Linux Virtual Delivery Agent 1903
Contents

What’s new 3
Fixed issues 4
Known issues 5
Third party notices 7
System requirements 7
Configure Delivery Controllers 10
Installation overview 12
Easy install 12
Use MCS to create Linux VMs 25
Install Linux Virtual Delivery Agent for RHEL/CentOS 38
Install Linux Virtual Delivery Agent for SUSE 68
Install Linux Virtual Delivery Agent for Ubuntu 91
Configure the Linux VDA 119
Integrate NIS with Active Directory 120
Publish applications 126
Print 128
File transfer 133
PDF printing 138
Configure graphics 139
Thinwire progressive display 147
Non-GRID 3D graphics 150
Configure policies 152
Policy support list 153
Configure IPv6 162
Configure CEIP 163
Configure USB redirection 167
Configure session reliability 176
Soft keyboard 178
Client IME 181
Support for multiple language inputs 182
Dynamic keyboard layout synchronization 183
Client IME user interface synchronization 185
HDX Insight 187
Adaptive transport 188
Tracing On 190
Shadow sessions 193
Support Citrix Workspace app for HTML5 199
Secure user sessions using SSL 200
Secure user sessions using DTLS 204
Pass-through authentication by using smart cards 205
Double-hop single sign-on authentication 216
Configure unauthenticated sessions 218
Configure LDAPS 220
Configure Xauthority 224
Federated Authentication Service 227
What’s new

March 28, 2019

What’s new in 1903

Version 1903 of the Linux VDA includes the following new features and enhancements:

Support for Ubuntu 18.04

Starting with this release, Citrix added Ubuntu 18.04 as a supported distribution. For more information, see Easy install and Install Linux Virtual Delivery Agent for Ubuntu.

Support for fast smart card

By default, fast smart card is enabled on the Linux VDA. Fast smart card is an improvement over the existing HDX PC/SC-based smart card redirection. It improves performance when smart cards are used in high-latency WAN situations. For more information, see Pass-through authentication by using smart cards.

ICA RTT metrics for Linux sessions are available in Citrix Director

ICA Round Trip Time (RTT) is the elapsed time from when you press a key until the response appears on the endpoint. To obtain the ICA RTT metric of a Linux session, find the session in Citrix Director and check the Session Details view (see the following screenshot).
To use this feature, you must:

- Use Citrix Director 1903 or later.
- Create the ICA round trip calculation and ICA round trip calculation interval policies in Citrix Studio. For more information, see Create a new policy using Studio.

Fixed issues

March 28, 2019

Compared to: Linux Virtual Delivery Agent 1811

Linux Virtual Delivery Agent 1903 contains the following fixes:

- Incorrect rules are used for keyboard mapping. [LC9684]
• The state of a previously active session on a VDA remains **Active** in Citrix Studio after the VDA is shut down. [LD0907]

**Known issues**

May 5, 2019

The following issues have been identified in this release:

• The Linux VDA does not work as expected when you use NVIDIA GRID 3D cards without enabling HDX 3D Pro. The issue occurs on RHEL 7.5 and earlier, SUSE 12.3 and earlier, and Ubuntu 16.04. The reason is that multiple OpenGL libraries cannot coexist in the graphics systems of these Linux distributions.

• The kworker and ksoftirqd kernel processes consume high CPU on RHEL 7.6. To resolve this issue, update the VHCL kernel module (see Knowledge Center article CTX253783). [LNXVDA-6498]

• Registration with the Delivery Controller might fail. The issue occurs on SUSE 12.3 systems with java-1_7_0-openjdk-1.7.0.201-43.18.1.x86_64 or a minor version later than OpenJDK 1.7.0.201 installed. [LNXVDA-6055]

To resolve the issue, run the following command to downgrade your Java runtime environment to java-1_7_0-openjdk-1.7.0.181-43.15.2.x86_64:

```bash
zypper install --oldpackage java-1_7_0-openjdk-1.7.0.181-43.15.2.x86_64
```

• An unexpected window appears during file download. The window does not affect the file download functionality and it automatically disappears after a while. [LNXVDA-5646]

• The default settings of PulseAudio cause the sound server program to exit after 20 seconds of inactivity. When PulseAudio exits, audio does not work. To work around this issue, set exit-idle-time=-1 in the /etc/pulse/daemon.conf file. [LNXVDA-5464]

• libtcmalloc 4.3.0 in SUSE 12.3 might cause processes to exit unexpectedly.

• The ctxhdx service might exit unexpectedly on the Ubuntu 16.04 and SUSE 12.3 VDAs. The issue occurs with the GNU C Library (glibc) Versions 2.22 through 2.24. The issue is fixed in glibc 2.25. If you are using the SUSE 12.3 distribution, you can install the patch that SUSE provides for fixing the issue. No fix is available for Ubuntu 16.04 at the time the Linux VDA 7.17 is released. [LNXVDA-4481]

• Sessions cannot be launched in Citrix Workspace app for Linux when SSL encryption is enabled. [RFLNX-1557]
• The indicator-datetime-service process does not consume the $TZ environment variable. When
the client and session locate in different time zones, the unity panel on Ubuntu 16.04 Unity Desk-
top does not show the time of the client. [LNXVDA-2128]

• Ubuntu graphics: In HDX 3D Pro, a black frame might appear around applications after resizing
the Desktop Viewer, or sometimes, the background can appear black.

• Printers created by the Linux VDA printing redirection might not be removed after logging out
of a session.

• CDM files are missing when a directory contains numerous files and subdirectories. This issue
might occur if the client side has too many files or directories.

• In this release, only UTF-8 encoding is supported for non-English languages.

• Citrix Workspace app for Android CAPS LOCK state might be reversed during session roaming.
The CAPS LOCK state can be lost when roaming an existing connection to Citrix Workspace app
for Android. As a workaround, use the Shift key on the extended keyboard to switch between
upper case and lower case.

• Shortcut keys with ALT do not always work when you connect to the Linux VDA using Citrix
Workspace app for Mac. Citrix Workspace app for Mac sends AltGr for both left and right Op-
tions/Alt keys by default. You can modify this behavior within the Citrix Workspace app settings
but the results vary with different applications.

• Registration fails when the Linux VDA is rejoined to the domain. The rejoining generates a fresh
set of Kerberos keys. But, the Broker might use a cached out-of-date VDA service ticket based
on the previous set of Kerberos keys. When the VDA tries to connect to the Broker, the Broker
might not be able to establish a return security context to the VDA. The usual symptom is that
the VDA registration fails.

This problem can eventually resolve itself when the VDA service ticket expires and is renewed.
But because service tickets are long-lived, it can take a long time.

As a workaround, clear the Broker’s ticket cache. Restart the Broker or run the following com-
mand on the Broker from a command prompt as Administrator:

```
  klist -li 0x3e4 purge
```

This command purges all service tickets in the LSA cache held by the Network Service principal
under which the Citrix Broker Service runs. It removes service tickets for other VDAs and poten-
tially other services. However, it is harmless – these service tickets can be reacquired from the
KDC when needed again.

• Audio plug-n-play is not supported. You can connect an audio capture device to the client ma-
chine before starting to record audio in the ICA session. If a capture device is attached after the
audio recording application has started, the application might become unresponsive and you must restart it. If a capture device is unplugged while recording, a similar issue might occur.

- Citrix Workspace app for Windows might experience audio distortion during audio recording.

Third party notices

February 11, 2019

Linux Virtual Desktop Version 1903 (PDF Download)

This release of the Linux VDA can include third party software licensed under the terms defined in the document.

System requirements

April 25, 2019

Linux distributions

Note:

System requirements for components not covered here (such as Citrix Workspace app) are described in their respective documentation sets.

The Linux VDA supports the following Linux distributions:

- SUSE Linux Enterprise:
  - Desktop 12 Service Pack 3
  - Server 12 Service Pack 3
- Red Hat Enterprise Linux
  - Workstation 7.6
  - Workstation 6.10
  - Workstation 6.9
  - Server 7.6
  - Server 6.10
  - Server 6.9
- CentOS Linux
Linux Virtual Delivery Agent 1903

- CentOS 7.6
- CentOS 6.10
- CentOS 6.9

- Ubuntu Linux
  - Ubuntu Desktop 18.04
  - Ubuntu Server 18.04
  - Ubuntu Live Server 18.04
  - Ubuntu Desktop 16.04
  - Ubuntu Server 16.04

- Pardus Linux
  - Pardus 17 (For information on the supported feature scope, see Knowledge Center article CTX238492.)

For a matrix of the Linux distributions and the Xorg versions that this version of the Linux VDA supports, see the following table. For more information, see XorgModuleABIVersions.

<table>
<thead>
<tr>
<th>Linux distribution</th>
<th>Xorg version</th>
</tr>
</thead>
<tbody>
<tr>
<td>RHEL 7.6, CentOS 7.6</td>
<td>1.20</td>
</tr>
<tr>
<td>RHEL 6.10/6.9, CentOS 6.10/6.9</td>
<td>1.17</td>
</tr>
<tr>
<td>Ubuntu 18.04</td>
<td>1.19</td>
</tr>
<tr>
<td>Ubuntu 16.04</td>
<td>1.18</td>
</tr>
<tr>
<td>SUSE 12.3</td>
<td>1.18</td>
</tr>
</tbody>
</table>

Do not use hwe xorg server 1.19 on Ubuntu 16.04.

In all cases, the supported processor architecture is x86-64.

**Note:**

Citrix’s support for a Linux OS platform and version expires when the support from the OS vendor expires.

**Important:**

Gnome and KDE desktops are supported in SUSE, RHEL, and CentOS. Unity desktop is supported in Ubuntu 16.04 and Gnome desktop is supported in Ubuntu 18.04. At least one desktop must be installed.
Citrix Virtual Desktops

The Linux VDA is compatible with all currently supported versions of Citrix Virtual Desktops. For information about the Citrix Virtual Desktops product lifecycle, and to find out when Citrix stops supporting specific versions of products, see the Citrix Product Lifecycle Matrix.

The configuration process for Linux VDAs differs slightly from Windows VDAs. However, any Delivery Controller farm is able to broker both Windows and Linux desktops.

Supported host platforms and virtualization environments

- XenServer
- VMware ESX and ESXi
- Microsoft Hyper-V
- Nutanix AHV
- Microsoft Azure Resource Manager
- Amazon Web Services (AWS)

Bare metal hosting is also supported.

Tip:
See the vendor’s documentation for the list of supported platforms.

Active Directory integration packages

The Linux VDA supports the following Active Directory integration packages or products:

- Samba Winbind
- Quest Authentication Services v4.1 or later
- Centrify DirectControl
- SSSD

Tip:
For the list of supported platforms, see the documentation from the vendors of the Active Directory integration packages.

HDX 3D Pro

The following hypervisors and NVIDIA GRID™ GPU are required to support HDX 3D Pro.
Hypervisors

- XenServer
- VMware ESX and ESXi
- Nutanix AHV

Note:
The hypervisors are compatible with certain Linux distributions.

GPU

The following GPUs are supported for GPU pass-through:

- NVIDIA GTX750Ti
- NVIDIA GRID™ - Tesla M60
- NVIDIA GRID™ - K2
- NVIDIA GRID™ - Tesla P40

The following GPUs are supported for vGPU:

- NVIDIA GRID™ - Tesla M60
- NVIDIA GRID™ - Tesla M10
- NVIDIA GRID™ - Tesla P40

Configure Delivery Controllers

March 25, 2019

XenDesktop 7.6 and earlier versions require changes to support the Linux VDA. For those versions, a hotfix or update script is required. The installation and verification instructions are provided in this article.

Update Delivery Controller configuration

For XenDesktop 7.6 SP2, apply Hotfix Update 2 to update the Broker for Linux Virtual Desktops. Hotfix Update 2 is available here:

- CTX142438: Hotfix Update 2 - For Delivery Controller 7.6 (32-bit) – English
- CTX142439: Hotfix Update 2 - For Delivery Controller 7.6 (64-bit) – English
For versions earlier than XenDesktop 7.6 SP2, you can use the PowerShell script named Update-BrokerServiceConfig.ps1 to update the Broker Service configuration. This script is available in the following package:

- citrix-linuxvda-scripts.zip

Repeat the following steps on every Delivery Controller in the farm:

1. Copy the Update-BrokerServiceConfig.ps1 script to the Delivery Controller machine.
2. Open a Windows PowerShell console in the context of the local administrator.
3. Browse to the folder containing the Update-BrokerServiceConfig.ps1 script.
4. Run the Update-BrokerServiceConfig.ps1 script:

```
1 .\Update-BrokerServiceConfig.ps1
```

**Tip:**
By default, PowerShell is configured to prevent the execution of PowerShell scripts. If the script fails to run, change the PowerShell execution policy before trying again:

```
1 Set-ExecutionPolicy Unrestricted
```

The Update-BrokerServiceConfig.ps1 script updates the Broker Service configuration file by using new WCF endpoints required by the Linux VDA and restarts the Broker Service. The script determines the location of the Broker Service configuration file automatically. A backup of the original configuration file is created in the same directory, with .prelinux appended to the file name.

These changes have no impact on the brokering of Windows VDAs configured to use the same Delivery Controller farm. A single Controller farm can manage and broker sessions for both Windows and Linux VDAs seamlessly.

**Note:**
The Linux VDA does not support Secure ICA for encryption. Enabling Secure ICA on the Linux VDA causes session launch failure.

**Verify Delivery Controller configuration**

When the required configuration changes have been applied to a Delivery Controller, the EndpointLinux string appears five times in the %PROGRAMFILES%\Citrix\Broker\Service\BrokerService.exe.config file.

From the Windows command prompt, log on as a local administrator to check:
Installation overview

March 25, 2019

There are three options for you to install the Linux VDA:

- **Easy install.** After installing the Linux VDA package on a machine, you can configure the running environment by using the ctxinstall.sh script. For more information, see [Easy install](#).
- **MCS.** You can use MCS to create Linux VMs in batches where the Linux VDA package is also installed. For more information, see [Use MCS to create Linux VMs](#).

  **Note:**
  When using MCS to create Linux VMs, do not use easy install on the template VM.

- **Manual installation.** You can use the following general steps to install the Linux VDA. Variations and specific commands are documented by distribution. For more information, see [Install Linux Virtual Delivery Agent for RHEL/CentOS](#), [Install Linux Virtual Delivery Agent for SUSE](#), and [Install Linux Virtual Delivery Agent for Ubuntu](#).

  1. Prepare for installation.
  2. Prepare the hypervisor.
  3. Add the Linux virtual machine (VM) to the Windows domain.
  4. Install the Linux VDA.
  5. Configure the Linux VDA.
  6. Create the machine catalog in Citrix Virtual Apps or Citrix Virtual Desktops.
  7. Create the delivery group in Citrix Virtual Apps or Citrix Virtual Desktops.

**Easy install**

March 25, 2019

Easy install is officially supported as of Version 7.13 of the Linux VDA. This feature helps you set up the running environment of the Linux VDA by installing the necessary packages and customizing the configuration files automatically.
Note:
Do not use easy install on the template VM when you use MCS to create Linux VMs.

Supported distributions

<table>
<thead>
<tr>
<th></th>
<th>Winbind</th>
<th>SSSD</th>
<th>Centrify</th>
</tr>
</thead>
<tbody>
<tr>
<td>RHEL 7.6</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>RHEL 6.10</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>RHEL 6.9</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>CentOS 7.6</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>CentOS 6.10</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>CentOS 6.9</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Ubuntu 18.04</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Ubuntu 16.04</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>SUSE 12.3</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Use easy install

To use this feature, do the following:

1. Prepare configuration information and the Linux machine.
2. Install the Linux VDA package.
   - Go to the Citrix website and download the appropriate Linux VDA package based on your Linux distribution.
3. Set up the runtime environment to complete the Linux VDA installation.

Step 1: Prepare configuration information and the Linux machine

Collect the following configuration information needed for easy install:

- Host name - Host name of the machine on which the Linux VDA is to be installed
- IP address of Domain Name Server
- IP address or string name of NTP Server
- Domain Name - The NetBIOS name of the domain
- Realm Name - The Kerberos realm name
- FQDN of Active Domain - Fully qualified domain name
Important:

- To install the Linux VDA, verify that the repositories are added correctly on the Linux machine.
- To launch a session, verify that the X Window system and desktop environments are installed.

Considerations

- The workgroup name is the domain name by default. To customize the workgroup in your environment, do the following:
  a. Create the /tmp/ctxinstall.conf file on the Linux VDA machine.
  b. Add the workgroup=<your workgroup> line to the file.
- Centrify does not support pure IPv6 DNS configuration. At least one DNS server using IPv4 is required in /etc/resolv.conf for adclient to find AD services properly.

Log:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ADSITE : Check that this machine's subnet is in a site known by AD : Failed</td>
</tr>
<tr>
<td>2</td>
<td>: This machine's subnet is not known by AD.</td>
</tr>
<tr>
<td>3</td>
<td>: We guess you should be in the site Site1.</td>
</tr>
</tbody>
</table>

This issue occurs due to the special configuration of Centrify. Do the following to resolve this issue:

a. Open Administrative Tools on the Delivery Controller.
b. Select Active Directory Sites and Services.
c. Add a correct subnet address for Subnets.

- Easy install supports pure IPv6 as of Linux VDA 7.16. This enhancement has the following precondition and limitation:
  - Your Linux repository must be configured to ensure that your machine can download required packages in pure IPv6 networks.
  - Centrify is not supported in pure IPv6 networks.

Note:

If your network is pure IPv6 and all your input is in proper IPv6 format, the VDA registers with the Delivery Controller through IPv6. If your network is a hybrid IPv4 and IPv6 world, the type of the first DNS IP address determines IPv4 or IPv6 is used for registration.

- If you choose Centrify as the method to join a domain, the ctxinstall.sh script needs the Centrify package. There are two ways for ctxinstall.sh to get the Centrify package:
- Easy install helps download the Centrify package from the Internet automatically. The following are the given URLs for each distribution:


- Fetch the Centrify package from a local directory. Do the following to designate the directory of the Centrify package:

  a. Create the /tmp/ctxinstall.conf file on the Linux VDA server if it does not exist.

  b. Add the “centrifypkgpath=<path name>” line to the file.

For example:

```bash
1  cat /tmp/ctxinstall.conf
2  set "centrifypkgpath=/home/mydir"
3  ls -ls  /home/mydir
4   9548  -r-xr-xr-x. 1 root root 9776688 May 13 2016
      adcheck-rhel4-x86_64
5   4140  -r--r--r--. 1 root root 4236714 Apr 21 2016
      centrifyda-3.3.1-rhel4-x86_64.rpm
6   33492 -r--r--r--. 1 root root 34292673 May 13 2016
      centrifydc-5.3.1-rhel4-x86_64.rpm
7    4  -rw-rw-r--. 1 root root 1168 Dec 1 2015
      centrifyd-install.cfg
8    756  -r--r--r--. 1 root root 770991 May 13 2016
      centrifydc-ldapproxy-5.3.1-rhel4-x86_64.rpm
9   268  -r--r--r--. 1 root root 271296 May 13 2016
      centrifydc-nis-5.3.1-rhel4-x86_64.rpm
10   1888 -r--r--r--. 1 root root 1930084 Apr 12 2016
      centrifydc-openssh-7.2p2-5.3.1-rhel4-x86_64.rpm
11   124 -rw-rw-r--. 1 root root 124543 Apr 19 2016
      centrify-suite.cfg
12    0  lrwxrwxrwx. 1 root root 10 Jun 9 2012 install-
      express.sh -> install.sh
13   332  -r-xr-xr-x. 1 root root 338292 Apr 10 2016 install
      .sh
14   12  -r--r--r--. 1 root root 11166 Apr 9 2015 release-
      notes-agent-rhel4-x86_64.txt
```
Step 2: Install the Linux VDA package

Run the following commands to set up the environment for the Linux VDA.

For RHEL and CentOS distributions:

```
1 sudo yum -y localinstall <PATH>/<Linux VDA RPM>
```

For Ubuntu distributions:

```
1 sudo dpkg -i <PATH>/<Linux VDA deb>
2 sudo apt-get install -f
```

For SUSE distributions:

```
1 zypper -i install <PATH>/<Linux VDA RPM>
```

Step 3: Set up the runtime environment to complete the installation

After installing the Linux VDA package, configure the running environment by using the ctxinstall.sh script. You can run the script in interactive mode or silent mode.

**Interactive mode:**

To do a manual configuration, run the following command and type the relevant parameter at each prompt.

```
1 sudo /opt/Citrix/VDA/sbin/ctxinstall.sh
```

**Silent mode:**

To use easy install in silent mode, set the following environment variables before running ctxinstall.sh.

- `CTX_EASYINSTALL_HOSTNAME` = host-name – Denotes the host name of the Linux VDA server.
- `CTX_EASYINSTALL_DNS` = ip-address-of-dns – IP address of DNS.
- `CTX_EASYINSTALL_NTPS` = address-of-ntps – IP address or string name of the NTP server.
- `CTX_EASYINSTALL_DOMAIN` = domain-name – The NetBIOS name of the domain.
• **CTX_EASYINSTALL_REALM**=realm-name – The Kerberos realm name.
• **CTX_EASYINSTALL_FQDN**=ad-fqdn-name
• **CTX_EASYINSTALL_ADINTEGRATIONWAY**=winbind | sssd | centrify – Denotes the Active Directory integration method.
• **CTX_EASYINSTALL_USERNAME**=domain-user-name – Denotes the name of the domain user; used to join the domain.
• **CTX_EASYINSTALL_PASSWORD**=password – Specifies the password of the domain user; used to join the domain.

The following variables are used by ctxsetup.sh:

• **CTX_XDL_SUPPORT_DDC_AS_CNAME**=Y | N – The Linux VDA supports specifying a Delivery Controller name using a DNS CNAME record.
• **CTX_XDL_DDC_LIST**=list-ddc-fqdns – The Linux VDA requires a space-separated list of Delivery Controller Fully Qualified Domain Names (FQDNs) to use for registering with a Delivery Controller. At least one FQDN or CNAME must be specified.
• **CTX_XDL_VDA_PORT**=port-number – The Linux VDA communicates with Delivery Controllers through a TCP/IP port.
• **CTX_XDL_REGISTER_SERVICE**=Y | N – The Linux Virtual Desktop services are started after machine startup.
• **CTX_XDL_ADD_FIREWALL_RULES**=Y | N – The Linux Virtual Desktop services require incoming network connections to be allowed through the system firewall. You can automatically open the required ports (by default ports 80 and 1494) in the system firewall for the Linux Virtual Desktop.
• **CTX_XDL_HDX_3D_PRO**=Y | N – Linux Virtual Desktop supports HDX 3D Pro, a set of graphics acceleration technologies designed to optimize the virtualization of rich graphics applications. HDX 3D Pro requires a compatible NVIDIA GRID graphics card to be installed. If HDX 3D Pro is selected, the VDA is configured for VDI desktops (single-session) mode - (that is, **CTX_XDL_VDI_MODE**=Y). HDX 3D Pro is not supported on SUSE. Ensure that the value is set to N for a SUSE platform.
• **CTX_XDL_VDI_MODE**=Y | N – Whether to configure the machine as a dedicated desktop delivery model (VDI) or hosted shared desktop delivery model. For HDX 3D Pro environments, set the value to Y.
• **CTX_XDL_SITE_NAME**=dns-name – The Linux VDA discovers LDAP servers through DNS. To limit the DNS search results to a local Site, specify a DNS Site name. If unnecessary, it can be set to <none>.
• **CTX_XDL_LDAP_LIST**=list-ldap-servers – The Linux VDA queries DNS to discover LDAP servers. If DNS cannot provide LDAP service records, you can provide a space-separated list of LDAP FQDNs with LDAP port. For example, ad1.mycompany.com:389. If unnecessary, it can be set to <none>.
• **CTX_XDL_SEARCH_BASE**=search-base-set – The Linux VDA queries LDAP through a search base set to the root of the Active Directory Domain (for example, DC=mycompany,DC=com). To im-
prove search performance, you can specify a search base (for example, OU=VDI,DC=mycompany,DC=com). If unnecessary, it can be set to `<none>`.

- **CTX_XDL_FAS_LIST** = list-fas-servers – The Federated Authentication Service (FAS) servers are configured through AD Group Policy. Because the Linux VDA does not support AD Group Policy, you can provide a semicolon-separated list of FAS servers instead. The sequence must be the same as configured in AD Group Policy. If any server address is removed, fill its blank with the `<none>` text string and keep the sequence of server addresses without any changes.

- **CTX_XDL_START_SERVICE** = Y | N – Whether or not the Linux VDA services are started when the configuration is complete.

If any parameters are not set, the installation rolls back to interactive mode, with a prompt for user input. The ctxinstall.sh script does not prompt for answers when all parameters are already set through the environment variables.

In silent mode, you must run the following commands to set environment variables and then run the ctxinstall.sh script.

```bash
export CTX_EASYINSTALL_HOSTNAME=host-name
export CTX_EASYINSTALL_DNS=ip-address-of-dns
export CTX_EASYINSTALL_NTPS=address-of-ntps
export CTX_EASYINSTALL_DOMAIN=domain-name
export CTX_EASYINSTALL_REALM=realm-name
export CTX_EASYINSTALL_FQDN=ad-fqdn-name
export CTX_EASYINSTALL_ADINTEGRATIONWAY=winbind | sssd | centrify
export CTX_EASYINSTALL_USERNAME=domain-user-name
export CTX_EASYINSTALL_PASSWORD=password
export CTX_XDL_SUPPORT_DDC_AS_CNAME=Y | N
export CTX_XDL_DDC_LIST=list-ddc-fqdns
export CTX_XDL_VDA_PORT=port-number
export CTX_XDL_REGISTER_SERVICE=Y | N
export CTX_XDL_ADD_FIREWALL_RULES=Y | N
```
When running the sudo command, type the -E option to pass the existing environment variables to the new shell it creates. Citrix recommends that you create a shell script file from the preceding commands with 

```
#!/bin/bash
```

as the first line.

Alternatively, you can specify all parameters by using a single command:

```
sudo -E /opt/Citrix/VDA/sbin/ctxinstall.sh
```
Troubleshooting

Use the information in this section to troubleshoot issues that can arise from using this feature.

Joining a domain by using SSSD fails

An error might occur when you attempt to join a domain, with the output resembling (verify logs for screen printing):

Step 6: join Domain! Enter ctxadmin’s password: Failed to join domain: failed to lookup DC info for domain 'CITRIXLAB.LOCAL' over rpc: The network name cannot be found

/var/log/xdl/vda.log:

```
1 2016-11-04 02:11:52.317 [INFO ] - The Citrix Desktop Service successfully obtained the following list of 1 delivery controller(s) with which to register: 'CTXDDC.citrixlab.local (10.158.139.214)'.
4 Check the following:- The system clock is in sync between this machine and the delivery controller.
5 - The Active Directory provider (e.g. winbind daemon) service is running and correctly configured.
6 - Kerberos is correctly configured on this machine.
7 If the problem persists, please refer to Citrix Knowledge Base articleCTX117248 for further information.
8 Error Details:
9 Exception 'General security error (An error occurred in trying to obtain a TGT: Client not found in Kerberos database (6))' of type 'class javax.xml.ws.soap.SOAPFaultException'.
```
Linux Virtual Delivery Agent 1903


Ensure that the system clock is in sync between this machine and the delivery controller.

Verify the NTP daemon is running on this machine and is correctly configured.

2016-11-04 02:11:52.364 [ERROR] - Could not register with any controllers. Waiting to try again in 120000 ms. Multi-forest - false

2016-11-04 02:11:52.365 [INFO ] - The Citrix Desktop Service failed to register with any controllers in the last 470 minutes.

/var/log/messages:

Nov 4 02:15:27 RH-WS-68 [sssd[ldap_child[14867]]]: Failed to initialize credentials using keytab [MEMORY:/etc/krb5.keytab]: Client 'RH-WS-68 $@CITRIXLAB.LOCAL' not found in Kerberos database. Unable to create GSSAPI-encrypted LDAP connection.

Nov 4 02:15:27 RH-WS-68 [sssd[ldap_child[14867]]]: Client 'RH-WS-68$@CITRIXLAB.LOCAL' not found in Kerberos database.

To resolve this issue:

1. Run the `rm -f /etc/krb5.keytab` command.
2. Run the `net ads leave $REALM -U $domain-administrator` command.
3. Remove the machine catalog and delivery group on the Delivery Controller.
5. Create the machine catalog and delivery group on the Delivery Controller.

Ubuntu desktop sessions show a gray screen

This issue occurs when you launch a session, which is then blocked in a blank desktop. In addition, the console of the server OS machine also shows a gray screen when you log on by using a local user account.

To resolve this issue:

1. `sudo apt-get update`
2. `sudo apt-get install unity-lightdm`
3. Add the following line to `/etc/lightdm/lightdm.conf`:
   ```
   greeter-show-manual-login=true
   ```

Launching Ubuntu desktop sessions fails due to the missing home directory

/var/log/xdl/hdx.log:
Tip:
The root cause of this issue is that the home directory is not created for the domain administrator.

To resolve this issue:

1. From a command line, type `pam-auth-update`.
2. In the resulting popup window, verify that **Create home directory login** is selected.

Session cannot launch or ends quickly with dbus error

/var/log/messages (for RHEL or CentOS):

```
1 Oct 27 04:17:16 CentOS7 citrix-ctxhdx[8978]: Session started for user CITRIXLAB\ctxadmin.
2 Oct 27 04:17:18 CentOS7 kernel: traps: gnome-session[19146] trap int3
   ip:7f89b3bde8d3 sp:7ff8c3409d0 error:0
3 Oct 27 04:17:18 CentOS7 gnome-session[19146]: ERROR: Failed to connect to system bus: Exhausted all available authentication mechanisms (tried: EXTERNAL, DBUS_COOKIE_SHA1, ANONYMOUS) (available: EXTERNAL, DBUS_COOKIE_SHA1, ANONYMOUS)#012aborting...
```
Or, alternately for Ubuntu distributions, use the log /var/log/syslog:

```
Nov 3 11:03:52 user01-HVM-domU pulseaudio[25326]: [pulseaudio] bluez5-util.c: Failed to get D-Bus connection: Did not receive a reply.
   Possible causes include: the remote application did not send a reply, the message bus security policy blocked the reply, the reply timeout expired, or the network connection was broken.
Nov 3 11:03:52 user01-HVM-domU pulseaudio[25352]: [pulseaudio] core-util.c: Failed to connect to system bus: Did not receive a reply.
   Possible causes include: the remote application did not send a reply, the message bus security policy blocked the reply, the reply timeout expired, or the network connection was broken.
Nov 3 11:03:52 user01-HVM-domU pulseaudio[25352]: message repeated 10 times: [ [pulseaudio] core-util.c: Failed to connect to system bus: Did not receive a reply. Possible causes include: the remote application did not send a reply, the message bus security policy blocked the reply, the reply timeout expired, or the network connection was broken.]
Nov 3 11:03:58 user01-HVM-domU citrix-ctxgfx [24693]: Exiting normally
```

Some groups or modules do not take effect until a restart. If the `dbus` error messages appear in the
log, Citrix recommends that you restart the system and retry.

**SELinux prevents SSHD from accessing the home directory**

The user can launch a session but cannot log on.

/var/log/ctxinstall.log:

```plaintext
Jan 25 23:30:31 yz-rhel72-1 setroubleshoot[3945]: SELinux is preventing /usr/sbin/sshd from setattr access on the directory /root. For complete SELinux messages. run sealert -l 32f52c1f-8ff9-4566-a698-963a79f16b81

Jan 25 23:30:31 yz-rhel72-1 python[3945]: SELinux is preventing /usr/sbin/sshd from setattr access on the directory /root.

***** Plugin catchall_boolean (89.3 confidence) suggests ***************

If you want to allow polyinstantiation to enabled

Then you must tell SELinux about this by enabling the 'polyinstantiation_enabled' boolean.

You can read 'None' man page for more details.

Do

    setsebool -P polyinstantiation_enabled 1

***** Plugin catchall (11.6 confidence) suggests ***************

If you believe that sshd should be allowed setattr access on the root directory by default.

Then you should report this as a bug.

You can generate a local policy module to allow this access.

Do

    allow this access for now by executing:

    # grep sshd /var/log/audit/audit.log | audit2allow -M mypol
```

© 1999-2019 Citrix Systems, Inc. All rights reserved.
To resolve this issue:

1. Disable SELinux by making the following change to /etc/selinux/config.

   SELINUX=disabled

2. Restart the VDA.

### Use MCS to create Linux VMs

**March 25, 2019**

Starting with the 7.18 release, you can use Citrix Machine Creation Services (MCS) to create Linux VMs.

**Note:**

When using MCS to create Linux VMs, do not use easy install on the template VM.

To use MCS to create Linux VMs, prepare a master image on your XenServer hypervisor, Microsoft Azure, or VMware vSphere (support for other hypervisors is currently not available). This process entails installing the VDA on the template VM; creating a Machine Catalog in Citrix Studio; creating a Delivery Group; and performing certain configuration tasks.

**Note:**

Unexpected results can occur if you try to prepare a master image on hypervisors other than XenServer.

### Use MCS to create Linux VMs on XenServer

**Step 1: Prepare a master image**

A master image contains the operating system, non-virtualized applications, VDA, and other software. To prepare a master image, do the following:

**Step 1a: Install XenServer Tools**

XenServer Tools must be installed on the template VM for each VM to be able to use the xe CLI or XenCenter. VM performance can be slow unless the tools are installed. Without the tools, you cannot do any of the following:

```bash
# semodule -i mypol.pp
```
• Cleanly shut down, restart, or suspend a VM.
• View the VM performance data in XenCenter.
• Migrate a running VM (through XenMotion).
• Create snapshots or snapshots with memory (checkpoints), and revert to snapshots.
• Adjust the number of vCPUs on a running Linux VM.

Important:
Running a VM without installing the XenServer Tools is not a supported configuration.

1. Run the following command to mount XenServer Tools named guest-tools.iso.

   ```bash
   sudo mount /dev/cdrom /mnt
   ```

2. Run the following command to install the xe-guest-utilities package based on your Linux distribution.

   **For RHEL/CentOS:**
   ```bash
   sudo rpm -i /mnt/Linux/xe-guest-utilities_{\package-version\} _all.rpm
   ```

   **For Ubuntu:**
   ```bash
   sudo dpkg -i /mnt/Linux/xe-guest-utilities_{\package-version\} _all.deb
   ```

   **For SUSE 12:**
   ```bash
   sudo rpm -i /mnt/Linux/xe-guest-utilities_{\package-version\} _all.rpm
   ```

3. Check the virtualization state of the template VM on the **General** tab in XenCenter. If XenServer Tools are installed correctly, the virtualization state is **Optimized** as shown below:
Step 1b: Install the Linux VDA package on the template VM

Based on your Linux distribution, run the following command to set up the environment for the Linux VDA:

For RHEL/CentOS:

```
1 sudo yum -y localinstall <PATH>/Linux VDA RPM
```

For Ubuntu:

```
1 sudo dpkg -i <PATH>/Linux VDA DEB
2
3 apt-get install -f
```

For SUSE 12:

```
1 sudo zypper -i install <PATH>/Linux VDA RPM
```

Step 1c: Enable repositories to install the tdb-tools package

For RHEL 7 server:

```
1 subscription-manager repos --enable=rhel-7-server-optional-rpms
```
For RHEL 7 workstation:

```bash
subscription-manager repos --enable=rhel-7-workstation-optional-rpms
```

Step 1d: Enable the EPEL repository to install ntfs-3g

For RHEL 6/CentOS 6:

```bash
```

For RHEL 7/CentOS 7:

```bash
```

Step 1e: Install ntfs-3g for SUSE 12

On the SUSE 12 platform, there is no repository providing ntfs-3g. You must download the source code, compile, and install ntfs-3g manually:

1. Install the GNU Compiler Collection (GCC) compiler system and the make package:

   ```bash
   sudo zypper install gcc
   sudo zypper install make
   ```

2. Download the ntfs-3g package from https://www.tuxera.com/community/open-source-ntfs-3g/.

3. Decompress the ntfs-3g package:

   ```bash
   sudo tar -xzvf ntfs-3g_ntfsprogs-<package version>.tgz
   ```

4. Enter the path to the ntfs-3g package:

   ```bash
   sudo cd ntfs-3g_ntfsprogs-<package version>
   ```

5. Install ntfs-3g:

   ```bash
   ./configure
   make
   make install
   ```
**Step 1f: Set up the runtime environment**

Before running `deploymcs.sh`, do the following:

- Change variables in `/var/xdl/mcs/mcs.conf`. The `mcs.conf` configuration file contains variables for setting MCS and the Linux VDA. The following are variables you can set as required:
  - `dns`: Sets the DNS IP address.
  - `AD_INTEGRATION`: Sets Winbind or SSSD (SSSD is not supported on SUSE).
  - `WORKGROUP`: Sets the workgroup name (case-sensitive) if it is configured in AD.

- On the template machine, add command lines to the `/var/xdl/mcs/mcs_local_setting.reg` file for writing or updating registry values as required. This action prevents the loss of data and settings every time an MCS-provisioned machine restarts.

Each line in the `/var/xdl/mcs/mcs_local_setting.reg` file is a command for setting or updating a registry value.

For example, you can add the following command lines to the `/var/xdl/mcs/mcs_local_setting.reg` file to write or update a registry value respectively:

```plaintext
create -k "HKLM\System\CurrentControlSet\Control\Citrix\VirtualChannels\Clipboard\ClipboardSelection" -t "REG_DWORD" -v "Flags" -d "0x00000003" --force

update -k "HKLM\System\CurrentControlSet\Control\Citrix\VirtualChannels\Clipboard\ClipboardSelection" -v "Flags" -d "0x00000003"
```

**Step 1g: Create a master image**

1. Run `/opt/Citrix/VDA/sbin/deploymcs.sh`.
2. On XenServer, shutdown the template VM. Create and name a snapshot of your master image.

**Step 2: Create a Machine Catalog**

In Citrix Studio, create a Machine Catalog and specify the number of VMs to create in the catalog. Do other configuration tasks as needed. For more information, see Create a machine catalog using Studio.

**Step 3: Create a Delivery Group**

A Delivery Group is a collection of machines selected from one or more Machine Catalogs. The Delivery Group specifies which users can use those machines, and the applications and desktops available to
those users. For more information, see Create Delivery Groups.

Use MCS to create Linux VMs on Azure

Step 1: Create a hosting connection to Azure in Citrix Studio

1. In Citrix Studio, choose Configuration > Hosting > Add Connection and Resources to create a connection to Azure.

2. Choose Microsoft Azure as the connection type.
3. Type the subscription ID of your Azure account and type your connection name.
A new connection appears in the hosting pane.

**Step 2: Prepare a master image**

Follow Step 1 (except Step 1a and Step 1c) of the preceding Use MCS to create Linux VMs on XenServer section to prepare a master image.

After you finish installing applications on the template VM, shut down the template VM from the Azure portal. Ensure that the power status of the template VM is **Stopped (deallocated)**.

Remember the name of the resource group here. You need the name to locate your master image on Azure.
Step 3: Create a Machine Catalog

In Citrix Studio, create a Machine Catalog and specify the number of VMs to create in the catalog. When creating the Machine Catalog, choose your master image from the resource group where the template VM belongs and find the VHD of the template VM, as shown below.

Do other configuration tasks as needed. For more information, see Knowledge Center article CTX219270 and Create a machine catalog using Studio.
Step 4: Create a Delivery Group

A Delivery Group is a collection of machines selected from one or more Machine Catalogs. The Delivery Group specifies which users can use those machines, and the applications and desktops available to those users. For more information, see Create Delivery Groups.

Use MCS to create Linux VMs on VMware vSphere

Step 1: Create a hosting connection to VMware in Citrix Studio

1. Install vCenter Server in the vSphere environment. For more information, see VMware vSphere.

2. In Citrix Studio, choose Configuration > Hosting > Add Connection and Resources to create a connection to VMware vSphere.

3. Choose VMware vSphere as the connection type.
4. Type the connection address (the vCenter Server URL) of your VMware account, your user name and password, and your connection name.
A new connection appears in the hosting pane.

**Step 2: Prepare a master image**

Follow Step 1 (except Step 1a and Step 1c) of the preceding **Use MCS to create Linux VMs on XenServer** section to prepare a master image.

After you finish installing applications on the template VM, shut down the template VM from the VMware. Take a snapshot of the template VM.

**Step 3: Create a Machine Catalog**

In Citrix Studio, create a Machine Catalog and specify the number of VMs to create in the catalog. When creating the Machine Catalog, choose your master image from the snapshot list, as shown below.
Do other configuration tasks as needed. For more information, see Knowledge Center article CTX219270 and Create a machine catalog using Studio.

**Step 4: Create a Delivery Group**

A Delivery Group is a collection of machines selected from one or more Machine Catalogs. The Delivery Group specifies which users can use those machines, and the applications and desktops available to those users. For more information, see Create Delivery Groups.

**(Not recommended) Use MCS to upgrade your Linux VDA**

To use MCS to upgrade your Linux VDA, do the following:

1. Upgrade your Linux VDA on the template machine:
   
   **For RHEL 7/CentOS 7:**
   
   ```
   sudo rpm -U LinuxVDA-1903.el7_x.rpm
   ```
For RHEL 6/CentOS 6:

```
sudo rpm -U LinuxVDA-1903.el6_x.rpm
```

For SUSE 12:

```
sudo rpm -U LinuxVDA-1903.sle12_x.rpm
```

For Ubuntu 16.04:

```
sudo dpkg -i LinuxVDA-1903.ubuntu16.04.deb
```

For Ubuntu 18.04:

```
sudo dpkg -i LinuxVDA-1903.ubuntu18.04.deb
```

2. Edit `/var/xdl/mcs/mcs.conf` and `/var/xdl/mcs/mcs_local_setting.reg`.

3. Take a new snapshot.

4. In Citrix Studio, select the new snapshot to update your Machine Catalog. Wait before each machine restarts. Do not restart a machine manually.

Install Linux Virtual Delivery Agent for RHEL/CentOS

May 14, 2019

You can choose to follow the steps in this article for manual installation or use easy install for automatic installation and configuration. Easy install saves time and labor and is less error-prone than the manual installation.

Note:
Use easy install only for fresh installations. Do not use easy install to update an existing installation.

Step 1: Prepare RHEL 7/CentOS 7, RHEL 6/CentOS 6 for VDA installation

Step 1a: Verify the network configuration

Citrix recommends that the network is connected and configured correctly before proceeding.
Step 1b: Set the host name

To ensure that the host name of the machine is reported correctly, change the `/etc/hostname` file to contain only the host name of the machine.

```
hostname
```

Step 1c: Assign a loopback address to the hostname

To ensure that the DNS domain name and FQDN of the machine are reported back correctly, change the following line of the `/etc/hosts` file to include the FQDN and host name as the first two entries:

```
127.0.0.1 hostname-fqdn hostname localhost localhost.localdomain localhost4 localhost4.localdomain4
```

For example:

```
127.0.0.1 vda01.example.com vda01 localhost localhost.localdomain localhost4 localhost4.localdomain4
```

Remove any other references to `hostname-fqdn` or `hostname` from other entries in the file.

**Note:**

The Linux VDA currently does not support NetBIOS name truncation. Therefore, the host name must not exceed 15 characters.

**Tip:**

Use a–z, A–Z, 0–9, and hyphen (-) characters only. Avoid underscores (_), spaces, and other symbols. Do not start a host name with a number and do not end with a hyphen. This rule also applies to Delivery Controller host names.

Step 1d: Check the host name

Verify that the host name is set correctly:

```
1 hostname
```

This command returns only the machine’s host name and not its fully qualified domain name (FQDN).

Verify that the FQDN is set correctly:

```
1 hostname -f
```

This command returns the FQDN of the machine.
Step 1e: Check name resolution and service reachability

Verify that you can resolve the FQDN and ping the domain controller and Delivery Controller:

```
1 nslookup domain-controller-fqdn
2 ping domain-controller-fqdn
3 nslookup delivery-controller-fqdn
4 ping delivery-controller-fqdn
```

If you cannot resolve the FQDN or ping either of these machines, review the steps before proceeding.

Step 1f: Configure clock synchronization

Maintaining accurate clock synchronization between the VDAs, Delivery Controllers, and domain controllers is crucial. Hosting the Linux VDA as a virtual machine can cause clock skew problems. For this reason, synchronizing time with a remote time service is preferred.

RHEL 6.x and earlier releases use the NTP daemon (ntpd) for clock synchronization, whereas an RHEL 7.x default environment uses the newer Chrony daemon (chronyd) instead. The configuration and operational process between the two services is similar.

Configure the NTP service (RHEL 6/CentOS 6 only)

As a root user, edit `/etc/ntp.conf` and add a server entry for each remote time server:

```
1 server peer1-fqdn-or-ip-address iburst
2 server peer2-fqdn-or-ip-address iburst
```

In a typical deployment, synchronize time from the local domain controllers and not directly from public NTP pool servers. Add a server entry for each Active Directory domain controller in the domain. Remove any other server entries listed including loopback IP address, localhost, and public server `*.pool.ntp.org` entries.

Save changes and restart the NTP daemon:

```
1 sudo /sbin/service ntpd restart
```
Configure the Chrony service (RHEL 7/CentOS 7 only)

As a root user, edit `/etc/chrony.conf` and add a server entry for each remote time server:

```
1 server peer1-fqdn-or-ip-address iburst
2 server peer2-fqdn-or-ip-address iburst
```

In a typical deployment, synchronize time from the local domain controllers and not directly from public NTP pool servers. Add a server entry for each Active Directory domain controller in the domain.

Remove any other server entries listed including loopback IP address, localhost, and public server `*.pool.ntp.org` entries.

Save changes and restart the Chrony daemon:

```
1 sudo /sbin/service chronyd restart
```

**Step 1g: Install OpenJDK**

The Linux VDA depends on OpenJDK. Typically, the runtime environment is installed as part of the operating system installation.

Confirm the correct version:

- RHEL 7/CentOS 7:

```
1 sudo yum info java-1.8.0-openjdk
```

- RHEL 6/CentOS 6:

```
1 sudo yum info java-1.7.0-openjdk
```

The prepackaged OpenJDK might be an earlier version. Update to the latest version as required:

- RHEL 7/CentOS 7:

```
1 sudo yum -y update java-1.8.0-openjdk
```

- RHEL 6/CentOS 6:

```
1 sudo yum -y update java-1.7.0-openjdk
```

Set the `JAVA_HOME` environment variable by adding the following line to the `~/.bashrc` file:

```
export JAVA_HOME=/usr/lib/jvm/java
```

Open a new shell and verify the version of Java:
```bash
java -version
```

**Tip:**
To avoid registration failure with the Delivery Controller, ensure that you installed only OpenJDK Version 1.7.0 or 1.8.0 in case of RHEL 6/CentOS 6 or only OpenJDK Version 1.8.0 in case of RHEL 7/CentOS 7. Remove all other versions of Java from your system.

**Step 1h: Install PostgreSQL**

The Linux VDA requires either PostgreSQL 8.4 or later on RHEL 6 or PostgreSQL 9.2 or later on RHEL 7. Install the following packages:

```bash
sudo yum -y install postgresql-server
sudo yum -y install postgresql-jdbc
```

The following post-installation step is required to initialize the database and to ensure that the service starts upon machine startup. This action creates database files under `/var/lib/pgsql/data`. The command differs between PostgreSQL 8 and 9:

- **RHEL 7 only: PostgreSQL 9**

```bash
sudo postgresql-setup initdb
```

- **RHEL 6 only: PostgreSQL 8**

```bash
sudo /sbin/service postgresql initdb
```

**Step 1i: Start PostgreSQL**

Start the service upon machine startup and start the service now:

- **RHEL 7 only: PostgreSQL 9**

```bash
sudo systemctl enable postgresql
sudo systemctl start postgresql
```

- **RHEL 6 only: PostgreSQL 8**
Check the version of PostgreSQL by using:

```bash
psql --version
```

Verify that the data directory is set by using the `psql` command-line utility:

```bash
sudo -u postgres psql -c 'show data_directory'
```

**Important:**

In this release, a new dependency is added for gperftools-libs, but it does not exist in the original repository. Add a new repository by using the `sudo rpm -ivh https://dl.fedoraproject.org/pub/epel/epel-release-latest-6.noarch.rpm` command. Only RHEL 6/CentOS 6 is impacted. Run the command before installing the Linux VDA package.

**Step 2: Prepare the hypervisor**

Some changes are required when running the Linux VDA as a virtual machine on a supported hypervisor. Make the following changes according to the hypervisor platform in use. No changes are required if you are running the Linux machine on bare metal hardware.

**Fix time synchronization on Citrix XenServer**

When the XenServer Time Sync feature is enabled, within each paravirtualized Linux VM you experience issues with the NTP and the XenServer, both of which try to manage the system clock. To avoid the clock becoming out of sync with other servers, ensure that the system clock within each Linux guest is synchronized with the NTP. This case requires disabling host time synchronization. No changes are required in HVM mode.

On some Linux distributions, if you are running a paravirtualized Linux kernel with XenServer Tools installed, you can check whether the XenServer Time Sync feature is present and enabled from within the Linux VM:

```bash
su -
3 cat /proc/sys/xen/independent_wallclock
```

This command returns 0 or 1:
• 0 - The time sync feature is enabled, and must be disabled.
• 1 - The time sync feature is disabled, and no further action is required.

If the /proc/sys/xen/indepent_wallclock file is not present, the following steps are not required.

If enabled, disable the time sync feature by writing 1 to the file:

```bash
sudo echo 1 > /proc/sys/xen/independent_wallclock
```

To make this change permanent and persistent after restart, edit the /etc/sysctl.conf file and add the line:

```
xen.independent_wallclock = 1
```

To verify these changes, restart the system:

```bash
su -
cat /proc/sys/xen/independent_wallclock
```

This command returns the value 1.

**Fix time synchronization on Microsoft Hyper-V**

The Linux VMs with Hyper-V Linux Integration Services installed can apply the Hyper-V time synchronization feature to use the time of the host operating system. To ensure that the system clock remains accurate, you must enable this feature alongside the NTP services.

From the management operating system:

1. Open the Hyper-V Manager console.
2. For the settings of a Linux VM, select Integration Services.
3. Ensure that **Time synchronization** is selected.

**Note:**

This approach is different from VMware and XenServer, where host time synchronization is disabled to avoid conflicts with NTP. Hyper-V time synchronization can coexist and supplement NTP time synchronization.

**Fix time synchronization on ESX and ESXi**

When the VMware Time Synchronization feature is enabled, within each paravirtualized Linux VM you experience issues with the NTP and the hypervisor, both of which try to synchronize the system clock. To avoid the clock becoming out of sync with other servers, ensure that the system clock within each Linux guest is synchronized with the NTP. This case requires disabling host time synchronization.
If you are running a paravirtualized Linux kernel with VMware Tools installed:

1. Open the vSphere Client.
2. Edit settings for the Linux VM.
3. In the Virtual Machine Properties dialog, open the Options tab.
4. Select VMware Tools.
5. In the Advanced box, clear Synchronize guest time with host.

Step 3: Add the Linux virtual machine (VM) to the Windows domain

The Linux VDA supports several methods for adding Linux machines to the Active Directory (AD) domain:

- Samba Winbind
- Quest Authentication Service
- Centrify DirectControl
- SSSD

Follow instructions based on your chosen method.

Note:
Session launches might fail when the same user name is used for the local account in the Linux VDA and the account in AD.

Samba Winbind

Install or update the required packages:

```bash
sudo yum -y install samba-winbind samba-winbind-clients krb5-workstation authconfig oddjob-mkhomedir
```

Enable Winbind daemon to start upon machine startup

The Winbind daemon must be configured to start upon machine startup:

```bash
sudo /sbin/chkconfig winbind on
```

Configure Winbind Authentication

Configure the machine for Kerberos authentication by using Winbind:
1  sudo authconfig --disablecache --disabledssd --disablessdauth --enablewinbind --enablewinbindauth --disablewinbindoffline --smbsecurity=ads --smbworkgroup=domain --smbrealm=REALM --krb5realm=REALM --krb5kdc=fqdn-of-domain-controller --winbindtemplateshell=/bin/bash --enablemkhomedir --updateall

Where REALM is the Kerberos realm name in uppercase and domain is the NetBIOS name of the domain.

If DNS-based lookup of the KDC server and realm name is required, add the following two options to the previous command:

--enablekrb5kdcdns --enablekrb5realmdns

Ignore any errors returned from the authconfig command about the winbind service failing to start. The errors can occur when authconfig tries to start the winbind service without the machine yet being joined to the domain.

Open /etc/samba/smb.conf and add the following entries under the [Global] section, but after the section generated by the authconfig tool:

```plaintext
kerberos method = secrets and keytab
winbind refresh tickets = true
```

The Linux VDA requires the system keytab file /etc/krb5.keytab to authenticate and register with the Delivery Controller. The previous kerberos method setting forces Winbind to create the system keytab file when the machine is first joined to the domain.

**Join Windows domain**

Your domain controller must be reachable and you must have an Active Directory user account with permissions to add computers to the domain:

1  sudo net ads join REALM -U user

REALM is the Kerberos realm name in uppercase, and user is a domain user who has permissions to add computers to the domain.

**Configure PAM for Winbind**

By default, the configuration for the Winbind PAM module (pam_winbind) does not enable Kerberos ticket caching and home directory creation. Open /etc/security/pam_winbind.conf and add or change the following entries under the [Global] section:
krb5_auth = yes
krb5_ccache_type = FILE
mkhomedir = yes

Ensure that any leading semi-colons from each setting are removed. These changes require restarting the Winbind daemon:

```
sudo /sbin/service winbind restart
```

Tip:
The winbind daemon stays running only if the machine is joined to a domain.

Open `/etc/krb5.conf` and change the following setting under the [libdefaults] section from KEYRING to FILE type:

```
default_ccache_name = FILE:/tmp/krb5cc_%{ uid }
```

**Verify domain membership**

The Delivery Controller requires that all VDA machines (Windows and Linux VDAs) have a computer object in Active Directory.

Run the `net ads` command of Samba to verify that the machine is joined to a domain:

```
sudo net ads testjoin
```

Run the following command to verify extra domain and computer object information:

```
sudo net ads info
```

**Verify Kerberos configuration**

To ensure that Kerberos is configured correctly for use with the Linux VDA, check that the system keytab file has been created and contains valid keys:

```
sudo klist -ke
```

This command displays the list of keys available for the various combinations of principal names and cipher suites. Run the Kerberos `kinit` command to authenticate the machine with the domain controller using these keys:

```
sudo kinit -k MACHINE\$@REALM
```
The machine and realm names must be specified in uppercase. The dollar sign ($) must be escaped with a backslash (\) to prevent shell substitution. In some environments, the DNS domain name is different from the Kerberos realm name. Ensure that the realm name is used. If this command is successful, no output is displayed.

Verify that the TGT ticket for the machine account has been cached using:

```
sudo klist
```

Examine the account details of the machine using:

```
sudo net ads status
```

**Verify user authentication**

Use the `wbinfo` tool to verify that domain users can authenticate with the domain:

```
wbinfo --krb5auth=domain\username\password
```

The domain specified here is the AD domain name, not the Kerberos realm name. For the bash shell, the backslash (\) character must be escaped with another backslash. This command returns a message indicating success or failure.

To verify that the Winbind PAM module is configured correctly, use a domain user account to log on to the Linux VDA. The domain user account has not been used before.

```
ssh localhost -l domain\username
id -u
```

Check that the tickets in the Kerberos credential cache are valid and not expired:

```
klist
```

Exit the session:

```
exit
```

A similar test can be performed by logging on to the Gnome or KDE console directly.

**Quest authentication service**

**Configure Quest on domain controller**
Assume that you have installed and configured the Quest software on the Active Directory domain controllers, and have been granted administrative privileges to create computer objects in Active Directory.

**Enable domain users to log on to Linux VDA machines**

To enable domain users to establish HDX sessions on a Linux VDA machine:

1. In the Active Directory Users and Computers management console, open Active Directory user properties for that user account.
2. Select the Unix Account tab.
3. Check Unix-enabled.
4. Set the Primary GID Number to the group ID of an actual domain user group.

**Note:**

These instructions are equivalent for setting up domain users for logon using the console, RDP, SSH, or any other remoting protocol.

**Configure Quest on Linux VDA**

**Work around SELinux policy enforcement**

The default RHEL environment has SELinux fully enforced. This enforcement interferes with the Unix domain socket IPC mechanisms used by Quest, and prevents domain users from logging on.

The convenient way to work around this issue is to disable SELinux. As a root user, edit `/etc/selinux/-config` and change the SELinux setting:

```
SELINUX=permissive
```

This change requires a machine restart:

```
1 reboot
```

**Important:**

Use this setting carefully. Reenabling SELinux policy enforcement after disabling can cause a complete lockout, even for the root user and other local users.

**Configure VAS daemon**

Auto-renewal of Kerberos tickets must be enabled and disconnected. Authentication (offline logon) must be disabled.
This command sets the renewal interval to nine hours (32,400 seconds) which is one hour less than the default 10-hour ticket lifetime. Set this parameter to a lower value on systems with a shorter ticket lifetime.

**Configure PAM and NSS**

To enable domain user logon through HDX and other services such as su, ssh, and RDP, run the following commands to manually configure PAM and NSS:

```bash
sudo /opt/quest/bin/vastool configure pam
sudo /opt/quest/bin/vastool configure nss
```

**Join Windows domain**

Join the Linux machine to the Active Directory domain using the Quest vastool command:

```bash
sudo /opt/quest/bin/vastool -u user join domain-name
```

The user is any domain user who has permissions to join computers to the Active Directory domain. The **domain-name** is the DNS name of the domain, for example, example.com.

**Verify domain membership**

The Delivery Controller requires that all VDA machines (Windows and Linux VDAs) have a computer object in Active Directory. To verify that a Quest-joined Linux machine is on the domain:

```bash
sudo /opt/quest/bin/vastool info domain
```

If the machine is joined to a domain, this command returns the domain name. If the machine is not joined to any domain, the following error appears:

```
ERROR: No domain could be found.
ERROR: VAS_ERR_CONFIG: at ctx.c:414 in _ctx_init_default_realm
default_realm not configured in vas.conf. Computer may not be joined to domain
```
Verify user authentication

To verify that Quest can authenticate domain users through PAM, use a domain user account to log on to the Linux VDA. The domain user account has not been used before.

```
1 ssh localhost -l domain\username
2 id -u
```

Check that a corresponding Kerberos credential cache file was created for the UID returned by the `id -u` command:

```
1 ls /tmp/krb5cc_uid
```

Check that the tickets in the Kerberos credential cache are valid and not expired:

```
1 /opt/quest/bin/vastool klist
```

Exit the session:

```
1 exit
```

A similar test can be performed by logging on to the Gnome or KDE console directly.

Centrify DirectControl

Join Windows domain

With the Centrify DirectControl Agent installed, join the Linux machine to the Active Directory domain using the Centrify `adjoin` command:

```
1 su -
2 adjoin -w -V -u user domain-name
```

The user parameter is any Active Directory domain user who has permissions to join computers to the Active Directory domain. The `domain-name` is the name of the domain to join the Linux machine to.

Verify domain membership

The Delivery Controller requires that all VDA machines (Windows and Linux VDAs) have a computer object in Active Directory. To verify that a Centrify-joined Linux machine is on the domain:

```
1 su -
2 adinfo
```
Check that the Joined to domain value is valid and the CentrifyDC mode returns connected. If the mode remains stuck in the starting state, then the Centrify client is experiencing server connection or authentication problems.

More comprehensive system and diagnostic information is available using:

```
1  adinfo --sysinfo all
2  adinfo --diag
```

To test connectivity to the various Active Directory and Kerberos services:

```
1  adinfo --test
```

### SSSD

Use the following information to set up SSSD. This section includes instructions for joining a Linux VDA machine to a Windows domain and provides guidance for configuring Kerberos authentication.

**Note:** If you are using SSSD, follow the instructions in this section.

### What is SSSD?

SSSD is a system daemon. Its primary function is to provide access to identify and authenticate remote resources through a common framework that can provide caching and offline support for the system. It provides both PAM and NSS modules, and in the future can support D-BUS based interfaces for extended user information. It also provides a better database to store local users and extended user data.

To set up SSSD on RHEL and CentOS, do the following:

1. Join the domain and create host keytab with Samba
2. Set up SSSD
3. Configure NSS/PAM
4. Verify the Kerberos configuration
5. Verify user authentication

### Required software

The Active Directory provider was first introduced with SSSD Version 1.9.0. If you are using an earlier version, follow the instructions provided in configuring the LDAP provider with Active Directory.

The following environments have been tested and verified when using the instructions included in this article:
• RHEL 7.6 or later
• CentOS 7.6 or later

Join the domain and create host keytab with Samba

SSSD does not provide Active Directory client functions for joining the domain and managing the system keytab file. There are a few methods for achieving the functions, including:

• adcli
• realmd
• Winbind
• Samba

The information in this section describes the Samba approach only. For realmd, see the RHEL or CentOS documentation. These steps must be followed before configuring SSSD.

On the Linux client with properly configured files:

• /etc/krb5.conf
• /etc/samba/smb.conf:

Configure the machine for Samba and Kerberos authentication:

```
sudo authconfig --smbsecurity=ads --smbworkgroup=domain --smbrealm=REALM --krb5realm=REALM --krb5kdc=fqdn-of-domain-controller --update
```

Where REALM is the Kerberos realm name in uppercase and domain is the short NetBIOS name of the Active Directory domain.

If DNS-based lookup of the KDC server and realm name is required, add the following two options to the preceding command:
```
--enablekrb5kdcdns --enablekrb5realmdns
```

Open /etc/samba/smb.conf and add the following entries under the [Global] section, but after the section generated by the authconfig tool:

```
kerberos method = secrets and keytab
```

Join the Windows domain. Ensure that your domain controller is reachable and you have an Active Directory user account with permissions to add computers to the domain:

```
sudo net ads join REALM -U user
```

REALM is the Kerberos realm name in uppercase and user is a domain user who has permissions to add computers to the domain.
Set up SSSD

Setting up SSSD consists of the following steps:

- Install the `sssd-ad` package on the Linux VDA.
- Make configuration changes to various files (for example, `sssd.conf`).
- Start the `sssd` service.

An example `sssd.conf` configuration (extra options can be added as needed):

```
[sssd]
config_file_version = 2
domains = ad.example.com
services = nss, pam

[domain/ad.example.com]
# Uncomment if you need offline logins
# cache_credentials = true

id_provider = ad
auth_provider = ad
access_provider = ad
ldap_id_mapping = true
ldap_schema = ad

# Should be specified as the lower-case version of the long version of the Active Directory domain.
ad_domain = ad.example.com

# Kerberos settings
krb5_ccachedir = /tmp
krb5_ccname_template = FILE:%d/krb5cc_%U

# Uncomment if service discovery is not working
# ad_server = server.ad.example.com

# Comment out if the users have the shell and home dir set on the AD side
default_shell = /bin/bash
fallback_homedir = /home/%d/%u

# Uncomment and adjust if the default principal SHORTNAME$@REALM is not available
# ldap_sasl_authid = host/client.ad.example.com@AD.EXAMPLE.COM
```

Replace `ad.example.com`, `server.ad.example.com` with the corresponding values. For more details,
see sssd-ad(5) - Linux man page.

Set the file ownership and permissions on sssd.conf:

```
chown root:root /etc/sssd/sssd.conf
chmod 0600 /etc/sssd/sssd.conf
restorecon /etc/sssd/sssd.conf
```

**Configure NSS/PAM**

**RHEL/CentOS:**

Use authconfig to enable SSSD. Install `oddjob-mkhomedir` to ensure that the home directory creation is compatible with SELinux:

```
1 authconfig --enablesssd --enablesssdauth --enablemkhomedir --update
2 sudo service sssd start
3 sudo chkconfig sssd on
```

**Verify Kerberos configuration**

Check that the system `keytab` file has been created and contains valid keys:

```
sudo klist -ke
```

This command displays the list of keys available for the various combinations of principal names and cipher suites. Run the Kerberos `kinit` command to authenticate the machine with the domain controller using these keys:

```
sudo kinit -k MACHINE\$@REALM
```

The machine and realm names must be specified in uppercase. The dollar sign ($) must be escaped with a backslash (\) to prevent shell substitution. In some environments, the DNS domain name is different from the Kerberos realm name. Ensure that the realm name is used. If this command is successful, no output is displayed.

Verify that the TGT ticket for the machine account has been cached using:

```
sudo klist
```
Verify user authentication

Use the `getent` command to verify that the logon format is supported and the NSS works:

```
1  sudo getent passwd DOMAIN\username
```

The `DOMAIN` parameter indicates the short version domain name. If another logon format is needed, verify by using the `getent` command first.

The supported logon formats are:

- Down-level logon name: `DOMAIN\username`
- UPN: `username@domain.com`
- NetBIOS Suffix format: `username@DOMAIN`

To verify that the SSSD PAM module is configured correctly, use a domain user account to log on to the Linux VDA. The domain user account has not been used before.

```
1  sudo ssh localhost -l DOMAIN\username
2  id -u
```

Check that a corresponding Kerberos credential cache file was created for the `uid` returned by the command:

```
1  ls /tmp/krb5cc_{uid}
```

Check that the tickets in the user’s Kerberos credential cache are valid and not expired:

```
1  klist
```

Install NVIDIA GRID drivers

Enabling HDX 3D Pro requires extra installation steps to install the requisite graphics drivers on the hypervisor and on the VDA machines.

Configure the following:

1. Citrix XenServer
2. VMware ESX

Follow the instructions for your chosen hypervisor.

**Citrix XenServer:**

This detailed section walks through the install and configuration of the NVIDIA GRID drivers on Citrix XenServer.
VMware ESX:

Follow the information contained in this guide to install and configure the NVIDIA GRID drivers for VMware ESX.

VDA machines:

Follow these steps to install and configure the drivers for each of the Linux VM guests:

1. Before starting, ensure that the Linux VM is shut down.
2. In XenCenter, add a GPU in GPU pass-through mode to the VM.
3. Start the RHEL VM.

To prepare the machine for the NVIDIA GRID drivers, run the following commands:

```bash
yum install gcc
yum install "kernel-devel-$\{uname -r\}"
systemctl set-default multi-user.target
```

Follow the steps in the Red Hat Enterprise Linux document to install the NVIDIA GRID driver.

Note:
During the GPU driver install, select the default ('no') for each question.

Important:
After GPU pass-through is enabled, the Linux VM is no longer accessible through XenCenter. Use SSH to connect.

```bash
nvidia-smi
```

Set the correct configuration for the card:
etc/X11/ctx-nvidia.sh

To take advantage of large resolutions and multi-monitor capabilities, you need a valid NVIDIA license. To apply for the license, follow the product documentation from “GRID Licensing Guide.pdf - DU-07757-001 September 2015.”

**Step 4: Install the Linux VDA**

You can do a fresh installation or upgrade an existing installation from the previous two versions and from an LTSR release.

**To do a fresh installation**

1. (Optional) Uninstall the old version
   
   If you installed an earlier version other than the previous two and an LTSR release, uninstall it before installing the new version.
   
   a) Stop the Linux VDA services:

   ```
   1 sudo /sbin/service ctxvda stop
   2
   3 sudo /sbin/service ctxhdx stop
   ```

   b) Uninstall the package:

   ```
   1 sudo rpm -e XenDesktopVDA
   ```

   **Note:**

   To execute a command, the full path is needed; alternately, you can add `/opt/Citrix/V-DA/sbin` and `/opt/Citrix/VDA/bin` to the system path.

2. Download the Linux VDA package

   Go to the Citrix website and download the appropriate Linux VDA package based on your Linux distribution.

3. Install the Linux VDA

   - Install the Linux VDA software using Yum:

   **For RHEL 7/CentOS 7:**

   ```
   1 sudo yum install -y LinuxVDA-1903.el7_x.rpm
   ```

   **For RHEL 6/CentOS 6:**
sudo yum install -y LinuxVDA-1903.el6_x.rpm

- Install the Linux VDA software using the RPM package manager. Before doing so, you must resolve the following dependencies:

For RHEL 7/CentOS 7:

sudo rpm -i LinuxVDA-1903.el7_x.rpm

For RHEL 6/CentOS 6:

sudo rpm -i LinuxVDA-1903.el6_x.rpm

RPM dependency list for RHEL 7/CentOS 7:

- postgresql-server >= 9.2
- postgresql-jdbc >= 9.2
- java-1.8.0-openjdk >= 1.8.0
- ImageMagick >= 6.7.8.9
- firewalld >= 0.3.9
- policycoreutils-python >= 2.0.83
- dbus >= 1.6.12
- dbus-x11 >= 1.6.12
- xorg-x11-server-utils >= 7.7
- xorg-x11-xinit >= 1.3.2
- libXpm >= 3.5.10
- libXrandr >= 1.4.1
- libXtst >= 1.2.2
- motif >= 2.3.4
- pam >= 1.1.8
For a matrix of the Linux distributions and the Xorg versions that this version of the Linux VDA supports, see System requirements.

**RPM dependency list for RHEL 6/CentOS 6:**

```
postgresql-jdbc >= 8.4
postgresql-server >= 8.4
```
java-1.7.0-openjdk >= 1.7.0
ImageMagick >= 6.5.4.7
GConf2 >= 2.28.0
system-config-firewall-base >= 1.2.27
policycoreutils-python >= 2.0.83
xorg-x11-server-utils >= 7.7
xorg-x11-xinit >= 1.0.9
ConsoleKit >= 0.4.1
dbus >= 1.2.24
dbus-x11 >= 1.2.24
libXpm >= 3.5.10
libXrandr >= 1.4.1
libXtst >= 1.2.2
openmotif >= 2.3.3
pam >= 1.1.1
util-linux-ng >= 2.17.2
bash >= 4.1
findutils >= 4.4
gawk >= 3.1
sed >= 4.2
cups >= 1.4.0
foomatic >= 4.0.0
Note:

After installing the Linux VDA on RHEL 7.x, run the `sudo yum install -y python-websockify` command. The purpose is to install python-websockify and x11vnc manually for using the session shadowing feature. For more information, see Shadowsessions.

To upgrade an existing installation

You can upgrade an existing installation from the previous two versions and from an LTSR release.

- To upgrade your software using Yum:

  **For RHEL 7/CentOS 7:**

  ```
  1  sudo yum install -y LinuxVDA-1903.el7_x.rpm
  ```

  **For RHEL 6/CentOS 6:**

  ```
  1  sudo yum install -y LinuxVDA-1903.el6_x.rpm
  ```

- To upgrade your software using the RPM package manager:

  **For RHEL 7/CentOS 7:**

  ```
  1  sudo rpm -U LinuxVDA-1903.el7_x.rpm
  ```
For RHEL 6/CentOS 6:

```bash
sudo rpm -U LinuxVDA-1903.el6_x.rpm
```

Important:

Restart the Linux VDA machine after upgrading the software.

**Step 5: Configure the Linux VDA**

After installing the package, you must configure the Linux VDA by running the `ctxsetup.sh` script. Before making any changes, the script verifies the environment and ensures that all dependencies are installed. If necessary, you can rerun the script at any time to change settings.

You can run the script manually with prompting, or automatically with preconfigured responses. Review Help about the script before proceeding:

```bash
sudo /opt/Citrix/VDA/sbin/ctxsetup.sh --help
```

**Prompted configuration**

Run a manual configuration with prompted questions:

```bash
sudo /opt/Citrix/VDA/sbin/ctxsetup.sh
```

**Automated configuration**

For an automated install, provide the options required by the setup script with environment variables. If all required variables are present, the script does not prompt for any information.

Supported environment variables include:

- **CTX_XDL_SUPPORT_DDC_AS_CNAME = Y | N** – The Linux VDA supports specifying a Delivery Controller name using a DNS CNAME record. Set to N by default.
- **CTX_XDL_DDC_LIST = list-ddc-fqdns** – The Linux VDA requires a space-separated list of Delivery Controller Fully Qualified Domain Names (FQDNs) to use for registering with a Delivery Controller. At least one FQDN or CNAME alias must be specified.
- **CTX_XDL_VDA_PORT = port-number** – The Linux VDA communicates with Delivery Controllers through a TCP/IP port, which is port 80 by default.
- **CTX_XDL_REGISTER_SERVICE = Y | N** - The Linux Virtual Desktop services are started after machine startup. The value is set to Y by default.
• **CTX_XDL_ADD_FIREWALL_RULES = Y | N** – The Linux Virtual Desktop services require incoming network connections to be allowed through the system firewall. You can automatically open the required ports (ports 80 and 1494 by default) in the system firewall for the Linux Virtual Desktop. Set to Y by default.

• **CTX_XDL_AD_INTEGRATION = 1 | 2 | 3 | 4** – The Linux VDA requires Kerberos configuration settings to authenticate with the Delivery Controllers. The Kerberos configuration is determined from the installed and configured Active Directory integration tool on the system. Specify the supported Active Directory integration method to use:
  - 1 – Samba Winbind
  - 2 – Quest Authentication Service
  - 3 – Centrify DirectControl
  - 4 – SSSD

• **CTX_XDL_HDX_3D_PRO = Y | N** – Linux Virtual Desktop supports HDX 3D Pro, a set of graphics acceleration technologies designed to optimize the virtualization of rich graphics applications. HDX 3D Pro requires a compatible NVIDIA GRID graphics card to be installed. If HDX 3D Pro is selected, the Virtual Delivery Agent is configured for VDI desktops (single-session) mode – (that is, CTX_XDL_VDI_MODE=Y). HDX 3D Pro is not supported on SUSE. Ensure that the value is set to N for a SUSE platform.

• **CTX_XDL_VDI_MODE = Y | N** – Whether to configure the machine as a dedicated desktop delivery model (VDI) or hosted shared desktop delivery model. For HDX 3D Pro environments, set this variable to Y. This variable is set to N by default.

• **CTX_XDL_SITE_NAME = dns-name** – The Linux VDA discovers LDAP servers through DNS. To limit the DNS search results to a local site, specify a DNS site name. This variable is set to <none> by default.

• **CTX_XDL_LDAP_LIST = list-ldap-servers** – The Linux VDA queries DNS to discover LDAP servers. If DNS cannot provide LDAP service records, you can provide a space-separated list of LDAP FQDNs with LDAP port. For example, ad1.mycompany.com:389. This variable is set to <none> by default.

• **CTX_XDL_SEARCH_BASE = search-base-set** – The Linux VDA queries LDAP through a search base set to the root of the Active Directory Domain (for example, DC=mycompany,DC=com). To improve search performance, you can specify a search base (for example, OU=VDI,DC=mycompany,DC=com). This variable is set to <none> by default.

• **CTX_XDL_FAS_LIST = list-fas-servers** – The Federated Authentication Service (FAS) servers are configured through AD Group Policy. Because the Linux VDA does not support AD Group Policy, you can provide a semicolon-separated list of FAS servers instead. The sequence must be the same as configured in AD Group Policy. If any server address is removed, fill its blank with the <none> text string and keep the sequence of server addresses without any changes.

• **CTX_XDL_START_SERVICE = Y | N** – Whether or not the Linux VDA services are started when the Linux VDA configuration is complete. Set to Y by default.
Set the environment variable and run the configure script:

```
export CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N
export CTX_XDL_DDC_LIST=list-ddc-fqdns
export CTX_XDL_VDA_PORT=port-number
export CTX_XDL_REGISTER_SERVICE=Y|N
export CTX_XDL_ADD_FIREWALL_RULES=Y|N
export CTX_XDL_AD_INTEGRATION=1|2|3|4
export CTX_XDL_VDI_MODE=Y|N
export CTX_XDL_SITE_NAME=dns-site-name | '<none>'
export CTX_XDL_LDAP_LIST=list-ldap-servers | '<none>'
export CTX_XDL_SEARCH_BASE=search-base-set | '<none>'
export CTX_XDL_FAS_LIST = list-fas-servers | '<none>'
export CTX_XDL_START_SERVICE=Y|N
sudo -E /opt/Citrix/VDA/sbin/ctxsetup.sh
```

When running the sudo command, type the -E option to pass the existing environment variables to the new shell it creates. Citrix recommends that you create a shell script file from the preceding commands with `#!/bin/bash` as the first line.

Alternatively, you can specify all parameters by using a single command:

```
sudo CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N 
CTX_XDL_DDC_LIST=list-ddc-fqdns 
CTX_XDL_VDA_PORT=port-number 
CTX_XDL_REGISTER_SERVICE=Y|N 
CTX_XDL_ADD_FIREWALL_RULES=Y|N 
```
Remove configuration changes

In some scenarios, you might have to remove the configuration changes made by the `ctxsetup.sh` script without uninstalling the Linux VDA package.

Review Help about this script before proceeding:

1. `sudo /opt/Citrix/VDA/sbin/ctxcleanup.sh --help`

To remove configuration changes:

1. `sudo /opt/Citrix/VDA/sbin/ctxcleanup.sh`

**Important:**

This script deletes all configuration data from the database and renders the Linux VDA inoperable.

**Configuration logs**

The `ctxsetup.sh` and `ctxcleanup.sh` scripts display errors on the console, with additional information written to the configuration log file `/tmp/xdl.configure.log`. Restart the Linux VDA services to have the changes take effect.
**Step 6: Run the Linux VDA**

After configuring the Linux VDA by using the `ctxsetup.sh` script, you can run the following commands to control the Linux VDA.

**Start the Linux VDA:**

To start the Linux VDA services:

```bash
1 sudo /sbin/service ctxhdx start
2 sudo /sbin/service ctxvda start
```

**Stop the Linux VDA:**

To stop the Linux VDA services:

```bash
1 sudo /sbin/service ctxvda stop
2 sudo /sbin/service ctxhdx stop
```

**Restart the Linux VDA:**

To restart the Linux VDA services:

```bash
1 sudo /sbin/service ctxvda stop
2 sudo /sbin/service ctxhdx restart
4 sudo /sbin/service ctxvda start
```

**Check the status of the Linux VDA:**

To check the running status of the Linux VDA services:

```bash
1 sudo /sbin/service ctxvda status
2 sudo /sbin/service ctxhdx status
```

**Step 7: Create the machine catalog in Citrix Virtual Apps or Citrix Virtual Desktops**

The process for creating machine catalogs and adding Linux VDA machines is similar to the traditional Windows VDA approach. For a more detailed description of how to complete these tasks, see [Create machine catalogs](#) and [Manage machine catalogs](#).

For creating machine catalogs that contain Linux VDA machines, there are a few restrictions that differentiate the process from creating machine catalogs for Windows VDA machines:
• For the operating system, select:
  – The Server OS option for a hosted shared desktops delivery model.
  – The Desktop OS option for a VDI dedicated desktop delivery model.
• Ensure that machines are set as not power managed.
• Do not mix Linux and Windows VDA machines in the same machine catalog.

Note:
Early versions of Citrix Studio did not support the notion of a “Linux OS.” However, selecting the Windows Server OS or Server OS option implies an equivalent hosted shared desktops delivery model. Selecting the Windows Desktop OS or Desktop OS option implies a single user per machine delivery model.

Tip:
If you remove and rejoin a machine to the Active Directory domain, you must remove and add the machine to the machine catalog again.

Step 8: Create the delivery group in Citrix Virtual Apps or Citrix Virtual Desktops

The process for creating a delivery group and adding machine catalogs containing Linux VDA machines is almost identical to Windows VDA machines. For a more detailed description of how to complete these tasks, see Create Delivery Groups.

For creating delivery groups that contain Linux VDA machine catalogs, the following restrictions apply:

• For the delivery type, select Desktops or Applications.
• Ensure that the AD users and groups you select have been properly configured to log on to the Linux VDA machines.
• Do not allow logon of unauthenticated (anonymous) users.
• Do not mix the delivery group with machine catalogs that contain Windows machines.

Important:
Publishing applications is supported with Linux VDA Version 1.4 and later. However, the Linux VDA does not support the delivery of desktops and apps to the same machine.

For information about how to create machine catalogs and delivery groups, see Citrix Virtual Apps and Desktops 7 1903.

Install Linux Virtual Delivery Agent for SUSE

March 25, 2019
You can choose to follow the steps in this article for manual installation or use easy install for automatic installation and configuration. Easy install saves time and labor and is less error-prone than the manual installation.

Note:
Use easy install only for fresh installations. Do not use easy install to update an existing installation.

**Step 1: Prepare for installation**

**Step 1a: Launch the YaST tool**

The SUSE Linux Enterprise YaST tool is used for configuring all aspects of the operating system.

To launch the text-based YaST tool:

```
1 su -
2 yast
```

Alternatively, launch the UI-based YaST tool:

```
1 su -
2 yast2 &
```

**Step 1b: Configure networking**

The following sections provide information on configuring the various networking settings and services used by the Linux VDA. Configuring networking is carried out via the YaST tool, not via other methods such as Network Manager. These instructions are based on using the UI-based YaST tool. The text-based YaST tool can be used but has a different method of navigation that is not documented here.

**Configure host name and DNS**

1. Open YaST Network Settings.
2. SLED 12 Only: On the Global Options tab, change the Network Setup Method to Wicked Service.
3. Open the Hostname/DNS tab.
4. Clear Change hostname via DHCP.
5. Check Assign Hostname to Loopback IP.

6. Edit the following to reflect your networking setup:
   - Host name – Add the DNS host name of the machine.
   - Domain name – Add the DNS domain name of the machine.
   - Name server – Add the IP address of the DNS server. It is typically the IP address of the AD Domain Controller.
   - Domain search list – Add the DNS domain name.

   **Note:**
   The Linux VDA currently does not support NetBIOS name truncation. Therefore, the host name must not exceed 15 characters.

   **Tip:**
   Use a–z, A–Z, 0–9, and hyphen (-) characters only. Avoid underscores (_), spaces, and other symbols. Do not start a host name with a number and do not end with a hyphen. This rule also applies to Delivery Controller host names.

**Disable multicast DNS**

On SLED only, the default settings have multicast DNS (mDNS) enabled, which can lead to inconsistent name resolution results. mDNS is not enabled on SLES by default, so no action is required.

To disable mDNS, edit `/etc/nsswitch.conf` and change the line containing:

```
hosts: files mdns_minimal [NOTFOUND=return] dns
```

To:

```
hosts: files dns
```

**Check the host name**

Verify that the host name is set correctly:

```
1 hostname
```

This command returns only the machine’s host name and not its fully qualified domain name (FQDN).

Verify that the FQDN is set correctly:

```
1 hostname -f
```

This command returns the machine’s FQDN.
Check name resolution and service reachability

Verify that you can resolve the FQDN and ping the domain controller and Delivery Controller:

```
1 nslookup domain-controller-fqdn
2 ping domain-controller-fqdn
3 nslookup delivery-controller-fqdn
4 ping delivery-controller-fqdn
```

If you cannot resolve the FQDN or ping either of these machines, review the steps before proceeding.

Step 1c: Configure the NTP service

It is crucial to maintain accurate clock synchronization between the VDAs, Delivery Controllers, and domain controllers. Hosting the Linux VDA as a virtual machine can cause clock skew problems. For this reason, maintaining time using a remote NTP service is preferred. Some changes might be required to the default NTP settings:

1. Open YaST NTP Configuration and select the General Settings tab.
2. In the Start NTP Daemon section, check Now and on Boot.
3. If present, select the Undisciplined Local Clock (LOCAL) item and click Delete.
4. Add an entry for an NTP server by clicking Add.
5. Select the Server Type and click Next.
6. Type the DNS name of the NTP server in the Address field. This service is normally hosted on the Active Directory domain controller.
7. Leave the Options field unchanged.
8. Click Test to check that the NTP service is reachable.
9. Click OK through the set of windows to save the changes.

Note:

For SLES 12 implementations, the NTP daemon might fail to start due to a known SUSE issue with AppArmor policies. Follow the resolution for additional information.

Step 1d: Install Linux VDA dependent packages

The Linux VDA software for SUSE Linux Enterprise depends on the following packages:

- PostgreSQL
  - SLED/SLES 12: Version 9.3 or later
Linux Virtual Delivery Agent 1903

- OpenJDK 1.7.0
- OpenMotif Runtime Environment 2.3.1 or later
- Cups
  - SLED/SLES 12: Version 1.6.0 or later
- Foomatic filters
  - SLED/SLES 12: Version 1.0.0 or later
- ImageMagick
  - SLED/SLES 12: Version 6.8 or later

Add repositories

Some required packages are not available in all SUSE Linux Enterprise repositories:

- SLED 12: PostgreSQL is available for SLES 12 but not SLED 12. ImageMagick is available via the SLE 12 SDK ISO or online repository.
- SLES 12: There are no issues. All packages are available. ImageMagick is available via the SLE 12 SDK ISO or online repository.

To resolve the issue, obtain missing packages from the media for the alternative edition of SLE from which you are installing. That is, on SLED install missing packages from the SLES media, and on SLES install missing packages from the SLED media. The following approach mounts both SLED and SLES ISO media files and adds repositories.

- On SLED 12, run the commands:

```
1 sudo mkdir -p /mnt/sles
2 3 sudo mount -t iso9660 path-to-iso/SLES-12-SP3-DVD-x86_64-GM-DVD1.iso /mnt/sles
4 5 sudo zypper ar -f /mnt/sles sles
```

- On SLED/SLES 12, run the commands:

```
1 sudo mkdir -p /mnt/sdk
2 3 sudo mount -t iso9660 path-to-iso/SLE-12-SP3-SDK-DVD-x86_64-GM-DVD1.iso /mnt/sdk
4 5 sudo zypper ar -f /mnt/sdk sdk
```

Install the Kerberos client
Install the Kerberos client for mutual authentication between the Linux VDA and the Delivery Controllers:

```
1 sudo zypper install krb5-client
```

The Kerberos client configuration depends on which Active Directory integration approach is used. See the following description.

**Install OpenJDK**

The Linux VDA depends on OpenJDK 1.7.0.

**Tip:**

To avoid registration failure with the Delivery Controller, ensure that you installed only OpenJDK Version 1.7.0. Remove all other versions of Java from your system.

On SUSE 12.3, if you have java-1_7_0-openjdk-1.7.0.201-43.18.1.x86_64 or a minor version later than OpenJDK 1.7.0.201 installed, run the `zypper install –oldpackage java-1_7_0-openjdk-1.7.0.181-43.15.2.x86_64` command to downgrade your Java runtime environment.

- **SLED:**
  1. On SLED, the Java runtime environment is typically installed with the operating system. Check whether it has been installed:

```
1 sudo zypper info java-1_7_0-openjdk
```

  2. Update to the latest version if the status is reported as out-of-date:

```
1 sudo zypper update java-1_7_0-openjdk
```

  3. Check the Java version:

```
1 java -version
```

- **SLES:**
  1. On SLES, install the Java runtime environment:

```
1 sudo zypper install java-1_7_0-openjdk
```

  2. Check the Java version:

```
1 java -version
```
Install PostgreSQL

On SLED/SLES 12, install the packages:

1. `sudo zypper install postgresql-init`
2. `sudo zypper install postgresql-server`
3. `sudo zypper install postgresql-jdbc`

Post-installation steps are required to initialize the database service and to ensure that PostgreSQL is started upon machine startup:

1. `sudo systemctl enable postgresql`
2. `sudo systemctl restart postgresql`

Database files locate at `/var/lib/pgsql/data`.

Remove repositories

With dependent packages installed, the alternative edition repositories set up earlier can now be removed and the media unmounted:

- on SLED 12, run the commands to remove the packages:

1. `sudo zypper rr sles`
2. `sudo umount /mnt/sles`
3. `sudo rmdir /mnt/sles`

- on SLED/SLES 12, run the commands to remove the packages:

1. `sudo zypper rr sdk`
2. `sudo umount /mnt/sdk`
3. `sudo rmdir /mnt/sdk`

Step 2: Prepare Linux VM for Hypervisor

Some changes are required when running the Linux VDA as a virtual machine on a supported hypervisor. Make the following changes according to the hypervisor platform in use. No changes are required if you are running the Linux machine on bare metal hardware.
Fix time synchronization on Citrix XenServer

If the XenServer Time Sync feature is enabled, within each paravirtualized Linux VM you experience issues with NTP and XenServer both trying to manage the system clock. To avoid the clock becoming out of sync with other servers, the system clock within each Linux guest must be synchronized with NTP. This case requires disabling host time synchronization. No changes are required in HVM mode.

On some Linux distributions, if you are running a paravirtualized Linux kernel with XenServer Tools installed, you can check whether the XenServer Time Sync feature is present and enabled from within the Linux VM:

```
1 su -
2 cat /proc/sys/xen/independent_wallclock
```

This command returns 0 or 1:

- 0 - The time sync feature is enabled, and must be disabled.
- 1 - The time sync feature is disabled, and no further action is required.

If the `/proc/sys/xen/independent_wallclock` file is not present, the following steps are not required.

If enabled, disable the time sync feature by writing 1 to the file:

```
1 sudo echo 1 > /proc/sys/xen/independent_wallclock
```

To make this change permanent and persistent after restart, edit the `/etc/sysctl.conf` file and add the line:

```sh
xen.independent_wallclock = 1
```

To verify these changes, restart the system:

```
1 reboot
```

After restart, check that the setting is correct:

```
1 su -
2 cat /proc/sys/xen/independent_wallclock
```

This command returns the value 1.

Fix time synchronization on Microsoft Hyper-V

Linux VMs with Hyper-V Linux Integration Services installed can apply the Hyper-V time synchronization feature to use the host operating system’s time. To ensure that the system clock remains accurate,
enable this feature alongside the NTP services.

From the management operating system:

1. Open the Hyper-V Manager console.
2. For the settings of a Linux VM, select Integration Services.
3. Ensure that Time synchronization is selected.

Note:
This approach is different from VMware and XenServer, where host time synchronization is disabled to avoid conflicts with NTP. Hyper-V time synchronization can coexist and supplement NTP time synchronization.

Fix time synchronization on ESX and ESXi

If the VMware Time Synchronization feature is enabled, within each paravirtualized Linux VM you experience issues with NTP and the hypervisor both trying to synchronize the system clock. To avoid the clock becoming out of sync with other servers, the system clock within each Linux guest must be synchronized with NTP. This case requires disabling host time synchronization.

If you are running a paravirtualized Linux kernel with VMware Tools installed:

1. Open the vSphere Client.
2. Edit settings for the Linux VM.
3. In the Virtual Machine Properties dialog, open the Options tab.
4. Select VMware Tools.
5. In the Advanced box, clear Synchronize guest time with host.

Step 3: Add the Linux virtual machine (VM) to the Windows domain

The Linux VDA supports several methods for adding Linux machines to the Active Directory (AD) domain:

- Samba Winbind
- Quest Authentication Service
- Centrify DirectControl

Follow instructions based on your chosen method.

Note:
Session launches might fail when the same user name is used for the local account in the Linux VDA and the account in AD.
**Samba Winbind**

**Join Windows domain**

Your domain controller must be reachable and you must have an Active Directory user account with permissions to add machines to the domain:

1. Open YaST Windows Domain Membership.
2. Make the following changes:
   - Set the Domain or Workgroup to the name of your Active Directory domain or the IP address of the domain controller. Ensure that the domain name is in uppercase.
   - Check Also Use SMB information for Linux Authentication.
   - Check Create Home Directory on Login.
   - Check Single Sign-on for SSH.
   - Ensure that Offline Authentication is not checked. This option is not compatible with the Linux VDA.
3. Click OK. If prompted to install some packages, click Install.
4. If a domain controller is found, it asks whether you want to join the domain. Click Yes.
5. When prompted, type the credentials of a domain user with permission to add computers to the domain and click OK.
6. A message indicating success is displayed.
7. If prompted to install some samba and krb5 packages, click Install.

YaST might have indicated that these changes require some services or the machine to be restarted. It is advisable to restart the machine:

```
1  su -
2  reboot
```

**SLED/SLES 12 Only: Patch Kerberos credential cache name**

SLED/SLES 12 has changed the default Kerberos credential cache name specification from the usual FILE:/tmp/krb5cc_%{uid} to DIR:/run/user/%{uid}/krb5cc. This new DIR caching method is not compatible with the Linux VDA and must be manually changed. As a root user, edit /etc/krb5.conf and add the following setting under the [libdefaults] section if not set:

```bash
default_ccache_name = FILE:/tmp/krb5cc_%{uid}
```
Verify domain membership

The Delivery Controller requires that all VDA machines (Windows and Linux VDAs) have a computer object in Active Directory.

Run the **net ads** command of Samba to verify that the machine is joined to a domain:

```
sudo net ads testjoin
```

Run the following command to verify extra domain and computer object information:

```
sudo net ads info
```

Verify Kerberos configuration

To ensure that Kerberos is configured correctly for use with the Linux VDA, check that the system keytab file has been created and contains valid keys:

```
sudo klist -ke
```

This command displays the list of keys available for the various combinations of principal names and cipher suites. Run the Kerberos kinit command to authenticate the machine with the domain controller using these keys:

```
sudo kinit -k MACHINE\@REALM
```

The machine and realm names must be specified in uppercase. The dollar sign ($) must be escaped with a backslash (\) to prevent shell substitution. In some environments, the DNS domain name is different from the Kerberos realm name. Ensure that the realm name is used. If this command is successful, no output is displayed.

Verify that the TGT ticket for the machine account has been cached using:

```
sudo klist
```

Examine the machine account details using:

```
sudo net ads status
```

Verify user authentication

Use the wbinfo tool to verify that domain users can authenticate with the domain:

```
wbinfo --krb5auth=domain\username\password
```
The domain specified here is the AD domain name, not the Kerberos realm name. For the bash shell, the backslash (\) character must be escaped with another backslash. This command returns a message indicating success or failure.

To verify that the Winbind PAM module is configured correctly, use a domain user account to log on to the Linux VDA. The domain user account has not been used before.

```bash
1 ssh localhost -l domain\username
2 id -u
```

Check that a corresponding Kerberos credential cache file was created for the uid returned by the `id -u` command:

```bash
1 ls /tmp/krb5cc_uid
```

Check that the tickets in the user's Kerberos credential cache are valid and not expired:

```bash
1 klist
```

Exit the session

```bash
1 exit
```

A similar test can be performed by logging on to the Gnome or KDE console directly.

**Quest authentication service**

**Configure Quest on domain controller**

Assume that you have installed and configured the Quest software on the Active Directory domain controllers, and have been granted administrative privileges to create computer objects in Active Directory.

**Enable domain users to log on to Linux VDA machines**

To enable domain users to establish HDX sessions on a Linux VDA machine:

1. In the Active Directory Users and Computers management console, open Active Directory user properties for that user account.
2. Select the **Unix Account** tab.
3. Check **Unix-enabled**.
4. Set the **Primary GID Number** to the group ID of an actual domain user group.
These instructions are equivalent for setting up domain users for logon using the console, RDP, SSH, or any other remoting protocol.

**Configure Quest on Linux VDA**

**Configure VAS daemon**

Auto-renewal of Kerberos tickets must be enabled and disconnected. Authentication (offline logon) must be disabled:

```
1  sudo /opt/quest/bin/vastool configure vas vasd auto-ticket-renew-interval 32400
2
3  sudo /opt/quest/bin/vastool configure vas vas_auth allow-disconnected-auth false
```

This command sets the renewal interval to nine hours (32,400 seconds) which is one hour less than the default 10-hour ticket lifetime. Set this parameter to a lower value on systems with a shorter ticket lifetime.

**Configure PAM and NSS**

To enable domain user logon through HDX and other services such as su, ssh, and RDP, run the following commands to manually configure PAM and NSS:

```
1  sudo /opt/quest/bin/vastool configure pam
2
3  sudo /opt/quest/bin/vastool configure nss
```

**Join Windows domain**

Join the Linux machine to the Active Directory domain using the Quest vastool command:

```
1  sudo /opt/quest/bin/vastool -u user join domain-name
```

The **user** is any domain user who has permissions to join computers to the Active Directory domain. The **domain-name** is the DNS name of the domain, for example, example.com.
Verify domain membership

The Delivery Controller requires that all VDA machines (Windows and Linux VDAs) have a computer object in Active Directory. To verify that a Quest-joined Linux machine is on the domain:

```
1 sudo /opt/quest/bin/vastool info domain
```

If the machine is joined to a domain, this command returns the domain name. If the machine is not joined to any domain, the following error appears:

```
ERROR: No domain could be found.
ERROR: VAS_ERR_CONFIG: at ctx.c:414 in _ctx_init_default_realm
default_realm not configured in vas.conf. Computer may not be joined to domain
```

Verify user authentication

To verify that Quest can authenticate domain users through PAM, use a domain user account to log on to the Linux VDA. The domain user account has not been used before.

```
1 ssh localhost -l domain\username
2 id -u
```

Check that a corresponding Kerberos credential cache file was created for the uid returned by the `id -u` command:

```
1 ls /tmp/krb5cc_uid
```

Check that the tickets in the Kerberos credential cache are valid and not expired:

```
1 /opt/quest/bin/vastool klist
```

Exit the session:

```
1 exit
```

A similar test can be performed by logging on to the Gnome or KDE console directly.

Centrify DirectControl

Join Windows domain

With the Centrify DirectControl Agent installed, join the Linux machine to the Active Directory domain using the Centrify `adjoin` command:
Verify domain membership

The Delivery Controller requires that all VDA machines (Windows and Linux VDAs) have a computer object in Active Directory. To verify that a Centrify-joined Linux machine is on the domain:

```
su -
adjoin -w -V -u user domain-name
```

Check that the **Joined to domain** value is valid and the **CentrifyDC mode** returns **connected**. If the mode remains stuck in the starting state, then the Centrify client is experiencing server connection or authentication problems.

More comprehensive system and diagnostic information is available using:

```
adinfo --sysinfo all
adinfo --diag
```

To test connectivity to the various Active Directory and Kerberos services:

```
adinfo --test
```

**Step 4: Install the Linux VDA**

**Step 4a: Uninstall the old version**

If you have previously installed a version of the Linux VDA earlier than V1.1, uninstall it before installing the new version.

1. Stop the Linux VDA services:

```
sudo /sbin/service ctxvda stop
sudo /sbin/service ctxhdx stop
```

2. Uninstall the package:
```
sudo rpm -e XenDesktopVDA
```

**Important:**
Upgrading from the latest two versions is supported.

**Note:**
Starting with Version 1.3, the installation path has changed. In previous releases, installation components were located in `/usr/local/`. The new location is `/opt/Citrix/VDA/`.

To execute a command, the full path is needed; alternatively, you can add `/opt/Citrix/VDA/sbin` and `/opt/Citrix/VDA/bin` to the system path.

**Step 4b: Download the Linux VDA package**

Go to the Citrix website and download the appropriate Linux VDA package based on your Linux distribution.

**Step 4c: Install the Linux VDA**

Install the Linux VDA software using Zypper:

**For SUSE 12:**
```
sudo zypper install LinuxVDA-1903.sle12_x.rpm
```

Install the Linux VDA software using the RPM package manager. Before doing so, resolve the following dependencies:

**For SUSE 12:**
```
sudo rpm -i LinuxVDA-1903.sle12_x.rpm
```

**Step 4d: Upgrade the Linux VDA (optional)**

You can upgrade the Linux VDA software from the previous two versions using the RPM package manager:

**For SUSE 12:**
```
sudo rpm -U LinuxVDA-1903.sle12_x.rpm
```

**RPM Dependency list for SUSE 12:**
postgresql-server >= 9.3
postgresql-jdbc >= 9.2
java-1.7.0-openjdk >= 1.7.0
ImageMagick >= 6.8
dbus-1 >= 1.8.8
dbus-1-x11 >= 1.8.8
libXpm4 >= 3.5.11
libXrandr2 >= 1.4.2
libXtst6 >= 1.2.2
motif >= 2.3
pam >= 1.1.8
bash >= 4.2
findutils >= 4.5
gawk >= 4.1
sed >= 4.2
cups >= 1.6.0
cups-filters-foomatic-rip >= 1.0.0
openssl >= 2.4
cyrus-sasl >= 2.1
cyrus-sasl-gssapi >= 2.1
libxml2 >= 2.9
python-requests >= 2.8.1
Step 5: Configure the Linux VDA

After installing the package, you must configure the Linux VDA by running the ctxsetup.sh script. Before making any changes, the script verifies the environment and ensures that all dependencies are installed. If necessary, you can rerun the script at any time to change settings.

You can run the script manually with prompting, or automatically with preconfigured responses. Review Help about the script before proceeding:

```
sudo /opt/Citrix/VDA/sbin/ctxsetup.sh -help
```

Prompted configuration

Run a manual configuration with prompted questions:

```
sudo /opt/Citrix/VDA/sbin/ctxsetup.sh
```

Automated configuration

For an automated installation, provide the options required by the setup script with environment variables. If all required variables are present, the script does not prompt for any information.

Supported environment variables include:
• **CTX_XDL_SUPPORT_DDC_AS_CNAME = Y | N** – The Linux VDA supports specifying a Delivery Controller name using a DNS CNAME record. Set to N by default.

• **CTX_XDL_DDC_LIST = list-ddc-fqdns** – The Linux VDA requires a space-separated list of Delivery Controller Fully Qualified Domain Names (FQDNs) to use for registering with a Delivery Controller. At least one FQDN or CNAME alias must be specified.

• **CTX_XDL_VDA_PORT = port-number** – The Linux VDA communicates with Delivery Controllers through a TCP/IP port, which is port 80 by default.

• **CTX_XDL_REGISTER_SERVICE = Y | N** - The Linux Virtual Desktop services are started after machine startup. The value is set to Y by default.

• **CTX_XDL_ADD_FIREWALL_RULES = Y | N** – The Linux Virtual Desktop services require incoming network connections to be allowed through the system firewall. You can automatically open the required ports (ports 80 and 1494 by default) in the system firewall for the Linux Virtual Desktop. Set to Y by default.

• **CTX_XDL_AD_INTEGRATION = 1 | 2 | 3 | 4** – The Linux VDA requires Kerberos configuration settings to authenticate with the Delivery Controllers. The Kerberos configuration is determined from the installed and configured Active Directory integration tool on the system. Specify the supported Active Directory integration method to use:
  - 1 – Samba Winbind
  - 2 – Quest Authentication Service
  - 3 – Centrify DirectControl
  - 4 – SSSD

• **CTX_XDL_HDX_3D_PRO = Y | N** – Linux Virtual Desktop supports HDX 3D Pro, a set of graphics acceleration technologies designed to optimize the virtualization of rich graphics applications. HDX 3D Pro requires a compatible NVIDIA GRID graphics card to be installed. If HDX 3D Pro is selected, the Virtual Delivery Agent is configured for VDI desktops (single-session) mode – (that is, CTX_XDL_VDI_MODE=Y). HDX 3D Pro is not supported on SUSE. Ensure that the value is set to N for a SUSE platform.

• **CTX_XDL_VDI_MODE = Y | N** – Whether to configure the machine as a dedicated desktop delivery model (VDI) or hosted shared desktop delivery model. For HDX 3D Pro environments, set this variable to Y. This variable is set to N by default.

• **CTX_XDL_SITE_NAME = dns-name** – The Linux VDA discovers LDAP servers through DNS. To limit the DNS search results to a local site, specify a DNS site name. This variable is set to `<none>` by default.

• **CTX_XDL_LDAP_LIST = list-ldap-servers** – The Linux VDA queries DNS to discover LDAP servers. If DNS cannot provide LDAP service records, you can provide a space-separated list of LDAP FQDNs with LDAP port. For example, ad1.mycompany.com:389. This variable is set to `<none>` by default.

• **CTX_XDL_SEARCH_BASE = search-base-set** – The Linux VDA queries LDAP through a search base set to the root of the Active Directory Domain (for example, DC=mycompany,DC=com). To
improve search performance, you can specify a search base (for example, OU=VDI,DC=mycompany,DC=com).
This variable is set to <none> by default.

- **CTX_XDL_FAS_LIST = list-fas-servers** – The Federated Authentication Service (FAS) servers are configured through AD Group Policy. Because the Linux VDA does not support AD Group Policy, you can provide a semicolon-separated list of FAS servers instead. The sequence must be the same as configured in AD Group Policy. If any server address is removed, fill its blank with the <none> text string and keep the sequence of server addresses without any changes.

- **CTX_XDL_START_SERVICE = Y | N** – Whether or not the Linux VDA services are started when the Linux VDA configuration is complete. Set to Y by default.

**Note:**
HDX 3D Pro is not currently available on SUSE.

Set the environment variable and run the configure script:

```
export CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N
export CTX_XDL_DDC_LIST=list-ddc-fqdns
export CTX_XDL_VDA_PORT=port-number
export CTX_XDL_REGISTER_SERVICE=Y|N
export CTX_XDL_ADD_FIREWALL_RULES=Y|N
export CTX_XDL_AD_INTEGRATION=1|2|3|4
export CTX_XDL_HDX_3D_PRO=Y|N
export CTX_XDL_VDI_MODE=Y|N
export CTX_XDL_SITE_NAME=dns-site-name | '<none>'
export CTX_XDL_LDAP_LIST=list-ldap-servers | '<none>'
export CTX_XDL_SEARCH_BASE=search-base-set | '<none>'
export CTX_XDL_FAS_LIST = list-fas-servers | '<none>'
export CTX_XDL_START_SERVICE=Y|N
sudo -E /opt/Citrix/VDA/sbin/ctxsetup.sh
```

When running the sudo command, type the -E option to pass the existing environment variables to
the new shell it creates. Citrix recommends that you create a shell script file from the preceding commands with 

```
#!/bin/bash
```

as the first line.

Alternatively, you can specify all parameters by using a single command:

```
1  sudo CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N \n2  CTX_XDL_DDC_LIST=list-ddc-fqdns \n3  CTX_XDL_VDA_PORT=port-number \n4  CTX_XDL_REGISTER_SERVICE=Y|N \n5  CTX_XDL_ADD_FIREWALL_RULES=Y|N \n6  CTX_XDL_AD_INTEGRATION=1|2|3|4 \n7  CTX_XDL_HDX_3D_PRO=Y|N \n8  CTX_XDL_VDI_MODE=Y|N \n9  CTX_XDL_SITE_NAME=dns-name \n10 CTX_XDL_LDAP_LIST=list-ldap-servers \n11 CTX_XDL_SEARCH_BASE=search-base-set \n12 CTX_XDL_FAS_LIST = list-fas-servers \n13 CTX_XDL_START_SERVICE=Y|N \n14 /opt/Citrix/VDA/sbin/ctxsetup.sh
```  

**Remove configuration changes**

In some scenarios, you might have to remove the configuration changes made by the `ctxsetup.sh` script without uninstalling the Linux VDA package.

Review Help about this script before proceeding:

```
1  sudo /usr/local/sbin/ctxcleanup.sh --help
```

To remove configuration changes:

```
1  sudo /usr/local/sbin/ctxcleanup.sh
```
Important:
This script deletes all configuration data from the database and renders the Linux VDA inoperable.

Configuration logs

The `ctxsetup.sh` and `ctxcleanup.sh` scripts display errors on the console, with additional information written to a configuration log file:

```
/tmp/xdl.configure.log
```

Restart the Linux VDA services to have the changes take effect.

Step 6: Run the Linux VDA

After configuring the Linux VDA by using the `ctxsetup.sh` script, you can run the following commands to control the Linux VDA.

Start the Linux VDA:

To start the Linux VDA services:

```
1  sudo /sbin/service ctxhdx start
2  sudo /sbin/service ctxvda start
```

Stop the Linux VDA:

To stop the Linux VDA services:

```
1  sudo /sbin/service ctxvda stop
2  sudo /sbin/service ctxhdx stop
```

Restart the Linux VDA:

To restart the Linux VDA services:

```
1  sudo /sbin/service ctxvda stop
2  sudo /sbin/service ctxhdx restart
3  sudo /sbin/service ctxvda start
5  sudo /sbin/service ctxvda start
```
Check the Linux VDA status:

To check the running status of the Linux VDA services:

```
1 sudo /sbin/service ctxvda status
2 sudo /sbin/service ctxhdx status
```

Step 7: Create the machine catalog in Citrix Virtual Apps or Citrix Virtual Desktops

The process for creating machine catalogs and adding Linux VDA machines is similar to the traditional Windows VDA approach. For a more detailed description of how to complete these tasks, see Create machine catalogs and Manage machine catalogs.

For creating machine catalogs that contain Linux VDA machines, there are a few restrictions that differentiate the process from creating machine catalogs for Windows VDA machines:

- For the operating system, select:
  - The Server OS option for a hosted shared desktops delivery model.
  - The Desktop OS option for a VDI dedicated desktop delivery model.
- Ensure that machines are set as not power managed.
- Do not mix Linux and Windows VDA machines in the same machine catalog.

Note:

Early versions of Citrix Studio did not support the notion of a “Linux OS.” However, selecting the Windows Server OS or Server OS option implies an equivalent hosted shared desktops delivery model. Selecting the Windows Desktop OS or Desktop OS option implies a single user per machine delivery model.

Tip:

If you remove and rejoin a machine to the Active Directory domain, you must remove and add the machine to the machine catalog again.

Step 8: Create the delivery group in Citrix Virtual Apps or Citrix Virtual Desktops

The process for creating a delivery group and adding machine catalogs containing Linux VDA machines is almost identical to Windows VDA machines. For a more detailed description of how to complete these tasks, see Create Delivery Groups.

For creating delivery groups that contain Linux VDA machine catalogs, the following restrictions apply:

- For delivery type, select Desktops or Applications.
• Ensure that the AD users and groups you select have been properly configured to log on to the Linux VDA machines.
• Do not allow logon of unauthenticated (anonymous) users.
• Do not mix the delivery group with machine catalogs that contain Windows machines.

Important:
Publishing applications is supported with Linux VDA Version 1.4 and later. However, the Linux VDA does not support the delivery of desktops and apps to the same machine.

For information about how to create machine catalogs and delivery groups, see Citrix Virtual Apps and Desktops 7 1903.

Install Linux Virtual Delivery Agent for Ubuntu

April 10, 2019

You can choose to follow the steps in this article for manual installation or use easy install for automatic installation and configuration. Easy install saves time and labor and is less error-prone than the manual installation.

Note:
Use easy install only for fresh installations. Do not use easy install to update an existing installation.

Step 1: Prepare Ubuntu for VDA installation

Step 1a: Verify the network configuration

Citrix recommends that the network is connected and configured correctly before proceeding.

If you are using a Ubuntu 18.04 Live Server, make the following change in the /etc/cloud/cloud.cfg configuration file before setting the host name:

preserve_hostname: true

Step 1b: Set the host name

To ensure that the host name of the machine is reported correctly, change the /etc/hostname file to contain only the host name of the machine.

hostname
**Step 1c: Assign a loopback address to the host name**

To ensure that the DNS domain name and FQDN of the machine are reported back correctly, change the following line of the `/etc/hosts` file to include the FQDN and host name as the first two entries:

```
127.0.0.1 hostname-fqdn hostname localhost
```

For example:

```
127.0.0.1 vda01.example.com vda01 localhost
```

Remove any other references to hostname-fqdn or hostname from other entries in the file.

**Note:**

The Linux VDA currently does not support NetBIOS name truncation. Therefore, the host name must not exceed 15 characters.

**Tip:**

Use a–z, A–Z, 0–9, and hyphen (-) characters only. Avoid underscores (_), spaces, and other symbols. Do not start a host name with a number and do not end with a hyphen. This rule also applies to Delivery Controller host names.

**Step 1d: Check the host name**

Verify that the host name is set correctly:

```
1 hostname
```

This command returns only the host name of the machine and not its FQDN.

Verify that the FQDN is set correctly:

```
1 hostname -f
```

This command returns the FQDN of the machine.

**Step 1e: Disable multicast DNS**

The default settings have multicast DNS (mDNS) enabled, which can lead to inconsistent name resolution results.

To disable mDNS, edit `/etc/nsswitch.conf` and change the line containing:

```
hosts: files mdns_minimal [NOTFOUND=return] dns
```

To:

```
hosts: files dns
```
**Step 1f: Check name resolution and service reachability**

Verify that you can resolve the FQDN and ping the domain controller and Delivery Controller:

```bash
1. nslookup domain-controller-fqdn
2. ping domain-controller-fqdn
3. nslookup delivery-controller-fqdn
4. ping delivery-controller-fqdn
```

If you cannot resolve the FQDN or ping either of these machines, review the steps before proceeding.

**Step 1g: Configure clock synchronization (chrony)**

Maintaining accurate clock synchronization between the VDAs, Delivery Controllers and domain controllers is crucial. Hosting the Linux VDA as a virtual machine can cause clock skew problems. For this reason, synchronizing time with a remote time service is preferred.

Install chrony:

```bash
1. apt-get install chrony
```

As a root user, edit `/etc/chrony/chrony.conf` and add a server entry for each remote time server:

```
server peer1-fqdn-or-ip-address iburst
server peer2-fqdn-or-ip-address iburst
```

In a typical deployment, synchronize time from the local domain controllers and not directly from public NTP pool servers. Add a server entry for each Active Directory domain controller in the domain.

Remove any other `server` or `pool` entries listed including loopback IP address, localhost, and public server `*.pool.ntp.org` entries.

Save changes and restart the Chrony daemon:

```bash
1. sudo systemctl restart chrony
```

**Step 1h: Install OpenJDK**

The Linux VDA depends on OpenJDK. Typically, the runtime environment is installed as part of the operating system installation.

On Ubuntu 16.04, install OpenJDK by using:

```bash
```
On Ubuntu 18.04, install OpenJDK by using:
```bash
sudo apt-get install -y openjdk-8-jdk
```

**Step 1i: Install PostgreSQL**

The Linux VDA requires PostgreSQL Version 9.x on Ubuntu:
```bash
sudo apt-get install -y postgresql
1 2
sudo apt-get install -y libpostgresql-jdbc-java
```

**Step 1j: Install Motif**

```bash
sudo apt-get install -y libxm4
```

**Step 1k: Install other packages**

```bash
sudo apt-get install -y libsasl2-2
1 2
sudo apt-get install -y libsasl2-modules-gssapi-mit
3 4
sudo apt-get install -y libldap-2.4-2
5 6
sudo apt-get install -y krb5-user
7 8
9 sudo apt-get install -y cups
```

**Step 1l: Install the following package (Ubuntu 18.04 only)**

```bash
sudo apt-get install -y libgtk2.0-0
```
**Step 2: Prepare the hypervisor**

Some changes are required when running the Linux VDA as a virtual machine on a supported hypervisor. Make the following changes according to the hypervisor platform in use. No changes are required if you are running the Linux machine on bare metal hardware.

**Fix time synchronization on Citrix XenServer**

When the XenServer Time Sync feature is enabled, within each paravirtualized Linux VM you experience issues with the NTP and the XenServer, both of which try to manage the system clock. To avoid the clock becoming out of sync with other servers, ensure that the system clock within each Linux guest is synchronized with the NTP. This case requires disabling host time synchronization. No changes are required in HVM mode.

On some Linux distributions, if you are running a paravirtualized Linux kernel with XenServer Tools installed, you can check whether the XenServer Time Sync feature is present and enabled from within the Linux VM:

```
1 su -
2 cat /proc/sys/xen/independent_wallclock
```

This command returns 0 or 1:

- **0** - The time sync feature is enabled, and must be disabled.
- **1** - The time sync feature is disabled, and no further action is required.

If the `/proc/sys/xen/independent_wallclock` file is not present, the following steps are not required.

If enabled, disable the time sync feature by writing 1 to the file:

```
1 sudo echo 1 > /proc/sys/xen/independent_wallclock
```

To make this change permanent and persistent after restart, edit the `/etc/sysctl.conf` file and add the line:

```
xen.independent_wallclock = 1
```

To verify these changes, restart the system:

```
1 su -
2 cat /proc/sys/xen/independent_wallclock
```

This command returns the value 1.
Fix time synchronization on Microsoft Hyper-V

Linux VMs with Hyper-V Linux Integration Services installed can leverage the Hyper-V time synchronization feature to use the host operating system’s time. To ensure that the system clock remains accurate, this feature must be enabled alongside NTP services.

From the management operating system:

1. Open the Hyper-V Manager console.
2. For the settings of a Linux VM, select Integration Services.
3. Ensure that Time synchronization is selected.

Note:
This approach is different from VMware and XenServer, where host time synchronization is disabled to avoid conflicts with NTP. Hyper-V time synchronization can coexist and supplement NTP time synchronization.

Fix time synchronization on ESX and ESXi

When the VMware Time Synchronization feature is enabled, within each paravirtualized Linux VM you experience issues with the NTP and the hypervisor, both of which try to synchronize the system clock. To avoid the clock becoming out of sync with other servers, ensure that the system clock within each Linux guest is synchronized with the NTP. This case requires disabling host time synchronization.

If you are running a paravirtualized Linux kernel with VMware Tools installed:

1. Open the vSphere Client.
2. Edit settings for the Linux VM.
3. In the Virtual Machine Properties dialog, open the Options tab.
4. Select VMware Tools.
5. In the Advanced box, clear Synchronize guest time with host.

Step 3: Add the Linux virtual machine (VM) to the Windows domain

The Linux VDA supports several methods for adding Linux machines to the Active Directory (AD) domain:

- Samba Winbind
- Quest Authentication Service
- Centrify DirectControl
- SSSD

Follow instructions based on your chosen method.
Note:
Session launches might fail when the same user name is used for the local account in the Linux VDA and the account in AD.

Samba Winbind

Install or update the required packages

```bash
sudo apt-get install winbind samba libnss-winbind libpam-winbind krb5-config krb5-locales krb5-user
```

Enable Winbind daemon to start on machine startup

The Winbind daemon must be configured to start on machine startup:

```bash
sudo systemctl enable winbind
```

Configure Kerberos

Open /etc/krb5.conf as a root user, and make the following settings:

```bash
[libdefaults]

default_realm = REALM
dns_lookup_kdc = false

[realms]
REALM = {
  admin_server = domain-controller-fqdn
  kdc = domain-controller-fqdn
}
```
The `domain-dns-name` property in this context is the DNS domain name, such as `example.com`. The `REALM` is the Kerberos realm name in uppercase, such as `EXAMPLE.COM`.

### Configure Winbind Authentication

Configure Winbind manually because Ubuntu does not have a tool like authconfig in RHEL and yast2 in SUSE.

Open `/etc/samba/smb.conf`, and make the following settings:

```
[global]
workgroup = WORKGROUP
security = ADS
realm = REALM
encrypt passwords = yes
idmap config *:range = 16777216-33554431
winbind trusted domains only = no
kerberos method = secrets and keytab
winbind refresh tickets = yes
template shell = /bin/bash
```

`WORKGROUP` is the first field in `REALM`, and `REALM` is the Kerberos realm name in uppercase.

### Configure nsswitch

Open `/etc/nsswitch.conf`, and append `winbind` to the following lines:

```
passwd: compat winbind
group: compat winbind
```
Join Windows Domain

Your domain controller must be reachable and you must have an Active Directory user account with permissions to add computers to the domain:

```
sudo net ads join REALM -U user
```

Where `REALM` is the Kerberos realm name in uppercase, and `user` is a domain user with permissions to add computers to the domain.

Restart winbind

```
sudo systemctl restart winbind
```

Configure PAM for Winbind

Run the following command and ensure that the **Winbind NT/Active Directory authentication** and **Create home directory on login** options are selected:

```
sudo pam-auth-update
```

Tip:
The winbind daemon stays running only if the machine is joined to a domain.

Verify Domain Membership

The Delivery Controller requires that all VDA machines, whether Windows or Linux, have a computer object in Active Directory.

Run the **net ads** command of Samba to verify that the machine is joined to a domain:

```
sudo net ads testjoin
```

Run the following command to verify extra domain and computer object information:

```
sudo net ads info
```

Verify Kerberos Configuration

To verify that Kerberos is configured correctly for use with the Linux VDA, check that the system **keytab** file has been created and contains valid keys:

```
sudo klist -ke
```
This command displays the list of keys available for the various combinations of principal names and cipher suites. Run the Kerberos kinit command to authenticate the machine with the domain controller using these keys:

```shell
sudo kinit -k MACHINE\$@REALM
```

The machine and realm names must be specified in uppercase. The dollar sign ($) must be escaped with a backslash (\) to prevent shell substitution. In some environments, the DNS domain name is different from the Kerberos realm name. Ensure that the realm name is used. If this command is successful, no output is displayed.

Verify that the TGT ticket for the machine account has been cached using:

```shell
sudo klist
```

Examine the account details of the machine using:

```shell
sudo net ads status
```

### Verify user authentication

Use the `wbinfo` tool to verify that domain users can authenticate with the domain:

```shell
wbinfo --krb5auth=domain\username\password
```

The domain specified here is the AD domain name, not the Kerberos realm name. For the bash shell, the backslash (\) character must be escaped with another backslash. This command returns a message indicating success or failure.

To verify that the Winbind PAM module is configured correctly, use a domain user account to log on to the Linux VDA. The domain user account has not been used before.

```shell
ssh localhost -l domain\username
id -u
```

Check that a corresponding Kerberos credential cache file was created for the uid returned by the `id -u` command:

```shell
ls /tmp/krb5cc_uid
```

Check that the tickets in the user’s Kerberos credential cache are valid and not expired:

```shell
klist
```
Exit the session:

```
exit
```

A similar test can be performed by logging on to the Gnome or KDE console directly.

**Tip:**

If you succeed in user authentication but cannot show your desktop when logging on with a domain account, restart the machine and then try again.

---

**Quest authentication service**

**Configure Quest on domain controller**

Assume that you have installed and configured the Quest software on the Active Directory domain controllers, and have been granted administrative privileges to create computer objects in Active Directory.

**Enable domain users to log on to Linux VDA machines**

To enable domain users to establish HDX sessions on a Linux VDA machine:

1. In the Active Directory Users and Computers management console, open Active Directory user properties for that user account.
2. Select the **Unix Account** tab.
3. Check **Unix-enabled**.
4. Set the **Primary GID Number** to the group ID of an actual domain user group.

**Note:**

These instructions are equivalent for setting up domain users for logon using the console, RDP, SSH, or any other remoting protocol.

---

**Configure Quest on Linux VDA**

**Work around SELinux policy enforcement**

The default RHEL environment has SELinux fully enforced. This enforcement interferes with the Unix domain socket IPC mechanisms used by Quest, and prevents domain users from logging on.

The convenient way to work around this issue is to disable SELinux. As a root user, edit `/etc/selinux/-config` and change the **SELinux** setting:

```bash
SELINUX=disabled
```
This change requires a machine restart:

```
reboot
```

**Important:**

Use this setting carefully. Reenabling SELinux policy enforcement after disabling can cause a complete lockout, even for the root user and other local users.

**Configure VAS daemon**

Auto-renewal of Kerberos tickets must be enabled and disconnected. Authentication (offline logon) must be disabled:

```
1  sudo /opt/quest/bin/vastool configure vas vasd auto-ticket-renew-interval 32400
2  sudo /opt/quest/bin/vastool configure vas vas_auth allow-disconnected-auth false
```

This command sets the renewal interval to nine hours (32,400 seconds) which is one hour less than the default 10-hour ticket lifetime. Set this parameter to a lower value on systems with a shorter ticket lifetime.

**Configure PAM and NSS**

To enable domain user logon through HDX and other services such as su, ssh, and RDP, run the following commands to manually configure PAM and NSS:

```
1  sudo /opt/quest/bin/vastool configure pam
2  sudo /opt/quest/bin/vastool configure nss
```

**Join Windows domain**

Join the Linux machine to the Active Directory domain using the Quest vastool command:

```
1  sudo /opt/quest/bin/vastool -u user join domain-name
```

The user is any domain user with permissions to join computers to the Active Directory domain. The domain-name is the DNS name of the domain, for example, example.com.
Linux Virtual Delivery Agent 1903

Verify domain membership
The Delivery Controller requires that all VDA machines, whether Windows or Linux, have a computer object in Active Directory. To verify that a Quest-joined Linux machine is on the domain:

```
sudo /opt/quest/bin/vastool info domain
```

If the machine is joined to a domain, this command returns the domain name. If the machine is not joined to any domain, the following error appears:

ERROR: No domain could be found.
ERROR: VAS_ERR_CONFIG: at ctx.c:414 in _ctx_init_defaultRealm
default_realm not configured in vas.conf. Computer may not be joined to domain

Verify user authentication
To verify that Quest can authenticate domain users through PAM, use a domain user account to log on to the Linux VDA. The domain user account has not been used before.

```
ssh localhost -l domain\username
id -u
```

Check that a corresponding Kerberos credential cache file was created for the UID returned by the `id -u` command:

```
ls /tmp/krb5cc_uid
```

Check that the tickets in the Kerberos credential cache are valid and not expired:

```
/opt/quest/bin/vastool klist
```

Exit the session:

```
exit
```

Centrify DirectControl

Join Windows domain
With the Centrify DirectControl Agent installed, join the Linux machine to the Active Directory domain using the Centrify adjoin command:
The user parameter is any Active Directory domain user with permissions to join computers to the Active Directory domain. The domain-name parameter is the name of the domain to join the Linux machine to.

**Verify domain membership**

The Delivery Controller requires that all VDA machines, whether Windows or Linux, have a computer object in Active Directory. To verify that a Centrify-joined Linux machine is on the domain:

```bash
su -
adjoin -w -V -u user domain-name
```

Check that the Joined to domain value is valid and the CentrifyDC mode returns connected. If the mode remains stuck in the starting state, then the Centrify client is experiencing server connection or authentication problems.

More comprehensive system and diagnostic information is available using:

```bash
adinfo --sysinfo all

adinfo --diag
```

To test connectivity to the various Active Directory and Kerberos services:

```bash
adinfo --test
```

**SSSD**

**Configure Kerberos**

Run the following command to install Kerberos:

```bash
sudo apt-get install krb5-user
```

To configure Kerberos, open /etc/krb5.conf as root and make the following settings:

```bash
[libdefaults]

default_realm = REALM
```
The domain-dns-name property in this context is the DNS domain name, such as example.com. The REALM is the Kerberos realm name in uppercase, such as EXAMPLE.COM.

Join the domain

SSSD must be configured to use Active Directory as its identity provider and Kerberos for authentication. However, SSSD does not provide AD client functions for joining the domain and managing the system keytab file. There are a few methods for domain joining:

- adcli
- samba
- realmd

Note:
This section only provides information for adcli and samba.

Use adcli to join the domain:

Install adcli:
Install the required package:

```
Paste 1
sudo apt-get install adcli
```

Join the domain with adcli:
Remove the old system keytab file and join the domain using:

```bash
1  su -
2  rm -rf /etc/krb5.keytab
3  adcli join domain-dns-name -U user -H hostname-fqdn
```

The **user** is a domain user with permissions to add machines to the domain. The **hostname-fqdn** is the host name in FQDN format for the machine.

The **-H** option is necessary for adcli to generate SPN in the format of host/hostname-fqdn@REALM, which the Linux VDA requires.

**Verify system keytab:**

The capabilities of the **adcli** tool are limited and do not provide a way to test whether a machine is joined to the domain. The best alternative to ensure that the system keytab file has been created:

```bash
1  sudo klist -ket
```

Verify that the timestamp for each key matches the time the machine was joined to the domain.

**Use samba to join the domain:**

**Install the package:**

```bash
1  sudo apt-get install samba
```

**Configure samba:**

Open `/etc/samba/smb.conf`, and make the following settings:

```bash
1  [global]
2  workgroup = WORKGROUP
3  security = ADS
4  realm = REALM
5  client signing = yes
6  client use spnego = yes
7  kerberos method = secrets and keytab
```

**WORKGROUP** is the first field in **REALM**, and REALM is the Kerberos realm name in uppercase.
Join the domain with samba:

Your domain controller must be reachable and you must have a Windows account with permissions to add computers to the domain.

```
sudo net ads join REALM -U user
```

Where **REALM** is the Kerberos realm name in uppercase, and **user** is a domain user with permissions to add computers to the domain.

Set up SSSD

Install or update required packages:

Install the required SSSD and configuration packages if not already installed:

```
sudo apt-get install sssd
```

If the packages are already installed, an update is recommended:

```
sudo apt-get install --only-upgrade sssd
```

**Note:**

By default, the install process in Ubuntu automatically configures **nsswitch.conf** and the PAM login module.

Configure SSSD

SSSD configuration changes are required before starting the SSSD daemon. For some versions of SSSD, the **/etc/sssd/sssd.conf** configuration file is not installed by default and must be manually created. As root, either create or open **/etc/sssd/sssd.conf** and make the following settings:

```
[sssd]
services = nss, pam
config_file_version = 2
domains = domain-dns-name
[domain/domain-dns-name]
id_provider = ad
```
access_provider = ad
auth_provider = krb5
krb5_realm = REALM

# Set krb5_renewable_lifetime higher if TGT renew lifetime is longer than 14 days
krb5_renewable_lifetime = 14d

# Set krb5_renew_interval to lower value if TGT ticket lifetime is shorter than 2 hours
krb5_renew_interval = 1h

krb5_ccachedir = /tmp
krb5_ccname_template = FILE:%d/krb5cc_%U

# This ldap_id_mapping setting is also the default value
ldap_id_mapping = true
override_homedir = /home/%d/%u
default_shell = /bin/bash
ad_gpo_map_remote_interactive = +ctxhdx

Note:

ldap_id_mapping is set to true so that SSSD itself takes care of mapping Windows SIDs to Unix UIDs. Otherwise, the Active Directory must be able to provide POSIX extensions. PAM service ctxhdx is added to ad_gpo_map_remote_interactive.

The domain-dns-name property in this context is the DNS domain name, such as example.com. The REALM is the Kerberos realm name in uppercase, such as EXAMPLE.COM. There is no requirement to configure the NetBIOS domain name.

Tip:

For information on these configuration settings, see the man pages for sssd.conf and sssd-ad.

The SSSD daemon requires that the configuration file must have owner read permission only:
Start SSSD daemon

Run the following commands to start the SSSD daemon now and to enable the daemon to start upon machine startup:

1. `sudo systemctl start sssd`
2. `sudo systemctl enable sssd`

PAM configuration

Run the following command and ensure that the SSSS authentication and Create home directory on login options are selected:

1. `sudo pam-auth-update`

Verify domain membership

The Delivery Controller requires that all VDA machines (Windows and Linux VDAs) have a computer object in Active Directory.

Use adcli to verify domain membership:

Show the domain information by running the following command:

1. `sudo adcli info domain-dns-name`

Use samba to verify domain membership:

Run the net ads command of Samba to verify that the machine is joined to a domain:

1. `sudo net ads testjoin`

Run the following command to verify extra domain and computer object information:

1. `sudo net ads info`

Verify Kerberos configuration

To verify that Kerberos is configured correctly for use with the Linux VDA, check that the system keytab file has been created and contains valid keys:
1 sudo klist -ke

This command displays the list of keys available for the various combinations of principal names and cipher suites. Run the Kerberos kinit command to authenticate the machine with the domain controller using these keys:

1 sudo kinit - k MACHINE\$@REALM

The machine and realm names must be specified in uppercase. The dollar sign ($) must be escaped with a backslash (\) to prevent shell substitution. In some environments, the DNS domain name is different from the Kerberos realm name. Ensure that the realm name is used. If this command is successful, no output is displayed.

Check that TGT ticket for the machine account has been cached using:

1 sudo klist

**Verify user authentication**

SSSD does not provide a command-line tool for testing authentication directly with the daemon, and can only be done via PAM.

To verify that the SSSD PAM module is configured correctly, use a domain user account to log on to the Linux VDA. The domain user account has not been used before.

1 ssh localhost -l domain\username
2 id -u
3 klist
4 exit

Check that the Kerberos tickets returned by the klist command are correct for that user and have not expired.

As a root user, check that a corresponding ticket cache file was created for the uid returned by the previous id -u command:

1 ls /tmp/krb5cc_uid

A similar test can be performed by logging on to KDE or Gnome Display Manager.
Step 4: Install the Linux VDA

Step 4a: Download the Linux VDA package

Go to the Citrix website and download the appropriate Linux VDA package based on your Linux distribution.

Step 4b: Install the Linux VDA

Install the Linux VDA software using the Debian package manager:

```bash
sudo dpkg -i LinuxVDA-1903.ubuntu16.04.deb
```

Debian dependency list for Ubuntu 18.04:

```plaintext
postgresql  >= 9.5
libpostgresql-jdbc-java  >= 9.2
openjdk-8-jdk  >= 1.8.0
gtk3-nocsd  >=3
imagemagick  >= 8:6.8.9.9
ufw  >= 0.35
ubuntu-desktop  >= 1.361
libxrandr2  >= 2:1.5.0
libxtst6  >= 2:1.2.2
libxm4  >= 2.3.4
util-linux  >= 2.27.1
bash  >= 4.3
findutils  >= 4.6.0
sed  >= 4.2.2
cups  >= 2.1
```
Debian dependency list for Ubuntu 16.04:

- postgresql >= 9.5
- libpostgresql-jdbc-java >= 9.2
- default-jdk >= 2:1.8
- imagemagick >= 8:6.8.9.9
- ufw >= 0.35
- ubuntu-desktop >= 1.361
- libxrandr2 >= 2:1.5.0
- libxtst6 >= 2:1.2.2
- libxm4 >= 2.3.4
- util-linux >= 2.27.1
- bash >= 4.3
- findutils >= 4.6.0
- sed >= 4.2.2
Step 4c: Configure the Linux VDA

After installing the package, you must configure the Linux VDA by running the ctxsetup.sh script. Before making any changes, the script verifies the environment and ensures that all dependencies are installed. If necessary, you can rerun the script at any time to change settings.

You can run the script manually with prompting, or automatically with preconfigured responses. Review Help about the script before proceeding:

```
sudo /opt/Citrix/VDA/sbin/ctxsetup.sh -help
```

**Prompted configuration**

Run a manual configuration with prompted questions:

```
sudo /opt/Citrix/VDA/sbin/ctxsetup.sh
```

**Automated configuration**

Note:
For a matrix of the Linux distributions and the Xorg versions that this version of the Linux VDA supports, see System requirements.
For an automated install, the options required by the setup script can be provided with environment variables. If all required variables are present, the script does not prompt the user for any information, allowing for a scripted installation process.

Supported environment variables include:

- **CTX_XDL_SUPPORT_DDC_AS_CNAME = Y | N** – The Linux VDA supports specifying a Delivery Controller name using a DNS CNAME record. Set to N by default.
- **CTX_XDL_DDC_LIST = list-ddc-fqdns** – The Linux VDA requires a space-separated list of Delivery Controller Fully Qualified Domain Names (FQDNs) to use for registering with a Delivery Controller. At least one FQDN or CNAME alias must be specified.
- **CTX_XDL_VDA_PORT = port-number** – The Linux VDA communicates with Delivery Controllers through a TCP/IP port, which is port 80 by default.
- **CTX_XDL_REGISTER_SERVICE = Y | N** – The Linux Virtual Desktop services are started after machine startup. Set to Y by default.
- **CTX_XDL_ADD_FIREWALL_RULES = Y | N** – The Linux Virtual Desktop services require incoming network connections to be allowed through the system firewall. You can automatically open the required ports (ports 80 and 1494 by default) in the system firewall for the Linux Virtual Desktop. Set to Y by default.
- **CTX_XDL_AD_INTEGRATION = 1 | 2 | 3 | 4** – The Linux VDA requires Kerberos configuration settings to authenticate with the Delivery Controllers. The Kerberos configuration is determined from the installed and configured Active Directory integration tool on the system. Specify the supported Active Directory integration method to use:
  - 1 – Samba Winbind
  - 2 – Quest Authentication Service
  - 3 – Centrify DirectControl
  - 4 – SSSD
- **CTX_XDL_HDX_3D_PRO = Y | N** – Linux Virtual Desktop supports HDX 3D Pro, a set of graphics acceleration technologies designed to optimize the virtualization of rich graphics applications. HDX 3D Pro requires a compatible NVIDIA GRID graphics card to be installed. If HDX 3D Pro is selected, the Virtual Delivery Agent is configured for VDI desktops (single-session) mode – (that is, CTX_XDL_VDI_MODE=Y).
- **CTX_XDL_VDI_MODE = Y | N** – Whether to configure the machine as a dedicated desktop delivery model (VDI) or hosted shared desktop delivery model. For HDX 3D Pro environments, set this variable to Y. This variable is set to N by default.
- **CTX_XDL_SITE_NAME = dns-name** – The Linux VDA discovers LDAP servers through DNS. To limit the DNS search results to a local site, specify a DNS site name. This variable is set to <none> by default.
- **CTX_XDL_LDAP_LIST = list-ldap-servers** – The Linux VDA queries DNS to discover LDAP servers. If DNS cannot provide LDAP service records, you can provide a space-separated list of LDAP FQDNs with LDAP port. For example, ad1.mycompany.com:389. This variable is set to
Linux Virtual Delivery Agent 1903

*<none>* by default.

- **CTX_XDL_SEARCH_BASE = search-base-set** – The Linux VDA queries LDAP through a search base set to the root of the Active Directory Domain (for example, DC=mycompany,DC=com). However, to improve search performance, you can specify a search base (for example, OU=VDI,DC=mycompany,DC=com). This variable is set to *<none>* by default.

- **CTX_XDL_FAS_LIST = list-fas-servers** – The Federated Authentication Service (FAS) servers are configured through AD Group Policy. Because the Linux VDA does not support AD Group Policy, you can provide a semicolon-separated list of FAS servers instead. The sequence must be the same as configured in AD Group Policy. If any server address is removed, fill its blank with the *<none>* text string and keep the sequence of server addresses without any changes.

- **CTX_XDL_START_SERVICE = Y | N** – Whether or not the Linux VDA services are started when the Linux VDA configuration is complete. Set to Y by default.

Set the environment variable and run the configure script:

```bash
export CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N
export CTX_XDL_DDC_LIST=list-ddc-fqdns
export CTX_XDL_VDA_PORT=port-number
export CTX_XDL_REGISTER_SERVICE=Y|N
export CTX_XDL_ADD_FIREWALL_RULES=Y|N
export CTX_XDL_AD_INTEGRATION=1|2|3|4
export CTX_XDL_HDX_3D_PRO=Y|N
export CTX_XDL_VDI_MODE=Y|N
export CTX_XDL_SITE_NAME=dns-site-name | '<none>'
export CTX_XDL_LDAP_LIST=list-ldap-servers | '<none>'
export CTX_XDL_SEARCH_BASE=search-base-set | '<none>'
export CTX_XDL_FAS_LIST = list-fas-servers | '<none>'
export CTX_XDL_START_SERVICE=Y|N
sudo -E /opt/Citrix/VDA/sbin/ctxsetup.sh
```

When running the sudo command, type the -E option to pass the existing environment variables to
the new shell it creates. Citrix recommends that you create a shell script file from the preceding commands with `#!/bin/bash` as the first line.

Alternatively, you can specify all parameters by using a single command:

```bash
sudo /opt/Citrix/VDA/sbin/ctxsetup.sh
```

### Remove configuration changes

In some scenarios, you might have to remove the configuration changes made by the `ctxsetup.sh` script without uninstalling the Linux VDA package.

Review Help about this script before proceeding:

```bash
sudo /opt/Citrix/VDA/sbin/ctxcleanup.sh --help
```

To remove configuration changes:

```bash
sudo /opt/Citrix/VDA/sbin/ctxcleanup.sh
```
Important:
This script deletes all configuration data from the database and renders the Linux VDA inoperable.

Configuration logs
The ctxsetup.sh and ctxcleanup.sh scripts display errors on the console, with additional information written to the configuration log file /tmp/xdl.configure.log.

Restart the Linux VDA services to have the changes take effect.

Uninstall the Linux VDA software
To check whether the Linux VDA is installed and to view the version of the installed package:

```
1 dpkg -l xendesktopvda
```
To view more detailed information:

```
1 apt-cache show xendesktopvda
```

Note:
Uninstalling the Linux VDA software deletes the associated PostgreSQL and other configuration data. However, the PostgreSQL package and other dependent packages that were set up before the installation of the Linux VDA are not deleted.

Tip:
The information in this section does not cover the removal of dependent packages including PostgreSQL.

Step 5: Run the Linux VDA

Once you have configured the Linux VDA using the ctxsetup.sh script, you use the following commands to control the Linux VDA.

Start the Linux VDA:
To start the Linux VDA services:

```
1 sudo systemctl start ctxhdx
2
3 sudo systemctl start ctxvda
```
Stop the Linux VDA:
To stop the Linux VDA services:

1  sudo systemctl stop ctxvda
2  sudo systemctl stop ctxhdx

Restart the Linux VDA:
To restart the Linux VDA services:

1  sudo systemctl stop ctxvda
2  3  sudo systemctl restart ctxhdx
4  5  sudo systemctl restart ctxvda

Check the Linux VDA status:
To check the running status of the Linux VDA services:

1  sudo systemctl status ctxvda
2  3  sudo systemctl status ctxhdx

Step 6: Create the machine catalog in Citrix Virtual Apps or Citrix Virtual Desktops

The process for creating machine catalogs and adding Linux VDA machines is similar to the traditional Windows VDA approach. For a more detailed description of how to complete these tasks, see Create machine catalogs and Manage machine catalogs.

For creating machine catalogs that contain Linux VDA machines, there are a few restrictions that differentiate the process from creating machine catalogs for Windows VDA machines:

- For the operating system, select:
  - The Server OS option for a hosted shared desktops delivery model.
  - The Desktop OS option for a VDI dedicated desktop delivery model.
- Ensure that machines are set as not power managed.
- Do not mix Linux and Windows VDA machines in the same machine catalog.

Note:

Early versions of Citrix Studio did not support the notion of a “Linux OS.” However, selecting the Windows Server OS or Server OS option implies an equivalent hosted shared desktops delivery
model. Selecting the Windows Desktop OS or Desktop OS option implies a single user per machine delivery model.

Tip:
If you remove and rejoin a machine to the Active Directory domain, you must remove and add the machine to the machine catalog again.

Step 7: Create the delivery group in Citrix Virtual Apps or Citrix Virtual Desktops

The process for creating a delivery group and adding machine catalogs containing Linux VDA machines is almost identical to Windows VDA machines. For a more detailed description of how to complete these tasks, see Create Delivery Groups.

For creating delivery groups that contain Linux VDA machine catalogs, the following restrictions apply:

- For delivery type, select Desktops. Linux VDA for Ubuntu does not support application delivery.
- Ensure that the AD users and groups you select have been properly configured to log on to the Linux VDA machines.
- Do not allow logon of unauthenticated (anonymous) users.
- Do not mix the delivery group with machine catalogs that contain Windows machines.

For information about how to create machine catalogs and delivery groups, see Citrix Virtual Apps and Desktops 7 1903.

Configure the Linux VDA

August 29, 2018

This chapter details features of the Linux VDA, including feature description, configuration, and troubleshooting.

Tip:
The xdlcollect Bash script used to collect logs is integrated into the Linux VDA software and located under /opt/Citrix/VDA/bin. After you install the Linux VDA, you can run the bash /opt/Citrix/VDA/bin/xdlcollect.sh command to collect logs.

After log collection completes, a compressed log file is generated in the same folder as the script. Xdlcollect can ask you whether or not to upload the compressed log file to Citrix Insight Services (CIS). If you agree, xdlcollect returns an upload_ID after the upload completes. The upload does
Integrate NIS with Active Directory

March 25, 2019

This article describes how to integrate NIS with Windows Active Directory (AD) on the Linux VDA by using SSSD. The Linux VDA is considered a component of Citrix Virtual Apps and Desktops. As a result, it fits tightly into the Windows AD environment.

Using NIS as a UID and GID provider instead of using AD requires that the account information (user name and password combinations) is the same in both AD and NIS.

Note:
Authentication is still performed by the AD server. NIS+ is not supported. If you use NIS as the UID and GID provider, the POSIX attributes from the Windows server are no longer used.

Tip:
This method represents a deprecated way to deploy the Linux VDA, which is used only for special use cases. For an RHEL/CentOS distribution, follow the instructions in Install Linux Virtual Delivery Agent for RHEL/CentOS. For an Ubuntu distribution, follow the instructions in Install Linux Virtual Delivery Agent for Ubuntu.

What is SSSD?
SSSD is a system daemon. Its primary function is to provide access to identify and authenticate remote resources through a common framework that can provide caching and offline support for the system. It provides both PAM and NSS modules, and in the future can support D-BUS based interfaces for extended user information. It also provides a better database to store local user accounts and extended user data.

Required software

The AD provider was first introduced with SSSD Version 1.9.0.

The following environments have been tested and verified when using the instructions included in this article:

- RHEL 7.6 or later
- CentOS 7.6 or later
**Integrate NIS with AD**

To integrate NIS with AD, do the following:

1. **Add the Linux VDA as a NIS client**
2. **Join the domain and create a host keytab using Samba**
3. **Set up SSSD**
4. **Configure NSS/PAM**
5. **Verify the Kerberos configuration**
6. **Verify user authentication**

### Add the Linux VDA as a NIS client

Configure the NIS client:

```
1 yum -y install ypbind rpcbind oddjob-mkhomedir
```

Set the NIS domain:

```
1 ypdomainname nis.domain
2 echo "NISDOMAIN=nis.domain" >> /etc/sysconfig/network
```

Add the IP address for the NIS server and client in `/etc/hosts`:

```
{ NIS server IP address } server.nis.domain nis.domain
```

Configure NIS by authconfig:

```
1 sudo authconfig --enablenis --nisdomain=nis.domain --nisserver=server.nis.domain --enablemkhomedir --update
```

The **nis.domain** represents the domain name of the NIS server. The **server.nis.domain** is the host name of the NIS server, which can also be the IP address of the NIS server.

Configure the NIS services:

```
1 sudo systemctl start rpcbind ybind
2 3 sudo systemctl enable rpcbind ybind
```

Ensure that the NIS configuration is correct:

```
1 ypwhich
```

Validate that the account information is available from the NIS server:
Linux Virtual Delivery Agent 1903

Note:
The **nisaccount** represents the real NIS account on the NIS server. Ensure that the UID, GID, home directory, and login shell are configured correctly.

### Join the domain and create a host keytab using Samba

SSSD does not provide AD client functions for joining the domain and managing the system keytab file. There are a few methods for achieving the functions, including:

- adcli
- realmd
- Winbind
- Samba

The information in this section describes the Samba approach only. For **realmd**, see the RHEL or CentOS vendor’s documentation. These steps must be followed before configuring SSSD.

### Join the domain and create host keytab using Samba:

On the Linux client with properly configured files:

- `/etc/krb5.conf`
- `/etc/samba/smb.conf`:

Configure the machine for Samba and Kerberos authentication:

```
1  sudo authconfig --smbsecurity=ads --smbworkgroup=domain --smbrealm=REALM --krb5realm=REALM --krb5kdc=fqdn-of-domain-controller --update
```

Where **REALM** is the Kerberos realm name in uppercase and **domain** is the NetBIOS name of the domain.

If DNS-based lookup of the KDC server and realm name is required, add the following two options to the preceding command:

```
--enablekrb5kcdcdns  --enablekrb5realmdns
```

Open `/etc/samba/smb.conf` and add the following entries under the `[Global]` section, but after the section generated by the `authconfig` tool:

```
kerberos method = secrets and keytab
```

Joining the Windows domain requires that your domain controller is reachable and you have an AD user account with permissions to add computers to the domain:
REALM is the Kerberos realm name in uppercase and user is a domain user who has permissions to add computers to the domain.

Set up SSSD

Setting up SSSD consists of the following steps:

- Install the sssd-ad and sssd-proxy packages on the Linux client machine.
- Make configuration changes to various files (for example, sssd.conf).
- Start the sssd service.

/etc/sssd/sssd.conf

An example sssd.conf configuration (more options can be added as needed):

```plaintext
[sssd]
config_file_version = 2
domains = example
services = nss, pam

[domain/example]
# Uncomment if you need offline logins
# cache_credentials = true
re_expression = (((?P<domain>[^\]\]+)\(\(?P<name>\.[^@]+\)\))|((?P<name>[^@]+)@\(\(?P<domain>\.[^@]+\)\))|^(?P<name>[@^\]\]+))

id_provider = proxy
proxy_lib_name = nis
auth_provider = ad
access_provider = ad

# Should be specified as the lower-case version of the long version of the Active Directory domain.
ad_domain = ad.example.com

# Kerberos settings
krb5_ccachedir = /tmp
krb5_ccname_template = FILE:%d/krb5cc_%U

# Uncomment if service discovery is not working
# ad_server = server.ad.example.com
```

© 1999-2019 Citrix Systems, Inc. All rights reserved.
Replace **ad.domain.com**, **server.ad.example.com** with the corresponding value. For more details, see the **sssd-ad(5)** - Linux man page.

Set the file ownership and permissions on **sssd.conf**:

```bash
chown root:root /etc/sssd/sssd.conf
chmod 0600 /etc/sssd/sssd.conf
restorecon /etc/sssd/sssd.conf
```

### Configure NSS/PAM

**RHEL/CentOS:**

Use **authconfig** to enable SSSD. Install **oddjob-mkhomedir** to ensure that the home directory creation is compatible with SELinux:

```bash
authconfig --enablesssd --enablesssdauth --enablemkhomedir --update
sudo systemctl start sssd
sudo systemctl enable sssd
```

**Tip:**

When configuring Linux VDA settings, consider that for SSSD, there has no special settings for the Linux VDA client. For additional solutions in the **ctxsetup.sh** script, use the default value.

### Verify the Kerberos configuration

To ensure that Kerberos is configured correctly for use with the Linux VDA, check that the system **keytab** file has been created and contains valid keys:

```
sudo klist -ke
```
This command displays the list of keys available for the various combinations of principal names and cipher suites. Run the Kerberos kinit command to authenticate the machine with the domain controller using these keys:

```
sudo kinit -k MACHINE\$@REALM
```

The machine and realm names must be specified in uppercase. The dollar sign ($) must be escaped with a backslash (\) to prevent shell substitution. In some environments, the DNS domain name is different from the Kerberos realm name. Ensure that the realm name is used. If this command is successful, no output is displayed.

Verify that the TGT ticket for the machine account has been cached using:

```
sudo klist -ke
```

**Verify user authentication**

Use the `getent` command to verify that the logon format is supported and whether the NSS works:

```
sudo getent passwd DOMAIN\username
```

The `DOMAIN` parameter indicates the short version domain name. If another logon format is needed, verify by using the `getent` command first.

The supported logon formats are:

- Down-level logon name: `DOMAIN\username`
- UPN: `username@domain.com`
- NetBIOS Suffix format: `username@DOMAIN`

To verify that the SSSD PAM module is configured correctly, use a domain user account to log on to the Linux VDA. The domain user account has not been used before.

```
sudo localhost -l DOMAIN\username
```

```
id -u
```

Check that a corresponding Kerberos credential cache file was created for the `uid` returned by the command:

```
ls /tmp/krb5cc_{
```

```
  uid }
```

Check that the tickets in the user’s Kerberos credential cache are valid and not expired:
Publish applications

November 9, 2018

With Linux VDA Version 7.13, Citrix added the seamless applications feature to all the supported Linux platforms. No specific installation procedures are required to use this feature.

Tip:
With Linux VDA version 1.4, Citrix added support for non-seamless published applications and session sharing.

Publish applications using Citrix Studio

You can publish applications installed on a Linux VDA when you create a delivery group or add applications to an existing delivery group. The process is similar to publishing applications installed on a Windows VDA. For more information, see the Citrix Virtual Apps and Desktops documentation (based on the version of Citrix Virtual Apps and Desktops being used).

Tip:
When configuring delivery groups, ensure that the delivery type is set to Desktop and applications or Applications.

Important:
PUBLISHING APPLICATIONS IS SUPPORTED WITH LINUX VDA VERSION 1.4 AND LATER. HOWEVER, THE LINUX VDA DOES NOT SUPPORT THE DELIVERY OF DESKTOPS AND APPS TO THE SAME MACHINE. TO ADDRESS THIS ISSUE, CITRIX RECOMMENDS THAT YOU CREATE SEPARATE DELIVERY GROUPS FOR APP AND DESKTOP DELIVERIES.

Note:
To use seamless applications, do not disable the seamless mode on StoreFront. The seamless mode is enabled by default. If you have already disabled it by setting “TWIMode=Off,” remove this setting instead of changing it to “TWIMode=On.” Otherwise you might not be able to launch a published desktop.
Troubleshooting

You might encounter that launching a published application takes more than two minutes and windows cannot show in seamless mode. If the issue occurs, verify that the seamless mode has been enabled on both the Linux VDA and StoreFront.

The command to check whether the seamless mode is enabled on the Linux VDA:

```
1 sudo /opt/Citrix/VDA/bin/ctxreg list -k "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Citrix" | grep "SeamlessEnabled"
```

If it shows “SeamlessEnabled = 0x00000000,” the seamless mode is disabled. To enable it, run the following command:

```
1 sudo /opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Citrix" -v "SeamlessEnabled" -d "0x00000001"
```

Known issues

The following known issues are identified during publishing applications:

- Non-seamless published applications fail to launch when the seamless mode is disabled on StoreFront but still enabled on the Linux VDA. Enable or disable the seamless mode on both the Linux VDA and StoreFront at the same time.
- Non-rectangular windows are not supported. The corners of a window might show the server-side background.
- Preview of the content of a window from a published application is not supported.
- Currently, the seamless mode supports the following Window Managers: Mutter, Metacity, and Compiz (Ubuntu 16.04). Kwin and other window managers are not supported. Ensure that your window manager is set a supported one.
- When you run multiple LibreOffice applications, only the one launched first shows on Citrix Studio because these applications share the process.
- Published Qt5-based applications like “Dolphin” might not show icons. To resolve the issue, see the article at https://wiki.archlinux.org/index.php/Qt#Configuration_of_Qt5_apps_under_environments_other_than_KDE_Plasma.
- All the taskbar buttons of published applications running in the same ICA session are combined in the same group. To resolve this issue, set the taskbar property not to combine taskbar buttons.
Print

May 6, 2019

This article provides information about printing best practices.

Installation

The Linux VDA requires both cups and foomatic filters. Execute the following commands based on your Linux distribution:

**RHEL 7 printing support:**

```
1  sudo yum -y install cups
2
3  sudo yum -y install foomatic-filters
```

**RHEL 6 printing support:**

```
1  sudo yum -y install cups
2
3  sudo yum -y install foomatic
```

Usage

You can print from both published desktops and published applications. Only the client-side default printer is mapped into a Linux VDA session. The printer name must be different for desktops and applications. Consider the following:

- For published desktops:
  CitrixUniversalPrinter:$CLIENT_NAME:dsk$SESSION_ID

- For published applications:
  CitrixUniversalPrinter:$CLIENT_NAME:app$SESSION_ID

Note:

If the same user opens both a published desktop and a published application, both printers are available to the session. Printing to a desktop printer in a published application session, or printing to an application printer in a published desktop fails.
Troubleshooting

Unable to print

There are various items to check when printing is not working correctly. The print daemon is a per-
session process and must be running for the length of the session. Verify that the printing daemon is
running.

```
ps -ef | grep ctxlpmngt
```

If the `ctxlpmngt` process is not running, manually start `ctxlpmngt` from a command line. If printing
is still not working, check CUPS framework. The `ctxcups` service is for printer management and com-
municates with the Linux CUPS framework. It is a single process per machine and can be checked by:

```
service ctxcups status
```

Extra log when printing CUPS

As one of the components of the Linux VDA, the method of how to get the log of a printing component
is similar to other components.

For RHEL, some extra steps are necessary to configure the CUPS service file. Otherwise, some logs
cannot get logged in `hdx.log`:

```
sudo service cups stop
3
sudo vi /etc/systemd/system/printer.target.wants/cups.service
4
PrivateTmp=false
6
sudo service cups start
8
sudo systemctl daemon-reload
```

Note:

This configuration is only for collecting the full printing log when an issue arises. Normally this
configuration is not recommended because it breaks CUPS security.
Print output is garbled

An incompatible printer driver can cause garbled output. A per-user driver configuration is available and can be configured by editing the ~/.CtxlpProfile$CLIENT_NAME configuration file:

```ini
[DEFAULT_PRINTER]
printernamex
model=
ppdpath=
drivertype=
```

Important:

The `printernamex` is a field containing the name of the current client-side default printer. It is a read-only value. Do not edit it.

The fields `ppdpath`, `model`, and `drivertype` cannot be set at the same time because only one takes effect for the mapped printer.

If the Universal Printer driver is not compatible with the client printer, configure the model of the native printer driver with the `model=` option. You can find the current model name of the printer by using the `lpinfo` command:

```
lpinfo -m
```

You can then set the model to match the printer:

```
Model=xerox/ph3115.ppd.gz
```

If the Universal Printer driver is not compatible with the client printer, configure the ppd file path of the native printer driver. The value of `ppdpath` is the absolute path of the native printer driver file.

For example, there is a `ppd driver` under /home/tester/NATIVE_PRINTER_DRIVER.ppd:
There are three types of Universal Printer Driver supplied by Citrix (postscript, pcl5, and pcl6). You can configure the driver type if no native printer driver is available.

For example, if the client default printer driver type is PCL5:

```
1 drivertype=pcl5
```

**Output size is zero**

Try different types of printers. And try a virtual printer like CutePDF and PDFCreator to find out whether this issue is related to the printer driver.

The print job depends on the printer driver of the client default printer. It's important to identify the type of the current active driver type. If the client printer is using a PCL5 driver but the Linux VDA chooses a Postscript driver, an issue can occur.

If the printer driver type is correct, you can identify the problem by performing the following steps:

1. Log on to a published desktop session.
2. Run the `vi ~/.CtxtlpProfile$CLIENT_NAME` command.
3. Add the following field to save the spool file on the Linux VDA:

```
1 deletespoolfile=no
```
4. Log off and back on to load the configuration changes.
5. Print the document to reproduce the issue. After printing, a spool file is saved under `/var/spool/cups-ctx/$logon_user/$spool_file`.
6. Verify whether the spool is empty. If the spool file is zero, it represents an issue. Contact Citrix Support (and provide the printing log) for more guidance.
7. If the spool size is not zero, copy the file to the client. The spool file content depends on the printer driver type of the client default printer. If the mapped printer (native) driver is postscript, the spool file can be opened in the Linux OS directly. Verify whether the content is correct.

   If the spool file is PCL, or if the client OS is Windows, copy the spool file to the client and print it by using the client-side printer. After completing this step, test it by using the other printer driver.
8. To change the mapped printer to another third-party printer driver, use the postscript client printer as an example:
   
   a) Log on to an active session and open a browser on the client desktop.
b) Open the printing management portal:

```
localhost:631
```

c) Choose the mapped printer **CitrixUniversalPrinter:$ClientName:app/dsk$SESSION_ID** and **Modify Printer**. This operation requires administrator privileges.

d) Retain the cups-ctx connection, then click Continue to change the printer driver.

e) In the Make and Model page, choose some other postscript driver instead of the Citrix UPD driver (for instance, Citrix Universal Driver Postscript). For example, if the CUPS-PDF virtual printer is installed, select the Generic CUPS-PDF Printer. Save the modification.

f) If this process succeeds, configure the ppd file path of the driver in **.CtxIpProfile$CLIENT_NAME** to allow the mapped printer to use this third-party driver.

**Known issues**

The following issues have been identified during printing on the Linux VDA:

**CTXPS driver is not compatible with some PLC printers**

If you encounter printing output corruption, set the printer driver to the native one provided by the manufacturer.

**Slow printing performance for large documents**

When you print a large document on a local client printer, the document is transferred over the server connection. On slow connections, the transfer can take a long time.

**Printer and print job notifications seen from other sessions**

Linux does not have the same session concept as the Windows operating system. Therefore, all users get system wide notifications. You can disable these notifications by changing the CUPS configuration file: **/etc/cups/cupsd.conf**.

Locate the current policy name configured in the file:

```
DefaultPolicy default
```

If the policy name is **default**, add the following lines to the default policy XML block:
File transfer

February 1, 2019

File transfer is supported between the Linux VDA and the client device. This feature is available when the client device runs a web browser that supports the HTML5 sandbox attribute. The HTML5 sandbox attribute allows users to access virtual desktops or web browser apps using Citrix Workspace app for HTML5 or Citrix Workspace app for Chrome. Within the published desktop or the web browser app sessions, you can use the toolbar of Citrix Workspace app to upload and download files between the Linux VDA and the client device. For example, you can click the **Upload** icon on the toolbar, choose a file on the client device, and upload the file to the Linux VDA.
Note:
This feature is available for RedHat 7.6, CentOS 7.6, SUSE 12.3, Ubuntu 16.04, and Ubuntu 18.04. To use this feature, ensure that the toolbar of Citrix Workspace app is enabled.

File transfer policies

You can use Citrix Studio to set the file transfer policies. By default, file transfer is enabled.

Policy descriptions:

- **Allow file transfer between desktop and client.** Allows or prevents users from transferring files between a Citrix Virtual Apps and Desktops session and their devices.

- **Download file from desktop.** Allows or prevents users from downloading files from a Citrix Virtual Apps and Desktops session to their device.

- **Upload file to desktop.** Allows or prevents users from uploading files from their device to a Citrix Virtual Apps and Desktops session.
Note:
To ensure that the Download file from desktop and Upload file to desktop policies take effect, set the Allow file transfer between desktop and client policy to Allowed.

Usage
To use the file transfer feature through Citrix Workspace app for HTML5:

1. In Citrix Studio, set the WebSockets connections policy to Allowed.

![Citrix Studio](image)

2. In Citrix Studio, enable file transfer through the file transfer policies described above.

3. In the Citrix StoreFront management console, click Stores, select the Manage Receiver for Web Sites node, and enable Citrix Receiver for HTML5 by selecting the Always use Receiver for HTML5 option.
4. Launch a virtual desktop or web browser app session. Upload and download files between the Linux VDA and your client device.

To use the file transfer feature through Citrix Workspace app for Chrome:

1. Enable file transfer through the file transfer policies described above.
2. Obtain Citrix Workspace app from the Chrome Web Store.
   - Skip this step if you already added Citrix Workspace app for Chrome to the Chrome Apps page.
     a) Type **Citrix Workspace for Chrome** in the search box of Google Chrome. Click the search icon.
     b) Among the search results, click the URL to the Chrome Web Store where Citrix Workspace app is available.
c) Click Add to Chrome to add Citrix Workspace app to Google Chrome.

3. Click Citrix Workspace app for Chrome on the Chrome Apps page.

4. Type the URL of your StoreFront store to connect.
   
   Skip this step if you typed the URL before.
5. Launch a virtual desktop or web browser app session. Upload and download files between the Linux VDA and your client device.

PDF printing

November 13, 2018

Using a version of Citrix Workspace app that supports PDF printing, you can print PDFs converted from within the Linux VDA sessions. Session print jobs are sent to the local machine where Citrix Workspace app is installed. On the local machine, you can open PDFs using your PDF viewer of choice and print them on your printer of choice.

The Linux VDA supports PDF printing on the following versions of Citrix Workspace app:

- Citrix Receiver for HTML5 Versions 2.4 through 2.6.9, Citrix Workspace app 1808 for HTML5 and later
- Citrix Receiver for Chrome Versions 2.4 through 2.6.9, Citrix Workspace app 1808 for Chrome and later
**Configuration**

Apart from using a version of Citrix Workspace app that supports PDF printing, enable the following policies in Citrix Studio:

- **Client Printer Redirection** (enabled by default)
- **Auto-create PDF Universal Printer** (disabled by default)

With these policies enabled, a print preview appears on the local machine for you to select a printer when you click Print within your launched session. See the Citrix Workspace app documentation for information about setting default printers.

**Configure graphics**

March 25, 2019

This article provides guidance for the Linux VDA graphics configuration and fine-tuning.

For more information, see System requirements and the Installation overview section.

**Configuration**

Thinwire is the display remoting technology used in the Linux VDA. The technology allows graphics generated on one machine to be transmitted, typically across a network, to another machine for display.

The **Use video codec for compression** graphics policy sets the default graphics mode and provides the following options for different use cases:

- **Use when preferred**. This setting is the default. No additional configuration is required. Keeping this setting ensures that Thinwire is selected for all Citrix connections, and optimized for scalability, bandwidth, and superior image quality for typical desktop workloads.

- **For the entire screen**. Delivers Thinwire with full-screen H.264 or H.265 to optimize for improved user experience and bandwidth, especially in cases with heavy use of 3D graphics.

- **For actively changing regions**. The adaptive display technology in Thinwire identifies moving images (video, 3D in motion), and uses H.264 only in the part of the screen where the image is moving. The **selective use of the H.264 video codec** enables HDX Thinwire to detect and encode parts of the screen that are frequently updated using the H.264 video codec, for example, video content. Still image compression (JPEG, RLE) and bitmap caching continue to be used for the rest of the screen, including text and photographic imagery. Users get the benefit of lower bandwidth and better quality for video content combined with lossless text or high quality imagery.
elsewhere. To enable this feature, change the policy setting **Use video codec for compression** to **Use when preferred** (default) or **For actively changing regions**. For more information, see Graphics policy settings.

Some other policy settings, including the following visual display policy settings can be used to fine-tune the performance of display remoting:

- **Preferred color depth for simple graphics**
- **Target frame rate**
• **Visual quality**

**Use H.264 for Build to Lossless in Thinwire**

By default, the **Build to Lossless** preference of the **Visual quality** policy setting is now H.264 instead of JPEG for moving images.

H.264 encoding offers superior image quality. The **Use video codec for compression** policy controls that preference, with the default being **Use when preferred**. To force **Build to Lossless** to use JPEG, set the **Use video codec for compression** policy to **Do not use video codec**. If your client does not support Selective H.264, **Build to Lossless** falls back to JPEG regardless of the policy settings. Citrix Receiver for Windows 4.9 through 4.12, Citrix Receiver for Linux 13.5 through 13.10, Citrix Workspace app 1808 for Windows and later, and Citrix Workspace app 1808 for Linux and later support Selective H.264. For more information about the **Visual quality** and **Use video codec for compression** policy settings, see [Visual display policy settings](#) and [Graphics policy settings](#).

**Support for H.265 video codec**

Starting with the 7.18 release, the Linux VDA supports the H.265 video codec for hardware acceleration of remote graphics and videos. You can use this feature on Citrix Receiver for Windows 4.10 through 4.12 and on Citrix Workspace app 1808 for Windows and later. To benefit from this feature, enable it on both the Linux VDA and on your client. If the GPU of your client does not support H.265 decoding using the DXVA interface, the H.265 Decoding for graphics policy setting is ignored and the session falls back to using the H.264 video codec. For more information, see [H.265 video encoding](#).

To enable H.265 hardware encoding on the VDA:

1. Enable the **Use hardware encoding for video codec** policy.
2. Enable the **Optimize for 3D graphics workload** policy
3. Ensure that the **Use video codec for compression** policy is default or set to **For the entire screen**.
4. Ensure that the **Visual quality** policy is **NOT** set to **Build to Lossless** or **Always Lossless**.

To enable H.265 hardware encoding on your client, see [H.265 video encoding](#).

**Support for YUV444 software encoding**

The Linux VDA supports YUV444 software encoding. The YUV encoding scheme assigns both brightness and color values to each pixel. In YUV, ‘Y’ represents the brightness, or ‘luma’ value, and ‘UV’ represents the color, or ‘chroma’ values. You can use this feature of the Linux VDA on Citrix Receiver for Windows 4.10 through 4.12 and on Citrix Workspace app 1808 for Windows and later.
Each unique Y, U, and V value comprises 8 bits, or one byte, of data. The YUV444 data format transmits 24 bits per pixel. The YUV422 data format shares U and V values between two pixels, which results in an average transmission rate of 16 bits per pixel. The following table shows an intuitive comparison between YUV444 and YUV420.

<table>
<thead>
<tr>
<th>YUV444</th>
<th>YUV420</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>Citrix</td>
<td>Citrix</td>
</tr>
<tr>
<td>Citrix</td>
<td>Citrix</td>
</tr>
<tr>
<td>Citrix</td>
<td>Citrix</td>
</tr>
<tr>
<td>Citrix</td>
<td>Citrix</td>
</tr>
<tr>
<td>Citrix</td>
<td>Citrix</td>
</tr>
<tr>
<td>Citrix</td>
<td>Citrix</td>
</tr>
</tbody>
</table>

To enable YUV444 software encoding on the VDA:

1. Enable the **Allow visually lossless compression** policy.
2. Ensure that the **Use video codec for compression** policy is set to **For the entire screen**.
3. Ensure that the **Visual quality** policy is set to **Always Lossless** or **Build to Lossless**.

**Adjust average bit rates based on bandwidth estimates**

Citrix enhances HDX 3D Pro hardware encoding by adjusting average bit rates based on bandwidth estimates.

When the HDX 3D Pro hardware encoding is in use, the VDA can intermittently estimate the bandwidth of the network and adjust the bit rates of encoded frames based on the bandwidth estimates. This new feature provides a mechanism to balance between sharpness and fluency.

This feature is enabled by default. To disable it, run the following command:

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "DisableReconfigureEncoder" -d "0x00000001" --force
```

In addition to using this feature, you can also run the following commands to adjust between sharpness and fluency. The **AverageBitRatePercent** and **MaxBitRatePercent** parameters set the percentage of bandwidth usage. The higher values you set, the sharper graphics and lower fluency you get. The recommended setting range is 50 to 100.

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "AverageBitRatePercent" -d "90" --force
```

© 1999-2019 Citrix Systems, Inc. All rights reserved. 143
Ted the average bit rate adjustment, when your screen holds still, the most recent frame stays in a low-quality state because no new frames are sent. Sharpening support can address this issue by reconfiguring and immediately sending the most recent frame at the highest quality.

For a full list of the policies supported by the Linux VDA Thinwire, see Policy support list.

For information on the configuration of multi-monitor support on the Linux VDA, see CTX220128.

Troubleshooting

Check which graphics mode is in use

Run the following command to check which graphics mode is in use (0 means TW+; 1 means full-screen video codec):

```bash
sudo /opt/Citrix/VDA/bin/ctxreg dump | grep GraphicsMode
```

The result resembles:

```plaintext
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"-v "GraphicsMode"-d "0x00000000"--force
```

Check whether H.264 is in use

Run the following command to check whether H.264 is in use (0 means not in use; 1 means in use):

```bash
sudo /opt/Citrix/VDA/bin/ctxreg dump | grep H264
```

The result resembles:

```plaintext
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"-v "H264"-d "0x00000000"--force
```

Check whether H.265 is in use

Run the following command to check whether full-screen H.265 is in use (0 means not in use; 1 means in use):
The result resembles:

```
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"-v "H265"-d "0x00000000"--force
```

**Check which YUV encoding scheme is in use**

Run the following command to check which YUV encoding scheme is in use (0 means YUV420; 1 means YUV422; 2 means YUV444):

**Note:** The value of YUVFormat is meaningful only when a video codec is in use.

```
sudo /opt/Citrix/VDA/bin/ctxreg dump | grep YUVFormat
```

The result resembles:

```
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"-v "YUVFormat"-d "0x00000000"--force
```

**Check whether YUV444 software encoding is in use**

Run the following command to check whether YUV444 software encoding is in use:

```
sudo /opt/Citrix/VDA/bin/ctxreg dump | grep Graphics
```

When YUV444 is in use, the result resembles:

```
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"-v "GraphicsMode"-d "0x00000001"--force
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"-v "H264"-d "0x00000001"--force
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"-v "HardwareEncoding"-d "0x00000000"--force
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"-v "YUVFormat"-d "0x00000002"--force
```

**Check whether hardware encoding is in use for 3D Pro**

Run the following command (0 means not in use; 1 means in use):
sudo /opt/Citrix/VDA/bin/ctxreg dump | grep HardwareEncoding

The results resemble:

create -k "HKLM\Software\Citrix\Ica\Session\1\Graphics"-t "REG_DWORD"-v "HardwareEncoding"-d "0x00000001"--force

Another way is to use the nvidia-smi command. The outputs resemble the following if hardware encoding is in use:
Verify that the NVIDIA GRID graphics driver is installed correctly

To verify that the NVIDIA GRID graphics driver is installed correctly, run `nvidia-smi`. The results resemble:

```plaintext
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 352.70  Driver Version: 352.70                                  |
+-----------------------------------------------------------------------------+
| GPU  Name     Persistence-M | Bus-Id   Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf Pwr:Usage/Cap | Memory-Usage | GPU-Util Compute M. |
+-----------------------------------------------------------------------------+
| 0 Tesla M60     Off | 0000:00:05.0 Off | 20C      Off |
| N/A   20C     P0  37W / 150W | 19MiB / 8191MiB | 0% |
+-----------------------------------------------------------------------------+
| Processes: | GPU Memory |
| GPU  PID Type Process name Usage | |
| No running processes found | |
+-----------------------------------------------------------------------------+

Set the correct configuration for the card:

`etc/X11/ctx-nvidia.sh`

HDX 3D Pro multi-monitor redraw issues

If you are seeing redraw issues on screens other than the primary monitor, check that the NVIDIA GRID license is available.
Check Xorg error logs

The log file of Xorg is named similar to `X org . { DISPLAY } . log` in the `/var/log/` folder.

Known issues and limitations

For vGPU, the XenServer local console shows the ICA desktop session screen

Workaround: Disable the VM’s local VGA console by running the following command:

```
xe vm-param-set uuid=<vm-uuid> platform:vgpu_extra_args="disable_vnc=1"
```

NVIDIA K2 graphics cards do not support YUV444 hardware encoding in pass-through mode

With Build to Lossless enabled through the policy setting, a black or gray screen appears when users are launching an app/desktop session with an NVIDIA K2 graphics card. The issue occurs because NVIDIA K2 graphics cards do not support YUV444 hardware encoding in pass-through mode. For more information, see Video Encode and Decode GPU Support Matrix.

Gnome 3 desktop popups slow when logging on

It is a limitation of Gnome 3 desktop session startup.

Some OpenGL/WebGL applications do not render well upon resizing the Citrix Workspace app window

Resizing the window of Citrix Workspace app changes the screen resolution. The NVIDIA proprietary driver changes some internal states and might require applications to respond accordingly. For example, the WebGL library element `lightgl.js` might spawn an error saying that `Rendering to this texture is not supported (incomplete frame buffer)`.

Thinwire progressive display

December 3, 2018

Session interactivity can degrade on low bandwidth or high latency connections. For example, on connections with less than 2 Mbps bandwidth or latency of more than 200 ms, scrolling on a webpage
Linux Virtual Delivery Agent 1903

can become slow, unresponsive, or choppy; keyboard and mouse operations can lag behind graphics updates.

Through version 7.17, you were able to use policy settings to reduce bandwidth consumption by configuring the session to **Low** visual quality, or setting a lower color depth (16-bit or 8-bit graphics). However, you had to know that a user was on a weak connection. HDX Thinwire could not dynamically adjust static image quality based on network conditions.

Starting with Version 7.18, HDX Thinwire, by default, switches to a progressive update mode when available bandwidth falls below 2 Mbps, or network latency exceeds 200 ms. In this mode:

- All static images are heavily compressed.
- Text quality is reduced.

For example, in the following graphic where progressive update mode is active, the letters **F** and **e** have blue artifacts, and the image is heavily compressed. This approach significantly reduces bandwidth consumption, which allows images and text to be received more quickly, and session interactivity improves.

**Features**

When you stop interacting with the session, the degraded images and text are progressively sharpened to lossless. For example, in the following graphic, the letters no longer contain blue artifacts, and the image appears at source quality.

**Features**

For images, sharpening uses a random block-like method. For text, individual letters or parts of words are sharpened. The sharpening process occurs over several frames. This approach avoids introducing a delay with a single large sharpening frame.
Transient imagery (video) is still managed with adaptive display or Selective H.264.

**How progressive mode is used**

By default, progressive mode is on standby for the **Visual quality** policy settings: **High**, **Medium** (default), and **Low**.

Progressive mode is forced off (not used) when:

- **Visual quality** = **Always Lossless** or **Build to Lossless**
- **Preferred color depth for simple graphics** = 8-bit
- **Use video codec for compression** = **For the entire screen** (when full-screen H.264 is desired)

When progressive mode is on standby, by default it is enabled when either of the following conditions occurs:

- Available bandwidth drops below 2 Mbps
- Network latency increases above 200 ms

After a mode switch occurs, a minimum of 10s is spent in that mode, even if the adverse network conditions are momentary.

**Change progressive mode behavior**

You can change the progressive mode behavior by running the following command:

```bash
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\SOFTWARE\CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "ProgressiveDisplay" -d "<value>" --force
```

where <value>:

- 0 = Always off (do not use under any circumstances)
- 1 = Automatic (toggle based on network conditions, default value)
- 2 = Always on

When in automatic mode (1), you can run either of the following commands to change the thresholds at which progressive mode is toggled:

```bash
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "ProgressiveDisplayBandwidthThreshold" -d "<value>" --force
```

where <value> is <threshold in Kbps> (default = 2,048)

Example: 4096 = toggle progressive mode on if bandwidth falls below 4 Mbps
where value is <threshold in ms> (default = 200)
Example: 100 = toggle progressive mode on if network latency drops below 100 ms.

**Non-GRID 3D graphics**

November 9, 2018

**Overview**

With this feature enhancement, the Linux VDA supports not only NVIDIA GRID 3D cards but also non-GRID 3D cards.

**Installation**

To use the non-GRID 3D graphics feature, you must install XDamage as a prerequisite. Typically, XDamage exists as an extension of XServer.

**Configuration**

**Xorg configuration files**

If your 3D card driver is NVIDIA, the configuration files are installed and set automatically.

**Other types of 3D cards**

If your 3D card driver is NOT NVIDIA, you must modify the four template configuration files installed under /etc/X11/:

- ctx-driver_name-1.conf
- ctx-driver_name-2.conf
- ctx-driver_name-3.conf
- ctx-driver_name-4.conf
Using `ctx-driver_name-1.conf` as an example, do the following to modify the template configuration files:

1. Replace `driver_name` with your actual driver name.

For example, if your driver name is “intel,” you can change the configuration file name to “ctx-intel-1.conf.”

2. Add the video driver information.

Each template configuration file contains a section named “Device,” which is commented out. This section describes the video driver information. Enable this section before adding your video driver information. To enable this section:

   a) See the 3D card guide provided by the manufacturer for configuration information. A native configuration file can be generated. Verify that your 3D card can work in a local environment with the native configuration file when you are not using a Linux VDA ICA session.

   b) Copy the “Device” section of the native configuration file to `ctx-driver_name-1.conf`.

3. Run the following command to set the registry key so that the Linux VDA can recognize the configuration file name set in Step 1.

   ```
   /opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Citrix\XDamage" -t "REG_SZ" -v "DriverName" -d "intel" --force
   ```

**Enable the non-GRID 3D graphics feature**

The non-GRID 3D graphics feature is disabled by default. You can run the following command to enable it by setting XDamageEnabled to 1.

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Citrix\XDamage" -t "REG_DWORD" -v "XDamageEnabled" -d 0x00000001 --force
```

**Troubleshooting**

**No or garbled graphic output**

If you can run 3D applications locally and all configurations are correct, missing or garbled graphic output is the result of a bug. Use `/opt/Citrix/VDA/bin/setlog` and set GFX_X11 to verbose to collect the trace information for debugging.
Hardware encoding does not work

This feature supports only software encoding.

Configure policies

March 25, 2019

Installation

Follow the installation articles to prepare the Linux VDA.

Dependencies

Ensure that you install these dependencies before installing the Linux VDA package.

RHEL/CentOS:

```
1 sudo yum -y install openldap
2 sudo yum -y install libxml2
3 sudo yum -y install cyrus-sasl
4 sudo yum -y install cyrus-sasl-gssapi
```

SLES/SELD:

```
1 sudo zypper install openldap2
2 sudo zypper install libxml2
3 sudo zypper install cyrus-sasl
4 sudo zypper install cyrus-sasl-gssapi
```

Ubuntu:

```
1 sudo apt-get install -y libldap-2.4-2
2 sudo apt-get install -y libsasl2-2
```
Configuration

Policy settings in Citrix Studio

To set policies in Citrix Studio, do the following:

1. Open Citrix Studio.
2. Select the Policies panel.
3. Click Create Policy.
4. Set the policy according to the Policy support list.

LDAP server setting on the VDA

The LDAP server setting on Linux VDA is optional for single domain environments but mandatory for multiple domain and multiple forest environments. This setting is necessary for the policy service to perform an LDAP search in these environments.

After installing the Linux VDA package, run the command:

```
1 /opt/Citrix/VDA/sbin/ctxsetup.sh
```

Type all the LDAP servers in the suggested format: space-separated list of LDAP Fully Qualified Domain Names (FQDNs) with the LDAP port (for example, ad1.mycompany.com:389 ad2.mycomany.com:389).

You can also run the ctxreg command to write this setting to the registry directly:

```
1 /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\VirtualDesktopAgent" -t "REG_SZ" -v "ListOfLDAPServers" -d "ad1.mycompany.com:389 ad2.mycomany.com:389" --force
```

Policy support list

March 25, 2019
## Linux VDA policy support list

<table>
<thead>
<tr>
<th>Studio Policy</th>
<th>Key Name</th>
<th>Type</th>
<th>Module</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use local time of client</td>
<td>UseLocalTimeOfClient</td>
<td>User</td>
<td>ICA\Time Zone Control</td>
<td>Use server time zone</td>
</tr>
<tr>
<td>ICA round trip calculation</td>
<td>IcaRoundTripCheckEnabled</td>
<td>Computer</td>
<td>ICA\End User Monitoring</td>
<td>Enabled (1)</td>
</tr>
<tr>
<td>ICA round trip calculation interval</td>
<td>IcaRoundTripCheckComputer</td>
<td>Computer</td>
<td>ICA\End User Monitoring</td>
<td>15</td>
</tr>
<tr>
<td>ICA round trip calculations for idle connections</td>
<td>IcaRoundTripCheckComputer</td>
<td>Computer</td>
<td>ICA\End User Monitoring</td>
<td>Disabled (0)</td>
</tr>
<tr>
<td>Overall session bandwidth limit</td>
<td>LimitOverallBw</td>
<td>User</td>
<td>ICA\Bandwidth</td>
<td>0</td>
</tr>
<tr>
<td>Audio redirection bandwidth limit</td>
<td>LimitAudioBw</td>
<td>User</td>
<td>ICA\Bandwidth</td>
<td>0</td>
</tr>
<tr>
<td>Audio redirection bandwidth limit percent</td>
<td>LimitAudioBwPerc</td>
<td>User</td>
<td>ICA\Bandwidth</td>
<td>0</td>
</tr>
<tr>
<td>Client USB device redirection bandwidth limit</td>
<td>LimitUSBBw</td>
<td>User</td>
<td>ICA\Bandwidth</td>
<td>0</td>
</tr>
<tr>
<td>Client USB device redirection bandwidth percent</td>
<td>LimitUSBBwPerc</td>
<td>User</td>
<td>ICA\Bandwidth</td>
<td>0</td>
</tr>
<tr>
<td>Clipboard redirection bandwidth limit</td>
<td>LimitClipbdBW</td>
<td>User</td>
<td>ICA\Bandwidth</td>
<td>0</td>
</tr>
<tr>
<td>Studio Policy</td>
<td>Key Name</td>
<td>Type</td>
<td>Module</td>
<td>Default Value</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-----------------------</td>
<td>--------</td>
<td>----------------</td>
<td>------------------</td>
</tr>
<tr>
<td>Clipboard redirection</td>
<td>LimitClipbdBWPer</td>
<td>User</td>
<td>ICA\Bandwidth</td>
<td>0</td>
</tr>
<tr>
<td>bandwidth limit percent</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>File redirection bandwidth</td>
<td>LimitCdmBw</td>
<td>User</td>
<td>ICA\Bandwidth</td>
<td>0</td>
</tr>
<tr>
<td>limit</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>File redirection bandwidth</td>
<td>LimitCdmBwPerce</td>
<td>User</td>
<td>ICA\Bandwidth</td>
<td>0</td>
</tr>
<tr>
<td>limit percent</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Printer redirection</td>
<td>LimitPrinterBw</td>
<td>User</td>
<td>ICA\Bandwidth</td>
<td>0</td>
</tr>
<tr>
<td>bandwidth limit</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Printer redirection</td>
<td>LimitPrinterBwPerce</td>
<td>User</td>
<td>ICA\Bandwidth</td>
<td>0</td>
</tr>
<tr>
<td>bandwidth limit percent</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WebSockets connections</td>
<td>AcceptWebSockets</td>
<td>Computer</td>
<td>ICA\WebSockets</td>
<td>Prohibited</td>
</tr>
<tr>
<td>WebSockets port number</td>
<td>WebSocketsPort</td>
<td>Computer</td>
<td>ICA\WebSockets</td>
<td>8008</td>
</tr>
<tr>
<td>WebSockets trusted origin</td>
<td>WSTrustedOriginServerList</td>
<td>Computer</td>
<td>ICA\WebSockets</td>
<td>*</td>
</tr>
<tr>
<td>server list</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICA keep alives</td>
<td>SendICAKeepAlive</td>
<td>Computer</td>
<td>ICA keep alive</td>
<td>Do not send ICA keep alive messages (0)</td>
</tr>
<tr>
<td>ICA keep alive timeout</td>
<td>ICAKeepAliveTime</td>
<td>Computer</td>
<td>ICA keep alive</td>
<td>60 seconds</td>
</tr>
<tr>
<td>ICA listener port number</td>
<td>IcaListenerPortNum</td>
<td>Computer</td>
<td>ICA</td>
<td>1494</td>
</tr>
<tr>
<td>HDX adaptive transport</td>
<td>HDXoverUDP</td>
<td>Computer</td>
<td>ICA</td>
<td>Preferred(2)</td>
</tr>
<tr>
<td>Session reliability</td>
<td>AcceptSessionReli</td>
<td>Computer</td>
<td>ICA\Session Reliability</td>
<td>Allowed(1)</td>
</tr>
<tr>
<td>connections</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Studio Policy</td>
<td>Key Name</td>
<td>Type</td>
<td>Module</td>
<td>Default Value</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>---------------------------------</td>
<td>-----------------------</td>
<td>-------------------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>Reconnection UI transparency level</td>
<td>ReconnectionUiTransparencyLevel</td>
<td>ICA\Auto Client Reconnect</td>
<td>80%</td>
<td></td>
</tr>
<tr>
<td>Session reliability port number</td>
<td>SessionReliabilityPort</td>
<td>Computer</td>
<td>ICA\Session Reliability</td>
<td>2598</td>
</tr>
<tr>
<td>Session reliability timeout</td>
<td>SessionReliabilityTimeout</td>
<td>Computer</td>
<td>ICA\Session Reliability</td>
<td>180s</td>
</tr>
<tr>
<td>Auto Client Reconnect</td>
<td>AllowAutoClientReconnect</td>
<td>User</td>
<td>ICA\Auto Client Reconnect</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Client audio redirection</td>
<td>AllowAudioRedirect</td>
<td>User</td>
<td>Audio</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Client printer redirection</td>
<td>AllowPrinterRedirect</td>
<td>User</td>
<td>Printing</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Auto-create PDF Universal Printer</td>
<td>AutoCreatePDFPrinter</td>
<td>User</td>
<td>Printing</td>
<td>Disabled (0)</td>
</tr>
<tr>
<td>Client clipboard redirection</td>
<td>AllowClipboardRedirect</td>
<td>User</td>
<td>Clipboard</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Client USB device redirection</td>
<td>AllowUSBRedirect</td>
<td>User</td>
<td>USB</td>
<td>Prohibited (0)</td>
</tr>
<tr>
<td>Client USB device redirection rules</td>
<td>USBDeviceRules</td>
<td>User</td>
<td>USB</td>
<td>&quot;\0&quot;</td>
</tr>
<tr>
<td>Moving image compression</td>
<td>MovingImageCompression</td>
<td>User</td>
<td>ThinWire</td>
<td>Enabled (1)</td>
</tr>
<tr>
<td>Extra color compression</td>
<td>ExtraColorCompression</td>
<td>User</td>
<td>ThinWire</td>
<td>Disabled (0)</td>
</tr>
<tr>
<td>Target minimum frame rate</td>
<td>TargetedMinimumFramesPerSecond</td>
<td>User</td>
<td>ThinWire</td>
<td>10 fps</td>
</tr>
<tr>
<td>Target frame rate</td>
<td>FramesPerSecond</td>
<td>User</td>
<td>ThinWire</td>
<td>30 fps</td>
</tr>
<tr>
<td>Visual quality</td>
<td>VisualQuality</td>
<td>User</td>
<td>ThinWire</td>
<td>Medium (3)</td>
</tr>
<tr>
<td>Studio Policy</td>
<td>Key Name</td>
<td>Type</td>
<td>Module</td>
<td>Default Value</td>
</tr>
<tr>
<td>--------------------------------------------------</td>
<td>------------------------</td>
<td>---------</td>
<td>-------------------------</td>
<td>----------------------------------------------------</td>
</tr>
<tr>
<td>Use video codec for compression</td>
<td>VideoCodec</td>
<td>User</td>
<td>ThinWire</td>
<td>Use when preferred (3)</td>
</tr>
<tr>
<td>Use hardware encoding for video codec</td>
<td>UseHardwareEncodingForVideoCodec</td>
<td>User</td>
<td>ThinWire</td>
<td>Enabled (1)</td>
</tr>
<tr>
<td>Allow visually lossless compression</td>
<td>AllowVisuallyLossless</td>
<td>User</td>
<td>ThinWire</td>
<td>Disabled (0)</td>
</tr>
<tr>
<td>Optimize for 3D graphics workload</td>
<td>OptimizeFor3dWorkload</td>
<td>User</td>
<td>ThinWire</td>
<td>Disabled (0)</td>
</tr>
<tr>
<td>Preferred color depth for simple graphics</td>
<td>PreferredColorDepth</td>
<td>User</td>
<td>ThinWire</td>
<td>24 bits per pixel (1)</td>
</tr>
<tr>
<td>Audio quality</td>
<td>SoundQuality</td>
<td>User</td>
<td>Audio</td>
<td>High – high definition audio (2)</td>
</tr>
<tr>
<td>Client microphone redirection</td>
<td>AllowMicrophoneRedir</td>
<td>User</td>
<td>Audio</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Maximum number of sessions</td>
<td>MaximumNumberOfSessions</td>
<td>Computer</td>
<td>Load Management</td>
<td>250</td>
</tr>
<tr>
<td>Concurrent logons tolerance</td>
<td>ConcurrentLogons</td>
<td>Computer</td>
<td>Load Management</td>
<td>2</td>
</tr>
<tr>
<td>Enable auto update of Controllers</td>
<td>EnableAutoUpdateOfControllers</td>
<td>Computer</td>
<td>Virtual Delivery Agent Settings</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Clipboard selection update mode</td>
<td>ClipboardSelectionUpdateMode</td>
<td>User</td>
<td>Clipboard</td>
<td>3</td>
</tr>
<tr>
<td>Primary selection update mode</td>
<td>PrimarySelectionUpdateMode</td>
<td>User</td>
<td>Clipboard</td>
<td>3</td>
</tr>
<tr>
<td>Studio Policy</td>
<td>Key Name</td>
<td>Type</td>
<td>Module</td>
<td>Default Value</td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>----------------------</td>
<td>--------</td>
<td>------------------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>Max speex quality</td>
<td>MaxSpeexQuality</td>
<td>User</td>
<td>Audio</td>
<td>5</td>
</tr>
<tr>
<td>Auto connect client drives</td>
<td>AutoConnectDrives</td>
<td>User</td>
<td>File redirection/CDM</td>
<td>Enabled (1)</td>
</tr>
<tr>
<td>Client optical drives</td>
<td>AllowCdromDrives</td>
<td>User</td>
<td>File redirection/CDM</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Client fixed drives</td>
<td>AllowFixedDrives</td>
<td>User</td>
<td>File redirection/CDM</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Client floppy drives</td>
<td>AllowFloppyDrives</td>
<td>User</td>
<td>File redirection/CDM</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Client network drives</td>
<td>AllowNetworkDrives</td>
<td>User</td>
<td>File redirection/CDM</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Client removable drives</td>
<td>AllowRemoveableDrives</td>
<td>User</td>
<td>File redirection/CDM</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Client drive redirection</td>
<td>AllowDriveRedir</td>
<td>User</td>
<td>File redirection/CDM</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Read-only client drive access</td>
<td>ReadOnlyMappedDrive</td>
<td>User</td>
<td>File redirection/CDM</td>
<td>Disabled (0)</td>
</tr>
<tr>
<td>Automatic keyboard display</td>
<td>AllowAutoKeyboardPopUp</td>
<td></td>
<td>MRVC</td>
<td>Disabled (0)</td>
</tr>
<tr>
<td>Allow file transfer between desktop and client</td>
<td>AllowFileTransferUser</td>
<td>User</td>
<td>File Transfer</td>
<td>Allowed</td>
</tr>
<tr>
<td>Download file from desktop</td>
<td>AllowFileDownloadUser</td>
<td>User</td>
<td>File Transfer</td>
<td>Allowed</td>
</tr>
<tr>
<td>Upload file to desktop</td>
<td>AllowFileUpload</td>
<td>User</td>
<td>File Transfer</td>
<td>Allowed</td>
</tr>
</tbody>
</table>

The following policies can be configured in Citrix Studio Version 7.12 and later.

- MaxSpeexQuality
  
  **Value (integer):** [0 - 10]
**Default value:** 5

**Details:**

Audio redirection encodes audio data with the Speex codec when audio quality is medium or low (see the policy Audio quality). Speex is a lossy codec, which means that it achieves compression at the expense of fidelity of the input speech signal. Unlike some other speech codecs, it is possible to control the tradeoff made between quality and bit rate. The Speex encoding process is controlled most of the time by a quality parameter that ranges from 0 to 10. The higher the quality is, the higher the bit rate.

The max Speex quality chooses the best Speex quality to encode audio data according to audio quality and bandwidth limit (see the policy Audio redirection bandwidth limit). If the audio quality is medium, the encoder is in wide band mode, which means a higher sampling rate. If the audio quality is low, the encoder is in narrow band mode, which means a lower sampling rate. The same Speex quality has different bit rates in different modes. The best Speex quality is when the largest value meets the following conditions:

- It is equal to or less than the max Speex quality.
- Its bit rate is equal to or less than the bandwidth limit.

**Related Settings:** Audio quality, Audio redirection bandwidth limit

- PrimarySelectionUpdateMode

  **Value (enum):** [0, 1, 2, 3]

  **Default value:** 3

  **Details:**

  Primary selection is used when you select data and paste it by pressing the middle mouse button.

  This policy controls whether primary selection changes on the Linux VDA and client can update the clipboard on each other. There are four value options:
Primary selection update mode

- **Selection changes are not updated on neither client nor host**
  Primary selection changes on the Linux VDA do not update the clipboard on the client. Primary selection changes on the client do not update the clipboard on the Linux VDA.

- **Host selection changes are not updated to client**
  Primary selection changes on the Linux VDA do not update the clipboard on the client. Primary selection changes on the client update the clipboard on the Linux VDA.

- **Client selection changes are not updated to host**
  Primary selection changes on the Linux VDA update the clipboard on the client. Primary selection changes on the client do not update the clipboard on the Linux VDA.

- **Selection changes are updated on both client and host**
  Primary selection changes on the Linux VDA update the clipboard on the client. Primary selection changes on the client update the clipboard on the Linux VDA. This option is the
Related Setting: Clipboard selection update mode

- ClipboardSelectionUpdateMode

Value (enum): [0, 1, 2, 3]

Default value: 3

Details:

Clipboard selection is used when you select some data and explicitly request it to be “copied” to the clipboard, such as by selecting “Copy” from the shortcut menu. Clipboard selection is primarily used in connection with Microsoft Windows clipboard operations while primary selection is unique to Linux.

This policy controls whether clipboard selection changes on the Linux VDA and client can update the clipboard on each other. There are four value options:

- Selection changes are not updated on neither client nor host

  Clipboard selection changes on the Linux VDA do not update the clipboard on the client. Clipboard selection changes on the client do not update the clipboard on the Linux VDA.
- **Host selection changes are not updated to client**
  Clipboard selection changes on the Linux VDA do not update the clipboard on the client. Clipboard selection changes on the client update the clipboard on the Linux VDA.

- **Client selection changes are not updated to host**
  Clipboard selection changes on the Linux VDA update the clipboard on the client. Clipboard selection changes on the client do not update the clipboard on the Linux VDA.

- **Selection changes are updated on both client and host**
  Clipboard selection changes on the Linux VDA update the clipboard on the client. Clipboard selection changes on the client update the clipboard on the Linux VDA. This option is the default value.

**Related Setting**: Primary selection update mode

**Note**: The Linux VDA supports both clipboard selection and primary selection. To control the copy and paste behaviors between the Linux VDA and the client, Citrix recommends that you set both clipboard selection update mode and primary selection update mode to the same value.

---

### Configure IPv6

November 9, 2018

The Linux VDA supports IPv6 to align with Citrix Virtual Apps and Desktops. When using this feature, consider the following:

- For dual stack environments, IPv4 is used unless IPv6 is explicitly enabled.
- If IPv6 is enabled in an IPv4 environment, the Linux VDA fails to function.

**Important**:  
- The whole network environment must be IPv6, not only for the Linux VDA.
- Centrify does not support pure IPv6.

No special setup tasks are required for IPv6 when you install the Linux VDA.

### Configure IPv6 for the Linux VDA

Before changing the configuration for the Linux VDA, ensure that your Linux virtual machine has previously worked in an IPv6 network. There are two registry keys related to IPv6 configuration:
OnlyUseIPv6ControllerRegistration must be set to 1 to enable IPv6 on the Linux VDA:

```
1 sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Policies\Citrix\VirtualDesktopAgent" -t "REG_DWORD" -v "OnlyUseIPv6ControllerRegistration" -d "0x00000001" --force
```

If the Linux VDA has more than one network interface, ControllerRegistrationIPv6Netmask can be used to specify which one is used for the Linux VDA registration:

```
1 sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Policies\Citrix\VirtualDesktopAgent" -t "REG_SZ" -v "ControllerRegistrationIPv6Netmask" -d "{{IPv6 netmask}}"
2 "IPv6 netmask "
3 "--force
```

Replace {{IPv6 netmask}} with the real netmask (for example, 2000::/64).

For more information about IPv6 deployment in Citrix Virtual Apps and Desktops, see IPv4/IPv6 support.

**Troubleshooting**

Check the basic IPv6 network environment and use ping6 to check whether AD and Delivery Controller are reachable.

**Configure CEIP**

November 15, 2018

When you participate in the Citrix Customer Experience Improvement Program (CEIP), anonymous statistics and usage information are sent to Citrix to help improve the quality and performance of Citrix products. In addition, a copy of the anonymous data is sent to Google Analytics (GA) for fast and efficient analysis.
Registry settings

By default, you automatically participate in the CEIP when you install the Linux VDA. The first upload of data occurs approximately seven days after you install the Linux VDA. You can change this default setting in the registry.

- **CEIPSwitch**

  Registry setting that enables or disables the CEIP (default = 0):
  
  Location: HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\CEIP
  
  Name: CEIPSwitch
  
  Value: 1 = disabled, 0 = enabled
  
  When unspecified, the CEIP is enabled.

  You can run the following command on a client to disable the CEIP:

  ```bash
  /opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\CEIP" -v "CEIPSwitch" -d "1"
  ```

- **GASwitch**

  Registry setting that enables or disables GA (default = 0):
  
  Location: HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\CEIP
  
  Name: GASwitch
  
  Value: 1 = disabled, 0 = enabled
  
  When unspecified, GA is enabled.

  You can run the following command on a client to disable GA:

  ```bash
  /opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\CEIP" -v "GASwitch" -d "1"
  ```

- **DataPersistPath**

  Registry setting that controls the data persisting path (default = /var/xdl/ceip):
  
  Location: HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\CEIP
  
  Name: DataPersistPath
  
  Value: String
  
  You can run the following command to set this path:

  ```bash
  /opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\CEIP" -v "DataPersistPath" -d "your_path"
  ```
If the path you configured does not exist or cannot be accessed, data is saved in the default path.

**CEIP data collected from the Linux VDA**

The following table gives an example of the types of anonymous information collected. The data does not contain any details that identify you as a customer.

<table>
<thead>
<tr>
<th>Data Point</th>
<th>Key Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Machine GUID</td>
<td>machine_guid</td>
<td>Identifying the machine where the data originates</td>
</tr>
<tr>
<td>AD solution</td>
<td>ad_solution</td>
<td>Text string denoting the machine’s domain joining method</td>
</tr>
<tr>
<td>Linux kernel version</td>
<td>kernel_version</td>
<td>Text string denoting the machine’s kernel version</td>
</tr>
<tr>
<td>LVDA version</td>
<td>vda_version</td>
<td>Text string denoting the installed version of the Linux VDA.</td>
</tr>
<tr>
<td>LVDA update or fresh install</td>
<td>update_or_fresh_install</td>
<td>Text string denoting the current Linux VDA package is being freshly installed or updated</td>
</tr>
<tr>
<td>LVDA installed method</td>
<td>install_method</td>
<td>Text string denoting that the current Linux VDA package is installed by using MCS, PVS, easy install, or manual installation.</td>
</tr>
<tr>
<td>HDX 3D pro enabled or not</td>
<td>hdx_3d_pro</td>
<td>Text string denoting whether HDX 3D Pro is enabled on the machine</td>
</tr>
<tr>
<td>VDI mode enabled or not</td>
<td>vdi_mode</td>
<td>Text string denoting whether VDI mode is enabled</td>
</tr>
<tr>
<td>System Locale</td>
<td>system_locale</td>
<td>Text string denoting the locale of this machine</td>
</tr>
<tr>
<td>Data Point</td>
<td>Key Name</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>-------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>LVDA key services last restart time</td>
<td>ctxhdctxvda</td>
<td>The last restart time of the ctxhd and ctxvda services, in the format of dd-hh:mm:ss, for example, 10-17:22:19</td>
</tr>
<tr>
<td>GPU type</td>
<td>gpu_type</td>
<td>Denoting the GPU type of the machine</td>
</tr>
<tr>
<td>CPU cores</td>
<td>cpu_cores</td>
<td>Integer denoting the number of CPU cores of the machine</td>
</tr>
<tr>
<td>CPU frequency</td>
<td>cpu_frequency</td>
<td>Float denoting the CPU frequency in MHz</td>
</tr>
<tr>
<td>Physical memory size</td>
<td>memory_size</td>
<td>Integer denoting the physical memory size in KB</td>
</tr>
<tr>
<td>Launched session number</td>
<td>session_launch</td>
<td>Integer denoting the number of sessions launched (logged on or reconnected) on the machine at the time we collect this data point</td>
</tr>
<tr>
<td>Linux OS name and version</td>
<td>os_name_version</td>
<td>Text string denoting the Linux OS name and version of the machine</td>
</tr>
<tr>
<td>Session key</td>
<td>session_key</td>
<td>Identifying the session where the data originates</td>
</tr>
<tr>
<td>Resource type</td>
<td>resource_type</td>
<td>Text string denoting the resource type of the launched session: desktop or &lt;appname&gt;</td>
</tr>
<tr>
<td>Active session time</td>
<td>active_session_time</td>
<td>Used to save the session’s active times. One session can have multiple active times because the session can disconnect/reconnect</td>
</tr>
<tr>
<td>Session duration time</td>
<td>session_duration_time</td>
<td>Used to save the session’s duration from logon to logoff</td>
</tr>
<tr>
<td>Data Point</td>
<td>Key Name</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>---------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Receiver client type</td>
<td>receiver_type</td>
<td>Integer denoting the type of Citrix Workspace app used to launch the session</td>
</tr>
<tr>
<td>Receiver client version</td>
<td>receiver_version</td>
<td>Text string denoting the version of Citrix Workspace app used to launch the session</td>
</tr>
<tr>
<td>Printing count</td>
<td>printing_count</td>
<td>Integer denoting the number of times the session uses the printing function</td>
</tr>
<tr>
<td>USB redirection count</td>
<td>usbRedirecting_count</td>
<td>Integer denoting the number of times the session uses a USB device</td>
</tr>
<tr>
<td>Gfx Provider type</td>
<td>gfx_provider_type</td>
<td>Text string denoting the graphics provider type of the session</td>
</tr>
<tr>
<td>Shadowing count</td>
<td>shadow_count</td>
<td>Integer denoting the number of times the session has been shadowed</td>
</tr>
<tr>
<td>User selected Language</td>
<td>ctxism_select</td>
<td>Composed long string that contains all languages that users have selected</td>
</tr>
<tr>
<td>Smartcard redirecting count</td>
<td>scardRedirecting_count</td>
<td>Integer denoting the number of times the session uses the smart card redirection, including session logon and the use of smart cards during the session</td>
</tr>
</tbody>
</table>

**Configure USB redirection**

March 25, 2019

USB devices are shared between Citrix Workspace app and the Linux VDA desktop. When a USB device
is redirected to the desktop, the user can use the USB device as if it were locally connected.

USB redirection includes three main areas of functionality:

- Open-source project implementation (VHCI)
- VHCI service
- USB service

**Open-source VHCI:**

This portion of the USB redirection feature develops a general USB device sharing system over an IP network. It consists of a Linux kernel driver and some user mode libraries that allow you to communicate with the kernel driver to get all the USB data. In the Linux VDA implementation, Citrix reuses the kernel driver of VHCI. However, all the USB data transfers between the Linux VDA and Citrix Workspace app are encapsulated in the Citrix ICA protocol package.

**VHCI service:**

The VHCI service is an open-source service provided by Citrix to communicate with the VHCI kernel module. This service works as a gateway between VHCI and the Citrix USB service.

**USB service:**

The USB service represents a Citrix module that manages all the virtualization and data transfers on the USB device.

**How USB redirection works**

Typically, if a USB device is redirected successfully to the Linux VDA, one or more device nodes are created in the system /dev path. Sometimes, however, the redirected device cannot be used by an active Linux VDA session. USB devices rely on drivers to function properly and some devices require special drivers. If drivers are not provided, the redirected USB devices are inaccessible to the active Linux VDA session. To ensure USB device connectivity, install the drivers and configure the system properly.

The Linux VDA supports a list of USB devices that are successfully redirected to and from the client. In addition, the device is properly mounted, especially the USB disk, allowing the user to access the disk without any additional configuration.

**Configure USB redirection**

A Citrix policy controls whether USB device redirection is enabled or disabled. In addition, the type of device can also be specified using a Delivery Controller policy. When configuring USB redirection for the Linux VDA, configure the following policy and rules:

- Client USB device redirection policy
• Client USB device redirection rules

Enable USB redirection policy

In Citrix Studio, enable (or disable) USB device redirection to and from the client (for workstation hosts only).

In the Edit Setting dialog:

1. Select **Allowed**.
2. Click **OK**.

Set USB redirection rules

After enabling the USB redirection policy, set redirection rules using Citrix Studio by specifying which devices are allowed (or denied) on the Linux VDA.

In the Client USB device redirection rules dialog:

1. Click **New** to add a redirection rule, or click **Edit** to review an existing rule.
2. After creating (or editing) a rule, click **OK**.

For more information about configuring generic USB redirection, see [Citrix Generic USB Redirection Configuration Guide](#).

**Troubleshoot USB redirection issues**

Use the information in this section to troubleshoot various issues that you might encounter when using the Linux VDA.

**No devices in the toolbar of Citrix Workspace app**

Sometimes, you might not be able to see devices listed in the toolbar of Citrix Workspace app, which indicates that no USB redirection is taking place. If you encounter the issue, verify the following:

- The policy is configured to allow USB redirection
- The Kernel module is compatible with your kernel
Note:
The Devices tab is not available in Citrix Workspace app for Linux.

Failed redirection when USB devices can be seen in the toolbar of Citrix Workspace app, but are labeled policy restricted

This issue occurs due to the device’s policy configuration. In such cases:

- Configure the Linux VDA policy to enable redirection
- Check whether any additional policy restrictions are configured in the registry of Citrix Workspace app. A device might be blocked by the registry setting of Citrix Workspace app. Check DeviceRules in the registry path to ensure that the device is not denied access by this setting:

  HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\ICA Client\GenericUSB

For more information, see How to Configure Automatic Redirection of USB Devices on the Citrix Support site.

A USB device is redirected successfully, but I cannot use it in my session

Usually, only supported USB devices can be redirected. Sometimes, however, other kinds of devices might be redirected to an active Linux VDA session. In these situations, for every redirected device, a node owned by the user is created in the system /dev path. However, it is the drivers and the configuration that determine whether the user can use the device successfully. If you find a device owned (plugged in) but inaccessible, add the device to an unrestricted policy.

Note:

In case of USB drives, the Linux VDA configures and mounts the disk. The user (and only the owner who installed it) can access the disk without any additional configuration. This might not be the case for devices that are not in the supported device list.
**Build the VHCI kernel module**

USB redirection depends on the VHCI kernel modules (`usb-vhci-hcd.ko` and `usb-vhci-iocif.ko`). These modules are part of the Linux VDA distribution (as part of the RPM package). They are compiled based on the official Linux distribution kernels and are noted in the following table:

<table>
<thead>
<tr>
<th>Supported Linux distribution</th>
<th>Kernel version</th>
</tr>
</thead>
<tbody>
<tr>
<td>RHEL 6.9</td>
<td>2.6.32-696.10.3.el6.x86_64</td>
</tr>
<tr>
<td>RHEL 7.6</td>
<td>3.10.0-957</td>
</tr>
<tr>
<td>SUSE 12.3</td>
<td>4.4.73-5-default</td>
</tr>
<tr>
<td>Ubuntu 18.04</td>
<td>4.15.0-42-generic</td>
</tr>
<tr>
<td>Ubuntu 16.04</td>
<td>4.4.0-45-generic</td>
</tr>
</tbody>
</table>

**Important:**

If the kernel of your machine is not compatible with the driver built by Citrix for the Linux VDA, the USB service might fail to start. In this case, you can use the USB redirection feature only if you build your own VHCI kernel modules.

**Verify whether your kernel is consistent with the modules built by Citrix**

On the command line, execute the following command to verify whether the kernel is consistent:

```
1 insmod /opt/Citrix/VDA/lib64/usb-vhci-hcd.ko
```

If the command executes successfully, the kernel module has loaded successfully and the version is consistent with the one installed by Citrix.

If the command executes with errors, the kernel is inconsistent with the Citrix module and must be rebuilt.

**Rebuild the VHCI kernel module**

If your kernel module is inconsistent with the Citrix version, do the following:

1. Download the LVDA source code from the [Citrix download site](#). Select the file contained in the section “*Linux Virtual Delivery Agent (sources)*.”
2. Restore files from the citrix-linux-vda-sources.zip file; you can get VHCI source files in `linux-vda-sources/vhci-hcd-1.15.tar.bz2`; you can restore VHCI files using `tar xvf vhci-hcd-1.15.tar.bz2`. 

© 1999-2019 Citrix Systems, Inc. All rights reserved.
3. Build the kernel module based on the header files and the `Module.symvers` file. Use the following steps to install the kernel header files and create `Module.symvers` based on the appropriate Linux distribution:

**RHEL/CentOS:**

```
1 yum install kernel-devel
```

**SUSE 12:**

```
1 zypper install kernel-devel
2
3 zypper install kernel-source
```

**Ubuntu 16.04:**

```
1 apt-get install linux-headers
```

**Tip:**

If the installation is successful, there is a kernel folder resembling:

```
/usr/src/kernels/3.10.0-327.10.1.el7.x86_64
```

4. In the `/usr/src/kernels/3.10.0-327.10.1.el7.x86_64` folder, verify that the `Module.symvers` file is present. If this file is not in the folder, build the kernel to get this file (for example, `make oldconfig; make prepare; make modules; make`) or copy it from `/usr/src/kernels/3.10.0-327.10.1.el7.x86_64-obj/x86_64/defaults/module.*`

5. In the `vhci-hcd-1.15/Makefile` file, change the Makefile of VCHI and set `KDIR` to the kernel directory:

```
1  # KDIR = $(BUILD_PREFIX)/lib/modules/$(KVERSION)/build
2  3 KDIR = /usr/src/kernels/3.10.0-327.10.1.el7.x86_64
```

6. In the folder `vhci-hcd-1.15/`, run `make` to build the VHCI kernel.

**Note:**

If the build was successful, `usb-vhci-hcd.ko` and `usb-vhci-iocifc.ko` are created in the `vhci-hcd-1.15/` folder.

7. Replace the kernel module with the newly built one: `cp -f usb-vhci-*.ko /opt/Citrix/VDA/lib64`

8. Restart the USB service: `service ctxusbsd restart`

9. Log off and back on to the session again. Check whether USB redirection is functioning.
Supported USB devices

The following devices have been verified to support this version of the Linux VDA. Other devices might be freely used, with unexpected results:

<table>
<thead>
<tr>
<th>USB mass storage device</th>
<th>VID:PID</th>
<th>File system</th>
</tr>
</thead>
<tbody>
<tr>
<td>Netac Technology Co., Ltd</td>
<td>0dd8:173c</td>
<td>FAT32</td>
</tr>
<tr>
<td>Kingston Datatraveler 101 II</td>
<td>0951:1625</td>
<td>FAT32</td>
</tr>
<tr>
<td>Kingston Datatraveler GT101 G2</td>
<td>1567:8902</td>
<td>FAT32</td>
</tr>
<tr>
<td>SanDisk SDCZ80 flash drive</td>
<td>0781:5580</td>
<td>FAT32</td>
</tr>
<tr>
<td>SanDisk Cruzer 16GB</td>
<td>1058:10B8</td>
<td>FAT32</td>
</tr>
<tr>
<td>WD HDD</td>
<td>0781:5567</td>
<td>FAT32</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>USB 3D mouse</th>
<th>VID:PID</th>
</tr>
</thead>
<tbody>
<tr>
<td>3DConnexion SpaceMouse Pro</td>
<td>046d: c62b</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>USB scanner</th>
<th>VID:PID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epson Perfection V330 photo</td>
<td>04B8: 0142</td>
</tr>
</tbody>
</table>

Known issues

- **Unable to unmount the redirected USB disk.** For the access control of all USB disks redirected from Citrix Workspace app, the Linux VDA manages all these devices under administrative privilege to ensure that only the owner can access the redirected device. As a result, the user cannot unmount the device without the administrative privilege.
• **File lost when you stop redirecting a USB disk.** If you redirect a USB disk to a session and try to modify it (for example, create some files on the disk), then stop redirecting it immediately using the toolbar of Citrix Workspace app, the file you modified or created can be lost. This issue occurs because when you write data to a file system, the system mounts the memory cache in the file system. The data is not written to the disk itself. If you stop redirecting using the toolbar of Citrix Workspace app, there is no time remaining for the data being flushed to the disk, which results in lost data. To resolve this issue, use the sync command in a terminal to flush data to the disk before stopping USB redirection.

```bash
[User1@rhel74work ~]$ sync
```

• **A kernel building error might occur on specific kernels of Ubuntu 16.** The error reads **implicit declaration of function ‘copy_to_user’** as shown below:

```
[usb-vc1iocifc.c:2165] error: implicit declaration of function ‘copy_to_user’
```

The error occurs due to header file changes in the kernels. As a workaround, add the `#include <linux/uaccess.h>` line to the `vhci-hcd-1.15/usb-vc1iocifc.c` file.

```
#include <linux/fs.h>
#include <linux/uaccess.h>
#include "usb-vc1iocifc.h"
```

• **A kernel building error might occur on kernel 4.15.0-29-generic of Ubuntu 16.** The error
reads `driver_attr_debug_output' undeclared as shown below:

```
error: 'driver_attr_debug_output' undeclared (first use in this function)
```

The error occurs when symbols are missing on the kernel. As a workaround, disable the macro definition for DEBUG in the `vhci-hcd-1.15/usb-vhci-iocifc.c` and `vhci-hcd-1.15/usb-vhci-hcd.c` files.

```
22
23  #define DEBUG
24
25  #include <linux/module.h>
```

### Configure session reliability

November 9, 2018

Citrix introduces the session reliability feature to all supported Linux platforms. Session reliability is enabled by default.

Session reliability reconnects ICA sessions seamlessly across network interruptions. For more information about session reliability, see Auto client reconnect and session reliability.

**Note:** Data transmitted through a session reliability connection is in plain text by default. For security purposes, Citrix recommends that you enable SSL encryption. For more information about SSL encryption, see Secure user sessions using SSL.

### Configuration

**Policy settings in Citrix Studio**

You can set the following policies for session reliability in Citrix Studio:

- Session reliability connections
- Session reliability timeout
- Session reliability port number
- Reconnection UI transparency level

For more information, see Session reliability policy settings and Auto client reconnect policy settings.

**Note:** After setting the Session reliability connections or Session reliability port number policy, restart the VDA service and the HDX service, in this order, for your settings to take effect.
**Settings on the Linux VDA**

- **Enable/disable the session reliability TCP listener**

By default, the session reliability TCP listener is enabled and listening on port 2598. To disable the listener, run the following command.

```bash
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\WinStations\cgp" -v "fEnableWinStation" -d "0x00000000"
```

**Note:** Restart the HDX service for your settings to take effect. Disabling the TCP listener does not disable session reliability. Session reliability is still available through other listeners (for example, SSL) if the feature is enabled through the **Session reliability connections** policy.

- **Session reliability port number**

You can also set the session reliability port number by using the following command (using port number 2599 as an example).

```bash
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\WinStations\cgp" -v "PortNumber" -d "2599"
```

**Note:** Restart the HDX service for your setting to take effect. If the port number has been set through the policy setting in Citrix Studio, your setting on the Linux VDA is ignored. Ensure that the firewall on the VDA is configured not to prohibit network traffic through the set port.

- **Server-to-client keep-alive interval**

Session reliability keep-alive messages are sent between the Linux VDA and the ICA client when there is no activity in the session (for example, no mouse movement, no screen update). The keep-alive messages are used to detect whether the client is still responsive. If there is no response from the client, the session is suspended until the client reconnects. This setting specifies the number of seconds between successive keep-alive messages. By default, this setting is not configured. To configure it, run the following command (using 10 seconds as an example).

```bash
/opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\XTEConfig" -t "REG_DWORD" -v "CgpServerToClientKeepAlive" -d "10" --force
```

- **Client-to-server keep-alive interval**

This setting specifies the number of seconds between successive keep-alive messages sent from the ICA client to the Linux VDA. By default, this setting is not configured. To configure it, run the following command (using 10 seconds as an example).
Troubleshooting

Unable to launch sessions after enabling session reliability through the policy setting.

To work around this issue, do the following:

1. Ensure that the VDA service and HDX service are restarted, in this order, after you enable session reliability through the policy setting in Citrix Studio.

2. Use the following command to verify that the session reliability TCP listener is running (using port 2598 as an example).

   ```
   netstat -an | grep 2598
   ```

   If there is no TCP listener on the session reliability port, enable the listener by using the following command.

   ```
   /opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\Win Stations\cgp" -v "fEnableWinStation" -d "0x00000001"
   ```

Soft keyboard

February 1, 2019

The soft keyboard feature is available in a Linux virtual desktop or application session. The soft keyboard shows or hides automatically when you enter or leave an input field.
Test mrvc
Note:
The feature is available for RHEL 7.6, CentOS 7.6, SUSE 12.3, Ubuntu 16.04, and Ubuntu 18.04. It is supported on Citrix Workspace app for iOS and for Android.

Enable and disable the feature

The feature is disabled by default. Use the `ctxreg` utility to enable or disable the feature. The feature configuration on a given Linux VDA applies to all sessions on that VDA.

To enable the feature:

1. Run the command:
   ```
   /opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\CurrentControlSet\Control\Citrix\VirtualChannels\MrVc" -v "Enabled" -d "0x00000001"
   ```

2. In Citrix Studio, set the Automatic keyboard display policy to Allowed.

3. (Optional) For RHEL 7 and CentOS 7, run the following command to configure Intelligent Input Bus (IBus) as the default IM service:
   ```
   echo "GTK_IM_MODULE=ibus" >> /etc/bashrc
   ```

To disable the feature, run the command:

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\CurrentControlSet\Control\Citrix\VirtualChannels\MrVc" -v "Enabled" -d "0x00000000"
```

Note:
The preceding settings take effect when you log on to a new session or log off and back on to the current session.

Limitations

- The feature might not work as expected with Google Chrome, LibreOffice, and other apps.
- To display the soft keyboard again after manually hiding it, click a non-input field and then the current input field again.
- The soft keyboard might not appear when you click from one input field to another in a web browser. To work around this issue, click a non-input field and then the target input field.
• The feature does not support Unicode characters and double-byte characters (such as Chinese, Japanese, and Korean characters).

• The soft keyboard is not available for password input fields.

• The soft keyboard might overlap the current input field. In this case, move the app window or scroll up your screen to move the input field to an accessible position.

• Due to compatibility issues between Citrix Workspace app and Huawei tablets, the soft keyboard appears on Huawei tablets even with a physical keyboard connected.

Client IME

August 29, 2018

Overview

Double-byte characters (such as Chinese, Japanese, and Korean characters) must be typed through an Input Method Editor (IME). The Client IME provides a way to type such characters by means of any IME that is compatible with Citrix Workspace app on the client side, such as the Windows native CJK IME.

Installation

This feature is installed automatically when you install the Linux VDA.

Usage

Open a Citrix Virtual Apps or Citrix Virtual Desktops session as per usual.

Change your input method as required on the client side to start using the Client IME.

Known issues

• You must double-click a cell in a Google spreadsheet before you can use the Client IME to type characters in the cell.

• The Client IME is not automatically disabled in Password fields.

• The IME user interface does not follow the cursor in input area.

• Client IME is not supported in a SUSE 11 distribution.
Support for multiple language inputs

December 3, 2018

As of the Linux VDA Version 1.4, Citrix has added support for published applications. Users can access a desired Linux application without the Linux desktop environment.

However, the native language bar on the Linux VDA was unavailable to the published application because the language bar is highly integrated with the Linux desktop environment. As a result, users were unable to input text in a language that requires IME such as Chinese, Japanese, or Korean. Further, it was also not possible for users to switch between keyboard layouts during an application session.

To address those issues, this feature provides a language bar for published applications that accept text input. The language bar enables users to select a server-side IME and to switch between keyboard layouts during an application session.

Configuration

You can use the ctxreg utility to enable or disable this feature (disabled by default). The feature configuration on a given Linux VDA server applies to all applications published on that VDA.

The configuration key is “HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\LanguageBar” and the type is DWORD.

To enable this feature, run the command:

```bash
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE \SYSTEM\CurrentControlSet\Control\Citrix\LanguageBar" -v "Enabled" -d "0x00000001"
```

To disable this feature, run the command:

```bash
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE \SYSTEM\CurrentControlSet\Control\Citrix\LanguageBar" -v "Enabled" -d "0x00000000"
```

Usage

The usage is straightforward.

1. Enable the feature.
2. Access a published application that can accept text input. A language bar appears in the session, alongside the application.

3. From the drop-down menu, select **Region & Language** to add the desired language (input source).

4. Select the IME or keyboard layout from the drop-down menu.

5. Type a language using the selected IME or keyboard layout.

**Note:**

- When you change a keyboard layout on the VDA-side language bar, ensure that the same keyboard layout is used on the client side (running Citrix Workspace app).
- The **accountsservice** package must be upgraded to Version 0.6.37 or later before you can perform settings in the **Region & Language** dialog box.

---

**Dynamic keyboard layout synchronization**

December 3, 2018
Previously, the keyboard layouts on the Linux VDA and on the client device had to be the same. For example, when the keyboard layout changed from English to French on the client device but not on the VDA, key mapping issues might occur and persist until the VDA changed to French too.

Starting with this release, Citrix is addressing the issue by automatically synchronizing the keyboard layout of the VDA with that of the client device. Anytime the keyboard layout on the client device changes, the layout on the VDA follows suit.

Tip:
This feature is supported on Citrix Workspace app for Windows and is compatible with both published apps and desktops.

Configuration

This feature is disabled by default. Use the `ctxreg` utility to enable or disable this feature. The feature configuration on a given Linux VDA applies to all sessions on that VDA.

The configure key is “HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\SyncKeyboardLayout” and the type is DWORD.

To enable this feature, run the command:

```
1 /opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Citrix\LanguageBar" -v "SyncKeyboardLayout" -d "0x00000001"
```

To disable this feature, run the command:

```
1 /opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Citrix\LanguageBar" -v "SyncKeyboardLayout" -d "0x00000000"
```

Usage

With this feature enabled, when the keyboard layout changes on the client device during a session, the keyboard layout of the session changes accordingly.

For example, if you change the keyboard layout on a client device to French (FR):

Then the keyboard layout of the Linux VDA session also changes to “fr.”
In an application session, you can see this automatic change if you have enabled the language bar:

![Image of language bar change in application session]

In a desktop session, you can see this automatic change in the task bar:

![Image of language bar change in desktop session]

**Client IME user interface synchronization**

December 3, 2018

**Overview**

To date, the client IME user interface (including the composition window and candidate window) was positioned in the upper left corner of the screen. It did not follow the cursor and sometimes was located far from the cursor in the text input area:
Starting with this release, Citrix enhances usability and further improves the seamless experience with the client IME as follows:

**Note:**

The feature is available for RHEL 7.x, SUSE 12.x, Ubuntu 16.04, and CentOS 7.x. It is supported on Citrix Workspace app for Windows and for Mac.

The feature installs automatically, but you must enable the feature before you can use it.

### Enable and disable the feature

The feature is disabled by default. Use the `ctxreg` utility to enable or disable the feature. The feature configuration on a given Linux VDA applies to all sessions on that VDA.

To enable the feature, run the command:

```bash
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Citrix\LanguageBar" -v "SyncClientIME" -d "0x00000001"
```

To disable the feature, run the command:

```bash
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Citrix\LanguageBar" -v "SyncClientIME" -d "0x00000000"
```
HDX Insight

February 1, 2019

Overview

HDX Insight is part of the Citrix Application Delivery Management (ADM) and is based on the popular industry standard AppFlow. It enables IT to deliver an exceptional user experience by providing unprecedented end-to-end visibility into the Citrix ICA traffic that passes through the Citrix ADC or CloudBridge application networking fabric.

In this release, the Linux VDA partially supports the HDX Insight feature. Because the End User Experience Management (EUEM) feature is not implemented, the data points related to time duration are not available.

Installation

No dependent packages need installation.

Usage

HDX Insight analyzes the ICA messages passed through the Citrix ADC between Citrix Workspace app and the Linux VDA.

You must set up an Insight Center environment with the Linux VDA and enable the HDX Insight feature. For the information on using the HDX Insight feature, see Use Cases: HDX Insight.

Troubleshooting

No data points are displayed

There might be two causes:

- HDX Insight is not configured correctly.
  
  For example, AppFlow is not enabled on the Citrix ADC or an incorrect Citrix ADC instance is configured on the Citrix ADM.

- The ICA Control Virtual Channel is not started on the Linux VDA.
  
  ps aux | grep -i ctxctl
  
  If ctxctl is not running, contact your administrator to report a bug to Citrix.
No application data points are displayed

Verify that the seamless virtual channel is enabled and a seamless application is launched for a while.

Known issue

Unable to display the data points related to time duration. Because the EUEM feature is not implemented, the data points related to time duration (such as ICA RTT) are unavailable and are displayed as N/A.

Adaptive transport

December 3, 2018

Previously available as an experimental feature, adaptive transport is a fully supported feature in this release.

Adaptive transport is a data transport mechanism for Citrix Virtual Apps and Desktops. It is faster, more scalable, improves application interactivity, and is more interactive on challenging long-haul WAN and internet connections. For more information about adaptive transport, see Adaptive transport.

Enable adaptive transport

In Citrix Studio, verify that the HDX Adaptive Transport policy is set to Preferred or Diagnostic mode. Preferred is selected by default.

- **Preferred**: Adaptive transport over Enlightened Data Transport (EDT) is used when possible, with fallback to TCP.
- **Diagnostic mode**: EDT is forced on and fallback to TCP is disabled.
Disable adaptive transport

To disable adaptive transport, set the **HDX Adaptive Transport** policy to **Off** in Citrix Studio.

**Troubleshooting**

**Check whether adaptive transport is enabled**

To check whether UDP listeners are running, run the following command.

```
netstat -an | grep "1494\|2598"
```

In normal circumstances, the output is similar to the following.
Tracing On

November 9, 2018

Overview

Collecting logs and reproducing issues slow down the diagnostics and degrade the user experience. This release provides the Tracing On feature to ease such efforts. Tracing is enabled for the Linux VDA by default.

Configuration

The ctxlogd daemon and the setlog utility are now included in the Linux VDA release package. By default, the ctxlogd daemon starts after you install and configure the Linux VDA.

ctxlogd daemon

All the other services that are traced depend on the ctxlogd daemon. You can stop the ctxlogd daemon if you do not want to keep the Linux VDA traced.

setlog utility

Tracing On is configured using the setlog utility, which is under the /opt/Citrix/VDA/bin/ path. Only the root user has the privilege to run it. You can use the GUI or run commands to view and change the configurations. Run the following command for help with the setlog utility:

```
1 setlog help
```
Values

By default, Log Output Path is set to /var/log/xdl/hdx.log, Max Log Size is set to 200 MB, and you can save up to two old log files under Log Output Path.

View the current setlog values:

```
1 setlog values
2 log_path (Log Output Path) = /var/log/xdl/hdx.log
3 log_size (Max Log Size (MiB)) = 200
4 log_count (Max Old Log Files) = 2
```

View or set a single setlog value:

```
1 setlog value <name> [<value>]
```

For example:

```
1 setlog value log_size 100
```

Levels

By default, the log level is set to Warnings.

View the log levels set for different components:

```
1 setlog levels
```

You can set all log levels (including Disable, Inherited, Verbose, Information, Warnings, Errors, and Fatal Errors) by using the following command:

```
1 setlog level <class> [<level>]
```

The <class> variable specifies one component of the Linux VDA. To cover all components, set it to all:

```
1 setlog level all error
2 Setting log class ALL to ERROR.
```
**Flags**

By default, the flags are set as follows:

```plaintext
setlog flags
DATE = true
TIME = true
NAME = true
PID = true
TID = false
SID = true
UID = false
GID = false
CLASS = false
LEVEL = false
FUNC = true
FILE = false
```

View the current flags:

```plaintext
setlog flags
```

View or set a single log flag:

```plaintext
setlog flag <flag> [state]
```

**Restore Defaults**

Revert all levels, flags, and values to the default settings:

```plaintext
setlog default
```
**Important:**

The ctxlogd service is configured using the `/var/xdl/.ctxlog` file, which only root users can create. Other users do not have write permission to this file. Citrix recommends that root users not give write permission to other users. Failure to comply can cause the arbitrary or malicious configuration to ctxlogd, which can affect server performance and therefore the user experience.

**Troubleshooting**

The ctxlogd daemon fails and you cannot restart the ctxlogd service when the `/var/xdl/.ctxlog` file is missing (for example, accidentally deleted).

`/var/log/messages`:

```plaintext
1 Apr 02:28:21 RH72 citrix-ctxlogd[17881]: Failed to open logging configuration file.
2
3 Apr 02:28:21 RH72 systemd: ctxlogd.service: main process exited, code =exited, status=1/FAILURE
4
5 Apr 02:28:21 RH72 systemd: Unit ctxlogd.service entered failed state.
6
7 Apr 02:28:21 RH72 systemd: ctxlogd.service failed.
```

To solve this issue, run `setlog` as a root user to recreate the `/var/xdl/.ctxlog` file. Then restart the ctxlogd service on which other services depend.

**Shadow sessions**

February 1, 2019

The session shadowing feature allows domain administrators to view users' ICA sessions in an intranet. The feature uses noVNC to connect to the ICA sessions and is supported only with RHEL 7.x and Ubuntu 16.04.

**Note:**

To use the session shadowing feature, the version of Citrix Director must be 7.16 or later.
Installation and configuration

Dependencies

Two new dependencies, python-websockify and x11vnc, are required for session shadowing. The python-websockify and x11vnc dependencies are automatically installed when you install the Linux VDA on Ubuntu 16.04. On RHEL 7.x, you must manually install python-websockify and x11vnc after you install the Linux VDA.

Run the following command on RHEL 7.x to install python-websockify and x11vnc (x11vnc version 0.9.13 or later).

```
sudo yum install -y python-websockify x11vnc
```

To resolve python-websockify and x11vnc, enable the following repositories on RHEL 7.x:

- **EPEL**

  The Extra Packages for Enterprise Linux (EPEL) repository is required for both python-websockify and x11vnc. Run the following command to enable the EPEL repository:

  ```
sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-$(rpm -E %({rhel}3)) -E %{%rhel} /v1/noarch.rpm
```

- **Optional RPMs**

  Run either of the following commands to enable the optional RPMs repository for installing some dependency packages of x11vnc:

  For workstation:

  ```
sudo subscription-manager repos --enable=rhel-7-workstation-optional-rpms
```

  For server:

  ```
sudo subscription-manager repos --enable=rhel-7-server-optional-rpms
```

Port

The session shadowing feature automatically selects available ports from within 6001-6099 to build up connections from the Linux VDA to Citrix Director. Therefore, the number of ICA sessions that you can shadow concurrently is limited to 99. Ensure that enough ports are available to meet your requirements, especially for multi-session shadowing.
The following table lists related registries:

<table>
<thead>
<tr>
<th>Registry</th>
<th>Description</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>EnableSessionShadowing</td>
<td>Enables or disables the session shadowing feature</td>
<td>1 (Enabled)</td>
</tr>
<tr>
<td>ShadowingUseSSL</td>
<td>Determines whether to encrypt the connection between the Linux VDA and Citrix Director</td>
<td>0 (Disabled)</td>
</tr>
</tbody>
</table>

Run the ctxreg command on the Linux VDA to change the registry values. For example, to disable session shadowing, run the following command:

```
1 /opt/Citrix/VDA/bin/ctxreg update -k "HKLM\Software\Citrix\VirtualDesktopAgent" -v "EnableSessionShadowing" -d 0x00000000
```

**SSL**

The noVNC connection between the Linux VDA and Citrix Director uses the WebSocket protocol. For session shadowing, whether ws:// or wss:// is chosen is determined by the previously mentioned “ShadowingUseSSL” registry. By default, ws:// is chosen. However, for security reasons, Citrix recommends that you use wss:// and install certificates on each Citrix Director client and on each Linux VDA server. Citrix disclaims any security responsibility for the Linux VDA session shadowing by using ws://.

**Obtain server and root SSL certificates**

Certificates must be signed by a trusted Certificate Authority (CA).

A separate server certificate (including the key) is required for each Linux VDA server on which you want to configure SSL. A server certificate identifies a specific computer, so you must know the Fully Qualified Domain Name (FQDN) of each server. For convenience, you can use a wildcard certificate for the whole domain instead. In this case, you must know at least the domain name.

In addition to installing a server certificate on each server, you must install a root certificate from the same CA on each Citrix Director client that communicates with the Linux VDA server. Root certificates are available from the same CAs that issue the server certificates. You can install server and client certificates from a CA that is bundled with your operating system, from an enterprise CA (a CA that your
organization makes accessible to you), or from a CA not bundled with your operating system. Consult the security team of your organization to find out which of the methods they require for obtaining certificates.

**Important:**

- The Common Name for a server certificate must be the exact FQDN of the Linux VDA server or at least the correct wildcard plus domain characters. For example, vda1.basedomain.com or *.basedomain.com.
- Hashing algorithms including the SHA1 and MD5 are too weak for signatures in digital certificates for some browsers to support. So SHA-256 is specified as the minimum standard.

**Install a root certificate on each Citrix Director client**

Session shadowing uses the same registry-based certificate store as IIS, so you can install root certificates using IIS or the Microsoft Management Console (MMC) Certificates snap-in. When you receive a certificate from a CA, you can restart the Web Server Certificate Wizard in IIS and the wizard installs the certificate. Alternatively, you can view and import certificates on the computer using the MMC and add the certificate as a standalone snap-in. Internet Explorer and Google Chrome import the certificates installed on your operation system by default. For Mozilla Firefox, you must import your root SSL certificates on the **Authorities** tab of Certificate Manager.

**Install a server certificate and its key on each Linux VDA server**

Name the server certificates “shadowingcert.*” and the key file “shadowingkey.*” ( * can indicate the format, for example, shadowingcert.csr and shadowingkey.key). Put server certificates and key files under the path `/etc/xdl/shadowingssl` and protect them properly with restricted permissions. An incorrect name or path makes the Linux VDA unable to find a specific certificate or key file and therefore causes connection failure with Citrix Director.

**Usage**

From Citrix Director, find the target session and click **Shadow** in the **Session Details** view to send a shadowing request to the Linux VDA.
After the connection initializes, a confirmation appears on the ICA session client (not the Citrix Director client) to ask the user for permission to shadow the session.

If the user clicks Yes, a window appears on the Citrix Director side, indicating that the ICA session is being shadowed.

For more information about the usage, see the Citrix Director Documentation.

Limitations

- Session shadowing is designed for use in an Intranet only. It does not work for external networks even connecting through Citrix Gateway. Citrix disclaims any responsibility for the Linux VDA session shadowing in an external network.
- With session shadowing enabled, a domain administrator can only view the ICA sessions, but has no permission to write or control it.
• After an administrator clicks **Shadow** from Citrix Director, a confirmation appears to ask the user for permission to shadow the session. A session can be shadowed only when the session user gives the permission.
• The previously mentioned confirmation has a timeout limitation, which is 20s. A shadowing request fails when the time runs out.
• One ICA session can be shadowed by only one administrator in one Citrix Director window. If an ICA session has been shadowed by administrator A and meanwhile, administrator B sends a shadowing request, the confirmation for getting the user permission reappears on the user device. If the user agrees, the shadowing connection for administrator A stops and a new shadowing connection is built for administrator B. It is the same if another shadowing request for the same ICA session is sent by the same administrator.
• To use session shadowing, install Citrix Director 7.16 or later.
• A Citrix Director client uses an FQDN rather than an IP address to connect to the target Linux VDA server. Therefore, the Citrix Director client must be able to resolve the FQDN of the Linux VDA server.

**Troubleshooting**

If session shadowing fails, perform debugging on both the Citrix Director client and the Linux VDA.

**On the Citrix Director client**

Through the developer tools of the browser, check the output logs on the **Console** tab. Or, check the response of the ShadowLinuxSession API on the **Network** tab. If the confirmation for getting the user permission appears but the connection fails to be built, manually ping the FQDN of the Linux VDA to verify that Citrix Director can resolve the FQDN. If there is an issue with the wss:// connection, check your certificates.

**On the Linux VDA**

Verify that the confirmation for getting the user permission appears in response to a shadowing request. If it does not, check the vda.log and hdx.log files for clues. To obtain the vda.log file, do the following:

1. Find the `/etc/xdl/ctx-vda.conf` file. Uncomment the following line to enable the vda.log configuration:
   ```
   Log4jConfig=" /etc/xdl/log4j.xml"
   ```
2. Open `/etc/xdl/log4j.xml`, locate the com.citrix.dmc part, and change “info” to “trace” as follows:
3. Run the service ctxvda restart command to restart the ctxvda service.

If there is an error during connection build-up:

1. Check for any firewall limitation that stops session shadowing from opening the port.
2. Verify that certificates and key files are named properly and put under the correct path if it is the SSL scenario.
3. Verify that there are enough ports left between 6001-6099 for new shadowing requests.

Support Citrix Workspace app for HTML5

November 15, 2018

Starting with this release, you can use Citrix Workspace app for HTML5 to access Linux virtual apps and desktops directly without connecting your client to Citrix Gateway. For information about Citrix Workspace app for HTML5, see the Citrix documentation.

Enable this feature

This feature is disabled by default. To enable it, do the following:

1. In Citrix StoreFront, enable Citrix Workspace app for HTML5.

   For the detailed procedure, see Step 1 of Knowledge Center article CTX208163.

2. Enable WebSocket connections.

   a) In Citrix Studio, set the WebSockets connections policy to Allowed.

   You can also set the other WebSocket policies. For a full list of the WebSocket policies, see WebSockets policy settings.

   b) On the VDA, restart the ctxvda service and the ctxhdx service, in this order, for your setting to take effect.
On the VDA, run the following command to check whether the WebSocket listener is running.

```
netstat -an | grep 8008
```

When the WebSocket listener is running, the command output is similar to the following:

```
tcp 0 0 :::8008 :::* LISTEN
```

**Note:** You can also enable TLS encryption to secure WebSocket connections. For information about enabling TLS encryption, see Secure user sessions using SSL.

---

**Secure user sessions using SSL**

December 3, 2018

As of Version 7.16, the Linux VDA supports SSL encryption for secure user sessions. SSL encryption is disabled by default.

**Enable SSL encryption**

To enable SSL encryption for secure user sessions, obtain certificates and enable SSL encryption on both the Linux VDA and the Delivery Controller (the Controller).

**Obtain certificates**

Obtain server certificates in PEM format and root certificates in CRT format from a trusted Certificate Authority (CA). A server certificate contains the following sections:

- Certificate
- Unencrypted private key
- Intermediate certificates (optional)

An example of a server certificate:

```
-----BEGIN CERTIFICATE-----
MIIDTTCCAgAwIBAgIJAjGAlwEBzAHMRMIEjAwIBAgIJCzAJBgNVBAYTAlVLMRIwEAYDVQQIEwJ

```

© 1999-2019 Citrix Systems, Inc. All rights reserved.
Enable SSL encryption

Enable SSL encryption on the Linux VDA

On the Linux VDA, use the enable_vdassl.sh tool to enable (or disable) SSL encryption. The tool is located in the /opt/Citrix/VDA/sbin directory. For information about options available in the tool, run the /opt/Citrix/VDA/sbin/enable_vdassl.sh -help command.

Tip: A server certificate must be installed on each Linux VDA server and root certificates must be installed on each Linux VDA server and client.

Enable SSL encryption on the Controller

Note:

- The Controller must use the Fully Qualified Domain Name (FQDN) of the Linux VDA. It cannot use the IP address, which is used by default, to connect to the target Linux VDA.
- You can enable SSL encryption only for entire delivery groups. You cannot enable SSL encryption for specific applications.

In a PowerShell window on the Controller, run the following commands in sequence to enable SSL encryption for the target delivery group and to make the Controller use the FQDN of the Linux VDA.

1. Asnp citrix.*
3. Set-BrokerSite –DnsResolutionEnabled $true

To disable SSL encryption on the Controller, run the following commands in sequence:

1. Asnp citrix.*
3. Set-BrokerSite –DnsResolutionEnabled $false
Troubleshooting

The following “Can’t assign requested address” error might occur in Citrix Workspace app for Windows when you try to access a published desktop session:

As a workaround, add an entry to the `hosts` file, which is similar to:

```
10.108.13.180 rhvm72work.citrixlab.local
```

Where

- `10.108.13.180` is the IP address of the Linux VDA.
- `rhvm72work.citrixlab.local` is the FQDN of the Linux VDA.

On Windows machines, the `hosts` file typically locates at `C:\Windows\System32\drivers\etc\hosts`.

Secure user sessions using DTLS

August 29, 2018

DTLS encryption is a fully supported feature starting with the 7.18 release. By default, this feature is enabled on the Linux VDA. For more information, see Transport Layer Security.

Enable DTLS encryption

Verify that adaptive transport is enabled

In Citrix Studio, verify that the `HDX Adaptive Transport` policy is set to Preferred or Diagnostic mode.
Enable SSL encryption on the Linux VDA

On the Linux VDA, use the `enable_vdassl.sh` tool to enable (or disable) SSL encryption. The tool is located at `/opt/Citrix/VDA/sbin`. For information about options available in the tool, run the `/opt/Citrix/VDA/sbin/enable_vdassl.sh --h` command.

Note:
Currently, the Linux VDA supports both DTLS 1.0 and DTLS 1.2. DTLS 1.2 requires Citrix Receiver for Windows 4.12, or Citrix Workspace app 1808 for Windows or later. If your client supports only DTLS 1.0 (for example, Citrix Receiver for Windows 4.11), set `SSLMinVersion` to `TLS_1.0` and `SSL-CipherSuite` to `COM` or `ALL` using the `enable_vdassl.sh` tool.

Pass-through authentication by using smart cards

April 4, 2019

Users can use a smart card connected to the client device for authentication when logging on to a Linux virtual desktop session. This feature is implemented through smart card redirection over the ICA smart card virtual channel. Users can also use the smart card within the session. Use cases include adding a digital signature to a document, encrypting or decrypting an email, or authenticating to a website that requires smart card authentication.

The Linux VDA uses the same configuration as the Windows VDA for this feature. For more information, see the following Configure the smart card environment section.

The availability of pass-through authentication by using smart cards depends on the following factors:

- The Linux VDA is installed on RHEL 7.6.
- Smart cards supported by CoolKey are used.
- Citrix Workspace app for Windows is used.

Note:
Smart card authentication to Citrix Gateway is not officially supported.

Install the Linux VDA software on RHEL 7.6

Install the Linux VDA software using the RPM package manager or easy install, see the Installation overview section.

After the VDA installation is complete, verify that the VDA can register to the Delivery Controller and the published Linux desktop sessions can be launched successfully using password authentication.
Ensure that CoolKey supports your smart card

CoolKey is a widely used smart card driver on RHEL. CoolKey supports four types of smart cards, which are CoolKey cards, CAC, PIV, and PKCS#15. But the number of cards that are formally supported and validated is still limited (see Smart Card Support in Red Hat Enterprise Linux).

In this article, the Yubikey 4 smart card is used as an example to illustrate the configuration. Yubikey 4 is an all-in-one USB CCID PIV device that can easily be purchased from Amazon or other retail vendors. The CoolKey driver supports Yubikey 4.

If your organization requires some other more advanced smart card, prepare a physical machine with RHEL 7.6 and the CoolKey package installed. For information about the CoolKey installation, see Install the smart card driver. Insert your smart card, and run the following command to verify that CoolKey supports your smart card:

```
1 pkcs11-tool --module libcoolkeypk11.so --list-slots
```

If CoolKey supports your smart card, the command output is similar to the following where slot information is contained.

```
[root@rhphy ~]# pkcs11-tool --module libcoolkeypk11.so --list-slots
Available slots:
Slot 0 (0x1): YubiCo Yubikey 4 CCID 00 00
  token label : user1
  token manufacturer :
  token model :
  token flags : login required, token initialized, PIN initialized, readonly
  hardware version : 0.0
  firmware version : 0.0
  serial num :
[root@rhphy ~]#
```
Configuration

Prepare a root certificate

A root certificate is used to verify the certificate on the smart card. Do the following to download and install a root certificate.

1. Obtain a root certificate in PEM format, typically from your CA server.
   You can run a command similar to the following to convert a DER file (*.crt, *.cer, *.der) to PEM. In the following command example, `certnew.cer` is a DER file.
   ```
   openssl x509 -inform der -in certnew.cer -out certnew.pem
   ```

2. Install the root certificate to the openssl directory. The `certnew.pem` file is used as an example.
   ```
   cp certnew.pem /etc/pki/CA/certs/
   ```

Configure the NSS database

The Linux VDA logon module relies on the NSS database to access smart cards and certificates. Do the following to configure the NSS database.

1. Add the previously mentioned root certificate to the NSS database.
   ```
   certutil -A -n "My Corp Root" -t "CT,C,C" -a -d /etc/pki/nssdb -i /etc/pki/CA/certs/certnew.pem
   ```

2. Run the following command to verify that the root certificate is added to the NSS database successfully.
   ```
   certutil -L -d /etc/pki/nssdb
   ```

   The command output is similar to the following if the root certificate is added successfully.
   ```
   Certificate Nickname
   Trust Attributes
   SSL,S/MIME,JAR/XPI
   My Corp Root
   C1, C
   ```

3. Check whether CoolKey is installed in the NSS PKCS#11 library.
   ```
   modutil -list -dbdir /etc/pki/nssdb
   ```

   The command output is similar to the following if the CoolKey module is installed.
If the CoolKey module is not installed, run the following command to install the module manually, and then check the installation again.

```
modutil -add "CoolKey PKCS #11 Module" -libfile libcoolkeypk11.so -dbdir /etc/pki/nssdb
```

**Configure the smart card environment**

You can use the ctxsmartlogon.sh script to configure the smart card environment or do the configuration manually.

- Use the ctxsmartlogon.sh script to configure the smart card environment

**Note:**

The ctxsmartlogon.sh script sets CoolKey as the default smart card driver and adds pkinit information to the default realm. You can change this setting through the `/etc/pam.d/smartcard-auth` and the `/etc/krb5.conf` configuration files.

Before using smart cards for the first time, run the ctxsmartlogon.sh script to configure the smart card environment.

```
sudo /opt/Citrix/VDA/sbin/ctxsmartlogon.sh
```

The results resemble the following:
To disable smart cards:

```bash
sudo /opt/Citrix/VDA/sbin/ctxsmartlogon.sh
```

The results resemble the following:

```
# ctxsmartlogon.sh sets up smart card logon for the Linux VDA, which
# includes automatic installation of the necessary packages and changes to
# the configuration files.
#
# Note:
# The ctxsmartlogon.sh sets CoolKey as the default smart card driver. You can
# change another smart card driver through the related configuration file.

Step 1: Enable smart card logon.
Do you want enable smart card logon? (y/n)[y] y

Step 2: Select the Active Directory integration method.
Please select which Active Directory integration method to use:
1: Winbind
2: SSSD
3: Centrify

Select one of the above options (1-3) [1] 1

Step 3: Install dependent packages.
Installing, please wait...
[krb5-pkinit][pam_krb5] already installed.
Package installed: [Success]

Step 4: Configure krb5.conf.
Specify the root CA path (e.g., /etc/pki/CA/certs/root.pem):/etc/pki/CA/certs/root.pem
/etc/krb5.conf configure successfully.

Step 5: Configure PAM files.
/etc/pam.d/ctxsmartlogon.conf configure successfully.
ctxsmartlogon.sh executed successfully. SmartCard is ready.
```

- Manually configure the smart card environment

The Linux VDA uses the same smart card environment with the Windows VDA. In the environment, multiple components must be configured, including the Domain Controller, Microsoft Certificate Authority (CA), Internet Information Services, Citrix StoreFront, and Citrix Workspace app. For information about the configuration based on the Yubikey 4 smart card, see Knowledge Center article CTX206156.

Before proceeding to the next step, ensure that all components are correctly configured, the private key and user certificate are downloaded to the smart card, and you can successfully log
on to the Windows VDA using the smart card.

Install the PC/SC Lite packages

PCSC Lite is an implementation of the Personal Computer/Smart Card (PC/SC) specification in Linux. It provides a Windows smart card interface for communicating to smart cards and readers. Smart card redirection in the Linux VDA is implemented on the PC/SC level.

Run the following command to install the PC/SC Lite packages.

```
1 yum install pcsc-lite pcsc-lite-ccid pcsc-lite-libs
```

Install the smart card driver

CoolKey is a widely used smart card driver on RHEL. If CoolKey is not installed, run the following command to install it.

```
1 yum install coolkey
```

Install the PAM modules for smart card authentication

Run the following command to install the pam_krb5 and krb5-pkinit modules.

```
1 yum install pam_krb5 krb5-pkinit
```

The pam_krb5 module is a pluggable authentication module that PAM-aware applications can use to check passwords and obtain ticket-granting tickets from the Key Distribution Center (KDC). The krb5-pkinit module contains the PKINIT plug-in that allows clients to obtain initial credentials from the KDC using a private key and a certificate.

Configure the pam_krb5 module

The pam_krb5 module interacts with the KDC to get Kerberos tickets using certificates in the smart card. To enable pam_krb5 authentication in PAM, run the following command:

```
1 authconfig --enablekrb5 --update
```

In the `/etc/krb5.conf` configuration file, add pkinit information according to the actual realm.
EXAMPLE.COM = {
    kdc = KDC.EXAMPLE.COM
    auth_to_local = RULE:[1:$1@$0]
    pkinit_anchors = FILE:/etc/pki/CA/certs/certnew.pem
    pkinit_kdc_hostname = KDC.EXAMPLE.COM
    pkinit_cert_match = ||<EKU>msScLogin,<KU>digitalSignature
    pkinit_eku_checking = kpServerAuth
    pkinit_kdc_hostname = KDC.EXAMPLE.COM
    pkinit_cert_match = ||<EKU>msScLogin,<KU>digitalSignature
    pkinit_eku_checking = kpServerAuth
}

The configuration file resembles the following after you add the pkinit information.

XD.LOCAL = {
    kdc = SXC-DOMAIN.XD.LOCAL
    auth_to_local = RULE:[1:$1@$0]
    pkinit_anchors = FILE:/etc/pki/CA/certs/certnew.pem
    pkinit_kdc_hostname = SXC-DOMAIN.XD.LOCAL
    pkinit_cert_match = ||<EKU>msScLogin,<KU>digitalSignature
    pkinit_eku_checking = kpServerAuth
}

**Configure PAM authentication**

PAM configuration files tell what modules are used for PAM authentication. To add pam_krb5 as an authentication module, add the following line to the /etc/pam.d/smartcard-auth file:

`auth [success=done ignore=ignore default=die] pam_krb5.so preauth_options=X509_user_identity=PKCS11:/usr/lib64/pkcs11/libcoolkeypk11.so`

The configuration file resembles the following after modification if SSSD is used.
The configuration file resembles the following after modification if Winbind is used.

```
# User changes will be destroyed the next time authconfig is run.
auth required pam_env.so
auth [success=1 default=die] pam_krb5.so preauth_options="K500_user identity=KRB511:/usr/lib/security/pam_krb5.so"
auth sufficient pam_permit.so
auth required pam_deny.so

account required pam_unix.so
account sufficient pam_localuser.so
account sufficient pam_unix.so uid < 1000 quiet
account [default=badsuccess user_unknown=ignore] pam_krb5.so
account [default=badsuccess user_unknown=ignore ignore ignore ignore] pam_krb5.so
account required pam_succeed_if.so service in crond quiet use_uid

password required pam_unix.so

session optional pam_keyinit.so revoke
session required pam_limits.so
session optional pam_unix.so
session optional pam_permit.so
session optional pam_xss.so
session optional pam_krb5.so
```

The configuration file resembles the following after modification if Centrify is used.

```
# User changes will be destroyed the next time authconfig is run.
auth required pam_env.so
auth [success=1 default=die] pam_krb5.so preauth_options="K500_user identity=KRB511:/usr/lib/security/pam_krb5.so"
auth sufficient pam_permit.so
auth required pam_deny.so

account required pam_unix.so
account sufficient pam_localuser.so
account sufficient pam_unix.so uid < 1000 quiet
account [default=badsuccess user_unknown=ignore] pam_krb5.so
account [default=badsuccess user_unknown=ignore ignore ignore ignore] pam_krb5.so
account required pam_succeed_if.so service in crond quiet use_uid

password required pam_unix.so

session optional pam_keyinit.so revoke
session required pam_limits.so
session optional pam_unix.so
session optional pam_permit.so
session optional pam_xss.so
session optional pam_krb5.so
```

(Optional) Single sign-on by using smart cards

Single sign-on (SSO) is a Citrix feature that implements pass-through authentication with virtual desktop and application launches. This feature reduces the number of times that users type their PIN. To use SSO with the Linux VDA, configure Citrix Workspace app. The configuration is the same with the Windows VDA. For more information, see Knowledge Center article CTX133982.

Enable the smart card authentication as follows when configuring the group policy in Citrix Workspace app.
Fast smart card logon

Fast smart card is an improvement over the existing HDX PC/SC-based smart card redirection. It improves performance when smart cards are used in high-latency WAN environments. For more information, see Smart cards.

The Linux VDA supports fast smart card on the following versions of Citrix Workspace app:

- Citrix Receiver for Windows 4.12
- Citrix Workspace app 1808 for Windows and later

To make the feature work as expected, run the following command on the Linux VDA:

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\CurrentControlSet\Control\Citrix\VirtualChannels\Scard" -v "PKCS11LibPath" -d "PATH"
```

This command sets the path of the PKCS#11 library related to your smart card. For example, set PATH to `/usr/lib64/pkcs11/libcoolkeypk11.so`.

Enable fast smart card logon on the client
Fast smart card logon is enabled by default on the VDA and disabled by default on the client. On the client, to enable fast smart card logon, include the following parameter in the default.ica file of the associated StoreFront site:

```plaintext
1 [WFClient]
2 SmartCardCryptographicRedirection=On
```

**Disable fast smart card logon on the client**

To disable fast smart card logon on the client, remove the `SmartCardCryptographicRedirection` parameter from the default.ica file of the associated StoreFront site.

**Usage**

**Log on to the Linux VDA by using a smart card**

Users can use a smart card to log on to the Linux VDA in both SSO and non-SSO scenarios.

- In the SSO scenario, users are automatically logged on to StoreFront by using the cached smart card certificate and PIN. When users launch a Linux virtual desktop session in StoreFront, the PIN is passed to the Linux VDA for smart card authentication.

- In the non-SSO scenario, users are prompted to select a certificate and type a PIN to log on to StoreFront.

When users launch a Linux virtual desktop session in StoreFront, a dialog box for logon to the Linux VDA appears as follows. The user name is extracted from the certificate in the smart card and users must type the PIN again for logon authentication.

This behavior is the same with the Windows VDA.
Reconnect to a session by using a smart card

To reconnect to a session, ensure that the smart card is connected to the client device. Otherwise, a gray caching window appears on the Linux VDA side and exits quickly because reauthentication fails without the smart card connected. No other prompt is provided in this case to remind you to connect the smart card.

On the StoreFront side, however, if a smart card is not connected when you try to reconnect to a session, the StoreFront web might give an alert as follows.

Limitation

Smart card removal policy

Now, the Linux VDA uses only the default behavior for smart card removal. When you remove the smart card after logging on to the Linux VDA successfully, the session still keeps connected and the session screen is not locked.

Support for other smartcards and the PKCS#11 library

Although only the CoolKey smart card is listed on our support list, you can try using other smart cards and the PKCS#11 library because Citrix is providing a generic smart card redirection solution. To switch to your specific smart card or the PKCS#11 library:

1. Replace all the `libcoolkeypk11.so` instances with your PKCS#11 library.
2. To set the path of your PKCS#11 library to the registry, run the following command:

   ```
   /opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\CurrentControlSet\Control\Citrix\VirtualChannels\Scard" -v "PKCS11LibPath" -d "PATH"
   ```

   where `PATH` points to your PKCS#11 library such as `/usr/lib64/pkcs11/libcoolkeypk11.so`
3. Disable fast smart card logon on the client.
Double-hop single sign-on authentication

December 12, 2018

The feature injects user credentials entered for accessing a StoreFront store to the AuthManager module of Citrix Workspace app for Linux and Citrix Receiver for Linux 13.10. After injection, you can use the client to access virtual desktops and applications from within a Linux virtual desktop session, without entering user credentials for a second time.

Note:
This feature is supported on Citrix Workspace app for Linux and Citrix Receiver for Linux 13.10.

To enable the feature:

1. On the Linux VDA, install Citrix Workspace app for Linux or Citrix Receiver for Linux 13.10.
   Download the app from the Citrix download page for Citrix Workspace app or for Citrix Receiver.
   The default installation path is /opt/Citrix/ICAClient/. If you install the app to a different path, set the ICAROOT environment variable to point to the actual installation path.

2. In the Citrix StoreFront management console, add the HTTP Basic authentication method for the target store.
3. Add the following key to the AuthManager configuration file ($ICAROOT/config/AuthManConfig.xml) for allowing the HTTP Basic authentication:

```xml
<Protocols>
  <HTTPBasic>
    <Enabled>True</Enabled>
  </HTTPBasic>
</Protocols>
```

4. Run the following commands to install the root certificate in the specified directory.

```bash
cp rootcert.pem $ICAROOT/keystore/cacerts/
$ICAROOT/util/ctx_rehash $ICAROOT/keystore/cacerts/
```

5. Run the following command to enable the feature:

```bash
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\CurrentControlSet\Control\Citrix" -v "LurSsonEnabled" -d "0x00000001"
```

6. Launch a Linux virtual desktop session and start Citrix Workspace app for Linux or Citrix Receiver for Linux 13.10 within that session.
You are prompted for a store account the first time you start Citrix Workspace app for Linux or Citrix Receiver for Linux 13.10 within a Linux virtual desktop session. Later on, you are automatically logged on to the store you specified earlier.

**Note:**

Enter an HTTPS URL as your store account.

Configure unauthenticated sessions

Use the information in this article to configure unauthenticated sessions. No special settings are required when installing the Linux VDA to use this feature.

**Note:**

When configuring unauthenticated sessions, consider that session prelaunch is not supported. Session prelaunch is also not supported on Citrix Workspace app for Android.

Create an unauthenticated store

You must create an unauthenticated store using StoreFront to support an unauthenticated session on the Linux VDA.
Enable unauthenticated users in a Delivery Group

After creating an unauthenticated store, enable unauthenticated users in a Delivery Group to support an unauthenticated session. To enable unauthenticated users in a Delivery Group, follow the instructions in the Citrix Virtual Apps and Desktops documentation.

Set the unauthenticated session idle time

An unauthenticated session has a default idle timeout of 10 minutes. This value is configured through the registry setting AnonymousUserIdleTime. Use the ctxreg tool to change this value. For example, to set this registry setting to five minutes:

```
sudo /opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\CurrentControlSet\Control\Citrix" -v AnonymousUserIdleTime -d 0x00000005
```

Set the maximum number of unauthenticated users

To set the maximum number of unauthenticated users, use the registry key MaxAnonymousUserNumber. This setting limits the number of unauthenticated sessions running on a single Linux VDA concurrently. Use the ctxreg tool to configure this registry setting. For example, to set the value to 32:

```
sudo /opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\CurrentControlSet\Control\Citrix" -v MaxAnonymousUserNumber -d 0x00000020
```

Important:

It is essential that you limit the number of unauthenticated sessions. Too many sessions being launched concurrently can cause problems on the VDA, including running out of available memory.

Troubleshooting

Consider the following when configuring unauthenticated sessions:

- Failed to log on to an unauthenticated session.

Verify that the registry was updated to include the following (set to 0):
Verify that the `ncsd` service is running and configured to enable `passwd` cache:

```
sudo /opt/Citrix/VDA/bin/ctxreg read -k "HKLM\System\CurrentControlSet \Control\Citrix" -v MaxAnonymousUserNumber
```

```
ps uax | grep nscd
cat /etc/nscd.conf | grep 'passwd' | grep 'enable-cache'
```

Set the `passwd` cache variable to `no` if it is enabled, then restart the `ncsd` service. You might need to reinstall the Linux VDA after changing this configuration.

- **The lock screen button is displayed in an unauthenticated session with KDE.**

  The lock screen button and menu are disabled by default in an unauthenticated session. However, they can still be displayed in KDE. In KDE, to disable the lock screen button and menu for a particular user, add the following lines to the configuration file `$Home/.kde/share/config/kdeglobals`. For example:

```
[KDE Action Restrictions]
action/lock_screen=false
```

However, if the `KDE Action Restrictions` parameter is configured as immutable in a global wide `kdeglobals` file such as `/usr/share/kde-settings/kde-profile/default/share/config/kdeglobals`, the user configuration has no effect.

To resolve this issue, try to modify the system-wide `kdeglobals` file to remove the `[Si]` tag at the `KDE Action Restrictions` section or directly use the system-wide configuration to disable the lock screen button and menu. For details about the KDE configuration, see the KDE System Administration/Kiosk/Keys page.

## Configure LDAPS

January 11, 2019

Secure LDAP (LDAPS) allows you to enable the Secure Lightweight Directory Access Protocol for your Active Directory managed domains to provide communication over SSL (Secure Socket Layer)/TLS (Transport Layer Security).

By default, LDAP communications between client and server applications are not encrypted. LDAP using SSL/TLS (LDAPS) enables you to protect the LDAP query content between Linux VDA and LDAP servers.

The following Linux VDA components have dependencies on LDAPS:
• Broker agent: Linux VDA registration to Delivery Controller
• Policy service: Policy evaluation

Configuring LDAPS involves:
• Enable LDAPS on the Active Directory (AD)/LDAP server
• Export the root CA for client use
• Enable/disable LDAPS on Linux VDA
• Configure LDAPS for third-party platforms
• Configure SSSD
• Configure Winbind
• Configure Centrify
• Configure Quest

Enable LDAPS on the AD/LDAP server

You can enable LDAP over SSL (LDAPS) by installing a properly formatted certificate from either a Microsoft certification authority (CA) or a non-Microsoft CA.

Tip:
LDAP over SSL/TLS (LDAPS) is automatically enabled when you install an Enterprise Root CA on a domain controller.

For more information about how to install the certificate and verify the LDAPS connection, see How to enable LDAP over SSL with a third-party certification authority on the Microsoft Support site.

When you have a multi-tier (such as a two-tier or three-tier) certificate authority hierarchy, you do not automatically have the appropriate certificate for LDAPS authentication on the domain controller.

For information about how to enable LDAPS for domain controllers using a multi-tier certificate authority hierarchy, see the LDAP over SSL (LDAPS) Certificate article on the Microsoft TechNet site.

Enable root certificate authority for client use

The client must be using a certificate from a CA that the LDAP server trusts. To enable LDAPS authentication for the client, import the root CA certificate to trust keystore.

For more information about how to export Root CA, see How to export Root Certification Authority Certificate on the Microsoft Support website.

Enable or disable LDAPS on the Linux VDA

To enable or disable LDAPS for Linux VDA, run the following script (while logged on as an administrator):
The syntax for this command includes the following:

- **Enable LDAP over SSL/TLS with the root CA certificate provided:**

  ```bash
  /opt/Citrix/VDA/sbin/enable_ldaps.sh -Enable pathToRootCA
  ```

- **Fallback to LDAP without SSL/TLS**

  ```bash
  /opt/Citrix/VDA/sbin/enable_ldaps.sh -Disable
  ```

The Java keystore dedicated for LDAPS is located in `/etc/xdI/.keystore`. Affected registry keys include:

```plaintext
<table>
<thead>
<tr>
<th>Registry Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HKLM\Software\Citrix\VirtualDesktopAgent\ListOfLDAPServers</td>
<td>List of LDAPServers</td>
</tr>
<tr>
<td>HKLM\Software\Citrix\VirtualDesktopAgent\ListOfLDAPServersForPolicy</td>
<td>List of LDAPServers for policy</td>
</tr>
<tr>
<td>HKLM\Software\Citrix\VirtualDesktopAgent\UseLDAPS</td>
<td>Use LDAPS</td>
</tr>
<tr>
<td>HKLM\Software\Policies\Citrix\VirtualDesktopAgent\Keystore</td>
<td>Keystore</td>
</tr>
</tbody>
</table>
```

### Configure LDAPS for third-party platform

Besides the Linux VDA components, several third-party software components that adhere to the VDA might also require secure LDAP, such as SSSD, Winbind, Centrify, and Quest. The following sections describe how to configure secure LDAP with LDAPS, STARTTLS, or SASL sign and seal.

**Tip:**

Not all of these software components prefer to use SSL port 636 to ensure secure LDAP. And most of the time, LDAPS (LDAP over SSL on port 636) cannot coexist with STARTTLS on 389.

#### SSSD

Configure the SSSD secure LDAP traffic on port 636 or 389 as per the options. For more information, see the SSSD LDAP Linux man page.

#### Winbind

The Winbind LDAP query uses the ADS method. Winbind supports only the StartTLS method on port 389. Affected configuration files are `ldap.conf` and `smb.conf`. Change the files as follows:

```plaintext
<table>
<thead>
<tr>
<th>File</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>ldap.conf</td>
<td>Modify the options as necessary.</td>
</tr>
<tr>
<td>smb.conf</td>
<td>Modify the options as necessary.</td>
</tr>
</tbody>
</table>
```
Alternately, secure LDAP can be configured by SASL GSSAPI sign and seal, but it cannot coexist with TLS/SSL. To use SASL encryption, change the `smb.conf` configuration:

```
# smb.conf:
ldap_ssl = off
ldap_ssl ads = no
client ldap_sasl.wrapping = seal
```

**Centrify**

Centrify does not support LDAPS on port 636. However, it does provide secure encryption on port 389. For more information, see the [Centrify site](#).

**Quest**

Quest Authentication Service does not support LDAPS on port 636, but it provides secure encryption on 389 using a different method.

**Troubleshooting**

The following issues might arise when you use this feature:

- **LDAPS service availability**
  
  Verify that the LDAPS connection is available on the AD/LDAP server. The port is on 636 by default.

- **Linux VDA registration failed when LDAPS is enabled**
  
  Verify that the LDAP server and ports are configured correctly. Check the Root CA Certificate first and ensure that it matches the AD/LDAP server.
• **Incorrect registry change by accident**

If the LDAPS related keys were updated by accident without using `enable_ldaps.sh`, it might break the dependency of LDAPS components.

• **LDAP traffic is not encrypted through SSL/TLS from Wireshark or any other network monitoring tools**

By default, LDAPS is disabled. Run `/opt/Citrix/VDA/sbin/enable_ldaps.sh` to force it.

• **There is no LDAPS traffic from Wireshark or any other networking monitoring tool**

LDAP/LDAPS traffic occurs when Linux VDA registration and Group Policy evaluation occur.

• **Failed to verify LDAPS availability by running ldp connect on the AD server**

Use the AD FQDN instead of the IP Address.

• **Failed to import Root CA certificate by running the `/opt/Citrix/VDA/sbin/enable_ldaps.sh` script**

Provide the full path of the CA certificate, and verify that the Root CA Certificate is the correct type. Generally speaking, it is supposed to be compatible with most of the Java Keytool types supported. If it is not listed in the support list, you can convert the type first. Citrix recommends the base64 encoded PEM format if you encounter a certificate format problem.

• **Failed to show the Root CA certificate with Keytool -list**

When you enable LDAPS by running `/opt/Citrix/VDA/sbin/enable_ldaps.sh`, the certificate is imported to `/etc/xdl/.keystore`, and the password is set to protect the keystore. If you forget the password, you can rerun the script to create a keystore.

---

**Configure Xauthority**

January 11, 2019

Environments that use interactive remoting using X11 display functionality (including xterm and gvim) are supported by the Linux VDA. This feature provides a security mechanism necessary to ensure secure communication between XClient and XServer.

There are two methods to secure permission for this secure communication:

• **Xhost**. By default, Xhost allows only the localhost XClient to communicate with XServer. If you choose to allow a remote XClient to access XServer, the Xhost command must be executed to grant permission on the specific machine. Or, you can alternately use `xhost +` to allow any XClient to connect to XServer.
• **Xauthority.** The .Xauthority file can be found in each user’s home directory. It is used to store credentials in cookies used by xauth for authentication of XServer. When an XServer instance (Xorg) is started, the cookie is used to authenticate connections to that specific display.

**How it works**

When Xorg starts up, a .Xauthority file is passed to the Xorg. This .Xauthority file contains the following elements:

- Display number
- Remote request protocol
- Cookie number

You can browse this file using the `xauth` command. For example:

```bash
# xauth -f ~/.Xauthority
# > list
# > us01msip06:107  MIT-MAGIC-COOKIE-1
  fb228d1b695729242616c5908f11624b
```

If XClient connects to the Xorg remotely, two prerequisites must be met:

- Set the `DISPLAY` environment variable to the remote XServer.
- Get the .Xauthority file which contains one of the cookie numbers in Xorg.

**Configure Xauthority**

To enable Xauthority on the Linux VDA for remote X11 display, you must create the following two registry keys:

```bash
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet\Control\Citrix\Xorg" -t "REG_DWORD" -v "XauthEnabled" -d "0x00000001" --force
```

After enabling Xauthority, pass the .Xauthority file to XClient manually or by mounting a shared home directory:
Linux Virtual Delivery Agent 1903

- **Pass the .Xauthority file to XClient manually**

After launching an ICA session, the Linux VDA generates the .Xauthority file for the XClient and stores the file in the logon user’s home directory. You can copy this .Xauthority file to the remote XClient machine, and set the DISPLAY and XAUTHORITY environment variables. DISPLAY is the display number stored in the .Xauthority file and XAUTHORITY is the file path of Xauthority. For an example, see the following command:

```
1 export DISPLAY={
2   Display number stored in the Xauthority file }
3
4 export XAUTHORITY={
5   the file path of .Xauthority }
```

**Note:**
If the XAUTHORITY environment variable is not set, the ~/.Xauthority file is used by default.

- **Pass the .Xauthority file to XClient by mounting a shared home directory**

The convenient way is to mount a shared home directory for the logon user. When the Linux VDA starts an ICA session, the .Xauthority file is created under the logon user’s home directory. If this home directory is shared with XClient, the user does not need to transmit this .Xauthority file to XClient manually. After the DISPLAY and XAUTHORITY environment variables are set correctly, the GUI is displayed in XServer desktop automatically.

**Troubleshooting**

If Xauthority does not work, follow the troubleshooting steps:

1. As an administrator with root privilege, retrieve all of Xorg cookies:

```
1 ps aux | grep -i xorg
```

This command displays the Xorg process and the parameters passed to Xorg while starting. Another parameter displays which .Xauthority file is used. For example:

```
1 /var/xdl/xauth/.Xauthority110
```

Display the cookies using the **Xauth** command:

```
1 Xauth -f /var/xdl/xauth/.Xauthority110
```

2. Use the **Xauth** command to show the cookies contained in ~/.Xauthority. For the same display number, the displayed cookies must be the same in the .Xauthority files of Xorg and XClient.
3. If the cookies are the same, check the remote display port accessibility by using the IP address of the Linux VDA (for example, 10.158.11.11) and the published desktop display number (for example, 160).

Run the following command on the XClient machine:

```bash
telnet 10.158.11.11 6160
```

The port number is the sum of 6000 + <display number>.

If this telnet operation fails, the firewall might be blocking the request.

---

**Federated Authentication Service**

March 25, 2019

**Overview**

The Citrix Federated Authentication Service (FAS) is a privileged component designed to integrate with Active Directory Certificate Services. It dynamically issues certificates for users, allowing them to log on to an Active Directory environment as if they had a smart card. This functionality allows StoreFront to use a broader range of authentication options, such as Security Assertion Markup Language (SAML) assertions. SAML is commonly used as an alternative to traditional Windows user accounts on the Internet.

The following diagram shows the Federated Authentication Service integrated with a Microsoft Certification Authority and providing support services to StoreFront and VDAs.
Trusted StoreFront servers contact the FAS when users request access to the Citrix environment. The FAS grants a ticket that allows a single Citrix Virtual Apps or Citrix Virtual Desktops session to authenticate with a certificate for that session. When a VDA needs to authenticate a user, it connects to the FAS and redeems the ticket. Only the FAS has access to the user certificate’s private key. The VDA must send each signing and decryption operation that it needs to perform with the certificate to the FAS.

Requirements

The Federated Authentication Service is supported on Windows servers (Windows Server 2008 R2 or later).

- Citrix recommends installing the FAS on a server that does not contain other Citrix components.
- The Windows Server must be secured. It will have access to a registration authority certificate and private key that allows it to automatically issue certificates for domain users, and it will have access to those user certificates and private keys.

In the Citrix Virtual Apps or Citrix Virtual Desktops Site:

- The Delivery Controllers must be minimum version 7.9.
- The StoreFront server must be minimum version 3.6 (version provided with the XenApp and XenDesktop 7.9 ISO).
- The Linux VDAs must be minimum version 7.18. Check that the Federated Authentication Service Group Policy configuration has been applied correctly to the VDAs before creating the Machine Catalog in the usual way. For more information, see the Configure Group Policy section in this article.

References:
Configure Windows for Certificate Logon

For information on configuring Windows for certificate logon, open Knowledge Center article CTX206156 to download and read the Smart_card_config_Citrix_Env.pdf file (hereinafter referred to as “the PDF file”). Perform the following steps according to the PDF file while noting the differences or complements that are given in each step. Pay special attention to the target machine you are operating on, for example, the AD, Delivery Controller, or StoreFront.

Set up a Windows domain (on AD)

Install domain controller roles

See the Installing Domain Controller Roles section of the PDF file.

During installation of the Active Directory Certificate Services, ensure that the following options are selected as shown below:
Select role services

Before You Begin
Installation Type
Server Selection
Server Roles
Features
AD CS
Role Services

Select the role services to install for Active Directory Certificate Services

Role services
- Certification Authority
- Certificate Enrollment Policy Web Service
- Certificate Enrollment Web Service
- Certification Authority Web Enrollment
- Network Device Enrollment Service
- Online Responder

Description
Certification Authority Web Enrollment provides a simple Web interface that allows users to perform tasks such as request and renew certificates, retrieve certificate revocation lists (CRLs), and enroll for smart card certificates.

Configure Active Directory Certificate Services on the...
Open http://localhost/certsrv/ to check whether it displays the following welcome page. If yes, the Active Directory Certificate Services are installed successfully.
**Prepare the Certificate Authority for smart card usage**

No complement. See the *Preparing the Certificate Authority for Smart card usage* section of the PDF file.

**Issue a Domain Controller certificate**

No complement. See the *Issuing a Domain Controller Certificate* section of the PDF file.

**Configure Microsoft IIS for HTTPS (on StoreFront)**

**Configure HTTPS on Microsoft IIS**

No complement. See the *Configuring HTTPS on Microsoft IIS* section of the PDF file.

**Non-domain joined computers**

See the *Non-Domain Joined Computers* section of the PDF file.

**Retrieve the CA Certificate from the Microsoft CA (on AD)**

No complement. See the *Retrieving the CA Certificate from the Microsoft CA* section of the PDF file.

**Install the trusted CA certificate on Windows**

No complement. See the *Installing the Trusted CA Certificate on Windows* section of the PDF file.

**Configure Citrix StoreFront (on StoreFront)**

See the *Creating the Store* section of the PDF file.

After the preceding IIS configuration, the base URL of the common store is forcibly set to https:// rather than http://. Because FAS does not share the store with smart cards, a new store is needed for FAS. The Linux VDA FAS is compatible with any StoreFront authentication methods. For example, the FAS store can be configured to use passwords or SAML, but cannot use both at the same time. When SAML is selected, the URL of StoreFront automatically redirects to IdP and the password authentication method is ignored.
Start Internet Explorer and open the URL of the FAS store (for example, https://mzgwy-ddc.xd.local/Citrix/FASWeb).

**Note**: The URL of the FAS store must have **Web** appended.
Install and set up FAS

The installation and setup process consists of the following steps:

1. Install the Federated Authentication Service
2. Enable the Federated Authentication Service plug-in on StoreFront servers
3. Configure Group Policy
4. Use the Federated Authentication Service administration console to: (a) Deploy the provided templates, (b) Set up certificate authorities, and (c) Authorize the Federated Authentication Service to use your certificate authority
5. Configure user rules

For instructions on each of the steps, see Federated Authentication Service. Note the following differences or complements in each of the steps. Pay special attention to the target machine you operating on, for example, the AD, Delivery Controller, StoreFront, or the FAS server.

Install the Federated Authentication Service (on the FAS server)

For security, Citrix recommends that the FAS be installed on a dedicated server that is secured in a similar way to a domain controller or certificate authority.

Enable the Federated Authentication Service plug-in on a StoreFront store (on StoreFront)

Ensure that the following command uses the same FAS store name that you typed when configuring StoreFront. For example, FAS is the store name in this example:

$StoreVirtualPath = “/Citrix/FAS”

Configure the Delivery Controller (on Delivery Controller)

To use the Federated Authentication Service, configure the Delivery Controller to trust the StoreFront servers that can connect to it: run the Set-BrokerSite -TrustRequestsSentToTheXmlServicePort $true PowerShell cmdlet. Sometimes, you might need to run asnp citrix* first.

Configure Group Policy (on the FAS server and on the AD)

You must be an administrator to be able to perform Steps 1–7 in this section. Step 1 must be done on the FAS server and Steps 2–7 must be done on the AD.

After you complete Steps 1–7, check in the Registry Editor of the FAS server to confirm that the FAS policy has been set.
Enable in-session certificate support

The Linux VDA does not support in-session certificates.

Use the Federated Authentication Service administration console (on the FAS server)

No complement. See the Federated Authentication Service article.

Deploy certificate templates (on the FAS server)

No complement. See the Federated Authentication Service article.

Set up Active Directory Certificate Services (on the FAS server)

No complement. See the Federated Authentication Service article.

Authorize the Federated Authentication Service (on the FAS server)

No complement. See the Federated Authentication Service article.
Configure user rules (on the FAS server)

No complement. See the Federated Authentication Service article.

For more information, see also the Delegated Enrollment Agents and Access Control List configuration parts in the Security considerations section of the Federated Authentication Service article.

Federated Authentication Service ADFS deployment

For information on how to deploy the ADFS IdP for Federated Authentication Service, see Federated Authentication Service ADFS deployment.

Configure the Linux VDA

Set FAS servers

For fresh Linux VDA installation, to use FAS, type the FQDN of each FAS server when you are asked for CTX_XDL_FAS_LIST during the execution of ctxinstall.sh or ctxsetup.sh. Because the Linux VDA does not support AD Group Policy, you can provide a semicolon-separated list of FAS servers instead. If any server address is removed, fill its blank with the <none> text string and keep the sequence of server addresses without any changes.

For upgrading an existing Linux VDA installation, you can rerun ctxsetup.sh to set the FAS servers. Or you can run the following commands to set the FAS servers and to restart the ctxvda service to make your setting take effect.

```
1  sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\VirtualDesktopAgent\Authentication\UserCredentialService" -t "REG_SZ" -v "Addresses" -d "<Your-FAS-Server-List>" -force
2 3  service ctxvda restart
```

To update the FAS servers through ctxreg, run the following commands:

```
1  sudo /opt/Citrix/VDA/bin/ctxreg update -k "HKLM\Software\Citrix\VirtualDesktopAgent\Authentication\UserCredentialService" -v "Addresses" -d "<Your-FAS-Server-List>"
2 3  service ctxvda restart
```
Install a root CA certificate

For the verification of users’ certificates, install the root CA certificate on the VDA. You can obtain the AD root certificate from the preceding Retrieve the CA Certificate from the Microsoft CA (on AD) step, or download its DER format from the root CA server http://CA-SERVER/certsrv.

Convert a DER file (.crt, .cer, .der) to PEM by running the command similar to the following:

```
sudo openssl x509 -inform der -in root.cer -out root.pem
```

Then, install the root CA certificate to the openssl directory by running the command similar to the following:

```
sudo cp root.pem /etc/pki/CA/certs/
```

**Note:**
Do not put the root CA certificate under the /root path. Otherwise, FAS does not have the read permission to the root CA certificate.

Configure FAS

Run the following script to configure FAS parameters:

```
sudo /opt/Citrix/VDA/sbin/ctxfascfg.sh
```

Two environment variables are added so that ctxfascfg.sh can be run in silent mode:

- **CTX_FAS_ADINTEGRATIONWAY=winbind | sssd | centrify** – Denotes the Active Directory integration method, which equals to CTX_EASYINSTALL_ADINTEGRATIONWAY when CTX_EASYINSTALL_ADINTEGRATIONWAY is specified. If CTX_EASYINSTALL_ADINTEGRATIONWAY is not specified, CTX_FAS_ADINTEGRATIONWAY uses its own value setting.

- **CTX_FAS_ROOT_CA_PATH - <root_CA_certificate>** – Specifies the full path of the root CA certificate.

Choose the correct Active Directory integration method and then type the correct path of the root CA certificate (for example, /etc/pki/CA/certs/root.pem).

The script then installs the krb5-pkinit and pam_krb5 packages and sets the relevant configuration files.

**Limitation**

- FAS supports limited platforms and AD integration methods, see the following matrix:
<table>
<thead>
<tr>
<th></th>
<th>Winbind</th>
<th>SSSD</th>
<th>Centriy</th>
</tr>
</thead>
<tbody>
<tr>
<td>RHEL 7.5</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Ubuntu 16.04 (4.13 kernel)</td>
<td>√</td>
<td>×</td>
<td>√</td>
</tr>
<tr>
<td>SLES 12.3</td>
<td>√</td>
<td>×</td>
<td>√</td>
</tr>
</tbody>
</table>

- FAS does not support lock screen yet. If you click the lock button in a session, you cannot log back on to the session again by using FAS.
- This release supports only the common FAS deployments summarized in the Federated Authentication Service architectures overview article, of which, Windows 10 Azure AD Join is excluded.

**Troubleshooting**

Before troubleshooting FAS, ensure that the Linux VDA is installed and configured correctly so that a non-FAS session can be launched successfully on the common store by using password authentication.

If there is nothing wrong with non-FAS sessions, set the HDX log level of the Login class to VERBOSE and the VDA log level to TRACE. For information on how to enable trace logging for the Linux VDA, see Knowledge Center article CTX220130.

**FAS server configuration error**

Launching a session from the FAS store fails and a window appears as shown below:

![Error window](image)

Check /var/log/xdl/hdx.log and find the error log similar to the following:

```
```
Solution

Run the following command to verify that the Citrix registry value "HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\VirtualDesktopAgent\Authentication\UserCredentialService" is set to <Your-FAS-Server-List>.

```bash
sudo /opt/Citrix/VDA/bin/ctxreg dump | grep "UserCredentialService"
```

If the existing setting is incorrect, follow the preceding Set FAS servers step to set it again.

Incorrect root CA certificate configuration

Launching a session from the FAS store fails. A gray window appears and disappears several seconds later.

Check /var/log/xdl/hdx.log and find the error log similar to the following:
<table>
<thead>
<tr>
<th>Time</th>
<th>Event</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>2018-03-27</td>
<td>citrix-ctxlogin: validate_user: pam_authenticate</td>
<td>err, can retry for user <a href="mailto:user1@CTXFAS.LAB">user1@CTXFAS.LAB</a></td>
</tr>
<tr>
<td></td>
<td></td>
<td>closing session and pam transaction</td>
</tr>
<tr>
<td></td>
<td></td>
<td>citrix-ctxlogin: validate_user: Exit</td>
</tr>
<tr>
<td></td>
<td></td>
<td>citrix-ctxlogin: LoginBoxValidate:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>citrix-ctxlogin: Audit_login_failure</td>
</tr>
</tbody>
</table>

**Solution**

Verify that the full path of the root CA certificate is set correctly in `/etc/krb5.conf`. The full path is similar to the following:

```
[realms]

EXAMPLE.COM = {
    ...... 
    pkinit_anchors = FILE:/etc/pki/CA/certs/root.pem 
    ...... 
}
```

If the existing setting is incorrect, follow the preceding **Install a root CA certificate** step to set it again. Alternatively, check whether the root CA certificate is valid.

**Shadow account mapping error**

FAS is configured by SAML authentication. The following error might appear after an ADFS user types the user name and password on the ADFS logon page.
This error indicates that the ADFS user has been verified successfully, but there is no shadow user configured on AD.

**Solution**

Set the Shadow Account on AD.

**ADFS not configured**

The following error appears during logon to the FAS store:

The cause is that the FAS store is configured to use SAML authentication while the ADFS deployment is missing.

**Solution**

Deploy the ADFS IdP for Federated Authentication Service. For more information, see Federated Authentication Service ADFS deployment.

**Related information**

- The common FAS deployments are summarized in the Federated Authentication Service architectures overview article.
- “How-to” articles are introduced in the Federated Authentication Service configuration and management article.
Known issue

When FAS is in use, you can fail when trying to launch a published desktop or app session with non-English characters.

Workaround

Right-click Manage Templates in the CA tool to change the Citrix_SmartcardLogon template from Build from this Active Directory information to Supply in the request as shown below:
Citrix Smartcard Logon Properties

- Supply in the request
- Build from the Active Directory information

Subject name format:
- None
- Include e-mail name in subject name
- Include this information in alternate subject name:
  - E-mail name
  - DNS name
  - User principal name (UPN)
  - Service principal name (SPN)

* Control is disabled due to compatibility settings.
Locations
Corporate Headquarters | 851 Cypress Creek Road Fort Lauderdale, FL 33309, United States
Silicon Valley | 4988 Great America Parkway Santa Clara, CA 95054, United States

© 2019 Citrix Systems, Inc. All rights reserved. Citrix, the Citrix logo, and other marks appearing herein are property of Citrix Systems, Inc. and/or one or more of its subsidiaries, and may be registered with the U.S. Patent and Trademark Office and in other countries. All other marks are the property of their respective owner(s).