDomain\computer2$
ADAccountSid : S-1-5-21-3751941309-1176885247-1409628468-3179
CpuCount : 1
CreationDate : 17/05/2012 09:35:22
Domain : steve.dum.local
ImageOutOfDate : False
Lock : True
MemoryMB : 512
ProvisioningSchemeName : XenPS
ProvisioningSchemeUid : 5135a865-ba49-4e5f-87f2-2d65ee7a4e51
Tag : Brokered
VMId : a830de93-ddc5-b763-dc1a-35580a31401c
VMName : IP0051
AssignedImage : 57fc60c3-eb9b-4d38-8646-0afceec85335
BootedImage : 57fc60c3-eb9b-4d38-8646-0afceec85335
HostingUnitUid : ea17840f-cf2d-4d80-94e0-3b752b32e0af
HypervisorConnectionUid : 99f9f826-31fc-4453-8ca0-9ba54306c3ac
IdentityDiskIndex : 1
LastBootTime : 17/05/2012 09:35:22
OSDiskIndex : 0
PersonalVDiskIndex : -2147483648
PersonalVDiskStorage :
StorageId : 33ad07a7-edd7-589b-716a-86cad4739f5e
Gets all the Virtual Machines that were provisioned into the Provisioning Scheme called 'MyScheme'.

```

----- **EXAMPLE 2** -----

```
C:\PS>Get-ProvVM -Locked $true
```

```

ADAccountName : MYDomain\computer1$
ADAccountSid : S-1-5-21-3751941309-1176885247-1409628468-3178
CpuCount : 1
CreationDate : 17/05/2012 09:35:30
Domain : steve.dum.local
ImageOutOfDate : False
Lock : True
MemoryMB : 512
ProvisioningSchemeName : XenPS
ProvisioningSchemeUid : 5135a865-ba49-4e5f-87f2-2d65ee7a4e51
Tag : Brokered
VMId : a830de93-ddc5-b763-dc1a-35580a31401c
VMName : IP0051
AssignedImage : 57fc60c3-eb9b-4d38-8646-0afceec85335
BootedImage : 57fc60c3-eb9b-4d38-8646-0afceec85335
HostingUnitUid : ea17840f-cf2d-4d80-94e0-3b752b32e0af
HypervisorConnectionUid : 99f9f826-31fc-4453-8ca0-9ba54306c3ac
IdentityDiskIndex : 1
LastBootTime : 17/05/2012 09:35:22

```

OSDiskIndex : 0

PersonalVDiskIndex : -2147483648

PersonalVDiskStorage :

StorageId : 33ad07a7-edd7-589b-716a-86cad4739f5e

Gets all the Virtual Machines that were locked, regardless of which Provisioning Scheme the VM is part of.

# Lock-ProvVM

May 10, 2016

Locks a VM.

## Syntax

```
Lock-ProvVM [-VMID] <String[]> -ProvisioningSchemeName <String> [-Tag <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Lock-ProvVM [-VMID] <String[]> -ProvisioningSchemeUid <Guid> [-Tag <String>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to 'lock' a virtual machine with a tag string. This stops other commands from being able to remove the virtual machine without unlocking the VM first. This is to enable consumers of the virtual machines to indicate that the VM is being used.

## Related topics

[UnLock-ProvVM](#)

[Get-ProvVM](#)

[Remove-ProvVM](#)

## Parameters

**-VMID**<String[]>

The virtual machine Id (hypervisor context).

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ProvisioningSchemeName**<String>

The name of the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ProvisioningSchemeUid**<Guid>

The unique identifier of the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Tag**<String>

The string to be held against the VM being locked.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Direct or typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

Citrix.MachineCreation.Sdk.ProvisionedVirtualMachine You can pipe an object containing a parameter called 'VMId' and 'ProvisioningSchemeName' to Lock-ProvVM

## Notes

In the case of failure, the following errors can result.

### Error Codes

-----

#### VMDoesNotExist

The specified VM cannot be located.

#### VMAlreadyLocked

The VM is already unlocked.

#### VMDoesNotExistForProvisioningScheme

The specified VM does exist in the hypervisor, but is not part of the specified provisioning scheme.

#### PermissionDenied

The user is not authorized to perform this operation

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the



XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS>Lock-ProvVM -provisioningSchemeName MyScheme -VMId bc79802c-ba6e-8de8-99e9-4c35d7ad24b4 -Tag LockedString
```

Locks the VM with the Id 'bc79802c-ba6e-8de8-99e9-4c35d7ad24b4' in the provisioning scheme 'MyScheme' with the tag 'LockedString'.

### ----- EXAMPLE 2 -----

```
C:\PS>Get-ProvVM -provisioningSchemeName MyScheme | Lock-ProvVM -Tag LockedString
```

Locks all the VMs in the provisioning scheme 'MyScheme' with the tag 'LockedString'.

# New-ProvScheme

May 10, 2016  
Creates a new provisioning scheme.

### Syntax

```
New-ProvScheme [-ProvisioningSchemeName] <String> -HostingUnitName <String> -IdentityPoolName <String> -MasterImageVM <String> [-VMCpuCount <Int32>] [-VMMemoryMB <Int32>] [-CleanOnBoot] [-UsePersonalVDiskStorage] [-Scope <String[]>] [-NoImagePreparation] [-NetworkMapping <Hashtable>] [-Metadata <Hashtable>] [-ServiceOffering <String>] [-SecurityGroup <String[]>] [-DedicatedTenancy] [-VhdTemplateSource <String>] [-VhdResultDestination <String>] [-AppScanResultsFile <String>] [-ResetAdministratorPasswords] [-RunAsynchronously] [-PurgeJobOnSuccess] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

New-ProvScheme [-ProvisioningSchemeName] <String> -HostingUnitUid <Guid> -IdentityPoolUid <Guid> -MasterImageVM <String> [-VMCpuCount <Int32>] [-VMMemoryMB <Int32>] [-CleanOnBoot] [-UsePersonalVDiskStorage] [-Scope <String[]>] [-NoImagePreparation] [-NetworkMapping <Hashtable>] [-Metadata <Hashtable>] [-ServiceOffering <String>] [-SecurityGroup <String[]>] [-DedicatedTenancy] [-VhdTemplateSource <String>] [-VhdResultDestination <String>] [-AppScanResultsFile <String>] [-ResetAdministratorPasswords] [-RunAsynchronously] [-PurgeJobOnSuccess] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

### Detailed Description

Lets you create a new provisioning scheme. The creation process makes a copy of the hard disk attached to a virtual machine snapshot and stores it in every storage location that the hosting unit referenced by the provisioning scheme defines. This is a long-running task and typically takes several minutes to complete (depending on the size of the hard disk that is being copied and the number of snapshots that the hard disk consists of).

A snapshot must be used rather than a VM, so that the content of the hard disk for the provisioning scheme can be easily determined.

As the snapshot is specified using a path into the Citrix Host Service Powershell Provider the Citrix Host Service Powershell snap-in must also be loaded for this cmdlet to be used.

This cmdlet requires information to be provided that is retrieved using other snap-ins that form part of the Citrix Machine Creation Services: Hosting Unit Service Snapin The snap-in that provides information about the hypervisors. AD Identity Service Snapin The snap-in that provides information about the identity pools.

The provisioning scheme is a collection of all of the data that is required to form a template against which virtual machines can be created. It therefore requires the following: Hosting Unit A reference to an item defined in the Host Service that defines the hypervisor, the network, and the storage to be used. Identity Pool A reference to the collection of Active Directory accounts that is used for virtual machines created from the provisioning scheme.

### Related topics

- [Get-ProvTask](#)
- [Get-ProvScheme](#)
- [Test-ProvSchemeNameAvailable](#)

### Parameters

**-ProvisioningSchemeName**<String>

The name of the provisioning scheme to be created. This must be a name that is not being used by an existing provisioning scheme, and it must not contain any of the following characters \;#.\*?=<>|[]0""

Required?	true
Default Value	
Accept Pipeline Input?	false

**-HostingUnitName**<String>

The name of the hosting unit used for the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-IdentityPoolName**<String>

The name for the identity pool used for the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-HostingUnitUid<Guid>**

The identifier for the hosting unit used for the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-IdentityPoolUid<Guid>**

The identifier of the identity pool used for the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-MasterImageVM<String>**

The path in the hosting unit provider to the virtual machine snapshot that is used. This identifies the hard disk to be used and the default values for the memory and processors. This must be a path to a Snapshot item in the same hosting unit that the hosting unit name or hosting unit UID refers to.

Valid paths are of the format; XDHyp:\HostingUnits\<HostingUnitName>\<path>\<VmName>.vm\<SnapshotName>.snapshot

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-VMCpuCount<Int32>**

The number of processors used by virtual machines created from the provisioning scheme.

Required?	false
Default Value	The number of CPUs assigned to the base image VM snapshot.
Accept Pipeline Input?	false

**-VMMemoryMB<Int32>**

The maximum amount of memory used by virtual machines created from the provisioning scheme.

Required?	false
Default Value	The amount of memory assigned to the base image VM snapshot.
Accept Pipeline Input?	false

**-CleanOnBoot<SwitchParameter>**

Indicates whether or not the virtual machines created from this provisioning scheme are reset to their initial condition each time they are started.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-UsePersonalVDiskStorage<SwitchParameter>**

Indicates whether or not personal vDisks are used for the VMs in this provisioning scheme.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### -Scope<String[]>

The administration scopes to be applied to the new provisioning scheme.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -NoImagePreparation<SwitchParameter>

Indicates that Image Preparation should not be performed on this Provisioning Scheme

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -NetworkMapping<Hashtable>

Specifies how the attached NICs are mapped to networks. If this parameter is omitted, provisioned VMs are created with a single NIC, which is mapped to the default network in the HostingUnit. If this parameter is supplied, machines are created with the number of NICs specified in the map, and each NIC is attached to the specified network.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -Metadata<Hashtable>

Specifies any metadata to be attached to the scheme when it is created.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -ServiceOffering<String>

The Service Offering to use when creating machines in Cloud Hypervisors.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -SecurityGroup<String[]>

The security groups to apply to machines created in Cloud Hypervisors

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DedicatedTenancy**<SwitchParameter>

Whether to use dedicated tenancy when creating machines in Cloud Hypervisors.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### **-VhdTemplateSource**<String>

A file path to a source VHD template to be used when performing Application Scanning during Image Preparation. The presence of this parameter in conjunction with VhdResultDestination implies that application scanning is to be performed

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-VhdResultDestination**<String>

A file path (including file name) where the VHD disk file containing the results of application scanning should be written.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AppScanResultsFile**<String>

File name to which the results of application scanning should be written.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ResetAdministratorPasswords**<SwitchParameter>

Indicates whether to reset the password for the administrator accounts on provisioned machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RunAsynchronously**<SwitchParameter>

Indicates whether or not the command returns before it is complete. If specified, the command returns an identifier for the task that was created. This task can be monitored using the get-ProvTask command.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### **-PurgeJobOnSuccess**<SwitchParameter>

Indicates that the task history is removed from the database when the task has finished. This can only be specified for tasks that are not run asynchronously.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

System.Guid

When the RunAsynchronously identifier is specified, this GUID is returned and provides the task identifier.

System.Management.Automation.PSCustomObject

This object provides details of the task that was run and contains the following information:

TaskId <Guid>

The identifier for the task that was performed.

Active <Boolean>

Indicates whether the task is still processing or is complete.

Host <string>

The name of the host on which the task is running or was run.

DateStarted <DateTime>

The date and time that the task was initiated.

Type <Citrix.XDInterServiceTypes.JobType>

The type of task. For new provisioning scheme tasks, this is always NewProvisioningScheme.

Metadata <Citrix.MachineCreation.Sdk.Metadata[]>

The list of metadata stored against the task. For new tasks, this is empty until metadata is added.

WorkflowStatus <System.Workflow.Runtime.WorkflowStatus>

Indicates the status of the workflow that is used to process the task.

ProvisioningSchemeName <string>

The name of the provisioning scheme that the task was for.

ProvisioningSchemeUid <Guid>

The unique identifier of the provisioning scheme that the task was for.

MasterImage <string>

The path (in the hosting unit provider) of the virtual machine snapshot that was used as the master VM image for the task.

IdentityPoolName <string>

The name of the identity pool (from the ADIdentity PowerShell snap-in) that the new provisioning scheme uses.

IdentityPoolUid <guid>

The unique identifier name of the identity pool (from the ADIdentity PowerShell snap-in) that the new provisioning scheme uses.

HostingUnitName <string>

The name of the hosting unit (from the Hosting Unit PowerShell snap-in) that the new provisioning scheme uses.

HostingUnitUid

The unique identifier of the hosting unit (from the Hosting Unit PowerShell snap-in) that the new provisioning scheme uses.

PersonalVdiskDriveLetter

The drive letter on which a personal vDisk is mounted (blank if the personal vDisk feature was not selected).

PersonalVdiskDriveSize

The size of any personal vDisk (zero if the personal vDisk feature was not selected).

Scopes <Citrix.Fma.Sdk.ServiceCoreScopeReference[]>

The delegated administration scopes to which the scheme will belong.

NetworkMap <Citrix.MachineCreation.Sdk.NetworkMap>

The list of NIC to network associations, if specified.

ProvisioningSchemeMetadata <Dictionary<string, string>>

The metadata to apply to the provisioning scheme, if specified.

TaskState <Citrix.MachineCreation.Sdk.NewProvisioningSchemeState>

The state of the task. This can be any of the following:

Processing

The task has begun but has not done anything yet.

LocatingResources,

The workflow is locating resources from other services.

HostingUnitNotFound

The task failed because the required hosting unit could not be located.

VirtualMachineSnapshotNotFound

The task failed because the required VM snapshot could not be located.

ConsolidatingMasterImage

The task is consolidating the master image.

ReplicatingConsolidatedImageToAllStorage

The task is replicating the consolidated master image.

StoringProvisioningScheme

The task is storing the provisioning scheme data in the database.

Finished

The task completed with no errors.

ProvisioningSchemeAlreadyExists

The task failed because a provisioning scheme with the same name already exists.

IdentityPoolNotFound

The task failed because the specified identity pool could not be found.

MasterVMImageIsNotPartOfProvisioningSchemeHostingUnit,

The task failed because the hosting unit from which the master image originated is not the same hosting unit that the provisioning scheme is using.

MasterVmlImageIsASnapshot

The task failed because the master VM path does not refer to a Snapshot item.

ProvisioningSchemeNotFound

The task failed because it could not find a provisioning scheme with the specified name.

TaskAlreadyRunningForProvisioningScheme

The task failed because a task for a provisioning scheme with the same name is already running.

InvalidMasterVMConfiguration

The task failed because the VM snapshot specified as the master has an invalid configuration.

#### InvalidMasterVMState

The task failed because the VM snapshot specified as the master is currently in an invalid state.

#### InsufficientResources

The task failed because the hypervisor did not have enough resources to complete the task.

#### DiskConsolidationFailed

The disk consolidation task failed. Details are in the task state information string.

#### StorageNotFound

The task failed because no associated storage was found in the hosting unit.

#### ConfigurationError

The task failed because the service is unable to contact one of the other services. This is because not all appropriate Configuration Service registrations have been performed.

#### Canceled

The task was stopped by user intervention (using Stop-ProvTask).

#### TaskStateInformation

Additional information about the current task state.

#### TaskProgress

The progress of the task 0-100%.

#### DiskSize

The size of the master image in GB

#### DedicatedTenancy

Whether to use dedicated tenancy when creating machines in Cloud Hypervisors.

#### Notes

Only one long-running task for each provisioning scheme can be processed at a time.

In case of failure, the following errors can result.

#### Error Codes

-----

#### JobCreationFailed

The requested task could not be started.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation. Communication with the database failed for various reasons.

#### MachineCreationServiceDoesNotSupportPersonalDisk

The service instance being used has not been upgraded to support the personal vDisk feature.

#### DatabaseMissingCapabilities

The database supporting the service instance being used has not been upgraded to support the personal vDisk feature.

#### CommunicationError

An error occurred while communicating with the service.

#### InvalidParameterCombination

Both PurgeJobOnSuccess and RunAsynchronously were specified. When running asynchronously, the cmdlet terminates before the job does, so it cannot clean up the completed job.

#### PermissionDenied

The user does not have administrative rights to perform this operation.



#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### ScopeNotFound

One or more of the scopes nominated for the new provisioning scheme do not exist.

#### WorkflowHostUnavailable

The task could not be started because the database connection is inactive.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used, or examine the XenDesktop logs. VhdParametersMustBeSupplied

When parameter VhdTemplateSource or VhdResultDestination is supplied, both parameters are required to be supplied.

The cmdlet is associated with a task of type NewProvisioningScheme, and while active will move through the following operations (CurrentOperation field)

#### ValidatingInputs

#### ConsolidatingMasterImage

#### PreparingMasterImage

#### ReplicatingMasterImage

#### CommittingScheme

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> New-ProvScheme -ProvisioningSchemeName XenPS -HostingUnitName XenHu -IdentityPoolName idPool1 -CleanOnBoot -MasterImageVM XDHyp:\HostingUnits\XenHU\Base.vm\Base.snapshot
```

```
TaskId : 90e93b9d-a225-4701-ad50-fa1546af35ac
Active : False
Host : MyHost
DateStarted : 17/05/2010 08:22:22
Type : NewProvisioningScheme
Metadata : {}
WorkflowStatus : Completed
ProvisioningSchemeName : XenPS
MasterImage : /Base.vm/Base.snapshot
IdentityPoolName : idPool1
IdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16
HostingUnitName : XenHU
HostingUnitUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501
PersonalVDiskDriveLetter :
PersonalVDiskDriveSize : 0
ProvisioningSchemeUid : 7585f0de-192e-4847-a6d8-22713c3a2f42
CurrentOperation :
TaskState : Finished
TaskStateInformation :
TaskProgress : 100
DiskSize : 24
Creates a new provisioning scheme with the name "XenPS" using the hosting unit "XenHu" and the identity pool "idPool1" from the master VM snapshot called "Base.snapshot".
```

##### ----- EXAMPLE 2 -----

```
C:\PS> New-ProvScheme -ProvisioningSchemeName XenPS -HostingUnitName XenHu -IdentityPoolName idPool1 -CleanOnBoot -MasterImageVM XDHyp:\HostingUnits\XenHU\Base.vm\Base.snapshot
```

#### Guid

```
6dd85fec-96cf-46b1-9cd4-d8ba7d06e85b
```

Creates a new provisioning scheme with the name "XenPS" using the hosting unit "XenHu" and the identity pool "idPool1" from the master VM snapshot called "Base.snapshot" asynchronously. To get the task details, use Get-ProvTask -TaskID <task id>

i.e.

```
C:\PS>Get-ProvTask -TaskID 6dd85fec-96cf-46b1-9cd4-d8ba7d06e85b
```

```
TaskId : 6dd85fec-96cf-46b1-9cd4-d8ba7d06e85b
```

```
Active : False
```

```
Host : MyHost
```

```
DateStarted : 17/05/2010 08:22:22
```

```
Type : NewProvisioningScheme
```

```
Metadata : {}
```

```
WorkflowStatus : Completed
ProvisioningSchemeName : XenPS
MasterImage : XDHyp:\HostingUnits\XenHU\Base.vm\Base.snapshot
IdentityPoolName : idPool1
IdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16
HostingUnitName : XenHU
HostingUnitUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501
PersonalVdiskDriveLetter :
PersonalVdiskDriveSize : 0
ProvisioningSchemeUid : 7585f0de-192e-4847-a6d8-22713c3a2f42
TaskState : Finished
TaskStateInformation :
TaskProgress : 100
DiskSize : 24
```

----- **EXAMPLE 3** -----

C:\PS>\$provScheme = New-ProvScheme -ProvisioningSchemeName XenPS2 -HostingUnitName XenHu -IdentityPoolName idPool1 -CleanOnBoot -MasterImageVM XDHyp:\HostingUnits\XenHU\Base  
Creates a new provisioning scheme with the name "XenPS2" using the hosting unit "XenHu" and the identity pool "idPool1" from the master VM snapshot called "Base.snapshot"; apply a 17GB  
personal vDisk. The personal vDisk is mapped as drive X. The operation runs synchronously, and the return value contains the task details

For example:

```
C:\PS>$provScheme
TaskId : d726222a-04b5-4098-b9ac-db85ed9d351b
Active : False
Host : MyHost
DateStarted : 12/09/2011 09:30:04
Type : NewProvisioningScheme
Metadata : {}
ProvisioningSchemeName : XenPS2
IdentityPoolName : idPool1
IdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16
HostingUnitName : XenHU
HostingUnitUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501
PersonalVdiskDriveLetter : X
PersonalVdiskDriveSize : 17
WorkflowStatus : Completed
MasterImage : XDHyp:\HostingUnits\XenHU\Base.vm\Base.snapshot
ProvisioningSchemeUid : 7585f0de-192e-4847-a6d8-22713c3a2f42
TaskState : Finished
TaskStateInformation :
TaskProgress : 100
DiskSize : 24
```

# New-ProvVM

May 10, 2016

Creates a new virtual machine.

## Syntax

```
New-ProvVM -ProvisioningSchemeName <String> -ADAccountName <String[]> [-FastBuild] [-NetworkMapping <Hashtable>] [-AutoAssignVLAN] [-SecurityGroup <String[]>] [-MachinesPerAssistant <Int32>] [-MaxAssistants <Int32>] [-RunAsynchronously] [-PurgeJobOnSuccess] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
New-ProvVM -ProvisioningSchemeUid <Guid> -ADAccountName <String[]> [-FastBuild] [-NetworkMapping <Hashtable>] [-AutoAssignVLAN] [-SecurityGroup <String[]>] [-MachinesPerAssistant <Int32>] [-MaxAssistants <Int32>] [-RunAsynchronously] [-PurgeJobOnSuccess] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Lets you create new virtual machines with the configuration specified by a provisioning scheme.

The virtual machines are created using all of the storage specified in the provisioning scheme. The storage is used in a round robin mechanism. For the duration of this task, the AD accounts are locked (by the AD Identity Service), which stops the same accounts being used by other virtual machine creation tasks.

## Related topics

[Get-ProvVM](#)

[Remove-ProvVM](#)

[Lock-ProvVM](#)

[Unlock-ProvVM](#)

## Parameters

**-ProvisioningSchemeName**<String>

The name of the provisioning scheme in which the virtual machines are created.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-ProvisioningSchemeUid**<Guid>

The unique identifier for the provisioning scheme in which the virtual machines are created.

--	--

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-ADAccountName**<String[]>

A list of the Active Directory account names that are used for the virtual machines. The accounts must be provided in a domain-qualified format. This parameter accepts Identity objects as returned by the New-AcctADAccount cmdlet, or any PSObject with string properties "Domain" and "ADAccountName".

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-FastBuild**<SwitchParameter>

Indicates whether or not the command can improve speed by using optimizations in the creation process. This may mean that machines you create cannot be booted until ResetVM operations have been performed by the Broker and Machine Identity Services.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-NetworkMapping**<Hashtable>

Specifies how the attached NICs are mapped to networks. If this parameter is omitted, provisioned VMs are created with network settings as inherited from the provisioning scheme. If this parameter is supplied, machines are created with the number of NICs specified in the map, and each NIC is attached to the specified network.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AutoAssignVLAN**<SwitchParameter>

Indicates whether or not the VLAN of the network should be automatically assigned to the new VM or if the VLAN from the master image should be used

Required?	false
Default Value	false
Accept Pipeline Input?	false

### **-SecurityGroup**<String[]>

The security groups to apply to machines created in Cloud Hypervisors

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-MachinesPerAssistant**<Int32>

The number of concurrent volume operations that may be performed by each volume worker helper machine

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-MaxAssistants**<Int32>

The maximum number of volume worker help machines that may be created for this operation.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-RunAsynchronously**<SwitchParameter>

Indicates whether or not the command returns before it is complete. If specified, the command returns an identifier for the task that was created. This task can be monitored using the get-ProvTask command.

Required?	false
Default Value	false
Accept Pipeline Input?	false

### **-PurgeJobOnSuccess**<SwitchParameter>

Indicates that the task history is removed from the database when the task has finished. This cannot be specified for tasks that are run asynchronously.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller to which the PowerShell snap-in connects. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

### System.Guid

When the RunAsynchronously identifier is specified, this GUID is returned and provides the task identifier.

### System.Management.Automation.PSCustomObject

This object provides details of the task that was run and contains the following information:

#### TaskId <Guid>

The identifier for the task that was performed.

#### Active <Boolean>

Indicates whether the task is still processing or is complete.

#### Host <string>

The name of the host on which the task is running or was run.

#### DateStarted <DateTime>

The date and time that the task was initiated.

#### Type <Citrix.XDInterServiceTypes.JobType>

The type of task. For newly provisioned VM tasks, this is always "NewVirtualMachine".

#### Metadata <Citrix.MachineCreation.Sdk.Metadata[]>

The list of metadata stored against the task. For new tasks this is empty until metadata is added.

#### WorkflowStatus <System.Workflow.Runtime.WorkflowStatus>

Indicates the status of the workflow that is used to process the task.

#### ProvisioningSchemeName <string>

The name of the provisioning scheme that the task was for.

#### ProvisioningSchemeUid <Guid>

The unique identifier of the provisioning scheme that the task was for.

#### MasterImage <string>

The path (in the hosting unit provider) of the virtual machine snapshot that was used as the master image for the task.

#### IdentityPoolName <string>

The name of the identity pool (from the ADIdentity PowerShell snap-in) that the new provisioning scheme uses.

IdentityPoolUid <guid>

The unique identifier name of the identity pool (from the ADIdentity PowerShell snap-in) that the new provisioning scheme uses.

HostingUnitName <string>

The name of the hosting unit (from the Hosting Unit PowerShell snap-in) that the new provisioning scheme uses.

HostingUnitUid

The unique identifier of the hosting unit (from the Hosting Unit PowerShell snap-in) that the new provisioning scheme uses.

TaskState <Citrix.DesktopUpdateManager.SDK.ProvisionVMState>

The state of the task. This can be any of the following:

Processing

Indicates that the task is in its initial state.

LocatingResources,

Indicates that the task is locating information from other services.

HostingUnitNotFound

Indicates that the required hosting unit could not be located.

ProvisioningSchemeNotFound

Indicates that the required provisioning scheme could not be located.

Provisioning

Indicates that the task is at the provisioning stage.

IdentityPoolNotFound

Indicates that the required identity pool could not be located.

TaskAlreadyRunningForProvisioningScheme

Indicates that the provisioning scheme already has another task running.

HypervisorInMaintenanceMode

Indicates that the hypervisor is not available for normal use.

NoAvailableStorage

Indicates that no storage is available for the hypervisor.

NoAvailableNetwork

Indicates that no network is available for the hypervisor.



Finalizing

Indicates that the task is finalizing.

Finished

Indicates that the task is complete.

FinishedWithErrors

Indicates that the task is complete but there were errors. Specific details of errors are included with each failed virtual machine.

Removing

Indicates that the task is removing virtual machines from the hypervisor.

Failed

The job failed for reasons specified in TaskStateInformation.

Canceled

Indicates that the task was stopped by user intervention (using Stop-ProvTask).

TaskStateInformation

Provides more detailed information about the current task state.

VirtualMachinesToCreateCount <int>

The total number of virtual machines that the task is trying to create.

VirtualMachinesCreatedCount <int>

The number of virtual machines that the task has created so far. Details of the machines that were created are in the CreatedVirtualMachines parameter.

VirtualMachinesCreationFailedCount <int>

The number of virtual machines that the task has failed to create. Details of the machines that were not created are in the FailedVirtualMachines parameter.

FailedVirtualMachines <Citrix.DesktopUpdateManager.SDK.VirtualMachineCreation[]>

ADAccountName <string>

The domain qualified AD Account name of the machine.

ADAccountSid <string>

The AD account SID of the machine account.

DiagnosticInformation <Citrix.MachineCreation.Sdk.ExceptionSurrogate[]>

A collection of handled error states which caused the provisioning to fail.

ExceptionType <string>

The type of exception this object represents

Message <string>

The exception message

Details <string>

The full exception content including stack trace

InnerException <Citrix.MachineCreation.Sdk.ExceptionSurrogate>

Information relating to any contributing error state

Status <string>

StatusAdditionalInformation <string>

VMId <string>

The virtual machine identifier within the hypervisor in which the VM resides.

VMName <string>

The display name of the virtual machine within the hypervisor in which the VM resides.

CreatedVirtualMachines <Citrix.DesktopUpdateManager.SDK.VirtualMachineCreation[]>

See FailedVirtualMachines for details of the object parameters.

## Notes

Only one long-running task in each provisioning scheme can be processed at a time.

In the case of failure, the following errors can result.

### Error Codes

-----

#### JobCreationFailed

The requested task could not be started.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation. Communication with the database failed for various reasons.

WorkflowHostUnavailable

The task could not be started because the database connection is inactive.

PermissionDenied

The user does not have administrative rights to perform this operation.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

CommunicationError

An error occurred while communicating with the service.

InvalidParameterCombination

Both PurgeJobOnSuccess and RunAsynchronously were specified.

When running asynchronously, the cmdlet terminates before the job does, so it cannot clean up the completed job.

PermissionDenied

The user does not have administrative rights to perform this operation.

ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used, or examine the XenDesktop logs.

The cmdlet is associated with a task of type NewVirtualMachine, and while active will move through the following operations (CurrentOperation field)

ValidatingInputs

CreatingVirtualMachines

ReleasingAccountLocks

Examples

----- **EXAMPLE 1** -----

```
C:\PS>New-provVM -ProvisioningSchemeName MyScheme -ADAccountName "domain\Account"
```

TaskId	: 4b49f13a-277c-4cb0-bc40-f088430cfe8a
Active	: False
Host	: DUMTESTDDC

```
DateStarted : 18/05/2010 10:54:32
Type : NewVirtualMachine
Metadata : {}
WorkflowStatus : Completed
MasterImage : /Base.vm/Base.snapshot
ProvisioningSchemeName : MyScheme
ProvisioningSchemeUid : 7585f0de-192e-4847-a6d8-22713c3a2f42
CurrentOperation :
TaskState : Finished
TaskStateInformation :
HostingUnitUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501
HostingUnitName : XenHU
IdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16
IdentityPoolName : idPool1
VirtualMachinesToCreateCount : 1
VirtualMachinesCreatedCount : 1
VirtualMachinesCreationFailedCount : 0
CreatedVirtualMachines : {DOMAIN\Account$}
FailedVirtualMachines : {}
ProvisioningJob : 6fa5dc7c-6d49-4616-8682-aeb1580866b3
ProvisioningStatus : Completed
```

Creates a new virtual machine using the AD account "domain\account" in the provisioning scheme "MyScheme".

#### ----- EXAMPLE 2 -----

```
C:\PS>New-provVM -ProvisioningSchemeName MyScheme -ADAccountName "domain\Account" -RunAsynchronously
```

Guid

----

```
6dd85fec-96cf-46b1-9cd4-d8ba7d06e85b
```

Creates a new virtual machine using the AD account "domain\account" in the provisioning scheme "MyScheme" asynchronously. To get the task details, use `Get-ProvTask -TaskID <task id>`

i.e.

```
C:\PS>Get-ProvTask -TaskID 6dd85fec-96cf-46b1-9cd4-d8ba7d06e85b
```

```
TaskId : 4b49f13a-277c-4cb0-bc40-f088430cfe8a
```

```
Active : False
```

```
Host : MyHost
```

```
DateStarted : 18/05/2010 10:54:32
```

```
Type : NewVirtualMachine
```

```
Metadata : {}
```

```
WorkflowStatus : Completed
```

```
MasterImage : /Base.vm/Base.snapshot
```

ProvisioningSchemeName : MyScheme  
ProvisioningSchemeUid : 7585f0de-192e-4847-a6d8-22713c3a2f42  
TaskState : Finished  
TaskStateInformation :  
HostingUnitUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501  
HostingUnitName : XenHU  
IdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16  
IdentityPoolName : idPool1  
VirtualMachinesToCreateCount : 1  
VirtualMachinesCreatedCount : 1  
VirtualMachinesCreationFailedCount : 0  
CreatedVirtualMachines : {DOMAIN\Account\$}  
FailedVirtualMachines : {}  
ProvisioningJob : 6fa5dc7c-6d49-4616-8682-aeb1580866b3  
ProvisioningStatus : Completed

----- **EXAMPLE 3** -----

C:\PS>\$accounts = Get-AcctAdAccount -IdentityPool MyPool -State Available  
C:\PS>New-provVM -ProvisioningSchemeName MyScheme -ADAccountName \$t -RunAsynchronously  
Creates a new virtual machine using all of the available AD accounts from the identity pool "MyPool" in the provisioning scheme "MyScheme" asynchronously. To get the task details, use Get-ProvTask -TaskID <task id>

For example:

C:\PS>Get-ProvTask -TaskID 6dd85fec-96cf-46b1-9cd4-d8ba7d06e85b

TaskId : 4b49f13a-277c-4cb0-bc40-f088430cfe8a

Active : False

Host : MyHost

DateStarted : 18/05/2010 10:54:32

Type : NewVirtualMachine

Metadata : {}

WorkflowStatus : Completed

MasterImage : /Base.vm/Base.snapshot

ProvisioningSchemeName : MyScheme  
ProvisioningSchemeUid : 7585f0de-192e-4847-a6d8-22713c3a2f42  
TaskState : Finished  
TaskStateInformation :  
HostingUnitUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501  
HostingUnitName : XenHU  
IdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16  
IdentityPoolName : idPool1  
VirtualMachinesToCreateCount : 1  
VirtualMachinesCreatedCount : 1  
VirtualMachinesCreationFailedCount : 0  
CreatedVirtualMachines : {DOMAIN\Account\$}  
FailedVirtualMachines : {}  
ProvisioningJob : 6fa5dc7c-6d49-4616-8682-aeb1580866b3  
ProvisioningStatus : Completed

# Publish-ProvMasterVmImage

May 10, 2016

Update the master image associated with the provisioning scheme.

## Syntax

```
Publish-ProvMasterVmImage [-ProvisioningSchemeName] <String> -MasterImageVM <String> [-DoNotStoreOldImage] [-VhdTemplateSource <String>] [-VhdResultDestination <String>] [-AppScanResultsFile <String>] [-RunAsynchronously] [-PurgeJobOnSuccess] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Publish-ProvMasterVmImage -ProvisioningSchemeUid <Guid> -MasterImageVM <String> [-DoNotStoreOldImage] [-VhdTemplateSource <String>] [-VhdResultDestination <String>] [-AppScanResultsFile <String>] [-RunAsynchronously] [-PurgeJobOnSuccess] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to update the hard disk image used to provision virtual machines. If the provisioning scheme is a 'CleanOnBoot' type, then the next time that virtual machines are started, their hard disks are updated to this new image. Regardless of the 'CleanOnBoot' type, all new virtual machines created after this command will use this new hard disk image.

A background task will be created to remove the old hard disk copies. You can locate and monitor this task using the Get-ProvTask cmdlet.

A snapshot is used rather than a VM, so that the content of the hard disk for the provisioning scheme can be easily determined.

As the snapshot is specified using a path into the Citrix Host Service Powershell Provider, the Citrix Host Service Powershell snap-in must also be loaded for this cmdlet to be used.

The previous hard disk image path is stored into the history (see Get-provSchemeMasterVMImageHistory). The data stored in the history allows for a roll back to be undertaken, to revert to the previous hard disk image if required.

## Related topics

[Get-ProvSchemeMasterVMImageHistory](#)

[Get-ProvScheme](#)

## Parameters

**-ProvisioningSchemeName**<String>

The provisioning scheme to which the hard disk image should be updated.

Required?	true
Default Value	

Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

#### **-ProvisioningSchemeUid**<Guid>

The provisioning scheme identifier to which the hard disk image should be updated.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-MasterImageVM**<String>

The path in the hosting unit provider to the virtual machine snapshot that will be used. This identifies the hard disk to be used and the default values for the memory and processors. This must be a path to a Snapshot Item in the same hosting unit that the hosting unit name or hosting unit Uid refers to.

Valid paths are of the format; XDHyp:\HostingUnits\<HostingUnitName>\<path>\<VmName>.vm\  
<SnapshotName>.snapshot

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

#### **-DoNotStoreOldImage**<SwitchParameter>

Specifies that the current image should not be added to the image history list for the provisioning scheme. This is useful when rolling back to a previous image.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-VhdTemplateSource**<String>

A file path to a source VHD template to be used when performing Application Scanning during Image Preparation. The presence of this parameter in conjunction with VhdResultDestination implies that application scanning is to be performed

--	--



Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-VhdResultDestination**<String>

A file path (including file name) where the VHD disk file containing the results of application scanning should be written.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AppScanResultsFile**<String>

File name to which the results of application scanning should be written.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RunAsynchronously**<SwitchParameter>

Indicates whether or not the cmdlet should return before it is complete. If specified, the command returns an identifier for the task that was created. You can monitor this task using the get-ProvTask cmdlet.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### **-PurgeJobOnSuccess**<SwitchParameter>

Indicates that the task history will be removed from the database once the task has finished. This cannot be specified for tasks that are run asynchronously.

Required?	false
Default Value	false
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

System.Guid

When "RunAsynchronously" is specified, this identifier is returned to provide the task identifier.

System.Management.Automation.PSCustomObject

This object provides details of the task run and contains the following information:

TaskId <Guid>

The identifier for the task that was performed.

Active <Boolean>

Indicates whether the task is still processing or is complete.

Host <string>

The name of the host on which the task is running or was run.

DateStarted <DateTime>

The date and time that the task was initiated.

Type <Citrix.XDInterServiceTypes.JobType>

The type of the task. For new provisioning scheme tasks this is always "NewProvisioningScheme".

Metadata <Citrix.MachineCreation.Sdk.Metadata[]>

The list of metadata stored against the task. For new tasks this will be empty until metadata is added.

WorkflowStatus <System.Workflow.Runtime.WorkflowStatus>

Indicates the status of the workflow that is being used to process the task.

ProvisioningSchemeName <string>

The name of the provisioning scheme that the task was for.

ProvisioningSchemeUid <Guid>

The unique identifier of the provisioning scheme that the task was for.

MasterImage <string>

The path (in the hosting unit provider) of the virtual machine snapshot that was used as the master image for the task.

IdentityPoolName <string>

The name of the identity pool (from the ADIdentity PowerShell snap-in) that the new provisioning scheme will use.

IdentityPoolUid <guid>

The unique identifier name of the identity pool (from the ADIdentity PowerShell snap-in) that the new provisioning scheme will use.

HostingUnitName <string>

The name of the hosting unit (from the Hosting Unit PowerShell snap-in) that the new provisioning scheme will use.

HostingUnitUid

The unique identifier of the hosting unit (from the Hosting Unit PowerShell snap-in) that the new provisioning scheme will use.

TaskState <Citrix.DesktopUpdateManager.SDK.NewProvisioningSchemeState>

The state of the task. This can be any of the following:

Processing

Indicates that the task has just begun and has not done anything yet.

LocatingResources,

Indicates that the workflow is locating resources from other services.

HostingUnitNotFound

Indicates that the task failed because the required hosting unit could not be located.

VirtualMachineSnapshotNotFound

Indicates that the task failed because the required snapshot could not be located.

ConsolidatingMasterImage

Indicates that the task is consolidating the master image.

ReplicatingConsolidatedImageToAllStorage

Indicates that the task is replicating the consolidated master image.

StoringProvisioningScheme

Indicates that the task is storing the provisioning scheme data to the database.

Finished

Indicates that the task has completed with no errors.

ProvisioningSchemeAlreadyExists

Indicates that the task failed because a provisioning scheme with the same name already exists.

IdentityPoolNotFound

Indicates that the task failed because the identity pool specified could not be found.

MasterVMImagesNotPartOfProvisioningSchemeHostingUnit,

Indicates that the hosting unit that the master image originated from is not the same hosting unit that the provisioning scheme is defined to use.

MasterVmImagesNotASnapshot

Indicates that the task failed because the master VM Image path does not refer to a Snapshot item.

ProvisioningSchemeNotFound

Could not find a provisioning scheme with the specified name.

TaskAlreadyRunningForProvisioningScheme

A task for a provisioning scheme with the same name is already running.

InvalidMasterVMConfiguration

The VM snapshot specified as the master had an invalid configuration.

InvalidMasterVMState

The VM snapshot specified as the master is currently in an invalid state.

InsufficientResources

Indicates that the task failed because the hypervisor did not have enough resources to complete the task.

StorageNotFound

Indicates that no associated storage was found in the hosting unit.

Canceled

Indicates that the task was stopped by user intervention (using Stop-ProvTask)

TaskStateInformation <string>

Holds string data providing more details about the current task state.

TaskProgress

The progress of the task 0-100%.

Notes

Only one long running task per provisioning scheme can be processed at any one time.

In the case of failure, the following errors can result.

Error Codes

-----

JobCreationFailed

The requested task could not be started.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

WorkflowHostUnavailable

The task could not be started because the database connection is inactive.

CommunicationError

An error occurred while communicating with the service.

PermissionDenied

The user does not have administrative rights to perform this operation.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

The cmdlet is associated with a task of type PublishImage, and while active will move through the following operations (CurrentOperation field)

ValidatingInputs

ConsolidatingMasterImage

PreparingMasterImage

ReplicatingMasterImage

CommittingScheme

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Publish-ProvMasterVMImage -ProvisioningSchemeName MyScheme -MasterImageVM XDHyp:\H
ostingUnits\HostUnit1\RhoneCC_baseXP.vm\base.snapshot
```

```
TaskId : 248f102f-073f-45fe-aea9-1819a6d6dd1f
Active : False
Host : MyHost
DateStarted : 17/05/2010 17:37:57
Type : PublishImage
Metadata : {}
WorkflowStatus : Completed
ProvisioningSchemeUid : 7585f0de-192e-4847-a6d8-22713c3a2f42
ProvisioningSchemeName : MyScheme
MasterImage : /RhoneCC_baseXP.vm/base.snapshot
HostingUnitName : HostUnit1
HostingUnitUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501
ADIdentityPoolName :
ADIdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16
```

CurrentOperation :

TaskState : Finished

TaskStateInformation :

Updates the master hard disk image for the provisioning Scheme "MyScheme" to use the "base.snapshot" hard disk image.

# Remove-ProvScheme

May 10, 2016

Removes a provisioning scheme

## Syntax

```
Remove-ProvScheme [-ProvisioningSchemeName] <String> [-ForgetVM] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-ProvScheme -ProvisioningSchemeUid <Guid> [-ForgetVM] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to remove a provisioning Scheme. The provisioning scheme must not contain any VMs unless the 'ForgetVM' option is specified.

If 'ForgetVM' is not specified, a cmdlet task is created that runs in the background to remove the hard disk copies that have been created for the provisioning scheme in hypervisor storage. Use the Get-ProvTask command to monitor the progress of this task.

## Related topics

[Get-ProvTask](#)

[New-ProvScheme](#)

## Parameters

**-ProvisioningSchemeName**<String>

The name of the provisioning scheme to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ProvisioningSchemeUid**<Guid>

The unique identifier for the provisioning scheme to be removed.

Required?	true
Default Value	



Accept Pipeline Input?	true (ByPropertyName)
------------------------	-----------------------

### **-ForgetVM**<SwitchParameter>

Indicates whether or not the VMs in the provisioning scheme should be left in the hypervisor and only the data held in the Machine Creation Services removed. If this is specified, it is up to the administrator of the hypervisor to remove the VMs and hard disk images using the tools provided by the hypervisor itself.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.MachineCreation.Sdk.ProvisioningScheme

This object provides details of the provisioning scheme and contains the following information:

ProvisioningSchemeUid

The unique identifier for the provisioning scheme.

ProvisioningSchemeName

The name of the provisioning scheme.

CpuCount

The number of processors that VMs will be created with when using this scheme.

MemoryMB

The maximum amount of memory that VMs will be created with when using this scheme.

MasterImageVM

The path within the hosting unit provider to the VM or snapshot of which the scheme is currently using a copy.

MasterImageVMDate

The date and time that the copy of the VM image was made for the scheme.

IdentityPoolUid

The unique identifier of the identity pool (from the ADIdentity PowerShell snap-in) that the scheme uses.

IdentityPoolName

The name of the identity pool (from the ADIdentity PowerShell snap-in) that the scheme uses.

HostingUnitUid

The unique identifier of the hosting unit (from the Hosting Unit PowerShell snap-in) that the new provisioning scheme will use.

HostingUnitName

The name of the hosting unit (from the hosting unit PowerShell snap-in) that the new provisioning scheme will use.

CleanOnBoot

Indicates whether the VMs created are to be reset to a clean state on each boot.

TaskId

The identifier of any current task that is running for the provisioning scheme.

## Notes

If the hosting unit referenced by the provisioning scheme no longer exists (i.e. it has been removed using the Hosting Unit PowerShell snap-in), the provisioning scheme data is deleted from the database without errors. However, the hard disks associated with the provisioning scheme cannot be removed and remain in the hypervisor.

In the case of failure, the following errors can result.

## Error Codes

-----

### IllegalParameter

One or more parameters are illegal or are not specified.

### ProvisioningSchemeNotFound

The specified provisioning scheme could not be located.

### UnableToRemoveProvisioningSchemeDueToAssociatedVM

The provisioning scheme contained VMs and the 'ForgetVM' parameter was not specified.

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

### CommunicationError

An error occurred while communicating with the service.

### PermissionDenied

The user does not have administrative rights to perform this operation.

### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

The cmdlet is associated with a task of type DisusedImageCleanUp, and while active will move through the following operations (CurrentOperation field)

Running

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Remove-ProvScheme -ProvisioningSchemeName $provScheme.ProvisioningSchemeName
```

Remove the empty provisioning scheme by name.

# Remove-ProvSchemeControllerAddress

May 10, 2016  
Removes metadata from a provisioning scheme.

## Syntax

Remove-ProvSchemeControllerAddress [-ProvisioningSchemeName] <String> [-ControllerAddress] <String[]> [-LoggingId <Guid>] [-AdminAddress <String>]  
[<CommonParameters>]

Remove-ProvSchemeControllerAddress -ProvisioningSchemeUid <Guid> -ControllerAddress <String[]> [-LoggingId <Guid>] [-AdminAddress <String>]  
[<CommonParameters>]

## Detailed Description

Removes the specified controller addresses from the specified object. Attempting to remove an address not present writes an error record to the pipeline.

## Related topics

- [Get-ProvScheme](#)
- [Add-ProvSchemeControllerAddress](#)

## Parameters

**-ProvisioningSchemeName**<String>

The name of the provisioning scheme from which metadata is to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ProvisioningSchemeUid**<Guid>

The identifier of the provisioning scheme from which metadata is to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-ControllerAddress**<String[]>

Specifies the array of DNS names to be removed from the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.MachineCreation.Sdk.ProvisioningScheme You can pipe an object containing a parameter called 'ProvisioningSchemeName' to Remove-ProvSchemeMetadata.

#### Notes

In the case of failure, the following errors can result.

#### Error Codes

-----

#### ProvisioningSchemeNotFound

The specified provisioning scheme could not be located.

#### ControllerAddressNotFound

The specified address was not associated with the provisioning scheme.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

#### Examples

----- **EXAMPLE 1** -----

```
C:\PS>Get-ProvScheme -ProvisioningSchemeName scheme1 | Remove-ProvSchemeControllerAddress
```

```
ProvisioningSchemeUid : 7585f0de-192e-4847-a6d8-22713c3a2f42
```

```
ProvisioningSchemeName : Scheme1
```

```
CpuCount : 1
```

```
MemoryMB : 1024
MasterImageVM : Base.vm/Base.snapshot
MasterImageVMDate : 17/05/2010 09:53:40
IdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16
IdentityPoolName : idPool1
HostingUnitUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501
HostingUnitName : HostUnit1
CleanOnBoot : True
TaskId : 00000000-0000-0000-0000-000000000000
Metadata : {}
ControllerAddress : {}
Remove all controller addresses from the provisioning scheme with the name "scheme1".
```

----- **EXAMPLE 2** -----

```
C:\PS>Remove-ProvSchemeControllerAddress -ProvisioningSchemeUid "01a4a008-8ce8-4165-ba9c-cdf15a6b0501" -ControllerAddress (ddcA.citrix.com,ddcC.citrix2.com)
```

```
ProvisioningSchemeUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501
ProvisioningSchemeName : Scheme2
CpuCount : 1
MemoryMB : 1024
MasterImageVM : Base.vm/Base.snapshot
MasterImageVMDate : 17/05/2010 09:53:40
IdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16
IdentityPoolName : idPool1
HostingUnitUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501
HostingUnitName : HostUnit1
CleanOnBoot : True
TaskId : 00000000-0000-0000-0000-000000000000
Metadata : {}
ControllerAddress : {ddcB.citrix.com}
Remove a subset of the controller address list from the provisioning scheme with the identifier "01a4a008-8ce8-4165-ba9c-cdf15a6b0501".
```

# Remove-ProvSchemeMasterVMImageHistory

May 10, 2016

Removes the history of provisioning scheme master image VMs.

## Syntax

```
Remove-ProvSchemeMasterVMImageHistory [-ProvisioningSchemeName] <String> [-VMImageHistoryUid <Guid>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ProvSchemeMasterVMImageHistory -ProvisioningSchemeUid <Guid> [-VMImageHistoryUid <Guid>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ProvSchemeMasterVMImageHistory -VMImageHistoryUid <Guid> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to remove the record of previously used master VMs or snapshots for provisioning schemes.

## Related topics

[Get-ProvSchemeMasterVMImageHistory](#)

[Publish-ProvMasterVMImage](#)

## Parameters

**-ProvisioningSchemeName**<String>

The name of the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ProvisioningSchemeUid**<Guid>

The unique identifier of the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	false



### **-VMImageHistoryUid**<Guid>

The unique identifier for the Image History item.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Notes

In the case of failure, the following errors can result.

#### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Remove-ProvSchemeMasterVMImageHistory -ProvisioningSchemeName MyScheme
```

Removes all history for the provisioning scheme called "MyScheme".

##### ----- EXAMPLE 2 -----

```
c:\PS>Get-ProvScheme | Remove-ProvSchemeMasterVMImageHistory
```

Removes all history for all provisioning schemes.

# Remove-ProvSchemeMetadata

May 10, 2016

Removes metadata from the given ProvisioningScheme.

## Syntax

```
Remove-ProvSchemeMetadata [-ProvisioningSchemeUid] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ProvSchemeMetadata [-ProvisioningSchemeUid] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ProvSchemeMetadata [-ProvisioningSchemeName] <String> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ProvSchemeMetadata [-ProvisioningSchemeName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ProvSchemeMetadata [-InputObject] <ProvisioningScheme[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ProvSchemeMetadata [-InputObject] <ProvisioningScheme[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given ProvisioningScheme.

## Related topics

[Add-ProvSchemeMetadata](#)

[Set-ProvSchemeMetadata](#)

## Parameters

**-ProvisioningSchemeUid**<Guid>

Id of the ProvisioningScheme

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-ProvisioningSchemeName**<String>

Name of the ProvisioningScheme

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

#### **-InputObject**<ProvisioningScheme[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create

high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-ProvScheme | Remove-ProvSchemeMetadata
Remove all metadata from all ProvisioningScheme objects.
```

# Remove-ProvSchemeScope

May 10, 2016

Remove the specified ProvisioningScheme(s) from the given scope(s).

## Syntax

```
Remove-ProvSchemeScope [-Scope] <String[]> -InputObject <ProvisioningScheme[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ProvSchemeScope [-Scope] <String[]> -ProvisioningSchemeUid <Guid[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ProvSchemeScope [-Scope] <String[]> -ProvisioningSchemeName <String[]> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The RemoveProvSchemeScope cmdlet is used to remove one or more ProvisioningScheme objects from the given scope(s).

There are multiple parameter sets for this cmdlet, allowing you to identify the ProvisioningScheme objects in different ways:

- ProvisioningScheme objects can be piped in or specified by the InputObject parameter
- The ProvisioningSchemeUid parameter specifies objects by ProvisioningSchemeUid
- The ProvisioningSchemeName parameter specifies objects by ProvisioningSchemeName (supports wildcards)

To remove a ProvisioningScheme from a scope you need permission to change the scopes of the ProvisioningScheme.

If the ProvisioningScheme is not in a specified scope, that scope will be silently ignored.

## Related topics

[Add-ProvSchemeScope](#)

[Get-ProvScopedObject](#)

## Parameters

**-Scope**<String[]>

Specifies the scopes to remove the objects from.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-InputObject**<ProvisioningScheme[]>

Specifies the ProvisioningScheme objects to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-ProvisioningSchemeUid**<Guid[]>

Specifies the ProvisioningScheme objects to be removed by ProvisioningSchemeUid.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-ProvisioningSchemeName**<String[]>

Specifies the ProvisioningScheme objects to be removed by ProvisioningSchemeName.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or



an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

None

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

### UnknownObject

One of the specified objects was not found.

### ScopeNotFound

One of the specified scopes was not found.

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

### PermissionDenied

You do not have permission to execute this command with the specified objects or scopes.

### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

### CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Remove-ProvSchemeScope Finance -ProvisioningSchemeUid 6702C5D0-C073-4080-A0EE-EC74CB537C52
```

Removes a single ProvisioningScheme from the 'Finance' scope.

### ----- EXAMPLE 2 -----

```
c:\PS>Remove-ProvSchemeScope Finance,Marketing -ProvisioningSchemeUid 6702C5D0-C073-4080-A0EE-EC74CB537C52
```

Removes a single ProvisioningScheme from multiple scopes.

### ----- EXAMPLE 3 -----

```
c:\PS>Get-ProvScheme | Remove-ProvSchemeScope Finance
```

Removes all visible ProvisioningScheme objects from the 'Finance' scope.

### ----- EXAMPLE 4 -----

```
c:\PS>Remove-ProvSchemeScope Finance -ProvisioningSchemeName A*
```

Removes ProvisioningScheme objects with a name starting with an 'A' from the 'Finance' scope.

# Remove-ProvServiceConfigurationData

May 10, 2016

Removes configuration data from the service.

## Syntax

```
Remove-ProvServiceConfigurationData [[-Name] <String>] [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to remove data from the MachineCreation Service configuration data.

## Related topics

[Set-ProvServiceConfigurationData](#)

[Get-ProvServiceConfigurationData](#)

## Parameters

**-Name**<String>

The name of the configuration data item to remove.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name

or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

In the case of failure the following errors can be produced.

### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

An error occurred while communicating with the service.

ExceptionThrown An unexpected error occurred. To locate more details see the Windows event logs on the controller being used, or examine the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
C:\PS>Remove-ProvServiceConfigurationData -Name "MyName"
```

Removes the configuration data item with name "MyName".

# Remove-ProvServiceMetadata

May 10, 2016

Removes metadata from the given Service.

## Syntax

```
Remove-ProvServiceMetadata [-ServiceHostId] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-ProvServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-ProvServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-ProvServiceMetadata [-InputObject] <Service[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Service.

## Related topics

[Set-ProvServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name**<String>

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples



----- **EXAMPLE 1** -----

```
c:\PS>Get-ProvService | % { Remove-ProvServiceMetadata -Map $_.MetadataMap }
```

Remove all metadata from all Service objects.

# Remove-ProvTask

May 10, 2016

Removes from the database completed tasks for the MachineCreation Service.

## Syntax

```
Remove-ProvTask [-TaskId] <Guid> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables completed tasks that have run within the MachineCreation Service to be removed from the database.

## Related topics

[Get-ProvTask](#)

[Add-ProvTaskMetadata](#)

[Remove-ProvTaskMetadata](#)

## Parameters

**-TaskId**<Guid>

Specifies the identifier for the task to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

System.Management.Automation.PSObject Objects containing the TaskId parameter can be piped to the Remove-ProvTask command.

## Notes

If the command fails, the following errors can result.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### InvalidWorkflow

The specified task could not be found.

#### InvalidWorkflowState

The task was not in an appropriate state for the requested operation.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-ProvTask -active $false | Remove-ProvTask
```

Success

Remove entries for all completed tasks from the MachineCreation Service.

# Remove-ProvTaskMetadata

May 10, 2016

Removes metadata from the given Task.

## Syntax

```
Remove-ProvTaskMetadata [-TaskId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-ProvTaskMetadata [-TaskId] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-ProvTaskMetadata [-InputObject] <Task[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-ProvTaskMetadata [-InputObject] <Task[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Task.

## Related topics

[Add-ProvTaskMetadata](#)

[Set-ProvTaskMetadata](#)

[Get-ProvTask](#)

[Remove-ProvTask](#)

[Stop-ProvTask](#)

[Switch-ProvTask](#)

## Parameters

**-TaskId**<Guid>

Id of the Task

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Task[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- **EXAMPLE 1** -----

```
c:\PS>Get-ProvTask | Remove-ProvTaskMetadata
```

Remove all metadata from all Task objects.



# Remove-ProvVM

May 10, 2016

Removes virtual machines.

## Syntax

```
Remove-ProvVM [-ProvisioningSchemeName] <String> -VMName <String[]> [-ForgetVM] [-RunAsynchronously] [-PurgeJobOnSuccess] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Remove-ProvVM [-ProvisioningSchemeUid] <Guid> -VMName <String[]> [-ForgetVM] [-RunAsynchronously] [-PurgeJobOnSuccess] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Lets you remove VMs from the Machine Creation Services and the hypervisor that they run on.

## Related topics

[Get-ProvVM](#)

[New-ProvVM](#)

## Parameters

**-ProvisioningSchemeName**<String>

The name of the provisioning scheme from which virtual machines will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ProvisioningSchemeUid**<Guid>

The unique identifier for the provisioning scheme from which the virtual machines are removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-VMName**<String[]>

List of VM names that will be removed from the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-ForgetVM**<SwitchParameter>

The named VMs will only be removed from the provisioning scheme database, but will remain in the hypervisor.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-RunAsynchronously**<SwitchParameter>

Indicates whether or not the command returns before it is complete. If this is specified, the command returns an identifier for the task that was created. This task can be monitored using the get-ProvTask command.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-PurgeJobOnSuccess**<SwitchParameter>

Indicates that the task history is removed from the database when the task finishes. This cannot be specified for tasks that are run asynchronously.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. When a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

##### System.Guid

When the RunAsynchronously identifier is specified, this GUID is returned and provides the task identifier.

##### System.Management.Automation.PSCustomObject

This object provides details of the task that was run and contains the following information:

##### TaskId <Guid>

The identifier for the task that was performed.

##### Active <Boolean>

Indicates whether the task is still processing or is complete.

##### Host <string>

The name of the host on which the task is running or was run.

##### DateStarted <DateTime>

The date and time that the task was initiated.

##### Type <Citrix.XDInterServiceTypes.JobType>

The type of task. For new remove-VM tasks, this is always "RemoveVirtualMachine".

Metadata <Citrix.MachineCreation.Sdk.Metadata[]>

The list of metadata stored for the task. For new tasks, this is empty until metadata is added.

WorkflowStatus <System.Workflow.Runtime.WorkflowStatus>

Indicates the status of the workflow that is used to process the task.

ProvisioningSchemeName <string>

The name of the provisioning scheme that the task is for.

ProvisioningSchemeUid <Guid>

The unique identifier of the provisioning scheme that the task is for.

TaskState <Citrix.DesktopUpdateManager.SDK.ProvisionVMState>

The state of the task. This can be any of the following:

Processing

Indicates that the task is in its initial state.

LocatingResources,

Indicates that the task is locating information from other services.

HostingUnitNotFound

Indicates that the required hosting unit could not be located.

ProvisioningSchemeNotFound

Indicates that the required provisioning scheme could not be located.

Provisioning

Indicates that the task is at the provisioning stage.

IdentityPoolNotFound

Indicates that the required identity pool could not be located.

TaskAlreadyRunningForProvisioningScheme

Indicates that the provisioning scheme already has another task running.

Finalizing

Indicates that the task is finalizing.

Finished

Indicates that the task is complete.

FinishedWithErrors

Indicates that the task is complete but there were errors. Specific details of errors are included with each failed virtual machine.

Removing

Indicates that the task is removing virtual machines from the hypervisor.

Failed

Job failed for reasons specified in TaskStateInformation.

Canceled

Indicates that the task was stopped by user intervention (using Stop-ProvTask)

TaskStateInformation

Provides more detailed information about the current task state.

VirtualMachinesToRemoveCount <int>

The total number of virtual machines that the task is trying to remove.

VirtualMachinesRemovedCount <int>

The number of virtual machines that the task has removed so far. Details of the machines that have been removed are in the RemovedVirtualMachines parameter.

VirtualMachinesFailedCount <int>

The number of virtual machines that the task has failed to remove. Details of the machines that failed are in the FailedVirtualMachines parameter.

FailedVirtualMachines <Citrix.DesktopUpdateManager.SDK.VirtualMachineCreation[]>

ADAccountName <string>

The domain-qualified AD Account name of the machine.

ADAccountSid <string>

The AD account SID of the machine account.

DiagnosticInformation <Citrix.MachineCreation.Sdk.ExceptionSurrogate[]>

A collection of handled error states which caused the provisioning to fail.

ExceptionType <string>

The type of exception this object represents

Message <string>

The exception message

Details <string>

The full exception content including stack trace

InnerException <Citrix.MachineCreation.Sdk.ExceptionSurrogate>

Information relating to any contributing error state

Status <string>

StatusAdditionalInformation <string>

VMId <string>

The virtual machine identifier in the hypervisor.

VMName <string>

The display name of the virtual machine in the hypervisor.

RemovedVirtualMachines <Citrix.DesktopUpdateManager.SDK.VirtualMachineCreation[]>

See FailedVirtualMachines for details of the object parameters.

ProgressEstimator

Gives an estimate of the number of virtual machines processed, averaging over virtual machines that were both partly and completely processed.

## Notes

Only one long-running task for each provisioning scheme can be processed at a time.

This task still operates if the hosting unit or VMs in the hypervisor are missing. This removes the data from the Citrix Machine Creation Services, but the VMs remain in the hypervisor.

In the case of failure, the following errors can result.

## Error Codes

-----

### JobCreationFailed

The requested task could not be started.

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation. Communication with the database failed for various reasons.

#### WorkflowHostUnavailable

The task could not be started because the database connection is inactive.

#### CommunicationError

An error occurred while communicating with the service.

#### InvalidParameterCombination

Both `PurgeJobOnSuccess` and `RunAsynchronously` were specified. When running asynchronously, the cmdlet terminates before the job does, so it cannot clean up the completed job.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

The cmdlet is associated with a task of type `RemoveVirtualMachine`, and while active will move through the following operations (`CurrentOperation` field)

#### ValidatingInputs

#### RemovingVirtualMachines

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>,(Get-ProvVM -ProvisioningSchemeName XenPS) | Remove-ProvVM XenPS
```

```
TaskId : cfb506a5-cc7e-4a49-ac7b-dd960029d0d3
Active : False
Host : DDC
DateStarted : 10/10/2012 16:39:45
Metadata : {}
WorkflowStatus : Completed
ProvisioningSchemeUid : e1afc8fb-3f52-42ea-9a17-305fdb0b6ee4
ProvisioningSchemeName : XenPS
TaskState : Finished
```

TaskStateInformation :  
VirtualMachinesToRemoveCount : 2  
RemovedVirtualMachines : {XD\IP0001\$, XD\IP0002\$}  
FailedVirtualMachines : {}  
VirtualMachinesRemovedCount : 2  
VirtualMachinesFailedCount : 0  
ProgressEstimator : 2  
Type : RemoveVirtualMachine  
Status : Finished  
CurrentOperation :  
TaskProgress : 100  
TaskExpectedCompletion : 10/10/2012 16:39:50  
LastUpdateTime : 10/10/2012 16:39:50  
ActiveElapsedTime : 5  
DateFinished : 10/10/2012 16:39:50  
TerminatingError :  
Remove all VMs from provisioning scheme XenPS



# Rename-ProvScheme

May 10, 2016

Renames a provisioning scheme.

## Syntax

```
Rename-ProvScheme [-ProvisioningSchemeName] <String> [-NewProvisioningSchemeName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Rename-ProvScheme -ProvisioningSchemeUid <Guid> [-NewProvisioningSchemeName] <String> [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to change the name of an existing provisioning scheme. The unique identifier for the provisioning scheme never changes after it has been created so, regardless of any name changes, the provisioning scheme can always be identified by its unique identifier.

## Related topics

[New-ProvScheme](#)

[Set-ProvScheme](#)

[Get-ProvScheme](#)

[Test-ProvSchemeNameAvailable](#)

## Parameters

**-ProvisioningSchemeName**<String>

The current name of the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-ProvisioningSchemeUid**<Guid>

The identifier for the provisioning scheme that is to be renamed.

Required?	true
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-NewProvisioningSchemeName**<String>

The new name for the provisioning scheme. This must be a name that is not being used by an existing provisioning scheme, and it must not contain any of the following characters \/:#.\*?=<>|[]()''''

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-PassThru**<SwitchParameter>

Defines whether or not the command returns a result showing the new state of the updated identity pool.

Required?	false
Default Value	true
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

Citrix.MachineCreation.Sdk.ProvisioningScheme

This object provides details of the provisioning scheme and contains the following information:

ProvisioningSchemeUid

The unique identifier for the provisioning scheme.

ProvisioningSchemeName

The name of the provisioning scheme.

CpuCount

The number of processors that VMs will be created with when using this scheme.

MemoryMB

The maximum amount of memory that VMs will be created with when using this scheme.

MasterImageVM

The path within the hosting unit provider to the VM or snapshot of which the scheme is currently using a copy.

MasterImageVMDate

The date and time that the copy of the VM image was made for the scheme.

IdentityPoolUid

The unique identifier of the identity pool (from the ADIdentity PowerShell snap-in) that the scheme uses.

IdentityPoolName

The name of the identity pool (from the ADIdentity PowerShell snap-in) that the scheme uses.

HostingUnitUid

The unique identifier of the hosting unit (from the Hosting Unit PowerShell snap-in) that the new provisioning scheme will use.

HostingUnitName

The name of the hosting unit (from the Hosting Unit PowerShell snap-in) that the new provisioning scheme will use.

CleanOnBoot

Indicates whether the VMs created are to be reset to a clean state on each boot.

TaskId

The identifier of any current task that is running for the provisioning scheme.

## Metadata

The metadata for the provisioning scheme.

## Notes

In the case of failure, the following errors can result.

## Error Codes

-----

### ProvisioningSchemeNotFound

The specified provisioning scheme could not be located.

### ProvisioningSchemeNameAlreadyExists

A provisioning scheme with the same name as the new provisioning scheme name already exists.

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

### CommunicationError

An error occurred while communicating with the service.

### PermissionDenied

The user does not have administrative rights to perform this operation.

### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

### ExceptionThrown

An unexpected error occurred. To locate more details, see the windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

C:\PS>Rename-ProvScheme -provisioningSchemeName CurrentName -NewProvisioningSchemeName NewName  
Renames a provisioning scheme from "currentName" to "NewName".

----- **EXAMPLE 2** -----

```
C:\PS>Rename-ProvScheme -provisioningSchemeName CurrentName -NewProvisioningSchemeName NewName -PassThru
```

```
ProvisioningSchemeUid : 7585f0de-192e-4847-a6d8-22713c3a2f42
```

```
ProvisioningSchemeName : NewName
```

```
CpuCount : 1
```

```
MemoryMB : 1024
```

```
MasterImageVM : /Base.vm
```

```
MasterImageVMDate : 17/05/2010 08:27:50
```

```
IdentityPoolUid : 03743136-e43b-4a87-af74-ab71686b3c16
```

```
IdentityPoolName : IdPool1
```

```
HostingUnitUid : 01a4a008-8ce8-4165-ba9c-cdf15a6b0501
```

```
HostingUnitName : XenHU
```

```
CleanOnBoot : True
```

```
TaskId :
```

```
Metadata : {}
```

Renames a provisioning scheme from "currentName" to "NewName" and displays the resulting state.

# Reset-ProvServiceGroupMembership

May 10, 2016  
Reloads the access permissions and configuration service locations for the MachineCreation Service.

## Syntax

Reset-ProvServiceGroupMembership [-ConfigServiceInstance] <ServiceInstance[]> [-LoggingId <Guid>] [-AdminAddress <String>]  
[<CommonParameters>]

## Detailed Description

Enables you to reload MachineCreation Service access permissions and configuration service locations. The Reset-ProvServiceGroupMembership command must be run on at least one instance of the service type (Prov) after installation and registration with the configuration service. Without this operation, the MachineCreation services will be unable to communicate with other services in the XenDesktop deployment. When the command is run, the services are updated when additional services are added to the deployment, provided that the configuration service is not stopped. The Reset-ProvServiceGroupMembership command can be run again to refresh this information if automatic updates do not occur when new services are added to the deployment. If more than one configuration service instance is passed to the command, the first instance that meets the expected service type requirements is used.

## Related topics

## Parameters

**-ConfigServiceInstance**<ServiceInstance[]>

Specifies the configuration service instance object that represents the service instance for the type 'InterService' that references a configuration service for the deployment.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

Citrix.MachineCreation.Sdk.ServiceInstance[] Service instances containing a ServiceInstance object that refers to the central configuration service interservice interface can be piped to the Reset-ProvServiceGroupMembership command.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### NoSuitableServiceInstance

None of the supplied service instance objects were suitable for resetting service group membership.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config | Reset-ProvServiceGroupMembership
```

Reset the service group membership for a service in a deployment where the configuration service is configured and running on the same machine as the service.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config -AdminAddress OtherServer.example.com | Reset-ProvServiceGroupmembership
```

Reset the service group membership for a service in a deployment where the configuration service that is configured and running on a machine named 'OtherServer.example.com'.



# Set-ProvDBConnection

May 10, 2016  
Configures a database connection for the MachineCreation Service.

## Syntax

Set-ProvDBConnection [-DBConnection] <String> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Configures a connection to a database in which the MachineCreation Service can store its state. The service will attempt to connect and start using the database immediately after the connection is configured. The database connection string is updated to the specified value regardless of whether it is valid or not. Specifying an invalid connection string prevents a service from functioning until the error is corrected.

After a connection is configured, you cannot alter it without first clearing it (by setting the connection to \$null).

You do not need to configure a database connection to use this command.

## Related topics

- [Get-ProvServiceStatus](#)
- [Get-ProvDBConnection](#)
- [Test-ProvDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be used by the MachineCreation Service. Passing in \$null will clear any existing database connection configured.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Force**<SwitchParameter>

If present, allows the local administrator to set the connection string to null when there are problems contacting the database or other services.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and

Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Set-ProvDBConnection command returns an object containing the status of the MachineCreation Service together with extra diagnostics information.

DBUnconfigured

The MachineCreation Service does not have a database connection configured.

DBRejectedConnection

The database rejected the logon attempt from the MachineCreation Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the MachineCreation Service schema has not been added to the database.

DBNotFound

The specified database could not be located with the configured connection string.

DBNewerVersionThanService

The version of the MachineCreation Service currently in use is incompatible with the version of the MachineCreation Service schema on the database. Upgrade the MachineCreation Service to a more recent version.

DBOlderVersionThanService

The version of the MachineCreation Service schema on the database is incompatible with the version of the MachineCreation Service currently in use. Upgrade the database schema to a more recent version.

DBVersionChangeInProgress

A database schema upgrade is currently in progress.

OK

The MachineCreation Service is running and is connected to a database containing a valid schema.

Failed

The MachineCreation Service has failed.

Unknown

The status of the MachineCreation Service cannot be determined.

Notes

If the command fails, the following errors can be returned.

Error Codes

-----

InvalidDBConnectionString

The database connection string has an invalid format.

DatabaseConnectionDetailsAlreadyConfigured

There was already a database connection configured. After a configuration is set, it can only be set to \$null.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Set-ProvDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Configures a database connection string for the MachineCreation Service.

----- **EXAMPLE 2** -----

```
c:\PS>Set-ProvDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Configures an invalid database connection string for the MachineCreation Service.

# Set-ProvScheme

May 10, 2016

Changes the parameter values for a provisioning scheme.

## Syntax

```
Set-ProvScheme [-ProvisioningSchemeName] <String> [-VMCpuCount <Int32>] [-VMMemoryMB <Int32>] [-ServiceOffering <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-ProvScheme [-ProvisioningSchemeName] <String> -IdentityPoolUid <Guid> [-VMCpuCount <Int32>] [-VMMemoryMB <Int32>] [-ServiceOffering <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-ProvScheme [-ProvisioningSchemeName] <String> -IdentityPoolName <String> [-VMCpuCount <Int32>] [-VMMemoryMB <Int32>] [-ServiceOffering <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-ProvScheme -ProvisioningSchemeUid <Guid> [-VMCpuCount <Int32>] [-VMMemoryMB <Int32>] [-ServiceOffering <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-ProvScheme -ProvisioningSchemeUid <Guid> -IdentityPoolName <String> [-VMCpuCount <Int32>] [-VMMemoryMB <Int32>] [-ServiceOffering <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-ProvScheme -ProvisioningSchemeUid <Guid> -IdentityPoolUid <Guid> [-VMCpuCount <Int32>] [-VMMemoryMB <Int32>] [-ServiceOffering <String>] [-PassThru] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability to update the parameters of an existing provisioning scheme.

This allows the following parameters to be updated:

Number of CPUs that will be used for VMs created from the provisioning scheme. Maximum amount of memory that will be used for VMs created from the provisioning scheme.

To change the name of the provisioning scheme see [Rename-ProvScheme](#).

## Related topics

[New-ProvScheme](#)

[Remove-ProvScheme](#)

[Get-ProvScheme](#)

[Rename-ProvScheme](#)

## Parameters

**-ProvisioningSchemeName**<String>

The name of the provisioning scheme to be updated.

Required?	true
-----------	------

Default Value	
Accept Pipeline Input?	false

#### **-IdentityPoolUid**<Guid>

The identifier of an identity pool to associate with the provisioning scheme, replacing the present one.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-IdentityPoolName**<String>

The name of an identity pool to associate with the provisioning scheme, replacing the present one.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-ProvisioningSchemeUid**<Guid>

The identifier of the provisioning scheme to be updated.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-VMCpuCount**<Int32>

The number of processors that virtual machines created from the provisioning scheme should use.

Required?	false

Default Value	
Accept Pipeline Input?	false

#### **-VMMemoryMB<Int32>**

The maximum amount of memory that virtual machines created from the provisioning scheme should use.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ServiceOffering<String>**

The name of the cloud service offering to use when creating machines.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-PassThru<SwitchParameter>**

Returns the affected record. By default, this cmdlet does not generate any output.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.MachineCreation.Sdk.ProvisioningScheme

This object provides details of the provisioning scheme and contains the following information:

ProvisioningSchemeUid

ProvisioningSchemeName

The name of the provisioning scheme.

CpuCount

The number of processors that VMs will be created with when using this scheme.

MemoryMB

The maximum amount of memory that VMs will be created with when using this scheme.

MasterImageVM

The path within the hosting unit provider to the VM or snapshot of which the scheme is currently using a copy.

MasterImageVMDate

The date and time that the copy of the VM image was made for the scheme.

IdentityPoolUid

The unique identifier of the identity pool (from the ADIdentity PowerShell snap-in) that the scheme uses.

IdentityPoolName

The name of the identity pool (from the ADIdentity PowerShell snap-in) that the scheme uses.

## HostingUnitUid

The unique identifier of the hosting unit (from the Hosting Unit PowerShell snap-in) that the new provisioning scheme will use.

## HostingUnitName

The name of the hosting unit (from the Hosting Unit PowerShell snap-in) that the new provisioning scheme will use.

## CleanOnBoot

Indicates whether the VMs created are to be reset to a clean state on each boot.

## TaskId

The identifier of any current task that is running for the provisioning scheme.

## Notes

In the case of failure, the following errors can result.

## Error Codes

-----

### ProvisioningSchemeNotFound

The specified provisioning scheme could not be located.

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

### CommunicationError

An error occurred while communicating with the service.

### PermissionDenied

The user does not have administrative rights to perform this operation.

### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

### HostingUnitNotFound



The hosting unit referenced by the provisioning scheme could not be resolved

ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

C:\PS> Set-ProvScheme -ProvisioningSchemeName MyScheme -VMCpuCount 2

Updates a provisioning scheme called "MyScheme" to use two processors on the VMs that are created from the provisioning scheme.

# Set-ProvSchemeMetadata

May 10, 2016

Adds or updates metadata on the given ProvisioningScheme.

## Syntax

```
Set-ProvSchemeMetadata [-ProvisioningSchemeUid] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-ProvSchemeMetadata [-ProvisioningSchemeUid] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Set-ProvSchemeMetadata [-ProvisioningSchemeName] <String> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-ProvSchemeMetadata [-ProvisioningSchemeName] <String> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Set-ProvSchemeMetadata [-InputObject] <ProvisioningScheme[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-ProvSchemeMetadata [-InputObject] <ProvisioningScheme[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

## Detailed Description

Provides the ability for additional custom data to be stored against given ProvisioningScheme objects.

## Related topics

[Add-ProvSchemeMetadata](#)

[Remove-ProvSchemeMetadata](#)

## Parameters

**-ProvisioningSchemeUid**<Guid>

Id of the ProvisioningScheme

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-ProvisioningSchemeName**<String>

Name of the ProvisioningScheme

Required?	true
Default Value	

Accept Pipeline Input?	true (ByValue, ByPropertyName)
------------------------	--------------------------------

#### **-InputObject**<ProvisioningScheme[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Map**<PObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the ProvisioningScheme specified. The property cannot contain any of the following characters \;#.\*?=<>|[]()"

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-ProvSchemeMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Set-ProvSchemeMetadata -ProvisioningSchemeUid 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
---	-----
property	value

Add metadata with a name of 'property' and a value of 'value' to the ProvisioningScheme with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Set-ProvServiceConfigurationData

May 10, 2016

Sets the value for the given key in the service configuration data.

## Syntax

```
Set-ProvServiceConfigurationData [-Name] <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability for additional custom data to be stored for the MachineCreation Service.

## Related topics

[Remove-ProvServiceConfigurationData](#)

[Get-ProvServiceConfigurationData](#)

## Parameters

**-Name**<String>

Specifies the key name of the metadata to be added. The key must be unique.

The Name cannot contain any of the following characters \/:#.\*?=<> | []()''''

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Value**<String>

Specifies the value for the name. If the name already exists, its value is updated.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create

high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

Citrix.MachineCreation.Sdk.ServiceConfigurationData

Set-ProvServiceConfigurationData returns an object containing the new definition of the configuration.

Name <string>

Specifies the name for the item of data.

Value <string>

Specifies the value of the data.

#### Notes

In the case of failure the following errors can be produced.

#### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

## CommunicationError

An error occurred while communicating with the service.

## ExceptionThrown

An unexpected error occurred. To locate more details see the Windows event logs on the controller being used, or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS>Set-ProvServiceConfigurationData -Name "customProperty1" -Value "value2"
```

Name	Value
-----	-----
customProperty1	value2

Set data with a name of 'customProperty1' and value of 'value2' to the service configuration.



# Set-ProvServiceMetadata

May 10, 2016

Adds or updates metadata on the given Service.

## Syntax

```
Set-ProvServiceMetadata [-ServiceHostId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-ProvServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-ProvServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-ProvServiceMetadata [-InputObject] <Service[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Allows you to store additional custom data against given Service objects.

## Related topics

[Remove-ProvServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the Service specified. The property cannot contain any of the following characters \/:#.\*?=<>|[]()''

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-ProvServiceMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

InvalidParameterCombination

The cmdlet parameters are inconsistent.

UnknownObject

One of the specified objects was not found.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- **EXAMPLE 1** -----

```
c:\PS>Set-ProvServiceMetadata -ServiceHostId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
---	----
property	value

Add metadata with a name of 'property' and a value of 'value' to the Service with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Set-ProvTaskMetadata

May 10, 2016

Adds or updates metadata on the given Task.

## Syntax

```
Set-ProvTaskMetadata [-TaskId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-ProvTaskMetadata [-TaskId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-ProvTaskMetadata [-InputObject] <Task[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-ProvTaskMetadata [-InputObject] <Task[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Use this cmdlet to store additional custom data against given Task objects.

## Related topics

[Add-ProvTaskMetadata](#)

[Remove-ProvTaskMetadata](#)

[Get-ProvTask](#)

[Stop-ProvTask](#)

[Remove-ProvTask](#)

[Switch-ProvTask](#)

## Parameters

**-TaskId**<Guid>

Id of the Task

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Task[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name<String>**

Specifies the property name of the metadata to be added. The property must be unique for the Task specified. The property cannot contain any of the following characters \ ; # . \* ? = < > | [ ] () ""

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Value<String>**

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-ProvTaskMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

### Notes

If the command fails, the following errors can result.

#### Error Codes

-----

InvalidParameterCombination

The cmdlet parameters are inconsistent.

## UnknownObject

One of the specified objects was not found.

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Set-ProvTaskMetadata -TaskId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
---	----
property	value

Add metadata with a name of 'property' and a value of 'value' to the Task with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.



# Stop-ProvTask

May 10, 2016

Stops currently running MachineCreation Service tasks.

## Syntax

```
Stop-ProvTask [-TaskId] <Guid> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables tasks currently running in the MachineCreation Service to be stopped. Once stopped, tasks cannot be restarted.

## Related topics

[Get-ProvTask](#)

[Remove-ProvTask](#)

[Switch-ProvTask](#)

## Parameters

**-TaskId**<Guid>

Specifies the identifier for the task to be stopped.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

System.Management.Automation.PSObject Objects containing the TaskId parameter can be piped to the Stop-ProvTask command.

## Notes

If the command fails, the following errors can result.

### Error Codes

-----

#### InvalidWorkflow

The specified task could not be found.

#### InvalidWorkflowHost

The specified task is executing on a different server.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

DatabaseError

There was a problem communicating with the database.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-ProvTask -active $true | Stop-ProvTask
```

Success

Terminate all tasks currently executing on the MachineCreation Service.

----- **EXAMPLE 2** -----

```
c:\PS>Stop-ProvTask -TaskId bd52e688-e40d-4790-83de-9f7633481454
```

Success

Terminate the named task executing on the MachineCreation Service.

# Switch-ProvTask

May 10, 2016

Moves all MachineCreation Service tasks from the current execution host to another.

## Syntax

```
Switch-ProvTask [-Host2] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables tasks running within the MachineCreation Service to be moved from one execution host to a different host.

Tasks can only execute on a single host. If the host is removed from a deployment, the tasks cannot continue to execute. These 'orphaned' tasks can be moved to a different host within the deployment so that they can continue to execute. All tasks must be moved; there is no mechanism to move individual tasks. Run the Switch-ProvTask command against the host to which the tasks are to be moved; that is, if you are not running the command directly on the host, use the AdminAddress parameter to specify the host to which tasks will be moved.

## Related topics

[Get-ProvTask](#)

[Stop-ProvTask](#)

[Remove-ProvTask](#)

## Parameters

**-Host2**<String>

Specifies the host from which the tasks should be removed.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Notes

If the command fails, the following errors can result.

#### Error Codes

-----

#### PartialData

Only a subset of the requested data was returned.

#### NoOp

The operation was successful but had no effect.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Switch-ProvTask -Host CRASHED

Success

Transfer the execution of tasks that had been executing on the MachineCreation Service instance on host CRASHED to the local instance.

# Test-ProvDBConnection

May 10, 2016

Tests a database connection for the MachineCreation Service.

## Syntax

```
Test-ProvDBConnection [-DBConnection] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Tests a connection to the database in which the MachineCreation Service can store its state. The service will attempt to connect to the database without affecting the current connection to the database.

You do not have to clear the connection to use this command.

## Related topics

[Get-ProvServiceStatus](#)

[Get-ProvDBConnection](#)

[Set-ProvDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be tested by the MachineCreation Service.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

### Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Test-ProvDBConnection command returns an object containing the status of the MachineCreation Service if the connection string of the specified data store were to be set to the string being tested, together with extra diagnostics information for the specified connection string.

### DBRejectedConnection

The database rejected the logon attempt from the MachineCreation Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

### InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the MachineCreation Service schema has not been added to the database.

### DBNotFound

The specified database could not be located with the configured connection string.

### DBNewerVersionThanService

The version of the MachineCreation Service currently in use is incompatible with the version of the MachineCreation Service schema on the database. Upgrade the MachineCreation Service to a more recent version.

### DBOlderVersionThanService

The version of the MachineCreation Service schema on the database is incompatible with the version of the MachineCreation Service currently in use. Upgrade the database schema to a more recent version.

### DBVersionChangeInProgress

A database schema upgrade is currently in progress.

### OK

The Set-ProvDBConnection command would succeed if it were executed with the supplied connection string.

### Failed

The MachineCreation Service has failed.

### Unknown

The status of the MachineCreation Service cannot be determined.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

### InvalidDBConnectionString

The database connection string has an invalid format.

### PermissionDenied



You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Test-ProvDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Tests a database connection string for the MachineCreation Service.

----- **EXAMPLE 2** -----

```
c:\PS>Test-ProvDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Tests an invalid database connection string for the MachineCreation Service.

# Test-ProvSchemeNameAvailable

May 10, 2016

Checks to ensure that the proposed name for a provisioning scheme is unused.

## Syntax

Test-ProvSchemeNameAvailable -ProvisioningSchemeName <String[]> [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Checks to ensure that the proposed name for a provisioning scheme is unused. This check is done without regard for scoping of existing provisioning schemes, so the names of inaccessible schemes are also checked.

## Related topics

[New-ProvScheme](#)

[Rename-ProvScheme](#)

## Parameters

**-ProvisioningSchemeName**<String[]>

The name or names of the provisioning scheme(s) to be tested.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

NameAvailability

Pairs of name and the availability of the name

## Notes

In the case of failure, the following errors can result.

### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### CommunicationError

An error occurred while communicating with the service.

#### PermissionDenied

The user does not have administrative rights to perform this operation.

#### ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

Test-ProvSchemeNameAvailable -ProvisioningSchemeName \$NewSchemeName

This tests whether the value of \$NewSchemeName is unique or not, and can be used to create a new provisioning scheme or rename an existing one without failing. True is returned if the name is good.

# Unlock-ProvScheme

May 10, 2016

Unlocks a Provisioning Scheme.

## Syntax

```
Unlock-ProvScheme [-ProvisioningSchemeName] <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Unlock-ProvScheme -ProvisioningSchemeUid <Guid> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to unlock a provisioning scheme that has the Id of a failed Task still associated with it. This allows another long-running task to operate on that scheme. The cmdlet will not unlock a scheme with a task still marked as being active. Use Stop-ProvTask (or, if the task is registered with a failed server, Switch-ProvTask and Stop-ProvTask) to stop any active task first.

## Related topics

[Get-ProvScheme](#)

[Stop-ProvTask](#)

## Parameters

**-ProvisioningSchemeName**<String>

The name of the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ProvisioningSchemeUid**<Guid>

The unique identifier of the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Notes

In the case of failure, the following errors can result.

#### Error Codes

-----

#### ProvisioningSchemeNotFound

The specified provisioning scheme could not be located.

#### NoOp

The specified provisioning scheme had no task object associated.

#### TaskActive

The task object associated with the provisioning scheme is still active.

#### DatabaseError

An error occurred in the service while attempting a database operation.

CouldNotQueryDatabase

The query required to get the database was not defined.

CommunicationError

An error occurred while communicating with the service.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

PermissionDenied

The user does not have administrative rights to perform this operation.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

C:\PS>Unlock-ProvScheme -provisioningSchemeName MyScheme

Unlocks the provisioning scheme 'MyScheme'.

# Unlock-ProvVM

May 10, 2016

Unlocks a VM.

## Syntax

```
Unlock-ProvVM [-VMID] <String[]> -ProvisioningSchemeName <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Unlock-ProvVM [-VMID] <String[]> -ProvisioningSchemeUid <Guid> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to unlock a virtual machine.

## Related topics

[Get-ProvVM](#)

[Lock-ProvVM](#)

## Parameters

**-VMID**<String[]>

The virtual machine Id (hypervisor context)

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ProvisioningSchemeName**<String>

The name of the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-ProvisioningSchemeUid**<Guid>

The unique identifier of the provisioning scheme.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller that the PowerShell snap-in connects to. You can provide this as a host name or an IP address.

Required?	false
Default Value	LocalHost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Input Type

Citrix.MachineCreation.Sdk.ProvisionedVirtualMachine You can pipe an object containing a parameter called 'VMId' and 'ProvisioningSchemeName' to Lock-ProvVM

#### Notes

In the case of failure, the following errors can result.

#### Error Codes

-----

VMDoesNotExist



The specified VM cannot be located.

VMAAlreadyUnLocked

The VM is already unlocked.

VMDoesNotExistForProvisioningScheme

The specified VM does exist in the hypervisor, but is not part of the specified provisioning scheme.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

CommunicationError

An error occurred while communicating with the service.

PermissionDenied

The user does not have administrative rights to perform this operation.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error

ExceptionThrown

An unexpected error occurred. To locate more details, see the Windows event logs on the controller being used or examine the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
C:\PS>Unlock-ProvVM -provisioningSchemeName MyScheme -VMId bc79802c-ba6e-8de8-99e9-4c35d7ad24b4
Unlocks the VM with the Id 'bc79802c-ba6e-8de8-99e9-4c35d7ad24b4' in the provisioning scheme 'MyScheme'.
```

### ----- EXAMPLE 2 -----

```
C:\PS>Get-ProvVM -provisioningSchemeName MyScheme | Unlock-ProvVM
Unlocks all the VMs in the provisioning scheme 'MyScheme'.
```

### ----- EXAMPLE 3 -----

```
C:\PS>Get-ProvVM -Locked $True | Unlock-ProvVM
```

Unlocks all the VMs that are currently locked.

# Citrix.Monitor.Admin.V1

May 10, 2016

## Overview

Name	Description
<a href="#">MonitorMonitorSnapin</a>	The Monitor Service PowerShell snap-in provides administrative functions for the Monitor Service.
<a href="#">Monitor Filtering</a>	Describes the common filtering options for XenDesktop cmdlets.

## Cmdlets

Name	Description
<a href="#">Get-MonitorConfiguration</a>	Gets the configuration settings currently being used by the Monitor Service.
<a href="#">Get-MonitorDataStore</a>	Gets details for each of the Monitor data stores.
<a href="#">Get-MonitorDBConnection</a>	Gets the database string for the specified data store used by the Monitor Service.
<a href="#">Get-MonitorDBSchema</a>	Gets a script that creates the Monitor Service database schema for the specified data store.
<a href="#">Get-MonitorDBVersionChangeScript</a>	Gets a script that updates the Monitor Service database schema.
<a href="#">Get-MonitorInstalledDBVersion</a>	Gets a list of all available database schema versions for the Monitor Service.
<a href="#">Get-MonitorService</a>	Gets the service record entries for the Monitor Service.
<a href="#">Get-MonitorServiceAddedCapability</a>	Gets any added capabilities for the Monitor Service on the controller.
<a href="#">Get-MonitorServiceInstance</a>	Gets the service instance entries for the Monitor Service.
<a href="#">Get-MonitorServiceStatus</a>	Gets the current status of the Monitor Service on the controller.
<a href="#">Remove-MonitorServiceMetadata</a>	Removes metadata from the given Service.
<a href="#">Reset-MonitorDataStore</a>	Refreshes the database string currently being used by the Monitor service.

Name	Description
<a href="#">Reset-MonitorServiceGroupMembership</a>	Resets the access permissions and configuration service locations for the Monitor Service.
<a href="#">Set-MonitorConfiguration</a>	Sets configuration settings that are used by the Monitor Service.
<a href="#">Set-MonitorDBConnection</a>	Configures a database connection for the Monitor Service.
<a href="#">Set-MonitorServiceMetadata</a>	Adds or updates metadata on the given Service.
<a href="#">Test-MonitorDBConnection</a>	Tests a database connection for the Monitor Service.

# about\_MonitorMonitorSnapin

May 10, 2016

## TOPIC

about\_MonitorMonitorSnapin

## SHORT DESCRIPTION

The Monitor Service PowerShell snap-in provides administrative functions for the Monitor Service.

## COMMAND PREFIX

All commands in this snap-in have the noun prefixed with 'Monitor'.

## LONG DESCRIPTION

The Monitor Service PowerShell snap-in enables both local and remote administration of the Monitor service. It provides facilities to query the Monitor service configuration settings and to modify those settings. It also provides the standard set of XenDesktop service cmdlets.

# about\_Monitor\_Filtering

May 10, 2016

## TOPIC

XenDesktop - Advanced Dataset Filtering

## SHORT DESCRIPTION

Describes the common filtering options for XenDesktop cmdlets.

## LONG DESCRIPTION

Some cmdlets operate on large quantities of data and, to reduce the overhead of sending all of that data over the network, many of the Get- cmdlets support server-side filtering of the results.

The conventional way of filtering results in PowerShell is to pipeline them into Where-Object, Select-Object, and Sort-Object, for example:

```
Get-<Noun> | Where { $_.Size = 'Small' } | Sort 'Date' | Select -First 10
```

However, for most XenDesktop cmdlets the data is stored remotely and it would be slow and inefficient to retrieve large amounts of data over the network and then discard most of it. Instead, many of the Get- cmdlets provide filtering parameters that allow results to be processed on the server, returning only the required results.

You can filter results by most object properties using parameters derived from the property name. You can also sort results or limit them to a specified number of records:

```
Get-<Noun> -Size 'Small' -SortBy 'Date' -MaxRecordCount 10
```

You can express more complex filter conditions using a syntax and set of operators very similar to those used by PowerShell expressions.

Those cmdlets that support filtering have the following common parameters:

-MaxRecordCount <int>

Specifies the maximum number of results to return.  
For example, to return only the first nine results use:

```
Get-<Noun> -MaxRecordCount 9
```

If not specified, only the first 250 records are returned, and if more are available, a warning is produced:

WARNING: Only first 250 records returned. Use -MaxRecordCount to retrieve more.

You can suppress this warning by using -WarningAction or by specifying a value for -MaxRecordCount.

To retrieve all records, specify a large number for -MaxRecordCount. As the value is an integer, you can use the following:

```
Get-<Noun> -MaxRecordCount [int]::MaxValue
```

-ReturnTotalRecordCount [<SwitchParameter>]

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. For example:

```
Get-<Noun> -MaxRecordCount 9 -ReturnTotalRecordCount
```

....

```
Get-<Noun> : Returned 9 of 10 items
```

```
At line:1 char:18
```

```
+ Get-<Noun> <<<< -MaxRecordCount 9 -ReturnTotalRecordCount
```

```
+ CategoryInfo : OperationStopped: (:) [Get-<Noun>], PartialDataException
```

```
+ FullyQualifiedErrorId : PartialData,Citrix.<SDKName>.SDK.Get<Noun>
```

The count can be accessed using the TotalAvailableResultCount property:

```
$count = $error[0].TotalAvailableResultCount
```

-Skip <int>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

-SortBy <string>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order, respectively. Ascending order is assumed if no prefix is present.

Sorting occurs before -MaxRecordCount and -Skip parameters are applied. For example, to sort by Name and then by Count (largest first) use:

```
-SortBy 'Name,-Count'
```

By default, sorting by an enumeration property uses the numeric value of the elements. You can specify a different sort order by qualifying the name with an ordered list of elements or their numeric values, or <null> to indicate the placement of null values.

Elements not mentioned are placed at the end in their numeric order. For example, to sort by two different enums and then by the object id:

```
-SortBy 'MyState(StateC,<null>,StateA,StateB),Another(0,3,2,1),Id'
```

-Filter <String>

This parameter lets you specify advanced filter expressions, and supports combination of conditions with -and and -or, and grouping with braces. For example:

```
Get-<Noun> -Filter 'Name -like "High*" -or (Priority -eq 1 -and Severity -ge 2)'
```

The syntax is close enough to PowerShell syntax that you can use script blocks in most cases. This can be easier to read as it reduces quoting:

```
Get-<Noun> -Filter { Count -ne $null }
```

The full -Filter syntax is provided below.

## EXAMPLES

Filtering by strings performs a case-insensitive wildcard match.

Separate parameters are combined with an implicit -and operator.

Normal PowerShell quoting rules apply, so you can use single or double quotes, and omit the quotes altogether for many strings. The order of parameters does not make any difference. The following are equivalent:

```
Get-<Noun> -Company Citrix -Product Xen*
Get-<Noun> -Company "citrix" -Product '[X]EN*'
Get-<Noun> -Product "Xen*" -Company "CITRIX"
Get-<Noun> -Filter { Company -eq 'Citrix' -and Product -like 'Xen*' }
```



See [about\\_Quoting\\_Rules](#) and [about\\_Wildcards](#) for details about PowerShell handling of quotes and wildcards.

To avoid wildcard matching or include quote characters, you can escape the wildcards using the normal PowerShell escape mechanisms (see [about\\_Escape\\_Characters](#)), or switch to a filter expression and the `-eq` operator:

```
Get-<Noun> -Company "Abc[*]" # Matches Abc*
Get-<Noun> -Company "Abc`*" # Matches Abc*
Get-<Noun> -Filter { Company -eq "Abc*" } # Matches Abc*
Get-<Noun> -Filter { Company -eq "A`"B`C`" } # Matches A"B'C
```

Simple filtering by numbers, booleans, and TimeSpans perform direct equality comparisons, although if the value is nullable you can also search for null values. Here are some examples:

```
Get-<Noun> -Uid 123
Get-<Noun> -Enabled $true
Get-<Noun> -Duration 1:30:40
Get-<Noun> -NullableProperty $null
```

More comparisons are possible using advanced filtering with `-Filter`:

```
Get-<Noun> -Filter 'Capacity -ge 10gb'
Get-<Noun> -Filter 'Age -ge 20 -and Age -lt 40'
Get-<Noun> -Filter 'VolumeLevel -like "[123]"'
Get-<Noun> -Filter 'Enabled -ne $false'
Get-<Noun> -Filter 'NullableProperty -ne $null'
```

You can check boolean values without an explicit comparison operator, and you can also combine them with `-not`:

```
Get-<Noun> -Filter 'Enabled' # Equivalent to 'Enabled -eq $true'
Get-<Noun> -Filter '-not Enabled' # Equivalent to 'Enabled -eq $false'
```

See [about\\_Comparison\\_Operators](#) for an explanation of the operators, but note that only a subset of PowerShell operators are supported (`-eq`, `-ne`, `-gt`, `-ge`, `-lt`, `-le`, `-like`, `-notlike`, `-in`, `-notin`, `-contains`, `-notcontains`).

Enumeration values can either be specified using typed values or the string name of the enumeration value:

```
Get-<Noun> -Shape [Shapes]::Square
Get-<Noun> -Shape Circle
```

With filter expressions, typed values can be specified with simple variables or quoted strings. They also support enumerations with wildcards:

```
$s = [Shapes]::Square
Get-<Noun> -Filter { Shape -eq $s -or Shape -eq "Circle" }
Get-<Noun> -Filter { Shape -like 'C*' }
```

By their nature, floating point values, DateTime values, and TimeSpan values are best suited to relative comparisons rather than just equality. DateTime strings are converted using the locale and time zone of the user device, but you can use ISO8601 format strings (YYYY-MM-DDThh:mm:ss.sTZD) to avoid ambiguity. You can also use standard PowerShell syntax to create these values:

```
Get-<Noun> -Filter { StartTime -ge "2010-08-23T12:30:00.0Z" }
$d = [DateTime]"2010-08-23T12:30:00.0Z"
Get-<Noun> -Filter { StartTime -ge $d }
$d = (Get-Date).AddDays(-1)
Get-<Noun> -Filter { StartTime -ge $d }
```

Relative times are quite common and, when using filter expressions, you can also specify DateTime values using a relative format:

```
Get-<Noun> -Filter { StartTime -ge '-2' } # Two days ago
Get-<Noun> -Filter { StartTime -ge '-1:30' } # Hour and a half ago
Get-<Noun> -Filter { StartTime -ge '-0:0:30' } # 30 seconds ago
```

## ARRAY PROPERTIES

When filtering against list or array properties, simple parameters perform a case-insensitive wildcard match against each of the members. With filter expressions, you can use the -contains and -notcontains operators. Unlike PowerShell, these perform wildcard matching on strings.

Note that for array properties the naming convention is for the returned property to be plural, but the parameter used to search for any match is singular. The following are equivalent (assuming Users is an array property):

```
Get-<Noun> -User Fred*
Get-<Noun> -Filter { User -like "Fred*" }
Get-<Noun> -Filter { Users -contains "Fred*" }
```

You can also use the singular form with -Filter to search using other operators:

```
Match if any user in the list is called "Frederick"
Get-<Noun> -Filter { User -eq "Frederick" }
Match if any user in the list has a name alphabetically below 'F'
Get-<Noun> -Filter { User -lt 'F' }
```

## COMPLEX EXPRESSIONS

When matching against multiple values, you can use a sequence of comparisons joined with -or operators, or you can use -in and -notin:

```
Get-<Noun> -Filter { Shape -eq 'Circle' -or Shape -eq 'Square' }
$shapes = 'Circle','Square'
Get-<Noun> -Filter { Shape -in $shapes }
$sides = 1..4
Get-<Noun> -Filter { Sides -notin $sides }
```

Braces can be used to group complex expressions, and override the default left-to-right evaluation of -and and -or. You can also use -not to invert the sense of any sub-expression:

```
Get-<Noun> -Filter { Size -gt 4 -or (Color -eq 'Blue' -and Shape -eq 'Circle') }
Get-<Noun> -Filter { Sides -lt 5 -and -not (Color -eq 'Blue' -and Shape -eq 'Circle') }
```

## PAGING

The simplest way to page through data is to use the -Skip and -MaxRecordCount parameters. So, to read the first three pages of data with 10 records per page, use:

```
Get-<Noun> -Skip 0 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 10 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 20 -MaxRecordCount 10 <other filtering criteria>
```

You must include the same filtering criteria on each call, and ensure that the data is sorted consistently.

The above approach is often acceptable, but as each call performs an independent query, data changes can result in records being skipped or appearing twice. One approach to improve this is to sort by a unique id field and then start the search for the next page at the unique id after the last unique id of the previous page. For example:

```
Get the first page
Get-<Noun> -MaxRecordCount 10 -SortBy SerialNumber

SerialNumber ...

A120004
A120007
... 7 other records ...
A120900

Get the next page
Get-<Noun> -MaxRecordCount 10 -Filter { FirstName -gt 'A120900' }

SerialNumber ...
```

-----  
A120901  
B220000  
...

## FILTER SYNTAX DEFINITION

<Filter> ::= <ScriptBlock> | <ComponentList>

<ScriptBlock> ::= "{" <ComponentList> "}"

<ComponentList> ::= <Component> <AndOrOperator> <ComponentList> |

<Component>

<Component> ::= <NotOperator> <Factor> |

<Factor>

<Factor> ::= "(" <ComponentList> ")" |

<PropertyName> <ComparisonOperator> <Value> |  
<PropertyName>

<AndOrOperator> ::= "-and" | "-or"

<NotOperator> ::= "-not" | "!"

<ComparisonOperator>

::= "-eq" | "-ne" | "-le" | "-ge" | "-lt" | "-gt" |  
"-like" | "-notlike" | "-contains" | "-notcontains" |  
"-in" | "-notin"

<PropertyName> ::= <simple name of property>

<Value> ::= <string literal> | <numeric literal> |

<scalar variable> | <array variable> |  
"\$null" | "\$true" | "\$false"

Numeric literals support decimal and hexadecimal literals, with optional multiplier suffixes (kb, mb, gb, tb, pb).

Dates and times can be specified as string literals. The current culture determines what formats are accepted. To avoid any ambiguity, use strings formatted to the ISO8601 standard. If not specified, the current time zone is used.

Relative date-time string literals are also supported, using a minus sign followed by a TimeSpan. For example, "-1:30" means 1 hour and 30 minutes ago.

# Get-MonitorConfiguration

May 10, 2016

Gets the configuration settings currently being used by the Monitor Service.

## Syntax

```
Get-MonitorConfiguration [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the configuration currently being used by the Monitor Service.

A site database connection must be configured for this command to be used.

## Related topics

[Set-MonitorConfiguration](#)

## Parameters

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Get-MonitorConfiguration returns the configuration settings.

## Examples

### ----- **EXAMPLE 1** -----

```
c:\PS>Get-MonitorConfiguration
```

Gets the current configuration for the Monitor Service.

# Get-MonitorDataStore

May 10, 2016

Gets details for each of the Monitor data stores.

## Syntax

```
Get-MonitorDataStore [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns an object for each of the Monitor data stores describing the connection string, data store name, db type, provider, schema name, and DB status.

A database connection must be configured in order for this command to be used if the service has a secondary data store. This is not required for the site data store.

## Related topics

[Reset-MonitorDataStore](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Monitor.Sdk.DataStoreConfiguration

An object describing the connection string, data store name, database type, provider, schema name and database status.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.



## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-MonitorDataStore
```

```
ConnectionString : Server=.\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True
DataStore : Site
DatabaseType : SqlServer
Provider : MSSQL
SchemaName : MonitorSiteSchema
Status : OK
```

```
ConnectionString :
DataStore : Secondary
DatabaseType : SqlServer
Provider : MSSQL
SchemaName : MonitorSecondarySchema
Status : DBUnconfigured
Get the database connection string for the Monitor Service.
```

# Get-MonitorDBConnection

May 10, 2016

Gets the database string for the specified data store used by the Monitor Service.

## Syntax

```
Get-MonitorDBConnection [-DataStore] <String> [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the database connection string for the specified data store.

If the returned string is blank, no valid connection string has been specified. In this case the service is running, but is idle and awaiting specification of a valid connection string.

## Related topics

[Get-MonitorServiceStatus](#)

[Get-MonitorDataStore](#)

[Set-MonitorDBConnection](#)

[Test-MonitorDBConnection](#)

## Parameters

**-DataStore**<String>

Specifies the logical name of the data store for the Monitor Service. Can be either be 'Site' or the logical name of the secondary data store.

Required?	false
Default Value	Site
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

system.string

The database connection string configured for the Monitor Service.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### NoDBConnections

The database connection string for the Monitor Service has not been specified.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-MonitorDBConnection
```

Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True  
Get the database connection string for the Monitor Service.

# Get-MonitorDBSchema

May 10, 2016

Gets a script that creates the Monitor Service database schema for the specified data store.

## Syntax

```
Get-MonitorDBSchema [-DatabaseName <String>] [-ServiceGroupName <String>] [-ScriptType <ScriptTypes>] [-LocalDatabase] [-Sid <String>] [-DataStore <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets SQL scripts that can be used to create a new Monitor Service database schema, add a new Monitor Service to an existing site, remove a Monitor Service from a site, or create a database server logon for a Monitor Service. If no Sid parameter is provided, the scripts obtained relate to the currently selected Monitor Service instance, otherwise the scripts relate to Monitor Service instance running on the machine identified by the Sid provided. When obtaining the Evict script, a Sid parameter must be supplied. The current service instance is that on the local machine, or that explicitly specified by the last usage of the -AdminAddress parameter to a Monitor SDK cmdlet. The service instance used to obtain the scripts does not need to be a member of a site or to have had its database connection configured. The database scripts support only Microsoft SQL Server, or SQL Server Express, and require Windows integrated authentication to be used. They can be run using SQL Server's SQLCMD utility, or by copying the script into an SQL Server Management Studio (SSMS) query window and executing the query. If using SSMS, the query must be executed in 'SMDCMD mode'. The ScriptType parameter determines which script is obtained. If ScriptType is not specified, or is FullDatabase, the script contains:

- o Creation of service schema
- o Creation of database server logon
- o Creation of database user
- o Addition of database user to Monitor Service roles

If ScriptType is Instance, the returned script contains:

- o Creation of database server logon
- o Creation of database user
- o Addition of database user to Monitor Service roles

If ScriptType is Evict, the returned script contains:

- o Removal of Monitor Service instance from database
- o Removal of database user

If ScriptType is Login, the returned script contains:

- o Creation of database server logon only

If the service uses two data stores they can exist in the same database. You do not need to configure a database before using this command.

## Related topics

[Get-MonitorDataStore](#)

[Set-MonitorDBConnection](#)

## Parameters

### **-DatabaseName**<String>

Specifies the name of the database for which the schema will be generated.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ServiceGroupName**<String>

Specifies the name of the service group to be used when creating the database schema. The service group is a collection of all the Monitor services that share the same database instance and are considered equivalent; that is, all the services within a service group can be used interchangeably.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-ScriptType**<ScriptTypes>

Specifies the type of database script returned. Available script types are:

#### Database

Returns a full database script that can be used to create a database schema for the Monitor Service in a database instance that does not already contain a schema for this service. The DatabaseName and ServiceGroupName parameters must be specified to create a script of this type.

#### Instance

Returns a permissions script that can be used to add further Monitor services to an existing database instance that already contains the full Monitor service schema, associating the services to the Service Group. The Sid parameter can optionally be specified to create a script of this type.

#### Login

Returns a database logon script that can be used to add the required logon accounts to an existing database instance that contains the Monitor Service schema. This is used primarily when creating a mirrored database environment. The DatabaseName parameter must be specified to create a script of this type.

#### Evict

Returns a script that can be used to remove the specified Monitor Service from the database entirely. The DatabaseName and Sid parameters must be specified to create a script of this type.

Required?	false
-----------	-------

Default Value	Database
Accept Pipeline Input?	false

#### **-LocalDatabase**<SwitchParameter>

Specifies whether the database script is to be used in a database instance run on the same controller as other services in the service group. Including this parameter ensures the script creates only the required permissions for local services to access the database schema for Monitor services.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Sid**<String>

Specifies the SID of the controller on which the Monitor Service instance to remove from the database is running.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-DataStore**<String>

Specifies the logical name of the data store for the Monitor Service. Can be either be 'Site' or the logical name of the secondary data store.

Required?	false
Default Value	Site
Accept Pipeline Input?	false

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

### System.string

A string containing the required SQL script for application to a database.

## Notes

The scripts returned support Microsoft SQL Server Express Edition, Microsoft SQL Server Standard Edition, and Microsoft SQL Server Enterprise Edition databases only, and are generated on the assumption that integrated authentication will be used.

If the ScriptType parameter is not included or set to 'FullDatabase', the full database script is returned, which will:

Create the database schema.

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist).

If the ScriptType parameter is set to 'Instance', the script will:

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist) and associate it with a user.

If the ScriptType parameter is set to 'Login', the script will:

Create the logon (providing the schema does not already exist) and associate it with a pre-existing user of the same name.

If the LocalDatabase parameter is included, the NetworkService account will be added to the list of accounts permitted to access the database. This is required only if the database is run on a controller.

If the command fails, the following errors can be returned.

### Error Codes

-----

#### GetSchemasFailed

The database schema could not be found.

#### ActiveDirectoryAccountResolutionFailed

The specified Active Directory account or Group could not be found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied



You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-MonitorDBSchema -DatabaseName MyDB -ServiceGroupName MyServiceGroup > c:\MonitorSchema.sql
```

Get the full database schema for site data store of the Monitor Service and copy it to a file called 'c:\MonitorSchema.sql'.

This script can then be used to create the schema in a pre-existing database named 'MyDB' that does not already contain a Monitor Service site schema.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-MonitorDBSchema -DatabaseName MyDB -scriptType Login > c:\MonitorLogins.sql
```

Get the logon scripts for the Monitor Service.

### ----- EXAMPLE 3 -----

```
c:\PS>Get-MonitorDBSchema -DatabaseName MyDB -ServiceGroupName MyServiceGroup -DataStore Secondary > c:\MonitorSchema.sql
```

Get the full database schema for the secondary data store of the Monitor Service and copy it to a file called 'c:\MonitorSecondarySchema.sql'.

This script can then be used to create the schema in a pre-existing database named 'MyDB' that does not already contain a Monitor Service secondary schema.

# Get-MonitorDBVersionChangeScript

May 10, 2016

Gets a script that updates the Monitor Service database schema.

## Syntax

```
Get-MonitorDBVersionChangeScript -DatabaseName <String> -TargetVersion <Version> [-DataStore <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns a database script that can be used to upgrade or downgrade the site or secondary schema for the Monitor Service from the current schema version to a different version.

## Related topics

[Get-MonitorInstalledDBVersion](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database instance to which the update applies.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-TargetVersion**<Version>

Specifies the version of the database you want to update to.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-DataStore**<String>

Specifies the logical name of the data store for the Monitor Service. Can be either be 'Site' or the logical name of the secondary data store.

--	--

Required?	false
Default Value	Site
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

System.Management.Automation.PSObject

A PSObject containing the required SQL script for application to a database.

### Notes

The PSObject returned by this cmdlet contains the following properties:

-- Script The raw text of the SQL script to apply the update, or null in the case when no upgrade path to the specified target version exists.

-- NeedExclusiveAccess Indicates whether all services in the service group must be shut down during the update or not.

-- CanUndo Indicates whether the generated script allows the updated schema to be reverted to the state prior to the update.

Scripts to update the schema version are stored in the database so any service in the service group can obtain these scripts. Extreme caution should be exercised when using update scripts. Citrix recommends backing up the database before attempting to upgrade the schema. Database update scripts may require exclusive use of the schema and so may not be able to execute while any Monitor services are running. However, this depends on the specific update being carried out.

After a schema update has been carried out, services that require the previous version of the schema may cease to operate. The ServiceState parameter reported by the Get-MonitorServiceStatus command provides information about service compatibility. For example, if the schema has been upgraded to a more recent version that a service cannot use, the service reports "DBNewerVersionThanService".

If the command fails, the following errors can be returned.

### Error Codes

-----

#### NoOp

The operation was successful but had no effect.

#### NoDBConnections

The database connection string for the Monitor Service has not been specified.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
C:\PS> $update = Get-MonitorDBVersionChangeScript -DatabaseName MyDb -TargetVersion 1.0.75.0
```

```
C:\PS> $update.Script > update_75.sql
```

Gets an SQL update script to update the current schema to version 1.0.75.0. The resulting update\_75.sql script is suitable for direct use with the SQL Server SQLCMD utility.

# Get-MonitorInstalledDBVersion

May 10, 2016

Gets a list of all available database schema versions for the Monitor Service.

## Syntax

```
Get-MonitorInstalledDBVersion [-Upgrade] [-Downgrade] [-DataStore <String>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Returns the current version of the Monitor Service database schema, if no flags are set, otherwise returns versions for which upgrade or downgrade scripts are available and have been stored in the database.

## Related topics

## Parameters

**-Upgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be updated should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Downgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be reverted should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-DataStore**<String>

Specifies the database connection logical name the schema script should be returned for. The parameter is optional.

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

#### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### Return Values

##### System.Version

The Get-MonitorInstalledDbVersion command returns objects containing the new definition of the Monitor Service database schema version.

##### Major <Integer>

##### Minor <Integer>

##### Build <Integer>

##### Revision <Integer>

#### Notes

If the command fails, the following errors can be returned.

##### Error Codes

-----

##### InvalidParameterCombination

Both the Upgrade and Downgrade flags were specified.

##### NoOp

The operation was successful but had no effect.

##### NoDBConnections

The database connection string for the Monitor Service has not been specified.

##### DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-MonitorInstalledDBVersion
```

Major	Minor	Build	Revision
-------	-------	-------	----------

----	----	----	-----
------	------	------	-------

5	6	0	0
---	---	---	---

Get the currently installed version of the Monitor Service database schema.

----- **EXAMPLE 2** -----

```
c:\PS>Get-MonitorInstalledDBVersion -Upgrade
```

Major	Minor	Build	Revision
-------	-------	-------	----------

----	----	----	-----
------	------	------	-------

6	0	0	0
---	---	---	---

Get the versions of the Monitor Service database schema for which upgrade scripts are supplied.

# Get-MonitorService

May 10, 2016

Gets the service record entries for the Monitor Service.

## Syntax

```
Get-MonitorService [-Metadata <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns instances of the Monitor Service that the service publishes. The service records contain account security identifier information that can be used to remove each service from the database.

A database connection for the service is required to use this command.

## Related topics

### Parameters

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>



When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Monitor\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See [about\\_Monitor\\_Filtering](#) for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.Monitor.Sdk.Service

The Get-MonitorServiceInstance command returns an object containing the following properties.

Uid <Integer>

Specifies the unique identifier for the service in the group. The unique identifier is an index number.

ServiceHostId <Guid>

Specifies the unique identifier for the service instance.

DNSName <String>

Specifies the domain name of the host on which the service runs.

MachineName <String>

Specifies the short name of the host on which the service runs.

CurrentState <Citrix.Fma.Sdk.ServiceCore.ServiceState>

Specifies whether the service is running, started but inactive, stopped, or failed.

LastStartTime <DateTime>

Specifies the date and time at which the service was last restarted.

LastActivityTime <DateTime>

Specifies the date and time at which the service was last stopped or restarted.

OSType

Specifies the operating system installed on the host on which the service runs.

OSVersion

Specifies the version of the operating system installed on the host on which the service runs.

ServiceVersion

Specifies the version number of the service instance. The version number is a string that reflects the full build version of the service.

DatabaseUserName <string>

Specifies for the service instance the Active Directory account name with permissions to access the database. This will be either the machine account or, if the database is running on a controller, the NetworkService account.

Sid <string>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

ActiveSiteServices <string[]>

Specifies the names of active site services currently running in the service. Site services are components that perform long-running background processing in some services. This field is empty for services that do not contain site services.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

UnknownObject

One of the specified objects was not found.

PartialData

Only a subset of the available data was returned.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

CouldNotQueryDatabase

The query required to get the database was not defined.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Get-MonitorService

Uid : 1  
ServiceHostId : aef6f464-f1ee-4042-a523-66982e0cecd0  
DNSName : MyServer.company.com  
MachineName : MYSERVER  
CurrentState : On  
LastStartTime : 04/04/2011 15:25:38  
LastActivityTime : 04/04/2011 15:33:39  
OSType : Win32NT  
OSVersion : 6.1.7600.0  
ServiceVersion : 5.1.0.0  
DatabaseUserName : NT AUTHORITY\NETWORK SERVICE  
SID : S-1-5-21-2316621082-1546847349-2782505528-1165  
ActiveSiteServices : {MySiteService1, MySiteService2...}  
Get all the instances of the Monitor Service running in the current service group.

# Get-MonitorServiceAddedCapability

May 10, 2016

Gets any added capabilities for the Monitor Service on the controller.

## Syntax

```
Get-MonitorServiceAddedCapability [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables updates to the Monitor Service on the controller to be detected.

You do not need to configure a database connection before using this command.

## Related topics

### Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.String

String containing added capabilities.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-MonitorServiceAddedCapability
```

Get the added capabilities of the Monitor Service.

# Get-MonitorServiceInstance

May 10, 2016

Gets the service instance entries for the Monitor Service.

## Syntax

```
Get-MonitorServiceInstance [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns service interfaces published by the instance of the Monitor Service. Each instance of a service publishes multiple interfaces with distinct interface types, and each of these interfaces is represented as a ServiceInstance object. Service instances can be used to register the service with a central configuration service so that other services can use the functionality.

You do not need to configure a database connection to use this command.

## Related topics

### Parameters

**-AdminAddress** <String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Monitor.Sdk.ServiceInstance

The Get-MonitorServiceInstance command returns an object containing the following properties.

**ServiceGroupUid** <Guid>

Specifies the unique identifier for the service group of which the service is a member.

**ServiceGroupName** <String>

Specifies the name of the service group of which the service is a member.

**ServiceInstanceUID** <Guid>

Specifies the unique identifier for registered service instances, which are service instances held by and obtained from a

central configuration service. Unregistered service instances do not have unique identifiers.

ServiceType <String>

Specifies the service instance type. For this service, the service instance type is always Monitor.

Address

Specifies the address of the service instance. The address can be used to access the service and, when registered in the central configuration service, can be used by other services to access the service.

Binding

Specifies the binding type that must be used to communicate with the service instance. In this release of XenDesktop, the binding type is always 'wcf\_HTTP\_kerb'. This indicates that the service provides a Windows Communication Foundation endpoint that uses HTTP binding with integrated authentication.

Version

Specifies the version of the service instance. The version number is used to ensure that the correct versions of the services are used for communications.

ServiceAccount <String>

Specifies the Active Directory account name for the machine on which the service instance is running. The account name is used to provide information about the permissions required for interservice communications.

ServiceAccountSid <String>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

InterfaceType <String>

Specifies the interface type. Each service can provide multiple service instances, each for a different purpose, and the interface defines the purpose. Available interfaces are:

SDK - for PowerShell operations

InterService - for operations between different services

Peer - for communications between services of the same type

Metadata <Citrix.Monitor.Sdk.Metadata[]>

The collection of metadata associated with registered service instances, which are service instances held by and obtained from a central configuration service. Metadata is not stored for unregistered service instances.

Notes

If the command fails, the following errors can be returned.

Error Codes

-----



## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-MonitorServiceInstance
```

```
Address : http://MyServer.com:80/Citrix/Monitor
Binding : wcf_HTTP_kerb
InterfaceType : SDK
Metadata :
MetadataMap :
ServiceAccount : ENG\MyAccount$
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164
ServiceGroupName : MyServiceGroup
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000
ServiceType : Monitor
Version : 1
```

```
Address : http://MyServer.com:80/Citrix/Monitor/IServiceApi
Binding : wcf_HTTP_kerb
InterfaceType : InterService
Metadata :
MetadataMap :
```

ServiceAccount : ENG\MyAccount  
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164  
ServiceGroupName : MyServiceGroup  
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d  
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000  
ServiceType : Monitor  
Version : 1

Get all instances of the Monitor Service running on the specified machine. For remote services, use the AdminAddress parameter to define the service for which the interfaces are required. If the AdminAddress parameter has not been specified for the runspace, service instances running on the local machine are returned.

# Get-MonitorServiceStatus

May 10, 2016

Gets the current status of the Monitor Service on the controller.

## Syntax

```
Get-MonitorServiceStatus [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables the status of the Monitor Service on the controller to be monitored. If the service has multiple data stores it will return the overall state as an aggregate of all the data store states. For example, if the site data store status is OK and the secondary data store status is DBUnconfigured then it will return DBUnconfigured.

## Related topics

[Get-MonitorDataStore](#)

[Set-MonitorDBConnection](#)

[Test-MonitorDBConnection](#)

[Get-MonitorDBConnection](#)

[Get-MonitorDBSchema](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Get-MonitorServiceStatus command returns an object containing the status of the Monitor Service together with extra diagnostics information.

DBUnconfigured

The Monitor Service does not have a database connection configured.

#### DBRejectedConnection

The database rejected the logon attempt from the Monitor Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

#### InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the Monitor Service schema has not been added to the database.

#### DBNotFound

The specified database could not be located with the configured connection string.

#### DBNewerVersionThanService

The version of the Monitor Service currently in use is incompatible with the version of the Monitor Service schema on the database. Upgrade the Monitor Service to a more recent version.

#### DBOlderVersionThanService

The version of the Monitor Service schema on the database is incompatible with the version of the Monitor Service currently in use. Upgrade the database schema to a more recent version.

#### DBVersionChangeInProgress

A database schema upgrade is currently in progress.

#### OK

The Monitor Service is running and is connected to a database containing a valid schema.

#### Failed

The Monitor Service has failed.

#### Unknown

(0) The service status cannot be determined.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-MonitorServiceStatus
```

## DBUnconfigured

Get the current status of the Monitor Service.

# Remove-MonitorServiceMetadata

May 10, 2016

Removes metadata from the given Service.

## Syntax

```
Remove-MonitorServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-MonitorServiceMetadata [-ServiceHostId] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-MonitorServiceMetadata [-InputObject] <Service[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-MonitorServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Service.

## Related topics

[Set-MonitorServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name<String>**

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples



----- **EXAMPLE 1** -----

```
c:\PS>Get-MonitorService | % { Remove-MonitorServiceMetadata -Map $_.MetadataMap }
Remove all metadata from all Service objects.
```

# Reset-MonitorDataStore

May 10, 2016

Refreshes the database string currently being used by the Monitor service.

## Syntax

```
Reset-MonitorDataStore [-DataStore] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the string for the database connection currently being used by the Monitor Service. Can only be called for secondary data stores.

There is no requirement for a database connection to be configured in order for this command to be used.

## Related topics

[Get-MonitorDataStore](#)

## Parameters

### **-DataStore**<String>

Specifies the database connection logical name to be used by the Monitor Service. Can be either be 'Site' or the logical name of the secondary data store. Specifying the site data store will display an error because this operation is not supported for site data stores.

Required?	true
Default Value	Site
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Monitor.Sdk.ServiceStatus

The status of the specified data store.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

### DatabaseError

An error occurred in the service while attempting a database operation.

### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

### PermissionDenied

You do not have permission to execute this command.

### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

### CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Reset-MonitorDataStore -DataStore Secondary
```

```
OK
```

Refresh the database connection string for the Monitor Service.

# Reset-MonitorServiceGroupMembership

May 10, 2016

Reloads the access permissions and configuration service locations for the Monitor Service.

## Syntax

```
Reset-MonitorServiceGroupMembership [-ConfigServiceInstance] <ServiceInstance[]> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Enables you to reload Monitor Service access permissions and configuration service locations. The Reset-MonitorServiceGroupMembership command must be run on at least one instance of the service type (Monitor) after installation and registration with the configuration service. Without this operation, the Monitor services will be unable to communicate with other services in the XenDesktop deployment. When the command is run, the services are updated when additional services are added to the deployment, provided that the configuration service is not stopped. The Reset-MonitorServiceGroupMembership command can be run again to refresh this information if automatic updates do not occur when new services are added to the deployment. If more than one configuration service instance is passed to the command, the first instance that meets the expected service type requirements is used.

## Related topics

### Parameters

**-ConfigServiceInstance**<ServiceInstance[]>

Specifies the configuration service instance object that represents the service instance for the type 'InterService' that references a configuration service for the deployment.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Input Type

Citrix.Monitor.Sdk.ServiceInstance[] Service instances containing a ServiceInstance object that refers to the central configuration service interservice interface can be piped to the Reset-MonitorServiceGroupMembership command.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### NoSuitableServiceInstance

None of the supplied service instance objects were suitable for resetting service group membership.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config | Reset-MonitorServiceGroupMembership
```

Reset the service group membership for a service in a deployment where the configuration service is configured and running on the same machine as the service.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config -AdminAddress OtherServer.example.com | Reset-MonitorServiceGroupmembership
```

Reset the service group membership for a service in a deployment where the configuration service that is configured and running on a machine named 'OtherServer.example.com'.



# Set-MonitorConfiguration

May 10, 2016

Sets configuration settings that are used by the Monitor Service.

## Syntax

```
Set-MonitorConfiguration [-GroomSessionsRetentionDays <Int32>] [-GroomFailuresRetentionDays <Int32>] [-GroomLoadIndexesRetentionDays <Int32>] [-GroomDeletedRetentionDays <Int32>] [-GroomSummariesRetentionDays <Int32>] [-GroomMachineHotfixLogRetentionDays <Int32>] [-GroomMinuteRetentionDays <Int32>] [-DataCollectionEnabled <Boolean>] [-FullPollStartHour <Int32>] [-ResolutionPollTimeHours <Int32>] [-SyncPollTimeHours <Int32>] [-DetailedSqlOutputEnabled <Boolean>] [-CollectHotfixDataEnabled <Boolean>] [-GroomApplicationInstanceRetentionDays <Int32>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Sets the configuration settings used by the Monitor Service. Use these settings to modify the behavior of the service.

A database connection need not be configured for this command to be used.

### Hinweis

For parameters whose default values depend on the product edition, as noted here, the values of these parameters are set to the new default values when the edition is changed.

## Related topics

[Get-MonitorConfiguration](#)

## Parameters

**-GroomSessionsRetentionDays**<Int32>

Determines how many days to keep Session and Connection records after the Session is terminated.

Required?	false
Default Value	7 for non-platinum, 90 for platinum.
Accept Pipeline Input?	false

**-GroomFailuresRetentionDays**<Int32>

Determines how many days to keep MachineFailureLog and ConnectionFailureLog records after these are created.

Required?	false
-----------	-------



Default Value	7 for non-platinum, 90 for platinum.
Accept Pipeline Input?	false

#### **-GroomLoadIndexesRetentionDays<Int32>**

Determines how many days to keep LoadIndex records after these are created.

Required?	false
Default Value	7 for non-platinum, 90 for platinum.
Accept Pipeline Input?	false

#### **-GroomDeletedRetentionDays<Int32>**

Determines how many days to keep Machine, Catalog, DesktopGroup and Hypervisor entities around that have a LifecycleState of 'Deleted'. This also deletes any related Session, Connection, Summary, Failure or LoadIndex records.

Required?	false
Default Value	7 for non-platinum, 90 for platinum.
Accept Pipeline Input?	false

#### **-GroomSummariesRetentionDays<Int32>**

Determines how many days to keep DesktopGroupSummary, FailureLogSummary and LoadIndexSummary records at the daily granularity.

Required?	false
Default Value	7 for non-platinum, 90 for platinum.
Accept Pipeline Input?	false

#### **-GroomMachineHotfixLogRetentionDays<Int32>**

Determines how many days to keep Machine-Hotfix history records at the daily granularity.

Required?	false
-----------	-------

Default Value	90
Accept Pipeline Input?	false

#### **-GroomMinuteRetentionDays<Int32>**

Determines how many days to keep minute data.

Required?	false
Default Value	3
Accept Pipeline Input?	false

#### **-DataCollectionEnabled<Boolean>**

Starts / stops data collection. Stopping data collection turns off polling, and does not persist operational event data to the database.

Required?	false
Default Value	True
Accept Pipeline Input?	false

#### **-FullPollStartHour<Int32>**

Hour of day when Full Poll should begin.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ResolutionPollTimeHours<Int32>**

Start time for the Resolution Poll worker.

Required?	false
-----------	-------

Default Value	
Accept Pipeline Input?	false

#### **-SyncPollTimeHours<Int32>**

Start time for Sync Poll worker.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-DetailedSqlOutputEnabled<Boolean>**

Determines if the SqlLog should be enabled to send SQL statements to the CDF Trace

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-CollectHotfixDataEnabled<Boolean>**

This setting determines if the hotfix inventory data should be collected and stored in the database or if it should be thrown away.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-GroomApplicationInstanceRetentionDays<Int32>**

Required?	false
Default Value	

Accept Pipeline Input?	false
------------------------	-------

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Examples

### ----- **EXAMPLE 1** -----

```
C:\PS>Set-MonitorConfiguration -GroomSessionsRetentionDays 5 -GroomFailuresRetentionDays 4 ...
```

Updates the settings in the site database with the newly specified values.

# Set-MonitorDBConnection

May 10, 2016

Configures a database connection for the Monitor Service.

## Syntax

```
Set-MonitorDBConnection [-DBConnection] <String> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [[-DataStore] <String>]
[<CommonParameters>]
```

## Detailed Description

Configures a connection to a database in which the Monitor Service can store its state. The service will attempt to connect and start using the database immediately after the connection is configured. The database connection string is updated to the specified value regardless of whether it is valid or not. Specifying an invalid connection string prevents a service from functioning until the error is corrected.

After a connection is configured, you cannot alter it without first clearing it (by setting the connection to \$null).

You do not need to configure a database connection to use this command.

## Related topics

[Get-MonitorServiceStatus](#)

[Get-MonitorDBConnection](#)

[Test-MonitorDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be used by the Monitor Service. Passing in \$null will clear any existing database connection configured.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Force**<SwitchParameter>

If present, allows the local administrator to set the connection string to null when there are problems contacting the database or other services.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

#### **-DataStore<String>**

Specifies the logical name of the data store for the Monitor Service. Can be either be 'Site' or the logical name of the secondary data store.

Required?	false
Default Value	Site
Accept Pipeline Input?	false

#### **Return Values**

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Set-MonitorDBConnection command returns an object containing the status of the Monitor Service together with extra diagnostics information.

#### **DBUnconfigured**

The Monitor Service does not have a database connection configured.

#### **DBRejectedConnection**

The database rejected the logon attempt from the Monitor Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

#### **InvalidDBConfigured**

The expected stored procedures are missing from the database. This may be because the Monitor Service schema has not been added to the database.

#### **DBNotFound**

The specified database could not be located with the configured connection string.

#### **DBNewerVersionThanService**

The version of the Monitor Service currently in use is incompatible with the version of the Monitor Service schema on the database. Upgrade the Monitor Service to a more recent version.

#### DBOlderVersionThanService

The version of the Monitor Service schema on the database is incompatible with the version of the Monitor Service currently in use. Upgrade the database schema to a more recent version.

#### DBVersionChangeInProgress

A database schema upgrade is currently in progress.

#### OK

The Monitor Service is running and is connected to a database containing a valid schema.

#### Failed

The Monitor Service has failed.

#### Unknown

The status of the Monitor Service cannot be determined.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### InvalidDBConnectionString

The database connection string has an invalid format.

#### DatabaseConnectionDetailsAlreadyConfigured

There was already a database connection configured. After a configuration is set, it can only be set to `Null`.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

----- **EXAMPLE 1** -----

```
c:\PS>Set-MonitorDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Configures a database connection string for the Monitor Service.

----- **EXAMPLE 2** -----

```
c:\PS>Set-MonitorDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Configures an invalid database connection string for the Monitor Service.



# Set-MonitorServiceMetadata

May 10, 2016

Adds or updates metadata on the given Service.

## Syntax

```
Set-MonitorServiceMetadata [-ServiceHostId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-MonitorServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-MonitorServiceMetadata [-InputObject] <Service[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

```
Set-MonitorServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Allows you to store additional custom data against given Service objects.

## Related topics

[Remove-MonitorServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name**<String>

Specifies the property name of the metadata to be added. The property must be unique for the Service specified. The property cannot contain any of the following characters \/:;#.\*?=<>|[]()"

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Value**<String>

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Map**<PSObject>

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

---

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-MonitorServiceMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

InvalidParameterCombination

The cmdlet parameters are inconsistent.

UnknownObject

One of the specified objects was not found.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Set-MonitorServiceMetadata -ServiceHostId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value

Key	Value
---	----
property	value

Add metadata with a name of 'property' and a value of 'value' to the Service with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Test-MonitorDBConnection

May 10, 2016

Tests a database connection for the Monitor Service.

## Syntax

```
Test-MonitorDBConnection [-DBConnection] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [[-DataStore] <String>] [<CommonParameters>]
```

## Detailed Description

Tests a connection to the database in which the Monitor Service can store its state. The service will attempt to connect to the database without affecting the current connection to the database.

You do not have to clear the connection to use this command.

## Related topics

[Get-MonitorServiceStatus](#)

[Get-MonitorDBConnection](#)

[Set-MonitorDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be tested by the Monitor Service.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

#### **-DataStore<String>**

Specifies the logical name of the data store for the Monitor Service. Can be either be 'Site' or the logical name of the secondary data store.

Required?	false
Default Value	Site
Accept Pipeline Input?	false

#### **Return Values**

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Test-MonitorDBConnection command returns an object containing the status of the Monitor Service if the connection string of the specified data store were to be set to the string being tested, together with extra diagnostics information for the specified connection string.

#### **DBRejectedConnection**

The database rejected the logon attempt from the Monitor Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

#### **InvalidDBConfigured**

The expected stored procedures are missing from the database. This may be because the Monitor Service schema has not been added to the database.

#### **DBNotFound**

The specified database could not be located with the configured connection string.

#### **DBNewerVersionThanService**

The version of the Monitor Service currently in use is incompatible with the version of the Monitor Service schema on the database. Upgrade the Monitor Service to a more recent version.

#### **DBOlderVersionThanService**

The version of the Monitor Service schema on the database is incompatible with the version of the Monitor Service currently in use. Upgrade the database schema to a more recent version.

#### **DBVersionChangeInProgress**

A database schema upgrade is currently in progress.

#### **OK**

The Set-MonitorDBConnection command would succeed if it were executed with the supplied connection string.

#### **Failed**

The Monitor Service has failed.

#### **Unknown**

The status of the Monitor Service cannot be determined.

#### **Notes**

If the command fails, the following errors can be returned.

#### Error Codes

-----

##### InvalidDBConnectionString

The database connection string has an invalid format.

##### DatabaseError

An error occurred in the service while attempting a database operation.

##### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

##### PermissionDenied

You do not have permission to execute this command.

##### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

##### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

##### CommunicationError

There was a problem communicating with the remote service.

##### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Test-MonitorDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Tests a database connection string for the Monitor Service.

##### ----- EXAMPLE 2 -----

```
c:\PS>Test-MonitorDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Tests an invalid database connection string for the Monitor Service.

# Citrix.Storefront.Admin.V1

May 10, 2016

## Overview

Name	Description
<a href="#">SfStorefrontSnapin</a>	This service short description
<a href="#">Sf Filtering</a>	Describes the common filtering options for XenDesktop cmdlets.

## Cmdlets

Name	Description
<a href="#">Add-SfServerToCluster</a>	Adds a new server to an existing cluster.
<a href="#">Add-SfStorefrontAddress</a>	Modifies a StoreFront address configuration by adding an additional StoreFront address to it.
<a href="#">Get-SfCluster</a>	Gets all Storefront clusters present in the site.
<a href="#">Get-SfDBConnection</a>	Gets the database string for the specified data store used by the Storefront Service.
<a href="#">Get-SfDBSchema</a>	Gets a script that creates the Storefront Service database schema for the specified data store.
<a href="#">Get-SfDBVersionChangeScript</a>	Gets a script that updates the Storefront Service database schema.
<a href="#">Get-SfInstalledDBVersion</a>	Gets a list of all available database schema versions for the Storefront Service.
<a href="#">Get-SfIsStorefrontInstalled</a>	Tells whether StoreFront Services and Privileged Service are installed.
<a href="#">Get-SfService</a>	Gets the service record entries for the Storefront Service.
<a href="#">Get-SfServiceAddedCapability</a>	Gets any added capabilities for the Storefront Service on the controller.
<a href="#">Get-SfServiceInstance</a>	Gets the service instance entries for the Storefront Service.
<a href="#">Get-SfServiceStatus</a>	Gets the current status of the Storefront Service on the controller.



Name	Description
<a href="#">Get-SfStorefrontAddress</a>	Sets the high-level description of a configuration for StoreFront addresses, based on a configuration byte array.
<a href="#">Get-SfTask</a>	Gets the task history for the Storefront Service.
<a href="#">New-SfCluster</a>	Creates new Storefront cluster with default set of services.
<a href="#">New-SfStorefrontAddress</a>	Creates a new StoreFront address configuration, specifying a single address.
<a href="#">Remove-SfServerFromCluster</a>	Removes server from the cluster.
<a href="#">Remove-SfServiceMetadata</a>	Removes metadata from the given Service.
<a href="#">Remove-SfTask</a>	Removes from the database completed tasks for the Storefront Service.
<a href="#">Remove-SfTaskMetadata</a>	Removes metadata from the given Task.
<a href="#">Reset-SfServiceGroupMembership</a>	Reloads the access permissions and configuration service locations for the Storefront Service.
<a href="#">Set-SfCluster</a>	Sets the parameters on the given cluster.
<a href="#">Set-SfDBConnection</a>	Configures a database connection for the Storefront Service.
<a href="#">Set-SfServiceMetadata</a>	Adds or updates metadata on the given Service.
<a href="#">Set-SfTaskMetadata</a>	Adds or updates metadata on the given Task.
<a href="#">Test-SfDBConnection</a>	Tests a database connection for the Storefront Service.

# about\_SfStorefrontSnapin

May 10, 2016

## TOPIC

about\_SfStorefrontSnapin

## SHORT DESCRIPTION

This service short description

## COMMAND PREFIX

All commands in this snap-in have the noun prefixed with 'Sf'.

## LONG DESCRIPTION

This service long description

# about\_Sf\_Filtering

May 10, 2016

## TOPIC

XenDesktop - Advanced Dataset Filtering

## SHORT DESCRIPTION

Describes the common filtering options for XenDesktop cmdlets.

## LONG DESCRIPTION

Some cmdlets operate on large quantities of data and, to reduce the overhead of sending all of that data over the network, many of the Get- cmdlets support server-side filtering of the results.

The conventional way of filtering results in PowerShell is to pipeline them into Where-Object, Select-Object, and Sort-Object, for example:

```
Get-<Noun> | Where { $_.Size = 'Small' } | Sort 'Date' | Select -First 10
```

However, for most XenDesktop cmdlets the data is stored remotely and it would be slow and inefficient to retrieve large amounts of data over the network and then discard most of it. Instead, many of the Get- cmdlets provide filtering parameters that allow results to be processed on the server, returning only the required results.

You can filter results by most object properties using parameters derived from the property name. You can also sort results or limit them to a specified number of records:

```
Get-<Noun> -Size 'Small' -SortBy 'Date' -MaxRecordCount 10
```

You can express more complex filter conditions using a syntax and set of operators very similar to those used by PowerShell expressions.

Those cmdlets that support filtering have the following common parameters:

-MaxRecordCount <int>

Specifies the maximum number of results to return.  
For example, to return only the first nine results use:

```
Get-<Noun> -MaxRecordCount 9
```

If not specified, only the first 250 records are returned, and if more are available, a warning is produced:

WARNING: Only first 250 records returned. Use -MaxRecordCount to retrieve more.

You can suppress this warning by using -WarningAction or by specifying a value for -MaxRecordCount.

To retrieve all records, specify a large number for -MaxRecordCount. As the value is an integer, you can use the following:

```
Get-<Noun> -MaxRecordCount [int]::MaxValue
```

-ReturnTotalRecordCount [<SwitchParameter>]

When specified, this causes the cmdlet to output an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. For example:

```
Get-<Noun> -MaxRecordCount 9 -ReturnTotalRecordCount
```

....

```
Get-<Noun> : Returned 9 of 10 items
```

```
At line:1 char:18
```

```
+ Get-<Noun> <<<< -MaxRecordCount 9 -ReturnTotalRecordCount
```

```
+ CategoryInfo : OperationStopped: (:) [Get-<Noun>], PartialDataException
```

```
+ FullyQualifiedErrorId : PartialData,Citrix.<SDKName>.SDK.Get<Noun>
```

The count can be accessed using the TotalAvailableResultCount property:

```
$count = $error[0].TotalAvailableResultCount
```

-Skip <int>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

-SortBy <string>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order, respectively. Ascending order is assumed if no prefix is present.

Sorting occurs before -MaxRecordCount and -Skip parameters are applied. For example, to sort by Name and then by Count (largest first) use:

```
-SortBy 'Name,-Count'
```

By default, sorting by an enumeration property uses the numeric value of the elements. You can specify a different sort order by qualifying the name with an ordered list of elements or their numeric values, or <null> to indicate the placement of null values.

Elements not mentioned are placed at the end in their numeric order. For example, to sort by two different enums and then by the object id:

```
-SortBy 'MyState(StateC,<null>,StateA,StateB),Another(0,3,2,1),Id'
```

-Filter <String>

This parameter lets you specify advanced filter expressions, and supports combination of conditions with -and and -or, and grouping with braces. For example:

```
Get-<Noun> -Filter 'Name -like "High*" -or (Priority -eq 1 -and Severity -ge 2)'
```

The syntax is close enough to PowerShell syntax that you can use script blocks in most cases. This can be easier to read as it reduces quoting:

```
Get-<Noun> -Filter { Count -ne $null }
```

The full -Filter syntax is provided below.

## EXAMPLES

Filtering by strings performs a case-insensitive wildcard match.

Separate parameters are combined with an implicit -and operator.

Normal PowerShell quoting rules apply, so you can use single or double quotes, and omit the quotes altogether for many strings. The order of parameters does not make any difference. The following are equivalent:

```
Get-<Noun> -Company Citrix -Product Xen*
```

```
Get-<Noun> -Company "citrix" -Product '[X]EN*'
```

```
Get-<Noun> -Product "Xen*" -Company "CITRIX"
```

```
Get-<Noun> -Filter { Company -eq 'Citrix' -and Product -like 'Xen*' }
```

See [about\\_Quoting\\_Rules](#) and [about\\_Wildcards](#) for details about PowerShell handling of quotes and wildcards.

To avoid wildcard matching or include quote characters, you can escape the wildcards using the normal PowerShell escape mechanisms (see [about\\_Escape\\_Characters](#)), or switch to a filter expression and the `-eq` operator:

```
Get-<Noun> -Company "Abc[*]" # Matches Abc*
Get-<Noun> -Company "Abc`*" # Matches Abc*
Get-<Noun> -Filter { Company -eq "Abc*" } # Matches Abc*
Get-<Noun> -Filter { Company -eq "A`"B`C`" } # Matches A"B'C
```

Simple filtering by numbers, booleans, and TimeSpans perform direct equality comparisons, although if the value is nullable you can also search for null values. Here are some examples:

```
Get-<Noun> -Uid 123
Get-<Noun> -Enabled $true
Get-<Noun> -Duration 1:30:40
Get-<Noun> -NullableProperty $null
```

More comparisons are possible using advanced filtering with `-Filter`:

```
Get-<Noun> -Filter 'Capacity -ge 10gb'
Get-<Noun> -Filter 'Age -ge 20 -and Age -lt 40'
Get-<Noun> -Filter 'VolumeLevel -like "[123]"'
Get-<Noun> -Filter 'Enabled -ne $false'
Get-<Noun> -Filter 'NullableProperty -ne $null'
```

You can check boolean values without an explicit comparison operator, and you can also combine them with `-not`:

```
Get-<Noun> -Filter 'Enabled' # Equivalent to 'Enabled -eq $true'
Get-<Noun> -Filter '-not Enabled' # Equivalent to 'Enabled -eq $false'
```

See [about\\_Comparison\\_Operators](#) for an explanation of the operators, but note that only a subset of PowerShell operators are supported (`-eq`, `-ne`, `-gt`, `-ge`, `-lt`, `-le`, `-like`, `-notlike`, `-in`, `-notin`, `-contains`, `-notcontains`).

Enumeration values can either be specified using typed values or the string name of the enumeration value:

```
Get-<Noun> -Shape [Shapes]::Square
Get-<Noun> -Shape Circle
```

With filter expressions, typed values can be specified with simple variables or quoted strings. They also support enumerations with wildcards:

```
$s = [Shapes]::Square
Get-<Noun> -Filter { Shape -eq $s -or Shape -eq "Circle" }
Get-<Noun> -Filter { Shape -like 'C*' }
```

By their nature, floating point values, DateTime values, and TimeSpan values are best suited to relative comparisons rather than just equality. DateTime strings are converted using the locale and time zone of the user device, but you can use ISO8601 format strings (YYYY-MM-DDThh:mm:ss.sTZD) to avoid ambiguity. You can also use standard PowerShell syntax to create these values:

```
Get-<Noun> -Filter { StartTime -ge "2010-08-23T12:30:00.0Z" }
$d = [DateTime]"2010-08-23T12:30:00.0Z"
Get-<Noun> -Filter { StartTime -ge $d }
$d = (Get-Date).AddDays(-1)
Get-<Noun> -Filter { StartTime -ge $d }
```

Relative times are quite common and, when using filter expressions, you can also specify DateTime values using a relative format:

```
Get-<Noun> -Filter { StartTime -ge '-2' } # Two days ago
Get-<Noun> -Filter { StartTime -ge '-1:30' } # Hour and a half ago
Get-<Noun> -Filter { StartTime -ge '-0:0:30' } # 30 seconds ago
```

## ARRAY PROPERTIES

When filtering against list or array properties, simple parameters perform a case-insensitive wildcard match against each of the members. With filter expressions, you can use the -contains and -notcontains operators. Unlike PowerShell, these perform wildcard matching on strings.

Note that for array properties the naming convention is for the returned property to be plural, but the parameter used to search for any match is singular. The following are equivalent (assuming Users is an array property):

```
Get-<Noun> -User Fred*
Get-<Noun> -Filter { User -like "Fred*" }
Get-<Noun> -Filter { Users -contains "Fred*" }
```

You can also use the singular form with -Filter to search using other operators:

```
Match if any user in the list is called "Frederick"
Get-<Noun> -Filter { User -eq "Frederick" }
Match if any user in the list has a name alphabetically below 'F'
Get-<Noun> -Filter { User -lt 'F' }
```

## COMPLEX EXPRESSIONS

When matching against multiple values, you can use a sequence of comparisons joined with -or operators, or you can use -in and -notin:

```
Get-<Noun> -Filter { Shape -eq 'Circle' -or Shape -eq 'Square' }
$shapes = 'Circle','Square'
Get-<Noun> -Filter { Shape -in $shapes }
$sides = 1..4
Get-<Noun> -Filter { Sides -notin $sides }
```

Braces can be used to group complex expressions, and override the default left-to-right evaluation of -and and -or. You can also use -not to invert the sense of any sub-expression:

```
Get-<Noun> -Filter { Size -gt 4 -or (Color -eq 'Blue' -and Shape -eq 'Circle') }
Get-<Noun> -Filter { Sides -lt 5 -and -not (Color -eq 'Blue' -and Shape -eq 'Circle') }
```

## PAGING

The simplest way to page through data is to use the -Skip and -MaxRecordCount parameters. So, to read the first three pages of data with 10 records per page, use:

```
Get-<Noun> -Skip 0 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 10 -MaxRecordCount 10 <other filtering criteria>
Get-<Noun> -Skip 20 -MaxRecordCount 10 <other filtering criteria>
```

You must include the same filtering criteria on each call, and ensure that the data is sorted consistently.

The above approach is often acceptable, but as each call performs an independent query, data changes can result in records being skipped or appearing twice. One approach to improve this is to sort by a unique id field and then start the search for the next page at the unique id after the last unique id of the previous page. For example:

```
Get the first page
Get-<Noun> -MaxRecordCount 10 -SortBy SerialNumber

SerialNumber ...

A120004
A120007
... 7 other records ...
A120900

Get the next page
Get-<Noun> -MaxRecordCount 10 -Filter { FirstName -gt 'A120900' }

SerialNumber ...
```



----- ---  
A120901  
B220000  
...

## FILTER SYNTAX DEFINITION

<Filter> ::= <ScriptBlock> | <ComponentList>

<ScriptBlock> ::= "{" <ComponentList> "}"

<ComponentList> ::= <Component> <AndOrOperator> <ComponentList> |

<Component>

<Component> ::= <NotOperator> <Factor> |

<Factor>

<Factor> ::= "(" <ComponentList> ")" |

<PropertyName> <ComparisonOperator> <Value> |  
<PropertyName>

<AndOrOperator> ::= "-and" | "-or"

<NotOperator> ::= "-not" | "!"

<ComparisonOperator>

::= "-eq" | "-ne" | "-le" | "-ge" | "-lt" | "-gt" |  
"-like" | "-notlike" | "-contains" | "-notcontains" |  
"-in" | "-notin"

<PropertyName> ::= <simple name of property>

<Value> ::= <string literal> | <numeric literal> |

<scalar variable> | <array variable> |  
"\$null" | "\$true" | "\$false"

Numeric literals support decimal and hexadecimal literals, with optional multiplier suffixes (kb, mb, gb, tb, pb).

Dates and times can be specified as string literals. The current culture determines what formats are accepted. To avoid any ambiguity, use strings formatted to the ISO8601 standard. If not specified, the current time zone is used.

Relative date-time string literals are also supported, using a minus sign followed by a TimeSpan. For example, "-1:30" means 1 hour and 30 minutes ago.

# Add-SfServerToCluster

May 10, 2016

Adds a new server to an existing cluster.

## Syntax

```
Add-SfServerToCluster -ClusterId <Guid> -ServerName <String> [-StorefrontUrl <Uri>] [-FarmName <String>] [-XmlServices <Uri[]>] [-RunAsynchronously <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Adds a new server to an existing cluster. Optionally updates Farm and Storefront Url. After operation succeeds, all servers are configured identically.

## Related topics

[Get-SfCluster](#)

[New-SfCluster](#)

[Remove-SfServerFromCluster](#)

[Set-SfCluster](#)

## Parameters

**-ClusterId**<Guid>

The id of the cluster to perform operation on.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-ServerName**<String>

The name of the server to join to existing cluster. The name must be one of the values returned by Get-SfCluster

Required?	true
Default Value	
Accept Pipeline Input?	false

**-StorefrontUri<Uri>**

The url that will be used by Receivers to contact Storefront. Http or https absolute urls are accepted.

Required?	false
Default Value	Server name and http binding.
Accept Pipeline Input?	false

**-FarmName<String>**

Name of the farm that will be used within Store service. Either both FarmName and XmlServices need to be specified or none of them.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-XmlServices<Uri[]>**

Collection of the url of xml services that will be used inside a farm. The urls needed to be http or https, be absolute and share the same schema and port. Either both FarmName and XmlServices need to be specified or none of them.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-RunAsynchronously<Boolean>**

If set, the command will run asynchronously.

Required?	false
Default Value	false
Accept Pipeline Input?	false

### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.Storefront.Sdk.Task or Citrix.Storefront.DataModel.Cluster

Returns cluster description or a task, if ran asynchronously.

### Examples

#### ----- **EXAMPLE 1** -----

```
Add-SfServerToCluster -ClusterId (Guid) -ServerName NewSfServer -RunAsynchronously $true
```

Adds "NewSfServer" to cluster with id (Guid).

# Add-SfStorefrontAddress

May 10, 2016

Modifies a StoreFront address configuration by adding an additional StoreFront address to it.

### Syntax

Add-SfStorefrontAddress [-ByteArray] <Byte[]> -Name <String> -Url <String> -Enabled <Boolean> -Description <String> [<CommonParameters>]

### Detailed Description

Use this command to transform an existing StoreFront configuration into a new configuration, where the new configuration contains one additional address. The original configuration is supplied as an input, along with the properties of the new StoreFront address being added. The cmdlet outputs the modified configuration, which can then be passed to the Citrix Broker Service using the Add-BrokerMachineConfiguration command.

This command does not, by itself, have any persistent effects within XenDesktop. To make the change persistent, the new configuration byte array must first be transformed into a machine configuration within the Citrix Broker Service. To do this, use the New-BrokerMachineConfiguration command. You can then use the Add-BrokerMachineConfiguration and Set-BrokerMachineConfiguration commands to fully associate the new configuration with a delivery group.

### Related topics

[New-SfStorefrontAddress](#)

[Get-SfStorefrontAddress](#)

[New-BrokerMachineConfiguration](#)

[Add-BrokerMachineConfiguration](#)

[Set-BrokerMachineConfiguration](#)

### Parameters

**-ByteArray**<Byte[]>

Specifies the initial configuration, on which the new configuration is based. All of the addresses in the original configuration are also present in the new configuration, along with the additional address specified. This configuration byte array is obtained from an earlier call to New-SfStorefrontAddress, or from Get-BrokerMachineConfiguration.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-Name**<String>

Specifies the name of the new StoreFront.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Url**<String>

Specifies the URL to the StoreFront, such as "https://mysite.com/Citrix/StoreWeb".

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Enabled**<Boolean>

Specifies if the new StoreFront address should be enabled for user access.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Description**<String>

Specifies a human-readable description of the new StoreFront.

Required?	true
Default Value	
Accept Pipeline Input?	false

Input Type

System.Byte[] This cmdlet accepts configurations as pipeline input, as an alternative to supplying the ByteArray parameter.

Return Values

System.Byte[]

This cmdlet outputs the new, modified configuration. This differs from the original configuration in that it contains the additional StoreFront address.

Examples

----- **EXAMPLE 1** -----

C:\PS> \$newConfiguration = Add-SfStorefrontAddress -ByteArray \$originalConfiguration -Url "https://mysite.com/Citrix/StoreWeb" -Description "This StoreFront delivers my corporate applications" -Name  
This command transforms the configuration byte array specified by \$originalConfiguration, adds the new StoreFront details, and stores the resulting configuration in \$newConfiguration.

# Get-SfCluster

May 10, 2016

Gets all Storefront clusters present in the site.

## Syntax

```
Get-SfCluster [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets all Storefront clusters present in the site. There is one special Cluster with null id that lists the servers that are not part to any cluster (they are available to join any).

## Related topics

[New-SfCluster](#)

[Add-SfServerToCluster](#)

[Remove-SfServerFromCluster](#)

[Set-SfCluster](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Storefront.DataModel.Cluster[]

Returns array of clusters available in current deployment.



# Get-SfDBConnection

May 10, 2016

Gets the database string for the specified data store used by the Storefront Service.

## Syntax

```
Get-SfDBConnection [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the database connection string for the specified data store.

If the returned string is blank, no valid connection string has been specified. In this case the service is running, but is idle and awaiting specification of a valid connection string.

## Related topics

[Get-SfServiceStatus](#)

[Set-SfDBConnection](#)

[Test-SfDBConnection](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

system.string

The database connection string configured for the Storefront Service.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

NoDBConnections

The database connection string for the Storefront Service has not been specified.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

c:\PS>Get-SfDBConnection

Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True  
Get the database connection string for the Storefront Service.

# Get-SfDBSchema

May 10, 2016

Gets a script that creates the Storefront Service database schema for the specified data store.

## Syntax

```
Get-SfDBSchema [-DatabaseName <String>] [-ServiceGroupName <String>] [-ScriptType <ScriptTypes>] [-LocalDatabase] [-Sid <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Gets SQL scripts that can be used to create a new Storefront Service database schema, add a new Storefront Service to an existing site, remove a Storefront Service from a site, or create a database server logon for a Storefront Service. If no Sid parameter is provided, the scripts obtained relate to the currently selected Storefront Service instance, otherwise the scripts relate to Storefront Service instance running on the machine identified by the Sid provided. When obtaining the Evict script, a Sid parameter must be supplied. The current service instance is that on the local machine, or that explicitly specified by the last usage of the -AdminAddress parameter to a Storefront SDK cmdlet. The service instance used to obtain the scripts does not need to be a member of a site or to have had its database connection configured. The database scripts support only Microsoft SQL Server, or SQL Server Express, and require Windows integrated authentication to be used. They can be run using SQL Server's SQLCMD utility, or by copying the script into an SQL Server Management Studio (SSMS) query window and executing the query. If using SSMS, the query must be executed in 'SMDCMD mode'. The ScriptType parameter determines which script is obtained. If ScriptType is not specified, or is FullDatabase, the script contains:

- o Creation of service schema
- o Creation of database server logon
- o Creation of database user
- o Addition of database user to Storefront Service roles

If ScriptType is Instance, the returned script contains:

- o Creation of database server logon
- o Creation of database user
- o Addition of database user to Storefront Service roles

If ScriptType is Evict, the returned script contains:

- o Removal of Storefront Service instance from database
- o Removal of database user

If ScriptType is Login, the returned script contains:

- o Creation of database server logon only

If the service uses two data stores they can exist in the same database. You do not need to configure a database before using this command.

## Related topics

[Set-SfDBConnection](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database for which the schema will be generated.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ServiceGroupName**<String>

Specifies the name of the service group to be used when creating the database schema. The service group is a collection of all the Storefront services that share the same database instance and are considered equivalent; that is, all the services within a service group can be used interchangeably.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ScriptType**<ScriptTypes>

Specifies the type of database script returned. Available script types are:

### Database

Returns a full database script that can be used to create a database schema for the Storefront Service in a database instance that does not already contain a schema for this service. The DatabaseName and ServiceGroupName parameters must be specified to create a script of this type.

### Instance

Returns a permissions script that can be used to add further Storefront services to an existing database instance that already contains the full Storefront service schema, associating the services to the Service Group. The Sid parameter can optionally be specified to create a script of this type.

### Login

Returns a database logon script that can be used to add the required logon accounts to an existing database instance that contains the Storefront Service schema. This is used primarily when creating a mirrored database environment. The

DatabaseName parameter must be specified to create a script of this type.

#### Evict

Returns a script that can be used to remove the specified Storefront Service from the database entirely. The DatabaseName and Sid parameters must be specified to create a script of this type.

Required?	false
Default Value	Database
Accept Pipeline Input?	false

#### -LocalDatabase<SwitchParameter>

Specifies whether the database script is to be used in a database instance run on the same controller as other services in the service group. Including this parameter ensures the script creates only the required permissions for local services to access the database schema for Storefront services.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### -Sid<String>

Specifies the SID of the controller on which the Storefront Service instance to remove from the database is running.

Required?	false
Default Value	
Accept Pipeline Input?	true (ByValue)

#### -AdminAddress<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

System.string

A string containing the required SQL script for application to a database.

## Notes

The scripts returned support Microsoft SQL Server Express Edition, Microsoft SQL Server Standard Edition, and Microsoft SQL Server Enterprise Edition databases only, and are generated on the assumption that integrated authentication will be used.

If the ScriptType parameter is not included or set to 'FullDatabase', the full database script is returned, which will:

Create the database schema.

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist).

If the ScriptType parameter is set to 'Instance', the script will:

Create the user and the role (providing the schema does not already exist).

Create the logon (providing the schema does not already exist) and associate it with a user.

If the ScriptType parameter is set to 'Login', the script will:

Create the logon (providing the schema does not already exist) and associate it with a pre-existing user of the same name.

If the LocalDatabase parameter is included, the NetworkService account will be added to the list of accounts permitted to access the database. This is required only if the database is run on a controller.

If the command fails, the following errors can be returned.

## Error Codes

-----

GetSchemasFailed

The database schema could not be found.

ActiveDirectoryAccountResolutionFailed

The specified Active Directory account or Group could not be found.

DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Get-SfDBSchema -DatabaseName MyDB -ServiceGroupName MyServiceGroup > c:\SfSchema.sql
```

Get the full database schema for site data store of the Storefront Service and copy it to a file called 'c:\SfSchema.sql'.

This script can then be used to create the schema in a pre-existing database named 'MyDB' that does not already contain a Storefront Service site schema.

##### ----- EXAMPLE 2 -----

```
c:\PS>Get-SfDBSchema -DatabaseName MyDB -scriptType Login > c:\StorefrontLogins.sql
```

Get the logon scripts for the Storefront Service.

# Get-SfDBVersionChangeScript

May 10, 2016

Gets a script that updates the Storefront Service database schema.

## Syntax

```
Get-SfDBVersionChangeScript -DatabaseName <String> -TargetVersion <Version> [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Returns a database script that can be used to upgrade or downgrade the site or secondary schema for the Storefront Service from the current schema version to a different version.

## Related topics

[Get-SfInstalledDBVersion](#)

## Parameters

**-DatabaseName**<String>

Specifies the name of the database instance to which the update applies.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-TargetVersion**<Version>

Specifies the version of the database you want to update to.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

--	--



Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.Management.Automation.PSObject

A PSObject containing the required SQL script for application to a database.

## Notes

The PSObject returned by this cmdlet contains the following properties:

-- Script The raw text of the SQL script to apply the update, or null in the case when no upgrade path to the specified target version exists.

-- NeedExclusiveAccess Indicates whether all services in the service group must be shut down during the update or not.

-- CanUndo Indicates whether the generated script allows the updated schema to be reverted to the state prior to the update.

Scripts to update the schema version are stored in the database so any service in the service group can obtain these scripts. Extreme caution should be exercised when using update scripts. Citrix recommends backing up the database before attempting to upgrade the schema. Database update scripts may require exclusive use of the schema and so may not be able to execute while any Storefront services are running. However, this depends on the specific update being carried out.

After a schema update has been carried out, services that require the previous version of the schema may cease to operate. The ServiceState parameter reported by the Get-SfServiceStatus command provides information about service compatibility. For example, if the schema has been upgraded to a more recent version that a service cannot use, the service reports "DBNewerVersionThanService".

If the command fails, the following errors can be returned.

## Error Codes

-----

### NoOp

The operation was successful but had no effect.

### NoDBConnections

The database connection string for the Storefront Service has not been specified.

### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- **EXAMPLE 1** -----

```
C:\PS> $update = Get-SfDBVersionChangeScript -DatabaseName MyDb -TargetVersion 1.0.75.0
```

```
C:\PS> $update.Script > update_75.sql
```

Gets an SQL update script to update the current schema to version 1.0.75.0. The resulting update\_75.sql script is suitable for direct use with the SQL Server SQLCMD utility.

# Get-SfInstalledDBVersion

May 10, 2016

Gets a list of all available database schema versions for the Storefront Service.

## Syntax

```
Get-SfInstalledDBVersion [-Upgrade] [-Downgrade] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns the current version of the Storefront Service database schema, if no flags are set, otherwise returns versions for which upgrade or downgrade scripts are available and have been stored in the database.

## Related topics

## Parameters

**-Upgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be updated should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-Downgrade**<SwitchParameter>

Specifies that only schema versions to which the current database version can be reverted should be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.

Accept Pipeline Input?	false
------------------------	-------

## Return Values

### System.Version

The Get-SfInstalledDbVersion command returns objects containing the new definition of the Storefront Service database schema version.

Major <Integer>

Minor <Integer>

Build <Integer>

Revision <Integer>

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

Both the Upgrade and Downgrade flags were specified.

#### NoOp

The operation was successful but had no effect.

#### NoDBConnections

The database connection string for the Storefront Service has not been specified.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-SfInstalledDBVersion
```

Major	Minor	Build	Revision
-------	-------	-------	----------

----	----	----	-----
------	------	------	-------

5	6	0	0
---	---	---	---

Get the currently installed version of the Storefront Service database schema.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-SfInstalledDBVersion -Upgrade
```

Major	Minor	Build	Revision
-------	-------	-------	----------

----	----	----	-----
------	------	------	-------

6	0	0	0
---	---	---	---

Get the versions of the Storefront Service database schema for which upgrade scripts are supplied.

# Get-SflsStorefrontInstalled

May 10, 2016

Tells whether StoreFront Services and Privileged Service are installed.

## Syntax

Get-SflsStorefrontInstalled [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

## Related topics

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

bool

True if both StoreFront Services and Privileged Service are installed, false otherwise.

# Get-SfService

May 10, 2016

Gets the service record entries for the Storefront Service.

## Syntax

```
Get-SfService [-Metadata <String>] [-Property <String[]>] [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns instances of the Storefront Service that the service publishes. The service records contain account security identifier information that can be used to remove each service from the database.

A database connection for the service is required to use this command.

## Related topics

### Parameters

#### **-Metadata**<String>

Gets records with matching metadata entries.

The value being compared with is a concatenation of the key name, a colon, and the value. For example: -Metadata "abc:x\*" matches records with a metadata entry having a key name of "abc" and a value starting with the letter "x".

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-Property**<String[]>

Specifies the properties to be returned. This is similar to piping the output of the command through Select-Object, but the properties are filtered more efficiently at the server.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Sf\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

#### **-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

#### **-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by -ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

#### **-Sort By**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false



### **-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Sf\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.Storefront.Sdk.Service

The Get-SfServiceInstance command returns an object containing the following properties.

Uid <Integer>

Specifies the unique identifier for the service in the group. The unique identifier is an index number.

ServiceHostId <Guid>

Specifies the unique identifier for the service instance.

DNSName <String>

Specifies the domain name of the host on which the service runs.

MachineName <String>

Specifies the short name of the host on which the service runs.

CurrentState <Citrix.Fma.Sdk.ServiceCore.ServiceState>

Specifies whether the service is running, started but inactive, stopped, or failed.

LastStartTime <DateTime>

Specifies the date and time at which the service was last restarted.

LastActivityTime <DateTime>

Specifies the date and time at which the service was last stopped or restarted.

OSType

Specifies the operating system installed on the host on which the service runs.

OSVersion

Specifies the version of the operating system installed on the host on which the service runs.

ServiceVersion

Specifies the version number of the service instance. The version number is a string that reflects the full build version of the service.

DatabaseUserName <string>

Specifies for the service instance the Active Directory account name with permissions to access the database. This will be either the machine account or, if the database is running on a controller, the NetworkService account.

Sid <string>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

ActiveSiteServices <string[]>

Specifies the names of active site services currently running in the service. Site services are components that perform long-running background processing in some services. This field is empty for services that do not contain site services.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

UnknownObject

One of the specified objects was not found.

PartialData

Only a subset of the available data was returned.

InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

CouldNotQueryDatabase

The query required to get the database was not defined.

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-SfService
```

```
Uid : 1
ServiceHostId : aef6f464-f1ee-4042-a523-66982e0cecd0
DNSName : MyServer.company.com
MachineName : MYSERVER
CurrentState : On
LastStartTime : 04/04/2011 15:25:38
LastActivityTime : 04/04/2011 15:33:39
OSType : Win32NT
OSVersion : 6.1.7600.0
ServiceVersion : 5.1.0.0
DatabaseUserName : NT AUTHORITY\NETWORK SERVICE
SID : S-1-5-21-2316621082-1546847349-2782505528-1165
ActiveSiteServices : {MySiteService1, MySiteService2...}
Get all the instances of the Storefront Service running in the current service group.
```

# Get-SfServiceAddedCapability

May 10, 2016

Gets any added capabilities for the Storefront Service on the controller.

## Syntax

```
Get-SfServiceAddedCapability [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables updates to the Storefront Service on the controller to be detected.

You do not need to configure a database connection before using this command.

## Related topics

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

System.String

String containing added capabilities.

## Notes

If the command fails, the following errors can be returned.

## Error Codes

-----

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-SfServiceAddedCapability
```

Get the added capabilities of the Storefront Service.

# Get-SfServiceInstance

May 10, 2016

Gets the service instance entries for the Storefront Service.

## Syntax

```
Get-SfServiceInstance [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns service interfaces published by the instance of the Storefront Service. Each instance of a service publishes multiple interfaces with distinct interface types, and each of these interfaces is represented as a ServiceInstance object. Service instances can be used to register the service with a central configuration service so that other services can use the functionality.

You do not need to configure a database connection to use this command.

## Related topics

### Parameters

**-AdminAddress** <String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Storefront.Sdk.ServiceInstance

The Get-SfServiceInstance command returns an object containing the following properties.

ServiceGroupUid <Guid>

Specifies the unique identifier for the service group of which the service is a member.

ServiceGroupName <String>

Specifies the name of the service group of which the service is a member.

ServiceInstanceUID <Guid>

Specifies the unique identifier for registered service instances, which are service instances held by and obtained from a

central configuration service. Unregistered service instances do not have unique identifiers.

ServiceType <String>

Specifies the service instance type. For this service, the service instance type is always Sf.

Address

Specifies the address of the service instance. The address can be used to access the service and, when registered in the central configuration service, can be used by other services to access the service.

Binding

Specifies the binding type that must be used to communicate with the service instance. In this release of XenDesktop, the binding type is always 'wcf\_HTTP\_kerb'. This indicates that the service provides a Windows Communication Foundation endpoint that uses HTTP binding with integrated authentication.

Version

Specifies the version of the service instance. The version number is used to ensure that the correct versions of the services are used for communications.

ServiceAccount <String>

Specifies the Active Directory account name for the machine on which the service instance is running. The account name is used to provide information about the permissions required for interservice communications.

ServiceAccountSid <String>

Specifies the Active Directory account security identifier for the machine on which the service instance is running.

InterfaceType <String>

Specifies the interface type. Each service can provide multiple service instances, each for a different purpose, and the interface defines the purpose. Available interfaces are:

SDK - for PowerShell operations

InterService - for operations between different services

Peer - for communications between services of the same type

Metadata <Citrix.Storefront.Sdk.Metadata[]>

The collection of metadata associated with registered service instances, which are service instances held by and obtained from a central configuration service. Metadata is not stored for unregistered service instances.

Notes

If the command fails, the following errors can be returned.

Error Codes

-----

## DatabaseError

An error occurred in the service while attempting a database operation.

## DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

## DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

## PermissionDenied

You do not have permission to execute this command.

## AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

## CommunicationError

There was a problem communicating with the remote service.

## ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-SfServiceInstance
```

```
Address : http://MyServer.com:80/Citrix/StorefrontContract
Binding : wcf_HTTP_kerb
InterfaceType : SDK
Metadata :
MetadataMap :
ServiceAccount : ENG\MyAccount$
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164
ServiceGroupName : MyServiceGroup
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000
ServiceType : Sf
Version : 1
```

```
Address : http://MyServer.com:80/Citrix/StorefrontContract/IServiceApi
Binding : wcf_HTTP_kerb
InterfaceType : InterService
Metadata :
MetadataMap :
```



ServiceAccount : ENG\MyAccount  
ServiceAccountSid : S-1-5-21-2406005612-3133289213-1653143164  
ServiceGroupName : MyServiceGroup  
ServiceGroupUid : a88d2f6b-c00a-4f76-9c38-e8330093b54d  
ServiceInstanceUid : 00000000-0000-0000-0000-000000000000  
ServiceType : Sf  
Version : 1

Get all instances of the Storefront Service running on the specified machine. For remote services, use the AdminAddress parameter to define the service for which the interfaces are required. If the AdminAddress parameter has not been specified for the runspace, service instances running on the local machine are returned.

# Get-SfServiceStatus

May 10, 2016

Gets the current status of the Storefront Service on the controller.

## Syntax

```
Get-SfServiceStatus [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables the status of the Storefront Service on the controller to be monitored. If the service has multiple data stores it will return the overall state as an aggregate of all the data store states. For example, if the site data store status is OK and the secondary data store status is DBUnconfigured then it will return DBUnconfigured.

## Related topics

[Set-SfDBConnection](#)

[Test-SfDBConnection](#)

[Get-SfDBConnection](#)

[Get-SfDBSchema](#)

## Parameters

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Get-SfServiceStatus command returns an object containing the status of the Storefront Service together with extra diagnostics information.

DBUnconfigured

The Storefront Service does not have a database connection configured.

DBRejectedConnection

The database rejected the logon attempt from the Storefront Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

#### InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the Storefront Service schema has not been added to the database.

#### DBNotFound

The specified database could not be located with the configured connection string.

#### DBNewerVersionThanService

The version of the Storefront Service currently in use is incompatible with the version of the Storefront Service schema on the database. Upgrade the Storefront Service to a more recent version.

#### DBOlderVersionThanService

The version of the Storefront Service schema on the database is incompatible with the version of the Storefront Service currently in use. Upgrade the database schema to a more recent version.

#### DBVersionChangeInProgress

A database schema upgrade is currently in progress.

#### OK

The Storefront Service is running and is connected to a database containing a valid schema.

#### Failed

The Storefront Service has failed.

#### Unknown

(0) The service status cannot be determined.

#### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-SfServiceStatus
```

DBUnconfigured

Get the current status of the Storefront Service.

# Get-SfStorefrontAddress

May 10, 2016  
Gets the high-level description of a configuration for StoreFront addresses, based on a configuration byte array.

### Syntax

Get-SfStorefrontAddress [-ByteArray] <Byte[]> [<CommonParameters>]

### Detailed Description

Use this command to convert a configuration byte array into a set of named property settings. The byte array will either have been retrieved from the Citrix Broker Service, or from the New-SfStorefrontAddress cmdlet.

### Related topics

- [New-SfStorefrontAddress](#)
- [Add-SfStorefrontAddress](#)

- New-BrokerMachineConfiguration
- Add-BrokerMachineConfiguration
- Set-BrokerMachineConfiguration

### Parameters

**-ByteArray**<Byte[]>

Specifies the low-level byte array (blob) to be interpreted.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### Input Type

System.Byte[] The cmdlet accepts the ByteArray parameter as pipeline input.

### Return Values

Citrix.Storefront.Sdk.SfStorefrontAddress

This cmdlet outputs one SfStorefrontAddress object for each address that is configured within the slot. Each object has the following properties:

- Name - Specifies the name of the StoreFront.
- Url - Specifies the URL to the StoreFront, such as "https://mysite.com/Citrix/StoreWeb".
- Enabled - Specifies whether the StoreFront is enabled for users.
- Description - Specifies the human-readable name of the StoreFront.

### Examples

----- **EXAMPLE 1** -----

```
C:\PS> $configuration = New-SfStorefrontAddress -Url "https://mysite.com/Citrix/StoreWeb" -Description "This StoreFront delivers my corporate applications" -Name "StoreFront1" -Enabled $true
C:\PS> Get-SfStorefrontAddress -ByteArray $configuration
```

Name	Url	Enabled	Description
StoreFront1	https://mysite.com/Citrix/StoreWeb	True	This StoreFront delivers my corporate applications.

This example shows a new configuration byte array being created to specify a single StoreFront address. The configuration byte array is then provided as input to the Get-SfStorefrontAddress command, which interprets and outputs the same fields.

# Get-SfTask

May 10, 2016

Gets the task history for the Storefront Service.

## Syntax

```
Get-SfTask [-TaskId] <Guid> [-ReturnTotalRecordCount] [-MaxRecordCount <Int32>] [-Skip <Int32>] [-SortBy <String>] [-Filter <String>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Returns a list of tasks that have run or are currently running within the Storefront Service.

## Related topics

[Remove-SfTask](#)

[Add-SfTaskMetadata](#)

[Remove-SfTaskMetadata](#)

## Parameters

**-TaskId**<Guid>

Specifies the task identifier to be returned.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-ReturnTotalRecordCount**<SwitchParameter>

When specified, the cmdlet outputs an error record containing the number of records available. This error record is additional information and does not affect the objects written to the output pipeline. See [about\\_Sf\\_Filtering](#) for details.

Required?	false
Default Value	False
Accept Pipeline Input?	false

**-MaxRecordCount**<Int32>

Specifies the maximum number of records to return.

Required?	false
Default Value	250
Accept Pipeline Input?	false

**-Skip**<Int32>

Skips the specified number of records before returning results. Also reduces the count returned by - ReturnTotalRecordCount.

Required?	false
Default Value	0
Accept Pipeline Input?	false

**-SortBy**<String>

Sorts the results by the specified list of properties. The list is a set of property names separated by commas, semi-colons, or spaces. Optionally, prefix each name with a + or - to indicate ascending or descending order. Ascending order is assumed if no prefix is present.

Required?	false
Default Value	The default sort order is by name or unique identifier.
Accept Pipeline Input?	false

**-Filter**<String>

Gets records that match a PowerShell-style filter expression. See about\_Sf\_Filtering for details.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### UnknownObject

One of the specified objects was not found.

#### PartialData

Only a subset of the available data was returned.

#### InvalidFilter

A filtering expression was supplied that could not be interpreted for this cmdlet.

#### CouldNotQueryDatabase

The query required to get the database was not defined.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.



AuthorizationError

There was a problem communicating with the Citrix Delegated Administration service.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

# New-SfCluster

May 10, 2016

Creates new Storefront cluster with default set of services.

## Syntax

```
New-SfCluster -ServerName <String> -FarmName <String> -XmlServices <Uri[]> [-StorefrontUrl <Uri>] [-RunAsynchronously <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

The server will have a default set of services containing fully-functionaly Storefront server with Authentication, Store, Receiver for Web and Desktop Appliance.

## Related topics

[Get-SfCluster](#)

[Add-SfServerToCluster](#)

[Remove-SfServerFromCluster](#)

[Set-SfCluster](#)

## Parameters

**-ServerName**<String>

The name of the server to build a cluster on. The name must be one of the values returned by Get-SfCluster

Required?	true
Default Value	
Accept Pipeline Input?	false

**-FarmName**<String>

Name of the farm that will be used within Store service.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-XmlServices**<Uri[]>

Collection of the url of xml services that will be used inside a farm. The urls needed to be http or https, be absolute and share the same schema and port.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-StorefrontUrl**<Uri>

The url that will be used by Receivers to contact Storefront. Http or https absolute urls are accepted.

Required?	false
Default Value	Server name and http binding.
Accept Pipeline Input?	false

**-RunAsynchronously**<Boolean>

If set, the command will run asynchronously.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Storefront.Sdk.Task or Citrix.Storefront.DataModel.Cluster

Returns cluster description or a task, if ran asynchronously.

## Examples

### ----- EXAMPLE 1 -----

**New-SfCluster -FarmName "XdSiteName" -XmlServices http://farm1,http://farm2 -ServerName SfServer -RunAsynchronously \$true**  
Creates a new cluster on server "SfServer". The server will have a default set of services containing fully-functionally Storefront server with Authentication, Store, Receiver for Web, Desktop Appliance site and the required infrastructure. The Store service will have a farm named "XdSiteName" that will contain servers farm1 and farm2, whose will be contacted using http on default port 80.

# New-SfStorefrontAddress

May 10, 2016  
Creates a new StoreFront address configuration, specifying a single address.

## Syntax

New-SfStorefrontAddress -Name <String> -Url <String> -Enabled <Boolean> -Description <String> [<CommonParameters>]

## Detailed Description

Use this command when you want to create a new StoreFront configuration byte array from scratch, rather than modifying an existing one. You must define the URL for the StoreFront, and some additional details.

This command does not, by itself, have any persistent effects within XenDesktop. To make the change persistent, the new configuration byte array must first be transformed into a machine configuration within the Citrix Broker Service. To do this, use the New-BrokerMachineConfiguration command. You can then use the Add-BrokerMachineConfiguration and Set-BrokerMachineConfiguration commands to fully associate the new configuration with a delivery group.

## Related topics

[Add-SfStorefrontAddress](#)

[Get-SfStorefrontAddress](#)

[New-BrokerMachineConfiguration](#)

[Add-BrokerMachineConfiguration](#)

[Set-BrokerMachineConfiguration](#)

## Parameters

**-Name**<String>

Specifies the name of the new StoreFront.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Url**<String>

Specifies the URL to the StoreFront, such as "https://mysite.com/Citrix/StoreWeb".

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Enabled**<Boolean>

Specifies if the new StoreFront address should be enabled for user access.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Description**<String>

Specifies a human-readable description of the new StoreFront.

Required?	true
Default Value	
Accept Pipeline Input?	false

Input Type

None

Return Values

System.Byte[]

The new configuration set, with all of the given modifications applied.

Examples

----- **EXAMPLE 1** -----

C:\PS> \$configuration = New-SfStorefrontAddress -Url "https://mysite.com/Citrix/StoreWeb" -Description "This StoreFront delivers my corporate applications" -Name "StoreFront1" -Enabled \$true

C:\PS> Get-SfStorefrontAddress -ByteArray \$configuration

Name	Url	Enabled	Description
------	-----	---------	-------------

StoreFront1	https://mysite.com/Citrix/StoreWeb	True	This StoreFront delivers my corporate applications.
-------------	------------------------------------	------	-----------------------------------------------------

This example shows a new configuration byte array being created to specify a single StoreFront address. The configuration byte array is then provided as input to the Get-SfStorefrontAddress command, which interprets and outputs the same fields.

# Remove-SfServerFromCluster

May 10, 2016

Removes server from the cluster.

## Syntax

```
Remove-SfServerFromCluster -ClusterId <Guid> -ServerName <String> [-StorefrontUrl <Uri>] [-FarmName <String>] [-XmlServices <Uri[]>] [-RunAsynchronously <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Removes server from the cluster and propagates information to other servers. The configuration of the server is wiped out, so the server can be reused.

## Related topics

[Get-SfCluster](#)

[New-SfCluster](#)

[Add-SfServerToCluster](#)

[Set-SfCluster](#)

## Parameters

**-ClusterId**<Guid>

The id of the cluster to perform operation on.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-ServerName**<String>

The name of the server to remove from cluster. The name must be one of the values returned by Get-SfCluster.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-StorefrontUri**<Uri>

The url that will be used by Receivers to contact Storefront. Http or https absolute urls are accepted.

Required?	false
Default Value	Server name and http binding.
Accept Pipeline Input?	false

#### **-FarmName**<String>

Name of the farm that will be used within Store service. Either both FarmName and XmlServices need to be specified or none of them.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-XmlServices**<Uri[]>

Collection of the url of xml services that will be used inside a farm. The urls needed to be http or https, be absolute and share the same schema and port. Either both FarmName and XmlServices need to be specified or none of them.

Required?	false
Default Value	
Accept Pipeline Input?	false

#### **-RunAsynchronously**<Boolean>

If set, the command will run asynchronously.

Required?	false
Default Value	false
Accept Pipeline Input?	false



### **-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

Citrix.Storefront.Sdk.Task or Citrix.Storefront.DataModel.Cluster

Returns cluster description or a task, if ran asynchronously.

### Examples

#### ----- **EXAMPLE 1** -----

Remove-SfServerFromCluster -ClusterId (Guid) -ServerName BrokenSfServer -RunAsynchronously \$true  
Removes Server "BrokenSfServer" from a Cluster with id (Guid).

# Remove-SfServiceMetadata

May 10, 2016

Removes metadata from the given Service.

## Syntax

```
Remove-SfServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-SfServiceMetadata [-ServiceHostId] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-SfServiceMetadata [-InputObject] <Service[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-SfServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Service.

## Related topics

[Set-SfServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{"name1" = "val1"; "name2" = "val2"}) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name<String>**

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-SfService | % { Remove-SfServiceMetadata -Map $_.MetadataMap }
```

Remove all metadata from all Service objects.

# Remove-SfTask

May 10, 2016

Removes from the database completed tasks for the Storefront Service.

## Syntax

```
Remove-SfTask [-TaskId] <Guid> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Enables completed tasks that have run within the Storefront Service to be removed from the database.

## Related topics

[Get-SfTask](#)

[Add-SfTaskMetadata](#)

[Remove-SfTaskMetadata](#)

## Parameters

**-TaskId**<Guid>

Specifies the identifier for the task to be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByPropertyName)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

System.Management.Automation.PSObject Objects containing the TaskId parameter can be piped to the Remove-SfTask command.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### ServiceStatusInvalidDb

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration service.

#### OperationDeniedByConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.



# Remove-SfTaskMetadata

May 10, 2016

Removes metadata from the given Task.

## Syntax

```
Remove-SfTaskMetadata [-TaskId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-SfTaskMetadata [-TaskId] <Guid> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-SfTaskMetadata [-InputObject] <Task[]> -Name <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Remove-SfTaskMetadata [-InputObject] <Task[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability to remove metadata from the given Task.

## Related topics

[Set-SfTaskMetadata](#)

## Parameters

**-TaskId**<Guid>

Id of the Task

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Task[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can be either a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]"). The properties whose names match keys in the map will be removed.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

### **-Name<String>**

The metadata property to remove.

Required?	true
Default Value	
Accept Pipeline Input?	false

### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

### **-AdminAddress<String>**

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### InvalidParameterCombination

The cmdlet parameters are inconsistent.

#### UnknownObject

One of the specified objects was not found.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Get-SfTask | % { Remove-SfTaskMetadata -Map $_.MetadataMap }
```

Remove all metadata from all Task objects.

# Reset-SfServiceGroupMembership

May 10, 2016  
Reloads the access permissions and configuration service locations for the Storefront Service.

## Syntax

Reset-SfServiceGroupMembership [-ConfigServiceInstance] <ServiceInstance[]> [-LoggingId <Guid>] [-AdminAddress <String>]  
[<CommonParameters>]

## Detailed Description

Enables you to reload Storefront Service access permissions and configuration service locations. The Reset-SfServiceGroupMembership command must be run on at least one instance of the service type (Sf) after installation and registration with the configuration service. Without this operation, the Storefront services will be unable to communicate with other services in the XenDesktop deployment. When the command is run, the services are updated when additional services are added to the deployment, provided that the configuration service is not stopped. The Reset-SfServiceGroupMembership command can be run again to refresh this information if automatic updates do not occur when new services are added to the deployment. If more than one configuration service instance is passed to the command, the first instance that meets the expected service type requirements is used.

## Related topics

### Parameters

**-ConfigServiceInstance**<ServiceInstance[]>

Specifies the configuration service instance object that represents the service instance for the type 'InterService' that references a configuration service for the deployment.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
-----------	-------

Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Input Type

Citrix.Storefront.Sdk.ServiceInstance[] Service instances containing a ServiceInstance object that refers to the central configuration service interservice interface can be piped to the Reset-SfServiceGroupMembership command.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

#### NoSuitableServiceInstance

None of the supplied service instance objects were suitable for resetting service group membership.

#### DatabaseError

An error occurred in the service while attempting a database operation.

#### DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

### ----- EXAMPLE 1 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config | Reset-SfServiceGroupMembership
```

Reset the service group membership for a service in a deployment where the configuration service is configured and running on the same machine as the service.

### ----- EXAMPLE 2 -----

```
c:\PS>Get-ConfigRegisteredServiceInstance -ServiceType Config -AdminAddress OtherServer.example.com | Reset-StServiceGroupmembership
```

Reset the service group membership for a service in a deployment where the configuration service that is configured and running on a machine named 'OtherServer.example.com'.

# Set-SfCluster

May 10, 2016

Sets the parameters on the given cluster.

## Syntax

```
Set-SfCluster -ClusterId <Guid> [-StorefrontUrl <Uri>] [-FarmName <String>] [-XmlServices <Uri[]>] [-RunAsynchronously <Boolean>] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]
```

## Detailed Description

Sets the parameters on the given cluster and propagate the changes to all servers within a given cluster.

## Related topics

[Get-SfCluster](#)

[New-SfCluster](#)

[Add-SfServerToCluster](#)

[Remove-SfServerFromCluster](#)

## Parameters

**-ClusterId**<Guid>

The id of the cluster to perform operation on.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-StorefrontUrl**<Uri>

The url that will be used by Receivers to contact Storefront. Http or https absolute urls are accepted.

Required?	false
Default Value	Server name and http binding.
Accept Pipeline Input?	false

**-FarmName**<String>



Name of the farm that will be used within Store service. Either both FarmName and XmlServices need to be specified or none of them.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-XmlServices**<Uri[]>

Collection of the url of xml services that will be used inside a farm. The urls needed to be http or https, be absolute and share the same schema and port. Either both FarmName and XmlServices need to be specified or none of them.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-RunAsynchronously**<Boolean>

If set, the command will run asynchronously.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

## **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Storefront.Sdk.Task or Citrix.Storefront.DataModel.Cluster

Returns cluster description or a task, if ran asynchronously.

## Examples

### ----- **EXAMPLE 1** -----

```
Set-SfCluster -StorefrontUrl http://SfUrl -ClusterId (Guid) -RunAsynchronously $true
```

Sets a Storefront Url in a Cluster with id (Guid). Propagates changes to all servers that are part of the cluster.

# Set-SfDBConnection

May 10, 2016  
Configures a database connection for the Storefront Service.

## Syntax

Set-SfDBConnection [-DBConnection] <String> [-Force] [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Configures a connection to a database in which the Storefront Service can store its state. The service will attempt to connect and start using the database immediately after the connection is configured. The database connection string is updated to the specified value regardless of whether it is valid or not. Specifying an invalid connection string prevents a service from functioning until the error is corrected.

After a connection is configured, you cannot alter it without first clearing it (by setting the connection to \$null).

You do not need to configure a database connection to use this command.

## Related topics

[Get-SfServiceStatus](#)

[Get-SfDBConnection](#)

[Test-SfDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be used by the Storefront Service. Passing in \$null will clear any existing database connection configured.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-Force**<SwitchParameter>

If present, allows the local administrator to set the connection string to null when there are problems contacting the database or other services.

Required?	false
Default Value	false
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level

operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Set-SfDBConnection command returns an object containing the status of the Storefront Service together with extra diagnostics information.

DBUnconfigured

The Storefront Service does not have a database connection configured.

DBRejectedConnection

The database rejected the logon attempt from the Storefront Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the Storefront Service schema has not been added to the database.

DBNotFound

The specified database could not be located with the configured connection string.

DBNewerVersionThanService

The version of the Storefront Service currently in use is incompatible with the version of the Storefront Service schema on the database. Upgrade the Storefront Service to a more recent version.

DBOlderVersionThanService

The version of the Storefront Service schema on the database is incompatible with the version of the Storefront Service currently in use. Upgrade the database schema to a more recent version.

DBVersionChangeInProgress

A database schema upgrade is currently in progress.

OK

The Storefront Service is running and is connected to a database containing a valid schema.

Failed

The Storefront Service has failed.

Unknown

The status of the Storefront Service cannot be determined.

## Notes

If the command fails, the following errors can be returned.

### Error Codes

-----

InvalidDBConnectionString

The database connection string has an invalid format.

DatabaseConnectionDetailsAlreadyConfigured

There was already a database connection configured. After a configuration is set, it can only be set to \$null.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

## Examples

----- **EXAMPLE 1** -----

```
c:\PS>Set-SfDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Configures a database connection string for the Storefront Service.

----- **EXAMPLE 2** -----

```
c:\PS>Set-SfDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Configures an invalid database connection string for the Storefront Service.

# Set-SfServiceMetadata

May 10, 2016

Adds or updates metadata on the given Service.

## Syntax

```
Set-SfServiceMetadata [-ServiceHostId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-SfServiceMetadata [-ServiceHostId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

```
Set-SfServiceMetadata [-InputObject] <Service[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-SfServiceMetadata [-InputObject] <Service[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress
<String>] [<CommonParameters>]
```

## Detailed Description

Allows you to store additional custom data against given Service objects.

## Related topics

[Remove-SfServiceMetadata](#)

## Parameters

**-ServiceHostId**<Guid>

Id of the Service

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Service[]>

Objects to which metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name<String>**

Specifies the property name of the metadata to be added. The property must be unique for the Service specified. The property cannot contain any of the following characters \/:;#.\*?=<>|[]()"

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Value<String>**

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
-----------	-------



Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-SfServiceMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

InvalidParameterCombination

The cmdlet parameters are inconsistent.

UnknownObject

One of the specified objects was not found.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- EXAMPLE 1 -----

```
c:\PS>Set-SfServiceMetadata -ServiceHostId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
---	----
property	value

Add metadata with a name of 'property' and a value of 'value' to the Service with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Set-SfTaskMetadata

May 10, 2016

Adds or updates metadata on the given Task.

## Syntax

```
Set-SfTaskMetadata [-TaskId] <Guid> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-SfTaskMetadata [-TaskId] <Guid> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-SfTaskMetadata [-InputObject] <Task[]> -Map <PSObject> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

```
Set-SfTaskMetadata [-InputObject] <Task[]> -Name <String> -Value <String> [-LoggingId <Guid>] [-AdminAddress <String>]
[<CommonParameters>]
```

## Detailed Description

Provides the ability for additional custom data to be stored against given Task objects.

## Related topics

[Remove-SfTaskMetadata](#)

## Parameters

**-TaskId**<Guid>

Id of the Task

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue, ByPropertyName)

**-InputObject**<Task[]>

Objects to which the metadata is to be added.

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Map<PSObject>**

Specifies a dictionary of (name, value)-pairs for the properties. This can either be a hashtable (created with @{ "name1" = "val1"; "name2" = "val2" }) or a string dictionary (created with new-object "System.Collections.Generic.Dictionary[String,String]").

Required?	true
Default Value	
Accept Pipeline Input?	true (ByValue)

#### **-Name<String>**

Specifies the property name of the metadata to be added. The property must be unique for the Task specified. The property cannot contain any of the following characters \ ; # . \* ? = < > | [ ] ( ) "

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-Value<String>**

Specifies the value for the property.

Required?	true
Default Value	
Accept Pipeline Input?	false

#### **-LoggingId<Guid>**

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	

Default Value	
Accept Pipeline Input?	false

### **-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
Default Value	Localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

### Return Values

System.Collections.Generic.Dictionary[String,String]

Set-SfTaskMetadata returns a dictionary containing the new (name, value)-pairs.

Key <string>

Specifies the name of the property.

Value <string>

Specifies the value for the property.

### Notes

If the command fails, the following errors can be returned.

#### Error Codes

-----

InvalidParameterCombination

The cmdlet parameters are inconsistent.

UnknownObject

One of the specified objects was not found.

DatabaseError

An error occurred in the service while attempting a database operation.

DatabaseNotConfigured

The operation could not be completed because the database for the service is not configured.

#### DataStoreException

An error occurred in the service while attempting a database operation - communication with the database failed for various reasons.

#### PermissionDenied

You do not have permission to execute this command.

#### AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

#### ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

#### CommunicationError

There was a problem communicating with the remote service.

#### ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

#### Examples

##### ----- **EXAMPLE 1** -----

```
c:\PS>Set-SfTaskMetadata -TaskId 4CECC26E-48E1-423F-A1F0-2A06DDD0805C -Name property -Value value
```

Key	Value
---	----
property	value

Add metadata with a name of 'property' and a value of 'value' to the Task with the identifier '4CECC26E-48E1-423F-A1F0-2A06DDD0805C'.

# Test-SfDBConnection

May 10, 2016  
Tests a database connection for the Storefront Service.

## Syntax

Test-SfDBConnection [-DBConnection] <String> [-LoggingId <Guid>] [-AdminAddress <String>] [<CommonParameters>]

## Detailed Description

Tests a connection to the database in which the Storefront Service can store its state. The service will attempt to connect to the database without affecting the current connection to the database.

You do not have to clear the connection to use this command.

## Related topics

[Get-SfServiceStatus](#)

[Get-SfDBConnection](#)

[Set-SfDBConnection](#)

## Parameters

**-DBConnection**<String>

Specifies the database connection string to be tested by the Storefront Service.

Required?	true
Default Value	
Accept Pipeline Input?	false

**-LoggingId**<Guid>

Specifies the identifier of the high-level operation this cmdlet call forms a part of. Citrix Studio and Director typically create high-level operations. PowerShell scripts can also wrap a series of cmdlet calls in a high-level operation by way of the Start-LogHighLevelOperation and Stop-LogHighLevelOperation cmdlets.

Required?	false
Default Value	
Accept Pipeline Input?	false

**-AdminAddress**<String>

Specifies the address of a XenDesktop controller the PowerShell snap-in will connect to. You can provide this as a host name or an IP address.

Required?	false
-----------	-------

Default Value	localhost. Once a value is provided by any cmdlet, this value becomes the default.
Accept Pipeline Input?	false

## Return Values

Citrix.Fma.Sdk.Utilities.Service.ServiceStatusInfo

The Test-SfDBConnection command returns an object containing the status of the Storefront Service if the connection string of the specified data store were to be set to the string being tested, together with extra diagnostics information for the specified connection string.

DBRejectedConnection

The database rejected the logon attempt from the Storefront Service. This may be because the service attempted to log on with invalid credentials or because a database has not been installed in the specified location.

InvalidDBConfigured

The expected stored procedures are missing from the database. This may be because the Storefront Service schema has not been added to the database.

DBNotFound

The specified database could not be located with the configured connection string.

DBNewerVersionThanService

The version of the Storefront Service currently in use is incompatible with the version of the Storefront Service schema on the database. Upgrade the Storefront Service to a more recent version.

DBOlderVersionThanService

The version of the Storefront Service schema on the database is incompatible with the version of the Storefront Service currently in use. Upgrade the database schema to a more recent version.

DBVersionChangeInProgress

A database schema upgrade is currently in progress.

OK

The Set-SfDBConnection command would succeed if it were executed with the supplied connection string.

Failed

The Storefront Service has failed.

Unknown

The status of the Storefront Service cannot be determined.

## Notes

If the command fails, the following errors can be returned.

Error Codes

-----



InvalidDBConnectionString

The database connection string has an invalid format.

PermissionDenied

You do not have permission to execute this command.

AuthorizationError

There was a problem communicating with the Citrix Delegated Administration Service.

ConfigurationLoggingError

The operation could not be performed because of a configuration logging error.

CommunicationError

There was a problem communicating with the remote service.

ExceptionThrown

An unexpected error occurred. For more details, see the Windows event logs on the controller or the XenDesktop logs.

Examples

----- **EXAMPLE 1** -----

```
c:\PS>Test-SfDBConnection -DBConnection "Server=serverName\SQLEXPRESS;Initial Catalog = databaseName; Integrated Security = True"
```

OK

Tests a database connection string for the Storefront Service.

----- **EXAMPLE 2** -----

```
c:\PS>Test-SfDBConnection -DBConnection "Invalid Connection String"
```

Invalid database connection string format.

Tests an invalid database connection string for the Storefront Service.

# Hinweise zu Drittanbietern

May 10, 2016

XenApp und XenDesktop 7.6 enthalten ggf. Software von Drittanbietern, die gemäß den in den folgenden Dokumenten aufgeführten Bestimmungen lizenziert ist:

- [XenApp 7.6 und XenDesktop 7.6 Hinweise zu Drittanbietern](#)
- [FlexNet Publisher Documentation Supplement: Software Licenses](#)