Linux Virtual Delivery Agent 7.15

Contents

What's new

September 21, 2022

Release date: July 07, 2022

What's new in 7.15

Cumulative Update 9 (CU9) is the latest release of the Linux VDA 7.15 LTSR. CU9 adds one fix compared to the Linux VDA 7.15 CU8.

PDF printing

Previously available as an experimental feature, PDF Printing is a fully supported feature in this release. It allows users of Citrix Receivers for Chrome and HTML5 to print PDFs converted from within their Linux VDA sessions.

System behavior change

As of this release, you do not have to run the ctxsetup.sh script after upgrading your Linux VDA.

Fixed issues

July 8, 2022

Fixed issues in CU9

- Uninstalling Linux VDAs on SUSE or RHEL might not delete empty folders in the /opt/Citrix/ location. [CVADHELP-18241]

Fixed issues in CU8

- With channel binding enabled, attempts to register a Linux VDA with the Delivery Controller might fail. [CVADHELP-14481]

Fixed issues in CU6

- A Linux session might become unresponsive if the mouse and the keyboard are not focused in the same window or the mouse fails to change focus. [CVADHELP-12768]
- Attempts to generically redirect a removable USB drive to a Linux VDA might fail. The issue occurs when the USB drive is NTFS (New Technology File System) formatted. [CVADHELP-13675]
- Linux VDAs might fail to reach frames per second as specified in the **Target frame rate** (FramesPerSecond) setting. The issue occurs when a GPU is installed on a Linux VDA. [CVADHELP-14267]

Fixed issues in CU5

- Attempts to copy and paste content between a client and a session using the clipboard feature might fail. [LD2047]
- When you launch a session on a Linux VDA and perform an action, the session might disconnect. [LD2257]

Fixed issues in CU4

- When you attempt to copy content from an endpoint and paste it into an application that is running on a Linux VDA, the content might not be copied. [LC8760]
- The keyboard might not work on a SUSE Linux Enterprise Server 11 Service Pack 4. As a result, the keystrokes are not shown on the screen and the keyboard layout is not set correctly. [LC9906]
- The **ctxctl** process might fail to run in a user session on a Linux VDA. [LD0353]

Fixed issues in CU3

- The Linux VDA might fail to apply Citrix policies. The issue occurs when you configure a policy to use the Access Control element connection type with NetScaler Gateway. [LC9842]

Fixed issues in CU2

- Registration of a Linux VDA using the Delivery Controller might fail intermittently. [LC7982]
- Citrix Director 7.13 that is running on a Red Hat Enterprise Linux Server 7.3 might not show the session details of the machine. The following error message appears:

  Cannot retrieve the data. [LC8204]

- A Linux VDA might register with the Delivery Controller and unregister after some time. [LC8205]
- Certain third-party applications that are used to check the session display of a Linux VDA might not display all pixels. [LC8419]
- When there are multiple LDAP Servers, attempts to launch an application on a Linux VDA might fail after policies are updated and a session times out. [LC8444]
- The ctxhdx process might exit unexpectedly with a **segfault** error when the session is connected to a Linux VDA. [LC8611]
- When using the Linux VDA 7.16 Early Access Release, the broker agent might fail to get the application name. This failure causes Director to display the error **Agent Requested**, after which re-registration starts. [LC9243]

Fixed issues in CU1

- A Linux VDA might register with the Delivery Controller and unregister after some time. [LC8205]
- Certain third-party applications that are used to check the session display of a Linux VDA might not display all pixels. [LC8419]
- When there are multiple LDAP Servers, attempts to launch an application on a Linux VDA might fail after policies are updated and a session times out. [LC8444]

Fixed issues In 7.15 LTSR

The following issues have been resolved in this release of the Linux VDA:

- Easy install might cause the Linux VDA to disconnect from the network when you enter the DNS IP address. [LNXVDA-2152]
- While playing a video, session roaming from Citrix Receiver for Windows to Citrix Receiver for Android fails. [LNXVDA-2164]

Known issues

July 7, 2022

The following issues have been identified in this release:

- Citrix Scout integrated with XenApp and XenDesktop 7.15 LTSR CU6 cannot collect logs from the Linux VDA 7.15. The Linux VDA 7.15 does not support the Citrix Telemetry Service that Citrix Scout uses to collect logs.

- 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 Desktop 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 Receiver 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 Receiver 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 Receiver for Mac. Citrix Receiver for Mac sends AltGr for both left and right Options/Alt keys by default. You can modify this behavior within the Citrix Receiver 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 command on the Broker from a command prompt as Administrator:

  ```
  klist -li 0x3e4 purge
  <!--NeedCopy-->
  ```

  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 potentially 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 machine 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 Receiver for Windows might experience audio distortion during audio recording.

Third party notices

October 20, 2021

Linux Virtual Desktop Version 7.15 (PDF Download)

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

System requirements

March 30, 2021

Linux distributions

The Linux VDA supports the following Linux distributions:

- SUSE Linux Enterprise:
  - Desktop 12 Service Pack 2
  - Server 12 Service Pack 2
  - Server 11 Service Pack 4
- Red Hat Enterprise Linux
  - Workstation 7.3
  - Workstation 6.9
  - Workstation 6.6
  - Server 7.3
  - Server 6.9
  - Server 6.6
- CentOS Linux
  - CentOS 7.3

  - CentOS 6.6

- Ubuntu Linux
  - Ubuntu Desktop 16.04 (with the 4.4.x kernel)
  - Ubuntu Server 16.04 (with the 4.4.x kernel)

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.

Linux distribution	Xorg version
RHEL 7.3, CentOS 7.3	1.17
RHEL 6.9	1.17
RHEL 6.6, CentOS 6.6	1.15
Ubuntu 16.04	1.18
SUSE 12.2	1.18
SUSE 11.4	1.6.5

Do not use HWE Xorg server 1.19 on Ubuntu 16.04.

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

> **Note:**
>
> Citrix' 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 only. At least one desktop must be installed.

XenDesktop

The Linux VDA is compatible with all currently supported versions of XenDesktop. For information about the XenDesktop 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.

> **Note:**
>
> The Linux VDA is incompatible with XenDesktop Version 7.0 or earlier.

Citrix Receiver

The following versions of Citrix Receiver are supported:

- Citrix Receiver for Universal Windows Platform Version 1.0
- Citrix Receiver for Windows Version 4.8 or later
- Citrix Receiver for Linux Version 13.5
- Citrix Receiver for Mac OSX Version 12.6
- Citrix Receiver for Android Version 3.11
- Citrix Receiver for iOS Version 7.2
- Citrix Receiver for Chrome Version 2.5
- Citrix Receiver for HTML5 Version 2.5 (only through Access Gateway)

Hypervisors

The following hypervisors for hosting Linux VDA guest VMs are supported:

- XenServer
- VMware ESX and ESXi
- Microsoft Hyper-V
- Nutanix AHV

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 and 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, Linux distributions, and NVIDIA GRID™ GPU are required to support HDX 3D Pro.

Hypervisors

The following hypervisors are supported:

- XenServer
- VMware ESX and ESXi
- Nutanix AHV

Linux distributions

The following Linux distributions support HDX 3D Pro:

- Red Hat Enterprise Linux - Workstation 7.3
- Red Hat Enterprise Linux - Server 7.3
- Red Hat Enterprise Linux - Workstation 6.9
- Red Hat Enterprise Linux - Server 6.9
- Red Hat Enterprise Linux - Workstation 6.6
- Red Hat Enterprise Linux - Server 6.6
- SUSE Linux Enterprise Desktop 12 Service Pack 2
- SUSE Linux Enterprise Server 12 Service Pack 2
- Ubuntu Linux Desktop 16.04
- Ubuntu Linux Server 16.04

GPU

The following GPUs are supported for GPU pass-through:

- NVIDIA GTX750Ti
- NVIDIA GRID - Tesla M60
- NVIDIA GRID - K2

The following GPUs are supported for vGPU:

- NVIDIA GRID - Tesla M60
- NVIDIA GRID - Tesla M10

Installation overview

January 16, 2019

Installing the Linux Virtual Delivery Agent (VDA) follows the same general steps for all supported Linux distributions.

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 XenApp or XenDesktop.
7. Create the delivery group in XenApp or XenDesktop.

Variations and specific commands are documented by distribution.

Easy install

June 10, 2022

Easy install is officially supported as of Version 7.13 of the Linux VDA. Easy install helps you set up the running environment of the Linux VDA by installing the necessary packages and customizing the configuration files automatically.

Supported distributions

	Winbind	SSSD	Centrify
RHEL 7.3	Yes	Yes	Yes
RHEL 6.9	Yes	Yes	Yes
RHEL 6.6	Yes	Yes	Yes

	Winbind	SSSD	Centrify
CentOS 7.3	Yes	Yes	Yes
Ubuntu 16.04	Yes	Yes	Yes
SUSE 12.2	Yes	No	Yes

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.

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:


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

For Ubuntu distributions:

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

For SUSE distributions:

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

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.

```
sudo /opt/Citrix/VDA/sbin/ctxinstall.sh
<!--NeedCopy-->
```

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 (ports 80 and 1494 by default) in the system firewall for the Linux Virtual Desktop.
- **CTX_XDL_HDX_3D_PRO**=Y | N –The Linux VDA supports HDX 3D Pro, a set of GPU acceleration technologies designed to optimize the virtualization of rich graphics applications. If HDX 3D Pro is selected, the VDA 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 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 improve 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_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.

```
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

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_START_SERVICE=Y | N

sudo -E /opt/Citrix/VDA/sbin/ctxinstall.sh
<!--NeedCopy-->
```

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_EASYINSTALL_HOSTNAME=host-name \

CTX_EASYINSTALL_DNS=ip-address-of-dns \

CTX_EASYINSTALL_NTPS=address-of-ntps \

```



```
CTX_EASYINSTALL_DOMAIN=domain-name \

CTX_EASYINSTALL_REALM=realm-name \

......

CTX_XDL_SEARCH_BASE=search-base-set \

CTX_XDL_START_SERVICE=Y \

/opt/Citrix/VDA/sbin/ctxinstall.sh
<!--NeedCopy-->
```

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.

- For Centrify on CentOS, easy install can fail at `adcheck`, the Centrify environment check tool, and report the following error:

  Log:

  ```
  ADSITE    : Check that this machine's subnet is in a site known by
      AD    : Failed
              : This machine's subnet is not known by AD.
              : We guess you should be in the site Site1.
  <!--NeedCopy-->
  ```

  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**.

- 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:

RHEL: wget http://edge.centrify.com/products/centrify-suite/2016-update-1/installers/centrify-suite-2016.1-rhel4-x86_64.tgz?_ga=1.178323680.558673738.1478847956

CentOS: wget http://edge.centrify.com/products/centrify-suite/2016-update-1/installers/centrify-suite-2016.1-rhel4-x86_64.tgz?_ga=1.186648044.558673738.1478847956

SUSE: wget http://edge.centrify.com/products/centrify-suite/2016-update-1/installers/centrify-suite-2016.1-suse10-x86_64.tgz?_ga=1.10831088.558673738.1478847956

Ubuntu: wget http://edge.centrify.com/products/centrify-suite/2016-update-1/installers/centrify-suite-2016.1-deb7-x86_64.tgz?_ga=1.178323680.558673738.1478847956

- 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:

```
cat /tmp/ctxinstall.conf
set  “ centrifypkgpath=/home/mydir ”
ls  -ls  /home/mydir
                9548 -r-xr-xr-x. 1 root root  9776688 May 13
     2016 adcheck-rhel4-x86_64
                4140 -r--r--r--. 1 root root  4236714 Apr 21
     2016 centrifyda-3.3.1-rhel4-x86_64.rpm
                33492 -r--r--r--. 1 root root 34292673 May
    13  2016 centrifydc-5.3.1-rhel4-x86_64.rpm
                4 -rw-rw-r--. 1 root root     1168 Dec  1
     2015 centrifydc-install.cfg
                756 -r--r--r--. 1 root root   770991 May 13
     2016 centrifydc-ldapproxy-5.3.1-rhel4-x86_64.rpm
                268 -r--r--r--. 1 root root   271296 May 13
     2016 centrifydc-nis-5.3.1-rhel4-x86_64.rpm
                1888 -r--r--r--. 1 root root  1930084 Apr 12
     2016 centrifydc-openssh-7.2p2-5.3.1-rhel4-x86_64.rpm
                124 -rw-rw-r--. 1 root root   124543 Apr 19
     2016 centrify-suite.cfg
                0 lrwxrwxrwx. 1 root root       10 Jul  9
     2012 install-express.sh -> install.sh
                 332 -r-xr-xr--. 1 root root   338292 Apr 10
     2016 install.sh
                12 -r--r--r--. 1 root root    11166 Apr  9
     2015 release-notes-agent-rhel4-x86_64.txt
                4 -r--r--r--. 1 root root     3732 Aug 24
     2015 release-notes-da-rhel4-x86_64.txt
                4 -r--r--r--. 1 root root     2749 Apr  7
     2015 release-notes-nis-rhel4-x86_64.txt
                12 -r--r--r--. 1 root root     9133 Mar 21
     2016 release-notes-openssh-rhel4-x86_64.txt
<!--NeedCopy-->
```


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:

```
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)'.
2016-11-04 02:11:52.362 [ERROR] - RegistrationManager.
    AttemptRegistrationWithSingleDdc: Failed to register with http://
    CTXDDC.citrixlab.local:80/Citrix/CdsController/IRegistrar. Error:
    General security error (An error occurred in trying to obtain a TGT:
     Client not found in Kerberos database (6))
2016-11-04 02:11:52.362 [ERROR] - The Citrix Desktop Service cannot
    connect to the delivery controller 'http://CTXDDC.citrixlab.local
    :80/Citrix/CdsController/IRegistrar' (IP Address '10.158.139.214')
Check the following:- The system clock is in sync between this machine
    and the delivery controller.
-  The Active Directory provider (e.g. winbind daemon) service is
    running and correctly configured.
-  Kerberos is correctly configured on this machine.
If the problem persists, please refer to Citrix Knowledge Base article
    CTX117248 for further information.
Error Details:
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'.
2016-11-04 02:11:52.362 [INFO ] - RegistrationManager.
    AttemptRegistrationWithSingleDdc: The current time for this VDA is
    Fri Nov 04 02:11:52 EDT 2016.
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.
<!--NeedCopy-->
```

/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.
4. Run /opt/Citrix/VDA/sbin/ctxinstall.sh.
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. Run the `sudo apt-get update` command.
2. Run the `sudo apt-get install unity lightdm` command.
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

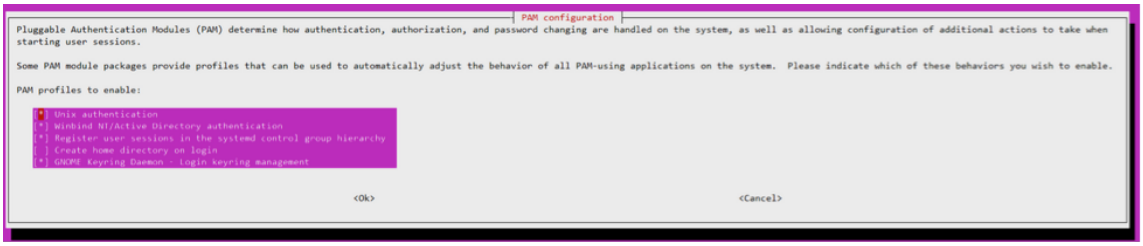
/var/log/xdl/hdx.log:

```
2016-11-02 13:21:19.015 <P22492:S1> citrix-ctxlogin: StartUserSession:
   failed to change to directory(/home/CITRIXLAB/ctxadmin) errno(2)

2016-11-02 13:21:19.017 <P22227> citrix-ctxhdx: logSessionEvent:
   Session started for user ctxadmin.

2016-11-02 13:21:19.023 <P22492:S1> citrix-ctxlogin: ChildPipeCallback:
    Login Process died: normal.

2016-11-02 13:21:59.217 <P22449:S1> citrix-ctxgfx: main: Exiting
   normally.
<!--NeedCopy-->
```


> **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):

```
Oct 27 04:17:16 CentOS7 citrix-ctxhdx[8978]: Session started for user
    CITRIXLAB\ctxadmin.

Oct 27 04:17:18 CentOS7 kernel: traps: gnome-session[19146] trap int3
    ip:7f89b3bde8d3 sp:7fff8c3409d0 error:0

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...

Oct 27 04:17:18 CentOS7 gnome-session: 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)

Oct 27 04:17:18 CentOS7 gnome-session: aborting...

Oct 27 04:17:18 CentOS7 citrix-ctxgfx[18981]: Exiting normally.

Oct 27 04:17:18 CentOS7 citrix-ctxhdx[8978]: Session stopped for user
    CITRIXLAB\ctxadmin.
<!--NeedCopy-->
```

Or, alternately for Ubuntu distributions, use the log /var/log/syslog:

```
Nov  3 11:03:52 user01-HVM-domU pulseaudio[25326]: [pulseaudio] pid.c:
    Stale PID file, overwriting.

```



```
3  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.
4
5  Nov  3 11:03:52 user01-HVM-domU pulseaudio[25326]: [pulseaudio] hashmap
       .c: Assertion 'h' failed at pulsecore/hashmap.c:116, function
       pa_hashmap_free(). Aborting.
6
7  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.
8
9  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.]
10
11 Nov  3 11:03:52 user01-HVM-domU pulseaudio[25352]: [pulseaudio] pid.c:
       Daemon already running.Nov  3 11:03:58 user01-HVM-domU citrix-ctxgfx
       [24693]: Exiting normally
12 <!--NeedCopy-->
```

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:

```
1  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
2
3  Jan 25 23:30:31 yz-rhel72-1 python[3945]: SELinux is preventing /usr/
       sbin/sshd from setattr access on the directory /root.
4
5  *****  Plugin catchall_boolean (89.3 confidence) suggests
       ******************
6
7  If you want to allow polyinstantiation to enabled
8
9     Then you must tell SELinux about this by enabling the '
       polyinstantiation_enabled' boolean.
```



```
10
11 You can read 'None' man page for more details.
12
13     Do
14
15        setsebool -P polyinstantiation_enabled 1
16
17 *****  Plugin catchall (11.6 confidence) suggests
    **************************
18
19 If you believe that sshd should be allowed setattr access on the root
    directory by default.
20
21 Then you should report this as a bug.
22
23 You can generate a local policy module to allow this access.
24
25       Do
26
27        allow this access for now by executing:
28
29        # grep sshd /var/log/audit/audit.log | audit2allow -M mypol
30
31 # semodule -i mypol.pp
32 <!--NeedCopy-->
```

To resolve this issue:

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

   SELINUX=disabled

2. Restart the VDA.

Install Linux Virtual Delivery Agent for RHEL/CentOS

June 10, 2022

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

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

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`=**hostname**

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

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

To ensure that the DNS domain name and Fully Qualified Domain Name (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.

> **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:


```
hostname
<!--NeedCopy-->
```

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

Verify that the FQDN is set correctly:

```
hostname -f
<!--NeedCopy-->
```

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:

```
nslookup domain-controller-fqdn

ping domain-controller-fqdn

nslookup delivery-controller-fqdn

ping delivery-controller-fqdn
<!--NeedCopy-->
```

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:

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

server peer2-fqdn-or-ip-address iburst
<!--NeedCopy-->
```


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:

```
sudo /sbin/service ntpd restart
<!--NeedCopy-->
```

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:

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

server peer2-fqdn-or-ip-address iburst
<!--NeedCopy-->
```

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:

```
sudo /sbin/service chronyd restart
<!--NeedCopy-->
```

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:

```
sudo yum info java-1.8.0-openjdk
<!--NeedCopy-->
```

- RHEL 6/CentOS 6:

```
sudo yum info java-1.7.0-openjdk
<!--NeedCopy-->
```

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

- RHEL 7/CentOS 7:

```
sudo yum -y update java-1.8.0-openjdk
<!--NeedCopy-->
```

- RHEL 6/CentOS 6:

```
sudo yum -y update java-1.7.0-openjdk
<!--NeedCopy-->
```

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:

```
java -version
<!--NeedCopy-->
```

> **Tip:**
>
> To avoid problems, 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 on 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:

```
sudo yum -y install postgresql-server

sudo yum -y install postgresql-jdbc
<!--NeedCopy-->
```

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 PostgreSQL 9:

- RHEL 7 only: PostgreSQL 9

```
sudo postgresql-setup initdb
<!--NeedCopy-->
```

- RHEL 6 only: PostgreSQL 8


```
sudo /sbin/service postgresql initdb
<!--NeedCopy-->
```

Step 1i: Start PostgreSQL

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

- RHEL 7 only: PostgreSQL 9

```
sudo systemctl enable postgresql

sudo systemctl start postgresql
<!--NeedCopy-->
```

- RHEL 6 only: PostgreSQL 8

```
sudo /sbin/chkconfig postgresql on

sudo /sbin/service postgresql start
<!--NeedCopy-->
```

Check the version of PostgreSQL by using:

```
psql --version
<!--NeedCopy-->
```

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

```
sudo -u postgres psql -c 'show data_directory'
<!--NeedCopy-->
```

> **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:

```
su -
cat /proc/sys/xen/independent_wallclock
<!--NeedCopy-->
```

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:

```
sudo echo 1 > /proc/sys/xen/independent_wallclock
<!--NeedCopy-->
```

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:

```
su -

cat /proc/sys/xen/independent_wallclock
<!--NeedCopy-->
```

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:

```
sudo yum -y install samba-winbind samba-winbind-clients krb5-
    workstation authconfig oddjob-mkhomedir
<!--NeedCopy-->
```

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

```
sudo /sbin/chkconfig winbind on
<!--NeedCopy-->
```

Configure Winbind Authentication Configure the machine for Kerberos authentication by using Winbind:

```
sudo authconfig --disablecache --disablesssd --disablesssdauth --
    enablewinbind --enablewinbindauth --disablewinbindoffline --
    smbsecurity=ads --smbworkgroup=domain --smbrealm=REALM --krb5realm=
    REALM --krb5kdc=fqdn-of-domain-controller --winbindtemplateshell=/
    bin/bash --enablemkhomedir --updateall
<!--NeedCopy-->
```

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:

```
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:

```
sudo net ads join REALM -U user
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

> **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
<!--NeedCopy-->
```

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

```
sudo net ads info
<!--NeedCopy-->
```


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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

Examine the account details of the machine using:

```
sudo net ads status
<!--NeedCopy-->
```

Verify user authentication Use the **wbinfo** tool to verify that domain users can authenticate with the domain:

```
wbinfo --krb5auth=domain\\username%password
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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


```
klist
<!--NeedCopy-->
```

Exit the session:

```
exit
<!--NeedCopy-->
```

A similar test can be performed by logging on to the Gnome or KDE console directly. Proceed to Step 4: Install the Linux VDA after the domain joining verification.

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:


```
reboot
<!--NeedCopy-->
```

> **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.

```
sudo /opt/quest/bin/vastool configure vas vasd auto-ticket-renew-
   interval 32400

sudo /opt/quest/bin/vastool configure vas vas_auth allow-disconnected-
   auth false
<!--NeedCopy-->
```

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:

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

sudo /opt/quest/bin/vastool configure nss
<!--NeedCopy-->
```

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

```
sudo /opt/quest/bin/vastool -u user join domain-name
<!--NeedCopy-->
```

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:


```
sudo /opt/quest/bin/vastool info domain
<!--NeedCopy-->
```

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.

```
ssh localhost -l domain\\username

id -u
<!--NeedCopy-->
```

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

```
ls /tmp/krb5cc_uid
<!--NeedCopy-->
```

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

```
/opt/quest/bin/vastool klist
<!--NeedCopy-->
```

Exit the session:

```
exit
<!--NeedCopy-->
```

A similar test can be performed by logging on to the Gnome or KDE console directly. Proceed to Step 4: Install the Linux VDA after the domain joining verification.

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:

```
su -
adjoin -w -V -u user domain-name
<!--NeedCopy-->
```


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:

```
su -

adinfo
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

Test connectivity to the various Active Directory and Kerberos services. Proceed to Step 4: Install the Linux VDA after the domain joining verification.

```
adinfo --test
<!--NeedCopy-->
```

SSSD

If you are using SSSD, follow the instructions in this section. This section includes instructions for joining a Linux VDA machine to a Windows domain and provides guidance for configuring Kerberos authentication.

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 7.3 or later/CentOS 7.3 or later
- Linux VDA Version 1.3 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. You can use `adcli`, `realmd`, `Winbind`, or `Samba` instead.

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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:

```
authconfig --enablesssd --enablesssdauth --enablemkhomedir – -update

sudo service sssd start

sudo chkconfig sssd on
<!--NeedCopy-->
```

Verify Kerberos configuration Check that the system **keytab** file has been created and contains valid keys:

```
sudo klist -ke
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

Verify user authentication Use the **getent** command to verify that the logon format is supported and the NSS works:

```
sudo getent passwd DOMAIN\\username
<!--NeedCopy-->
```

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 ssh localhost  – l DOMAIN\\username

id -u
<!--NeedCopy-->
```

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

```
ls /tmp/krb5cc_{
 uid }

<!--NeedCopy-->
```

Check that the tickets in the user's Kerberos credential cache are valid and not expired. Proceed to Step 4: Install the Linux VDA after the domain joining verification.

```
klist
<!--NeedCopy-->
```

Step 4: Install the Linux VDA

Step 4a: Uninstall the old version

If you have previously installed an earlier version of the Linux VDA, uninstall it before installing the new version.

1. Stop the Linux VDA services:

   ```
   sudo /sbin/service ctxvda stop

   sudo /sbin/service ctxhdx stop
   <!--NeedCopy-->
   ```

2. Uninstall the package:

   ```
   sudo rpm -e XenDesktopVDA
   <!--NeedCopy-->
   ```

> **Note:**
>
> Upgrading from the previous two versions is supported.

> **Note:**
>
> Starting with Version 1.3, the installation path changed. In previous releases, installation components were located in **/usr/local/**. The new location is **/opt/Citrix/VDA/**.
>
> To run a command, the full path is needed; alternately, 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 Yum:

For RHEL 7/CentOS 7:

```
sudo yum install -y XenDesktopVDA-7.15.0.404-1.el7_3.x86_64.rpm
<!--NeedCopy-->
```

For RHEL 6.9:

```
sudo yum install -y XenDesktopVDA-7.15.0.404-1.el6_9.x86_64.rpm
<!--NeedCopy-->
```

For RHEL 6.6/CentOS 6.6:

```
sudo yum install -y XenDesktopVDA-7.15.0.404-1.el6_6.x86_64.rpm
<!--NeedCopy-->
```

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 XenDesktopVDA-7.15.0.404-1.el7_3.x86_64.rpm
<!--NeedCopy-->
```

For RHEL 6.9:

```
sudo rpm -i XenDesktopVDA-7.15.0.404-1.el6_9.x86_64.rpm
<!--NeedCopy-->
```

For RHEL 6.6/CentOS 6.6:

```
sudo rpm -i XenDesktopVDA-7.15.0.404-1.el6_6.x86_64.rpm
<!--NeedCopy-->
```


RPM dependency list for RHEL 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

util-linux >= 2.23.2

bash >= 4.2

findutils >= 4.5

gawk >= 4.0

sed >= 4.2

cups >= 1.6.0

foomatic-filters >= 4.0.9

openldap >= 2.4

cyrus-sasl >= 2.1

cyrus-sasl-gssapi >= 2.1

libxml2 >= 2.9

```



```
python-requests >= 2.6.0

gperftools-libs >= 2.4

xorg-x11-server-Xorg >= 1.17

xorg-x11-server-Xorg < 1.18

rpmlib(FileDigests) <= 4.6.0-1

rpmlib(PayloadFilesHavePrefix) <= 4.0-1

rpmlib(CompressedFileNames) <= 3.0.4-1

rpmlib(PayloadIsXz) <= 5.2-1
<!--NeedCopy-->
```

RPM dependency list for RHEL 6.9:

```
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

openldap >= 2.4

cyrus-sasl >= 2.1

cyrus-sasl-gssapi >= 2.1

libxml2 >= 2.7

python-requests >= 2.6.0

gperftools-libs >= 2.0

xorg-x11-server-Xorg >= 1.17

xorg-x11-server-Xorg < 1.18

rpmlib(FileDigests) <= 4.6.0-1

rpmlib(PayloadFilesHavePrefix) <= 4.0-1

rpmlib(CompressedFileNames) <= 3.0.4-1

rpmlib(PayloadIsXz) <= 5.2-1
<!--NeedCopy-->
```

RPM dependency list for RHEL 6.6/CentOS 6.6:

```
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

openldap >= 2.4

cyrus-sasl >= 2.1

cyrus-sasl-gssapi >= 2.1

libxml2 >= 2.7

python-requests >= 2.6.0

gperftools-libs >= 2.0

xorg-x11-server-Xorg >= 1.15

xorg-x11-server-Xorg < 1.16

rpmlib(FileDigests) <= 4.6.0-1
```



```
66
67 rpmlib(PayloadFilesHavePrefix) <= 4.0-1
68
69 rpmlib(CompressedFileNames) <= 3.0.4-1
70
71 rpmlib(PayloadIsXz) <= 5.2-1
72 <!--NeedCopy-->
```

Step 4d: Upgrade the Linux VDA (optional)

You can upgrade the Linux VDA software from versions 7.14 and 7.13 using Yum:

For RHEL 7/CentOS 7:

```
1 sudo yum install -y XenDesktopVDA-7.15.0.404-1.el7_3.x86_64.rpm
2 <!--NeedCopy-->
```

For RHEL 6.9:

```
1 sudo yum install -y XenDesktopVDA-7.15.0.404-1.el6_9.x86_64.rpm
2 <!--NeedCopy-->
```

For RHEL 6.6/CentOS 6.6:

```
1 sudo yum install -y XenDesktopVDA-7.15.0.404-1.el6_6.x86_64.rpm
2 <!--NeedCopy-->
```

Upgrade the Linux VDA software using the RPM package manager:

For RHEL 7/CentOS 7:

```
1 sudo rpm -U XenDesktopVDA-7.15.0.404-1.el7_3.x86_64.rpm
2 <!--NeedCopy-->
```

For RHEL 6.9:

```
1 sudo rpm -U XenDesktopVDA-7.15.0.404-1.el6_9.x86_64.rpm
2 <!--NeedCopy-->
```

For RHEL 6.6/CentOS 6.6:

```
1 sudo rpm -U XenDesktopVDA-7.15.0.404-1.el6_6.x86_64.rpm
2 <!--NeedCopy-->
```

> **Important:**
>
> Restart the Linux VDA machine after upgrading the software.

Step 5: 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:

```
yum install gcc

yum install "kernel-devel-$(uname -r)"

systemctl set-default multi-user.target
<!--NeedCopy-->
```

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.


```
nvidia-smi

+-----------------------------------------------------+
| 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 |                  Off |
| N/A   20C    P0    37W / 150W |     19MiB /  8191MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| 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`

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 6: 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
<!--NeedCopy-->
```

Prompted configuration

Run a manual configuration with prompted questions:

```
sudo /opt/Citrix/VDA/sbin/ctxsetup.sh
<!--NeedCopy-->
```


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** –The Linux VDA supports HDX 3D Pro, a set of GPU acceleration technologies designed to optimize the virtualization of rich graphics applications. 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 **<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_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_HDX_3D_PRO=Y|N

export CTX_XDL_VDI_MODE=Y|N

export CTX_XDL_SITE_NAME=dns-name

export CTX_XDL_LDAP_LIST=list-ldap-servers

export CTX_XDL_SEARCH_BASE=search-base-set

export CTX_XDL_START_SERVICE=Y|N

sudo -E /opt/Citrix/VDA/sbin/ctxsetup.sh
<!--NeedCopy-->
```

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 \
```



```
8
9  CTX_XDL_ADD_FIREWALL_RULES=Y|N \
10
11 CTX_XDL_AD_INTEGRATION=1|2|3|4 \
12
13 CTX_XDL_HDX_3D_PRO=Y|N \
14
15 CTX_XDL_VDI_MODE=Y|N \
16
17 CTX_XDL_SITE_NAME=dns-name \
18
19 CTX_XDL_LDAP_LIST=list-ldap-servers \
20
21 CTX_XDL_SEARCH_BASE=search-base-set \
22
23 CTX_XDL_START_SERVICE=Y|N \
24
25 /opt/Citrix/VDA/sbin/ctxsetup.sh
26 <!--NeedCopy-->
```

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
2 <!--NeedCopy-->
```

To remove configuration changes:

```
1 sudo /opt/Citrix/VDA/sbin/ctxcleanup.sh
2 <!--NeedCopy-->
```

> **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 7: 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:

```
sudo /sbin/service ctxhdx start

sudo /sbin/service ctxvda start
<!--NeedCopy-->
```

Stop the Linux VDA:

To stop the Linux VDA services:

```
sudo /sbin/service ctxvda stop

sudo /sbin/service ctxhdx stop
<!--NeedCopy-->
```

Restart the Linux VDA:

To restart the Linux VDA services:

```
sudo /sbin/service ctxvda stop

sudo /sbin/service ctxhdx restart

sudo /sbin/service ctxvda start
<!--NeedCopy-->
```

Check the status of the Linux VDA:

To check the running status of the Linux VDA services:

```
sudo /sbin/service ctxvda status

sudo /sbin/service ctxhdx status
<!--NeedCopy-->
```

Step 8: Create the machine catalog in XenApp or XenDesktop

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.
- Because MCS is not supported for Linux VDAs, choose PVS or the **Another service or technology** (existing images) deployment method.
- 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 9: Create the delivery group in XenApp or XenDesktop

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.

Install Linux Virtual Delivery Agent for SUSE

January 4, 2024

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:

```
su -

yast
<!--NeedCopy-->
```

Alternatively, launch the UI-based YaST tool:

```
su -

yast2 &
<!--NeedCopy-->
```

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:

```
hostname
<!--NeedCopy-->
```

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

Verify that the FQDN is set correctly:

```
hostname -f
<!--NeedCopy-->
```


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:

```
nslookup domain-controller-fqdn

ping domain-controller-fqdn

nslookup delivery-controller-fqdn

ping delivery-controller-fqdn
<!--NeedCopy-->
```

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 11: Version 9.1 or later
  - SLED/SLES 12: Version 9.3 or later
- OpenJDK 1.7.0
- OpenMotif Runtime Environment 2.3.1 or later
- Cups
  - SLED/SLES 11: Version 1.3.7 or later
  - SLED/SLES 12: Version 1.6.0 or later
- Foomatic filters
  - SLED/SLES 11: Version 3.0.0 or later
  - SLED/SLES 12: Version 1.0.0 or later
- ImageMagick
  - SLED/SLES 11: Version 6.4.3.6 or later
  - SLED/SLES 12: Version 6.8 or later

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

- SLED 11: PostgreSQL is available for SLES 11 but not SLED 11.
- SLES 11: OpenJDK and OpenMotif are available for SLED 11 but not SLES 11.
- 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 11, run the commands:

```
sudo mkdir -p /mnt/sles

sudo mount -t iso9660 path-to-iso/SLES-11-SP4-DVD-x86_64-GM-DVD1.iso /
   mnt/sles

sudo zypper ar -f /mnt/sles sles
<!--NeedCopy-->
```


- On SLES 11, run the commands:

```
sudo mkdir -p /mnt/sled

sudo mount -t iso9660 path-to-iso/SLED-11-SP4-DVD-x86_64-GM-DVD1.iso /
   mnt/sled

sudo zypper ar -f /mnt/sled sled
<!--NeedCopy-->
```

- On SLED 12, run the commands:

```
sudo mkdir -p /mnt/sles

sudo mount -t iso9660 path-to-iso/SLES-12-SP2-DVD-x86_64-GM-DVD1.iso /
   mnt/sles

sudo zypper ar -f /mnt/sles sles
<!--NeedCopy-->
```

- On SLED/SLES 12, run the commands:

```
sudo mkdir -p /mnt/sdk

sudo mount -t iso9660 path-to-iso/SLE-12-SP3-SDK-DVD-x86_64-GM-DVD1.iso
    /mnt/sdk

sudo zypper ar -f /mnt/sdk sdk
<!--NeedCopy-->
```

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

```
sudo zypper install krb5-client
<!--NeedCopy-->
```

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 problems, ensure that you installed only OpenJDK Version 1.7.0. Remove all other versions of Java on your system.

- **SLED:**

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

```
sudo zypper info java-1_7_0-openjdk
<!--NeedCopy-->
```

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

```
sudo zypper update java-1_7_0-openjdk
<!--NeedCopy-->
```

3. Check the Java version:

```
java -version
<!--NeedCopy-->
```

- **SLES:**

1. On SLES, install the Java runtime environment:

```
sudo zypper install java-1_7_0-openjdk
<!--NeedCopy-->
```

2. Check the Java version:

```
java -version
<!--NeedCopy-->
```

Install PostgreSQL

- On SLED/SLES 11, install the packages:

```
 sudo zypper install libecpg6

 sudo zypper install postgresql-init

 sudo zypper install postgresql

 sudo zypper install postgresql-server

 sudo zypper install postgresql-jdbc
 <!--NeedCopy-->
```

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

```
 sudo /sbin/insserv postgresql

 sudo /etc/init.d/postgresql restart
 <!--NeedCopy-->
```


- On SLED/SLES 12, install the packages:

```
sudo zypper install postgresql-init

sudo zypper install postgresql-server

sudo zypper install postgresql-jdbc
<!--NeedCopy-->
```

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

```
sudo systemctl enable postgresql

sudo systemctl restart postgresql
<!--NeedCopy-->
```

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 11, run the commands to remove the packages:

```
sudo zypper rr sles

sudo umount /mnt/sles

sudo rmdir /mnt/sles
<!--NeedCopy-->
```

- on SLES 11, run the commands to remove the packages:

```
sudo zypper rr sled

sudo umount /mnt/sled

sudo rmdir /mnt/sled
<!--NeedCopy-->
```

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

```
sudo zypper rr sles

sudo umount /mnt/sles

sudo rmdir /mnt/sles
<!--NeedCopy-->
```

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


```
sudo zypper rr sdk

sudo umount /mnt/sdk

sudo rmdir /mnt/sd
<!--NeedCopy-->
```

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:

```
su -



cat /proc/sys/xen/independent_wallclock
<!--NeedCopy-->
```

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:

```
sudo echo 1 > /proc/sys/xen/independent_wallclock
<!--NeedCopy-->
```

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:

```
reboot
<!--NeedCopy-->
```

After restart, check that the setting is correct:

```
su -
cat /proc/sys/xen/independent_wallclock
<!--NeedCopy-->
```

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.

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. We recommend you restart the machine:


```
su -

reboot
<!--NeedCopy-->
```

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:

```
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
<!--NeedCopy-->
```

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

```
sudo net ads info
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

Examine the machine account details using:

```
sudo net ads status
<!--NeedCopy-->
```

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

```
wbinfo --krb5auth=domain\\username%password
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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

```
ls /tmp/krb5cc_uid
<!--NeedCopy-->
```

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

```
klist
<!--NeedCopy-->
```

Exit the session

```
exit
<!--NeedCopy-->
```

A similar test can be performed by logging on to the Gnome or KDE console directly. Proceed to Step 4: Install the Linux VDA after the domain joining verification.

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

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

```
sudo /opt/quest/bin/vastool configure vas vasd auto-ticket-renew-
   interval 32400

sudo /opt/quest/bin/vastool configure vas vas_auth allow-disconnected-
   auth false
<!--NeedCopy-->
```

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:

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

sudo /opt/quest/bin/vastool configure nss
<!--NeedCopy-->
```


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

```
sudo /opt/quest/bin/vastool -u user join domain-name
<!--NeedCopy-->
```

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:

```
sudo /opt/quest/bin/vastool info domain
<!--NeedCopy-->
```

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.

```
ssh localhost -l domain\\username

id -u
<!--NeedCopy-->
```

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

```
ls /tmp/krb5cc_uid
<!--NeedCopy-->
```

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

```
/opt/quest/bin/vastool klist
<!--NeedCopy-->
```

Exit the session.

```
exit
<!--NeedCopy-->
```


A similar test can be performed by logging on to the Gnome or KDE console directly. Proceed to Step 4: Install the Linux VDA after the domain joining verification.

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:

```
su -

adjoin -w -V -u user domain-name
<!--NeedCopy-->
```

The **user** 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:

```
su -

adinfo
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

Test connectivity to the various Active Directory and Kerberos services.

```
adinfo --test
<!--NeedCopy-->
```

Proceed to Step 4: Install the Linux VDA after the domain joining verification.

Step 4: Install the Linux VDA

Step 4a: 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.

1. Stop the Linux VDA services:

```
sudo /sbin/service ctxvda stop

sudo /sbin/service ctxhdx stop
<!--NeedCopy-->
```

2. Uninstall the package:

```
sudo rpm -e XenDesktopVDA
<!--NeedCopy-->
```

> **Important:**
>
> Upgrading from the previous two versions is supported.
>
> **Note:**
>
> Installation components are located in **/opt/Citrix/VDA/**.
>
> To run 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 XenDesktopVDA-7.15.0.404-1.sle12_2.x86_64.rpm
<!--NeedCopy-->
```

For SUSE 11:

```
sudo zypper install XenDesktopVDA-7.15.0.404-1.sle11_4.x86_64.rpm
<!--NeedCopy-->
```


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

For SUSE 12:

```
sudo rpm -i XenDesktopVDA-7.15.0.404-1.sle12_2.x86_64.rpm
<!--NeedCopy-->
```

For SUSE 11:

```
sudo rpm -i XenDesktopVDA-7.15.0.404-1.sle11_4.x86_64.rpm
<!--NeedCopy-->
```

Step 4d: Upgrade the Linux VDA (optional)

You can upgrade the Linux VDA software from Versions 7.14 and 7.13 using the RPM package manager:

For SUSE 12:

```
sudo rpm -U XenDesktopVDA-7.15.0.404-1.sle12_2.x86_64.rpm
<!--NeedCopy-->
```

For SUSE 11:

```
sudo rpm -U XenDesktopVDA-7.15.0.404-1.sle11_4.x86_64.rpm
<!--NeedCopy-->
```

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

openldap2 >= 2.4

cyrus-sasl >= 2.1

cyrus-sasl-gssapi >= 2.1

libxml2 >= 2.9

python-requests >= 2.8.1

rpmlib(PayloadFilesHavePrefix) <= 4.0-1

rpmlib(CompressedFileNames) <= 3.0.4-1

rpmlib(PayloadIsLzma) <= 4.4.6-1
<!--NeedCopy-->
```

RPM Dependency list for SUSE 11:

```
postgresql-server >= 9.1.

postgresql-jdbc >= 9.1

java-1_7_0-openjdk >= 1.7.0.6

ImageMagick >= 6.4.3.6

ConsoleKit >= 0.2.10

dbus-1 >= 1.2.10

dbus-1-x11 >= 1.2.10

xorg-x11-libXpm >= 7.4

xorg-x11-libs >= 7.4

openmotif-libs >= 2.3.1

pam >= 1.1.5
```



```
libdrm >= 2.4.41

libpixman-1-0 >= 0.24.4

Mesa >= 9.0

openssl >= 0.9.8j

xorg-x11 >= 7.4

xorg-x11-fonts-core >= 7.4

xorg-x11-libXau >= 7.4

xorg-x11-libXdmcp >= 7.4

bash >= 3.2

findutils >= 4.4

gawk >= 3.1

sed >= 4.1

cups >= 1.3.7

foomatic-filters >= 3.0.0

openldap2 >= 2.4

cyrus-sasl >= 2.1

cyrus-sasl-gssapi >= 2.1

libxml2 >= 2.7

python-requests >= 2.0.1

rpmlib(PayloadFilesHavePrefix) <= 4.0-1

rpmlib(CompressedFileNames) <= 3.0.4-1

rpmlib(PayloadIsLzma) <= 4.4.6-1
<!--NeedCopy-->
```

> **Important:**
>
> Restart the Linux VDA machine after upgrading.

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
<!--NeedCopy-->
```

Prompted configuration

Run a manual configuration with prompted questions:

```
sudo /opt/Citrix/VDA/sbin/ctxsetup.sh
<!--NeedCopy-->
```

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** –The Linux VDA supports HDX 3D Pro, a set of GPU acceleration technologies designed to optimize the virtualization of rich graphics applications. 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 **<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_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_HDX_3D_PRO=Y|N

export CTX_XDL_VDI_MODE=Y|N

export CTX_XDL_SITE_NAME=dns-name

```



```
export CTX_XDL_LDAP_LIST=list-ldap-servers

export CTX_XDL_SEARCH_BASE=search-base-set

export CTX_XDL_START_SERVICE=Y|N

sudo -E /opt/Citrix/VDA/sbin/ctxsetup.sh
<!--NeedCopy-->
```

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 \

CTX_XDL_AD_INTEGRATION=1|2|3|4 \

CTX_XDL_HDX_3D_PRO=Y|N \

CTX_XDL_VDI_MODE=Y|N \

CTX_XDL_SITE_NAME=dns-name \

CTX_XDL_LDAP_LIST=list-ldap-servers \

CTX_XDL_SEARCH_BASE=search-base-set \

CTX_XDL_START_SERVICE=Y|N \

/opt/Citrix/VDA/sbin/ctxsetup.sh
<!--NeedCopy-->
```

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:

```
sudo /usr/local/sbin/ctxcleanup.sh --help
<!--NeedCopy-->
```


To remove configuration changes:

```
sudo /usr/local/sbin/ctxcleanup.sh
<!--NeedCopy-->
```

> **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:

```
sudo /sbin/service ctxhdx start

sudo /sbin/service ctxvda start
<!--NeedCopy-->
```

Stop the Linux VDA:

To stop the Linux VDA services:

```
sudo /sbin/service ctxvda stop

sudo /sbin/service ctxhdx stop
<!--NeedCopy-->
```

Restart the Linux VDA:

To restart the Linux VDA services:

```
sudo /sbin/service ctxvda stop

sudo /sbin/service ctxhdx restart

```



```
sudo /sbin/service ctxvda start
<!--NeedCopy-->
```

Check the Linux VDA status:

To check the running status of the Linux VDA services:

```
sudo /sbin/service ctxvda status

sudo /sbin/service ctxhdx status
<!--NeedCopy-->
```

Step 7: Create the machine catalog in XenApp or XenDesktop

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.
- Because MCS is not supported for Linux VDAs, choose PVS or the **Another service or technology** (existing images) deployment method.
- 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 XenApp or XenDesktop

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 com-

plete 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.

Install Linux Virtual Delivery Agent for Ubuntu

June 10, 2022

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

Ensure 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 host name

To ensure that the DNS domain name and Fully Qualified Domain Name (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
2 <!--NeedCopy-->
```

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

Verify that the FQDN is set correctly:

```
1 hostname -f
2 <!--NeedCopy-->
```

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:

```
nslookup domain-controller-fqdn

ping domain-controller-fqdn

nslookup delivery-controller-fqdn

ping delivery-controller-fqdn
<!--NeedCopy-->
```

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:

```
apt-get install chrony
<!--NeedCopy-->
```

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:

```
sudo systemctl restart chrony
<!--NeedCopy-->
```


Step 1h: Install OpenJDK

The Linux VDA depends on OpenJDK. Typically, the runtime environment is installed as part of the operating system installation. Check whether it has been installed with:

```
sudo apt-get install -y default-jdk
<!--NeedCopy-->
```

Step 1i: Install PostgreSQL

The Linux VDA requires PostgreSQL Version 9.x on Ubuntu 16.04:

```
sudo apt-get install -y postgresql

sudo apt-get install -y libpostgresql-jdbc-java
<!--NeedCopy-->
```

Step 1j: Install Motif

```
sudo apt-get install -y libxm4
<!--NeedCopy-->
```

Step 1k: Install other packages

```
sudo apt-get install -y libsasl2-2

sudo apt-get install -y libsasl2-modules-gssapi-mit

sudo apt-get install -y libldap-2.4-2

sudo apt-get install -y krb5-user

sudo apt-get install -y cups
<!--NeedCopy-->
```

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:

```
su -

cat /proc/sys/xen/independent_wallclock
<!--NeedCopy-->
```

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:

```
sudo echo 1 > /proc/sys/xen/independent_wallclock
<!--NeedCopy-->
```

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:

```
su -

cat /proc/sys/xen/independent_wallclock
<!--NeedCopy-->
```

This command returns the value 1.

Fix time synchronization on Microsoft Hyper-V

Linux VMs with Hyper-V Linux Integration Services installed can use 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.

Samba Winbind

Install or update the required packages

```
sudo apt-get install winbind samba libnss-winbind libpam-winbind krb5-config krb5-locales krb5-user
<!--NeedCopy-->
```

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

```
sudo systemctl enable winbind
<!--NeedCopy-->
```

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

```
[libdefaults]

default_realm = REALM

dns_lookup_kdc = false



[realms]

REALM = {


admin_server = domain-controller-fqdn

kdc = domain-controller-fqdn

 }




[domain_realm]

domain-dns-name = REALM

.domain-dns-name = REALM
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```


> **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
<!--NeedCopy-->
```

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

```
sudo net ads info
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

Examine the account details of the machine using:

```
sudo net ads status
<!--NeedCopy-->
```

Verify user authentication Use the **wbinfo** tool to verify that domain users can authenticate with the domain:


```
wbinfo --krb5auth=domain\\username%password
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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

```
ls /tmp/krb5cc_uid
<!--NeedCopy-->
```

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

```
klist
<!--NeedCopy-->
```

Exit the session.

```
exit
<!--NeedCopy-->
```

A similar test can be performed by logging on to the Gnome or KDE console directly. Proceed to Step 4: Install the Linux VDA after the domain joining verification.

> **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:

```
SELINUX=disabled
```

This change requires a machine restart:

```
reboot
<!--NeedCopy-->
```

> **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:

```
sudo /opt/quest/bin/vastool configure vas vasd auto-ticket-renew-
    interval 32400

sudo /opt/quest/bin/vastool configure vas vas_auth allow-disconnected-
    auth false
<!--NeedCopy-->
```

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:

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

sudo /opt/quest/bin/vastool configure nss
<!--NeedCopy-->
```

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

```
sudo /opt/quest/bin/vastool -u user join domain-name
<!--NeedCopy-->
```

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.

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
<!--NeedCopy-->
```

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.

```
ssh localhost -l domain\\username

id -u
<!--NeedCopy-->
```

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

```
ls /tmp/krb5cc_uid
<!--NeedCopy-->
```


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

```
/opt/quest/bin/vastool klist
<!--NeedCopy-->
```

Exit the session.

```
exit
<!--NeedCopy-->
```

Proceed to Step 4: Install the Linux VDA after the domain joining verification.

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:

```
su -

adjoin -w -V -u user domain-name
<!--NeedCopy-->
```

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:

```
su -

adinfo
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

Test connectivity to the various Active Directory and Kerberos services.


```
adinfo --test
<!--NeedCopy-->
```

Proceed to Step 4: Install the Linux VDA after the domain joining verification.

SSSD

Configure Kerberos Run the following command to install Kerberos:

```
sudo apt-get install krb5-user
<!--NeedCopy-->
```

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

```
[libdefaults]

 default_realm = REALM

 dns_lookup_kdc = false

[realms]

 REALM = {


  admin_server = domain-controller-fqdn

  kdc = domain-controller-fqdn

  }


[domain_realm]

 domain-dns-name = REALM

 .domain-dns-name = REALM
<!--NeedCopy-->
```

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. You can use `adcli`, `realmd`, or `Samba` instead.

> **Note:**
>
> This section only provides information for `adcli` and `Samba`.

Use adcli to join the domain:

Install adcli:

Install the required package:

```
sudo apt-get install adcli
<!--NeedCopy-->
```

Join the domain with adcli:

Remove the old system keytab file and join the domain using:

```
su -

rm -rf /etc/krb5.keytab

adcli join domain-dns-name -U user -H hostname-fqdn
<!--NeedCopy-->
```

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:

```
sudo klist -ket
<!--NeedCopy-->
```

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:

```
sudo apt-get install samba
<!--NeedCopy-->
```

Configure samba:

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

```
[global]

```



```
    workgroup = WORKGROUP

    security = ADS

    realm = REALM

    client signing = yes

    client use spnego = yes

    kerberos method = secrets and keytab
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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

```
sudo apt-get update sssd
<!--NeedCopy-->
```

> **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]
```



```
2
3  services = nss, pam
4
5  config_file_version = 2
6
7  domains = domain-dns-name
8
9  [domain/domain-dns-name]
10
11 id_provider = ad
12
13 access_provider = ad
14
15 auth_provider = krb5
16
17 krb5_realm = REALM
18
19 # Set krb5_renewable_lifetime higher if TGT renew lifetime is longer
      than 14 days
20
21 krb5_renewable_lifetime = 14d
22
23 # Set krb5_renew_interval to lower value if TGT ticket lifetime is
      shorter than 2 hours
24
25 krb5_renew_interval = 1h
26
27 krb5_ccachedir = /tmp
28
29 krb5_ccname_template = FILE:%d/krb5cc_%U
30
31 # This ldap_id_mapping setting is also the default value
32
33 ldap_id_mapping = true
34
35 override_homedir = /home/%d/%u
36
37 default_shell = /bin/bash
38
39 ad_gpo_map_remote_interactive = +ctxhdx
40 <!--NeedCopy-->
```

> **Note:**
>
> ldap_id_mapping is set to **true** so that SSSD itself takes care of mapping Windows SIDs to Unix UIDs. Otherwise, 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:

```
sudo chmod 0600 /etc/sssd/sssd.conf
<!--NeedCopy-->
```

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

```
sudo systemctl start sssd

sudo systemctl enable sssd
<!--NeedCopy-->
```

PAM configuration Run the following command and ensure that the **SSS authentication** and **Create home directory on login** options are selected:

```
sudo pam-auth-update
<!--NeedCopy-->
```

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:

```
sudo adcli info domain-dns-name
<!--NeedCopy-->
```

Use samba to verify domain membership:

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

```
sudo net ads testjoin
<!--NeedCopy-->
```

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

```
sudo net ads info
<!--NeedCopy-->
```


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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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:

```
sudo klist
<!--NeedCopy-->
```

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.

```
ssh localhost -l domain\\username

id -u

klist

exit
<!--NeedCopy-->
```

Check that the Keberos 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:

```
ls /tmp/krb5cc_uid
<!--NeedCopy-->
```

A similar test can be performed by logging on to KDE or Gnome Display Manager. Proceed to Step 4: Install the Linux VDA after the domain joining verification.

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:

```
sudo dpkg -i xendesktopvda_7.15.0.404-1.ubuntu16.04_amd64.deb
<!--NeedCopy-->
```

Debian dependency list for Ubuntu:

```
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

cups >= 2.1

libldap-2.4-2 >= 2.4.42

libsasl2-modules-gssapi-mit >= 2.1.~

python-requests >= 2.9.1

```



```
libgoogle-perftools4 >= 2.4~
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

Prompted configuration Run a manual configuration with prompted questions:

```
sudo /opt/Citrix/VDA/sbin/ctxsetup.sh
<!--NeedCopy-->
```

Automated configuration 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** –The Linux VDA supports HDX 3D Pro, a set of GPU acceleration technologies designed to optimize the virtualization of rich graphics applications. 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 **<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_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_HDX_3D_PRO=Y|N

export CTX_XDL_VDI_MODE=Y|N
```



```
16
17 export CTX_XDL_SITE_NAME=dns-name
18
19 export CTX_XDL_LDAP_LIST=list-ldap-servers
20
21 export CTX_XDL_SEARCH_BASE=search-base-set
22
23 export CTX_XDL_START_SERVICE=Y|N
24
25 sudo -E /opt/Citrix/VDA/sbin/ctxsetup.sh
26 <!--NeedCopy-->
```

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 \
2
3  CTX_XDL_DDC_LIST=list-ddc-fqdns \
4
5  CTX_XDL_VDA_PORT=port-number \
6
7  CTX_XDL_REGISTER_SERVICE=Y|N \
8
9  CTX_XDL_ADD_FIREWALL_RULES=Y|N \
10
11 CTX_XDL_AD_INTEGRATION=1|2|3|4 \
12
13 CTX_XDL_HDX_3D_PRO=Y|N \
14
15 CTX_XDL_VDI_MODE=Y|N \
16
17 CTX_XDL_SITE_NAME=dns-name \
18
19 CTX_XDL_LDAP_LIST=list-ldap-servers \
20
21 CTX_XDL_SEARCH_BASE=search-base-set \
22
23 CTX_XDL_START_SERVICE=Y|N \
24
25 /opt/Citrix/VDA/sbin/ctxsetup.sh
26 <!--NeedCopy-->
```

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
```



```
<!--NeedCopy-->
```

To remove configuration changes:

```
sudo /opt/Citrix/VDA/sbin/ctxcleanup.sh
<!--NeedCopy-->
```

> **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:

```
dpkg -l xendesktopvda
<!--NeedCopy-->
```

To view more detailed information:

```
apt-cache show xendesktopvda
<!--NeedCopy-->
```

To uninstall the Linux VDA software:

```
dpkg -r xendesktopvda
<!--NeedCopy-->
```

> **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:

```
sudo systemctl start ctxhdx

sudo systemctl start ctxvda
<!--NeedCopy-->
```

Stop the Linux VDA:

To stop the Linux VDA services:

```
sudo systemctl stop ctxvda

sudo systemctl stop ctxhdx
<!--NeedCopy-->
```

Restart the Linux VDA:

To restart the Linux VDA services:

```
sudo systemctl stop ctxvda

sudo systemctl restart ctxhdx

sudo systemctl restart ctxvda
<!--NeedCopy-->
```

Check the Linux VDA status:

To check the running status of the Linux VDA services:

```
sudo systemctl status ctxvda

sudo systemctl status ctxhdx
<!--NeedCopy-->
```

Step 6: Create the machine catalog in XenApp or XenDesktop

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.
- Because MCS is not supported for Linux VDAs, choose PVS or the **Another service or technology** (existing images) deployment method.
- 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 XenApp or XenDesktop

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.

Configure the Linux VDA

June 18, 2020

This section details the features of the Linux VDA, including feature description, configuration, and troubleshooting.

Integrate NIS with Active Directory

January 11, 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 XenApp & XenDesktop. 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 7.3 or later/CentOS 7.3 or later
- Linux VDA Version 1.3 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:

```
yum –y install ypbind rpcbind oddjob-mkhomedir
<!--NeedCopy-->
```

Set the NIS domain:

```
ypdomainname nis.domain
echo "NISDOMAIN=nis.domain" >> /etc/sysconfig/network
<!--NeedCopy-->
```

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:

```
sudo authconfig --enablenis --nisdomain=nis.domain --nisserver=server.
   nis.domain --enablemkhomedir --update
<!--NeedCopy-->
```

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:

```
sudo systemctl start rpcbind ypbind

sudo systemctl enable rpcbind ypbind
<!--NeedCopy-->
```

Ensure that the NIS configuration is correct:

```
ypwhich
<!--NeedCopy-->
```

Validate that the account information is available from the NIS server:

```
getent passwd nisaccount
<!--NeedCopy-->
```


> **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:

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

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:

```
--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
```

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:

```
sudo net ads join REALM -U user
<!--NeedCopy-->
```


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):

```
[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

# 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
<!--NeedCopy-->
```


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**:

```
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:

```
authconfig --enablesssd --enablesssdauth --enablemkhomedir --update

sudo systemctl start sssd

sudo systemctl enable sssd
<!--NeedCopy-->
```

> **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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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

```
ls /tmp/krb5cc_{
 uid }

<!--NeedCopy-->
```

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

```
klist
<!--NeedCopy-->
```

Publish applications

June 24, 2022

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:

```
sudo /opt/Citrix/VDA/bin/ctxreg list -k "HKEY_LOCAL_MACHINE\System\
   CurrentControlSet\Control\Citrix"  |  grep "SeamlessEnabled"
<!--NeedCopy-->
```


If it shows "SeamlessEnabled = 0x00000000,"the seamless mode is disabled. To enable it, run the following command:

```
sudo /opt/Citrix/VDA/bin/ctxreg update  -k "HKEY_LOCAL_MACHINE\System\
   CurrentControlSet\Control\Citrix"  -v "SeamlessEnabled" -d "0
   x00000001"
<!--NeedCopy-->
```

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 (CentOS7.3\RHEL7.3\SUSE12.2), Metacity (CentOS6.6\RHEL6.6\SUSE 11.4), 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.
- 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

June 18, 2020

This article provides information about printing best practices.

Installation

The Linux VDA requires both **cups** and **foomatic** filters. Run the following commands based on your Linux distribution:

RHEL 7 printing support:

```
sudo yum –y install cups

sudo yum -y install foomatic-filters
<!--NeedCopy-->
```

RHEL 6 printing support:

```
sudo yum –y install cups

sudo yum -y install foomatic
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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
<!--NeedCopy-->
```

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.lo**:

```
sudo service cups stop

sudo vi /etc/systemd/system/printer.target.wants/cups.service

PrivateTmp=false

sudo service cups start

sudo systemctl daemon-reload
<!--NeedCopy-->
```

> **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:

```
[DEFAULT_PRINTER]

printername=

model=

ppdpath=

drivertype=
<!--NeedCopy-->
```


> **Important:**
>
> The **printername** 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

...

xerox/ph3115.ppd.gz Xerox Phaser 3115, SpliX V. 2.0.0

xerox/ph3115fr.ppd.gz Xerox Phaser 3115, SpliX V. 2.0.0

xerox/ph3115pt.ppd.gz Xerox Phaser 3115, SpliX V. 2.0.0
<!--NeedCopy-->
```

You can then set the model to match the printer:

```
Model=xerox/ph3115.ppd.gz
<!--NeedCopy-->
```

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:

```
ppdpath=/home/tester/NATIVE_PRINTER_DRIVER.ppd
<!--NeedCopy-->
```

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:

```
drivertype=pcl5
<!--NeedCopy-->
```

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:

To identify this issue:

1. Log on to the ICA session desktop.
2. vi ~/.CtxlProfile$CLIENT_NAME
3. Add the following field to the save pool file on the Linux VDA:

```
deletespoolfile=no
<!--NeedCopy-->
```

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
   <!--NeedCopy-->
   ```

   c) Choose the mapped printer **CitrixUniversalPrinter:$ClientName:app/dek$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 **.CtxlpProfile$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:

```
<Policy default>

     # Job/subscription privacy...

     JobPrivateAccess default

     JobPrivateValues default

     SubscriptionPrivateAccess default

     SubscriptionPrivateValues default

     ... ...

     <Limit Create-Printer-Subscription>

```



```
            Require user @OWNER

            Order deny,allow

      </Limit>

      <Limit All>

            Order deny,allow

      </Limit>

</Policy>
<!--NeedCopy-->
```

PDF printing

March 22, 2021

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
- Citrix Workspace app 1905 for Windows 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

August 19, 2022

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 parameters

There are several graphics-related configuration parameters under **HKEY_LOCAL_MACHINE\System\CurrentCon** that you can tune with the **ctxreg** utility.

How to enable Thinwire Plus

Thinwire plus is enabled by default, for both standard VDA and 3D Pro.

How to enable H.264

In addition to the operating system requirement, H.264 has a minimum requirement for the Citrix Workspace app (formerly Citrix Receiver) version. If the client does not meet the requirements, it falls back to Thinwire Plus.

Operating system	Minimum requirement for H.264
Windows	3.4 or later
Mac OS X	11.8 or later
Linux	13.0 or later
Android	3.5
iOS	5.9
Chrome OS	1.4

The latest feature matrix for Citrix Workspace app is available at https://docs.citrix.com/en-us/citrix-workspace-app/citrix-workspace-app-feature-matrix.html.

Run the following command to advertise H.264 encoding on the VDA:

```
1 sudo ctxreg create -k "HKLM\System\CurrentControlSet\Control\Citrix\
     Thinwire" -t "REG_DWORD" -v "AdvertiseH264" -d "0x00000001" --force
2 <!--NeedCopy-->
```


How to enable hardware encoding in HDX 3D Pro

For the HDX 3D Pro, the **AdvertiseH264** setting only enables software H.264 encoding.
Run the following command to enable hardware encoding:

```
sudo ctxreg create -k "HKLM\System\CurrentControlSet\Control\Citrix\
    Thinwire" -t "REG_DWORD" -v "HardwareEncoding" -d "0x00000001" --
    force
<!--NeedCopy-->
```

> **Note:**
>
> If you get the `ctxreg command can't be found` error, use the `ctxreg` command with a full path. For example, use `sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet\Control\Citrix\Thinwire"-t "REG_DWORD" -v "AdvertiseH264"-d "0x00000001"–force` instead of `sudo ctxreg create -k "HKLM\System\CurrentControlSet\Control\Citrix\Thinwire"-t "REG_DWORD"-v "AdvertiseH264"-d "0x00000001"–force`.

How to tune Thinwire Plus for lower bandwidth

- MaxColorDepth

  ```
  Default 0x20, type DWORD
  <!--NeedCopy-->
  ```

  This option specifies the color depth of graphics transferred through the Thinwire protocol to the client.

  To save bandwidth, set it to 0x10 (which represents the preferred color depth for simple graphics) or to 0x8 (the experimental low bandwidth mode).

- Quality

  Visual quality

  ```
  Default: 0x1(medium), type: DWORD, valid values: 0x0(low), 0x1(
      medium), 0x2(high), 0x3(build to lossless), 0x4 always
      lossless.
  <!--NeedCopy-->
  ```

  To save bandwidth, set Quality to 0x0(low).

- More parameters

  - TargetFPS

    Target frame rate


```
Default: 0x1e (30), Type: DWORD
<!--NeedCopy-->
```

- MinFPS

  Target minimum frame rate

```
  Default: 0xa (10), Type: DWORD
  <!--NeedCopy-->
```

- MaxScreenNum

  Maximum number of monitors the client can have

```
  Default: 0x2, Type: DWORD
  <!--NeedCopy-->
```

  For a standard VDA, you can set a maximum value of up to 10. For 3D Pro, the maximum value allowed is 4.

Troubleshooting

Check which encoding is in use

Run the following command to check whether H.264 encoding is in use (**1** means H.264; **0** means TW+):

```
sudo ctxreg dump | grep H264
<!--NeedCopy-->
```

The results resemble:

```
create -k "HKLM\Software\Citrix\Ica\Session\1\Graphics"-t "REG_DWORD"
-v "H264"-d "0x00000001"--force
```

```
create -k "HKLM\System\CurrentControlSet\Control\Citrix\Thinwire"-t "
REG_DWORD"-v "AdvertiseH264"-d "0x00000001"--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
<!--NeedCopy-->
```

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:

```
Tue Apr 12 10:42:03 2016
+-----------------------------------------------------+
| NVIDIA-SMI 361.28     Driver Version: 361.28        |
|-------------------------------+----------------------+----------------------+

| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile
   Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util
   Compute M. |
|===============================+======================+======================|

|   0  GRID K1             Off  | 0000:00:05.0     Off |
                      N/A |
| N/A   42C    P0    14W /  31W |    207MiB /  4095MiB |      8%
   Default |
+-------------------------------+----------------------+----------------------+


+-----------------------------------------------------------------------------+

| Processes:                                                       GPU
   Memory |
|  GPU       PID  Type  Process name
   Usage      |
|=============================================================================|

|    0      2164  C+G   /usr/local/bin/ctxgfx
   106MiB |
|    0      2187    G   Xorg
    85MiB |
+-----------------------------------------------------------------------------+

<!--NeedCopy-->
```

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:

```
+-----------------------------------------------------+
| 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 |
                    Off |
| N/A   20C    P0    37W / 150W |     19MiB /  8191MiB |      0%
   Default |
+-------------------------------+----------------------+----------------------+


+-----------------------------------------------------------------------------+
| Processes:                                                       GPU
   Memory |
|  GPU       PID  Type  Process name
   Usage      |
|=============================================================================|
|  No running processes found
                                 |
+-----------------------------------------------------------------------------+

<!--NeedCopy-->
```

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 **Xorg.{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"
<!--NeedCopy-->
```


NVENC API is not supported in vGPU profiles other than 8Q

NVIDIA Tesla M60 card vGPU profiles other than 8Q do not support cuda, as a result, NVENC API and Citrix 3D Pro hardware encoding are not available.

NVIDIA K2 graphics cards do not support YUV444 hardware encoding in pass-through mode

This is a limitation of NVIDIA K2 graphics cards.

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 Receiver window

Resizing the Citrix Receiver window 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)`'.

Non-GRID 3D graphics

February 20, 2024

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.
- Set `CTX_XDL_HDX_3D_PRO` to `Y` when installing the Linux VDA. For information about environment variables, see Step 3: Set up the runtime environment to complete the installation.

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.

```
1 /opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\System\
     CurrentControlSet\Control\Citrix\XDamage" -t "REG_SZ" -v "
     DriverName" -d "intel" --force
2 <!--NeedCopy-->
```


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
<!--NeedCopy-->
```

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

June 18, 2020

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:

```
sudo yum -y install openldap

sudo yum -y install libxml2

```



```
sudo yum -y install cyrus-sasl

sudo yum -y install cyrus-sasl-gssapi
<!--NeedCopy-->
```

SLES/SELD:

```
sudo zypper install openldap2

sudo zypper install libxml2

sudo zypper install cyrus-sasl

sudo zypper install cyrus-sasl-gssapi
<!--NeedCopy-->
```

Ubuntu:

```
sudo apt-get install -y libldap-2.4-2

sudo apt-get install -y libsasl2-2

sudo apt-get install -y libsasl2-modules-gssapi-mit
<!--NeedCopy-->
```

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.

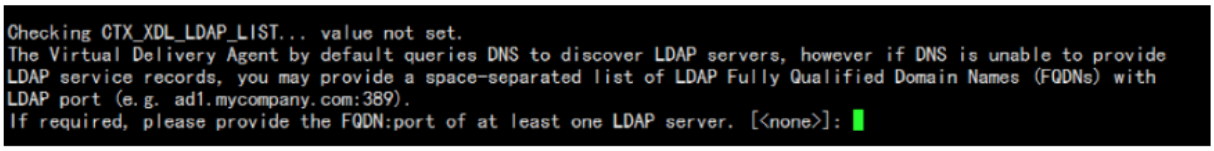
LDAP server setting on the VDA

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:

```
/opt/Citrix/VDA/sbin/ctxsetup.sh
<!--NeedCopy-->
```


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:

```
/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
<!--NeedCopy-->
```

The following policies apply only to the Linux VDA and can be configured only in Citrix Studio Version 7.12 and later:

- ClipboardSelectionUpdateMode
- PrimarySelectionUpdateMode
- MaxSpeexQuality

These policies are described in the Policy support list. If you are using Citrix Studio Version 7.11 or earlier, you must configure these policies locally on the Linux VDA using the **ctxreg** command.

> **Note:**
>
> The values are restricted to a certain range. For detailed descriptions, see the Policy support list.

Policy support list

June 18, 2020

Linux VDA policy support list

Studio Policy	Key Name	Type	Module	Default Value
ICA keep alives	SendICAKeepAlives	Computer	ICA\Keep Alive	Do not send ICA keep alive messages (0)

Studio Policy	Key Name	Type	Module	Default Value
ICA keep alive timeout	ICAKeepAliveTimeout	Computer	ICA\Keep Alive	60 seconds
ICA listener port number	IcaListenerPortNumber	Computer	ICA	1494
Audio redirection bandwidth limit	LimitAudioBw	User	Audio	0 Kbps
Client audio redirection	AllowAudioRedirection	User	Audio	Allowed (1)
Client printer redirection	AllowPrinterRedir	User	Printing	Allowed (1)
Client clipboard redirection	AllowClipboardRedir	User	Clipboard	Allowed(1)
Client USB device redirection	AllowUSBRedir	User	USB	Prohibited (0)
Client USB device redirection rules	USBDeviceRules	User	USB	“\0”
Moving image compression	MovingImageCompressionConfiguration	User	Thinwire	Enabled (1)
Target minimum frame rate	TargetedMinimumFramesPerSecond	User	Thinwire	10 fps
Target frame rate	FramesPerSecond	User	Thinwire	30 fps
Visual quality	VisualQuality	User	Thinwire	Medium (3)
Use video codec for compression	VideoCodec	User	Thinwire	Use when preferred (3)
Use hardware encoding for video codec	UseHardwareEncodingForVideoCodec	User	Thinwire	Enabled (1)
Preferred color depth for simple graphics	PreferredColorDepth	User	Thinwire	24 bits per pixel (1)
Audio quality	SoundQuality	User	Audio	High –high definition audio (2)
Client microphone redirection	AllowMicrophoneRedir	User	Audio	Allowed (1)

Studio Policy	Key Name	Type	Module	Default Value
Maximum number of sessions	MaximumNumberOfSessions	Computer	Load Management	250
Concurrent logons tolerance	ConcurrentLogonsTolerance	Computer	Load Management	2
Enable auto update of Controllers	EnableAutoUpdateOfControllers	Computer	Virtual Delivery Agent Settings	Allowed (1)
Clipboard selection update mode	ClipboardSelectionUpdateMode	User	Clipboard	3
Primary selection update mode	PrimarySelectionUpdateMode	User	Clipboard	3
Max Speex quality	MaxSpeexQuality	User	Audio	5
Auto connect client drives	AutoConnectDrives	User	ICA\File Redirection	Enabled (1)
Client optical drives	AllowCdromDrives	User	ICA\File Redirection	Allowed (1)
Client fixed drives	AllowFixedDrives	User	ICA\File Redirection	Allowed (1)
Client floppy drives	AllowFloppyDrives	User	ICA\File Redirection	Allowed (1)
Client network drives	AllowNetworkDrives	User	ICA\File Redirection	Allowed (1)
Client removable drives	AllowRemoveableDrives	User	ICA\File Redirection	Allowed (1)
Client drive redirection	AllowDriveRedir	User	ICA\File Redirection	Allowed (1)
Read-only client drive access	ReadOnlyMappedDrive	User	ICA\File Redirection	Disabled (0)

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

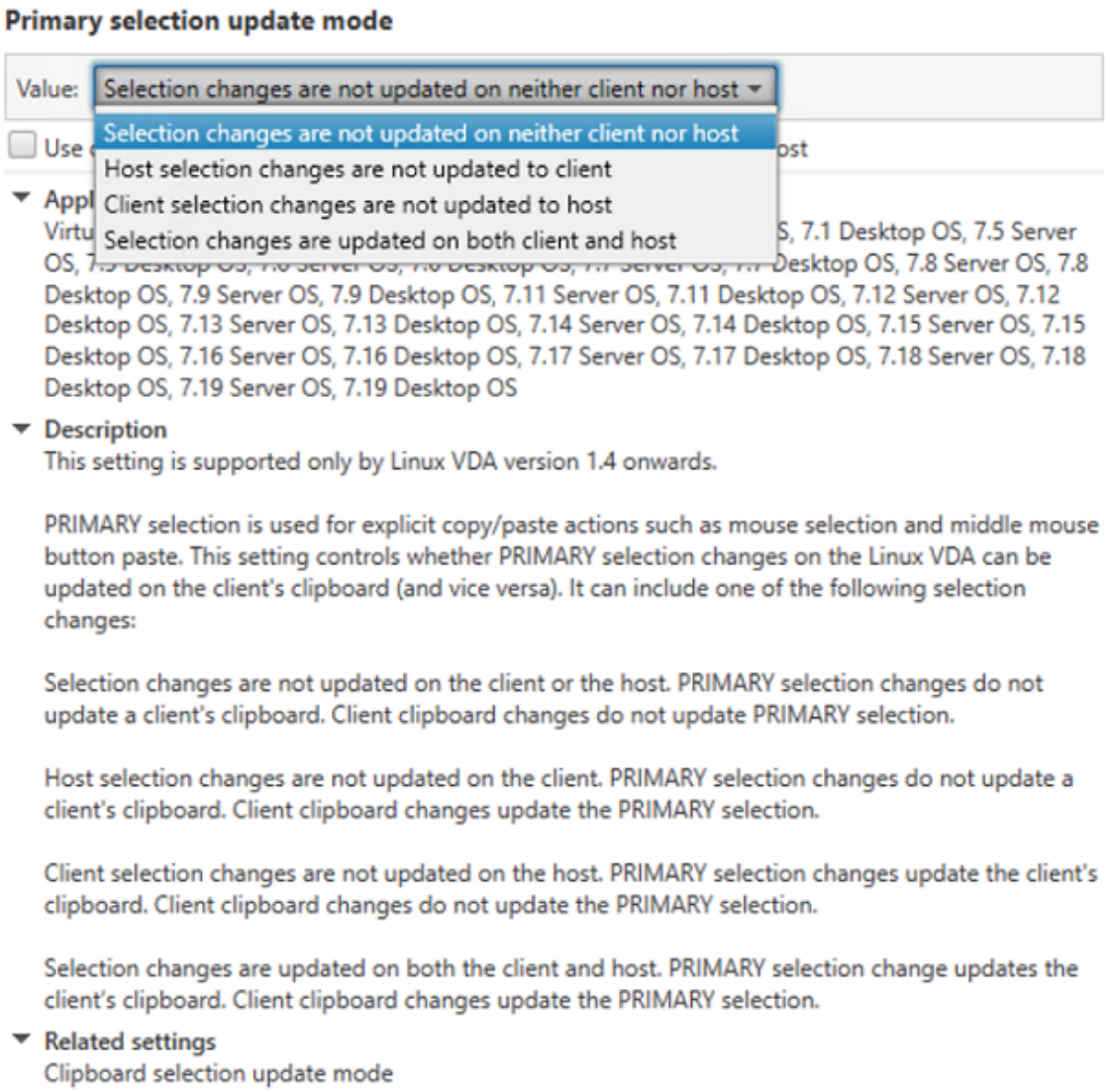
  Value (enum): [0, 1, 2, 3]

  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:



- **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

default value.

Related Setting: Clipboard selection update mode

- ClipboardSelectionUpdateMode

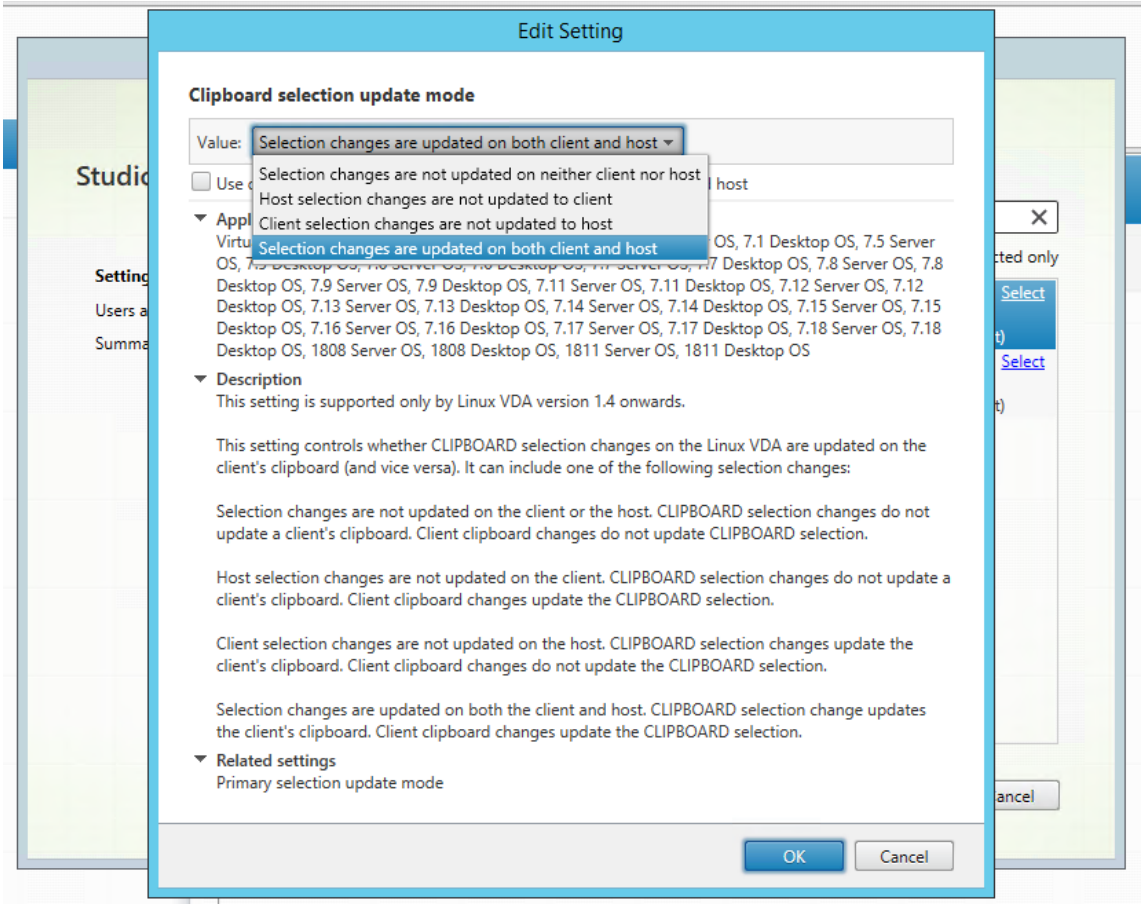
  Value (enum): [0, 1, 2, 3]

  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, we recommend that you set both clipboard selection update mode and primary selection update mode to the same value.

Configure IPv6

January 11, 2019

The Linux VDA supports IPv6 to align with XenApp and XenDesktop. 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:


```
1 “HKLM\Software\Policies\Citrix\VirtualDesktopAgent” -t “REG_DWORD”
    -v “OnlyUseIPv6ControllerRegistration”
2
3 “HKLM\Software\Policies\Citrix\VirtualDesktopAgent” -t “REG_DWORD”
    -v “ControllerRegistrationIPv6Netmask”
4 <!--NeedCopy-->
```

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
2 <!--NeedCopy-->
```

If the Linux VDA has more than one network interfaces, **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 "{
2  IPv6 netmask }
3  " --force
4 <!--NeedCopy-->
```

Replace **{IPv6 netmask}** with the real netmask (for example, 2000::/64).

For more information about IPv6 deployment in XenApp and XenDesktop, see IPv4/IPv6 support.

Troubleshooting

Check the basic IPv6 network environment and use ping6 to check whether AD and Delivery Controller are reachable.

Configure Citrix Customer Experience Improvement Program (CEIP)

February 9, 2021

When you participate in the CEIP, anonymous statistics and usage information are sent to Citrix to help improve the quality and performance of Citrix products.

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:

```
1 /opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SOFTWARE\
     Citrix\CEIP" -v "CEIPSwitch" -d "1"
2 <!--NeedCopy-->
```

- **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:

```
1 /opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SOFTWARE\
     Citrix\CEIP" -v "DataPersistPath" -d "your_path"
2 <!--NeedCopy-->
```

If the configured path 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.

Data Point	Key Name	Description
Machine GUID	machine_guid	Identifying the machine where the data originates
AD solution	ad_solution	Text string denoting the machine's domain joining method

Data Point	Key Name	Description
Linux kernel version	kernel_version	Text string denoting the machine's kernel version
LVDA version	vda_version	Text string denoting the installed version of the Linux VDA.
LVDA update or fresh install	update_or_fresh_install	Text string denoting the current Linux VDA package is being freshly installed or updated
LVDA installed method	install_method	Text string denoting that the current Linux VDA package is installed by using MCS, PVS, easy install, or manual installation.
HDX 3D pro enabled or not	hdx_3d_pro	Text string denoting whether HDX 3D Pro is enabled on the machine
VDI mode enabled or not	vdi_mode	Text string denoting whether VDI mode is enabled
LVDA key services last restart time	ctxhdx ctxvda	The last restart time of the `ctxhdx` and `ctxvda` services, in the format of dd-hh:mm:ss, for example, 10-17:22:19
GPU type	gpu_type	Denoting the GPU type of the machine
CPU cores	cpu_cores	Integer denoting the number of CPU cores of the machine
CPU frequency	cpu_frequency	Float denoting the CPU frequency in MHz
Physical memory size	memory_size	Integer denoting the physical memory size in KB
Active session number	active_session_number	Integer denoting the number of active sessions on the machine at the time we collect this data point
Linux OS name and version	os_name_version	Text string denoting the Linux OS name and version of the machine

Data Point	Key Name	Description
Session key	session_key	Identifying the session where the data originates
Reconnect time cost	econnect_time_cost	Used to save the session's reconnect time cost. The size of the array is 5 where we keep track of the current value, the minimum value, the maximum value, the running sum, and the update count of this data point.
Active session time	active_session_time	Used to save the session's active times. One session can have multiple active times because the session can disconnect/reconnect.
Session duration time	session_duration_time	Used to save the session's duration from logon to logoff
Receiver client type	receiver_type	Integer denoting the type of Citrix Receiver used to launch the session
Receiver client version	receiver_version	Text string denoting the version of Citrix Receiver used to launch the session
Printing count	printing_count	Integer denoting the number of times the session uses the printing function
USB redirection count	usb_redirecting_count	Integer denoting the number of times the session uses a USB device

Configure USB redirection

February 20, 2021

USB devices are shared between Citrix Receiver 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 Receiver 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 is not usable for 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.

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:

> **Note:**
>
> The Linux VDA supports only USB 2.0 protocols.

USB mass storage device	VID:PID	File system
Netac Technology Co., Ltd	0dd8:173c	FAT32
Kingston Datatraveler 101 II	0951:1625	FAT32
Kingston Datatraveler GT101 G2	1567:8902	FAT32
SanDisk SDCZ80 flash drive	0781:5580	FAT32
WD HDD	1058:10B8	FAT32

USB 3D mouse	VID:PID
3DConnexion SpaceMouse Pro	046d: c62b

USB scanner	VID:PID
Epson Perfection V330 photo	04B8: 0142

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:

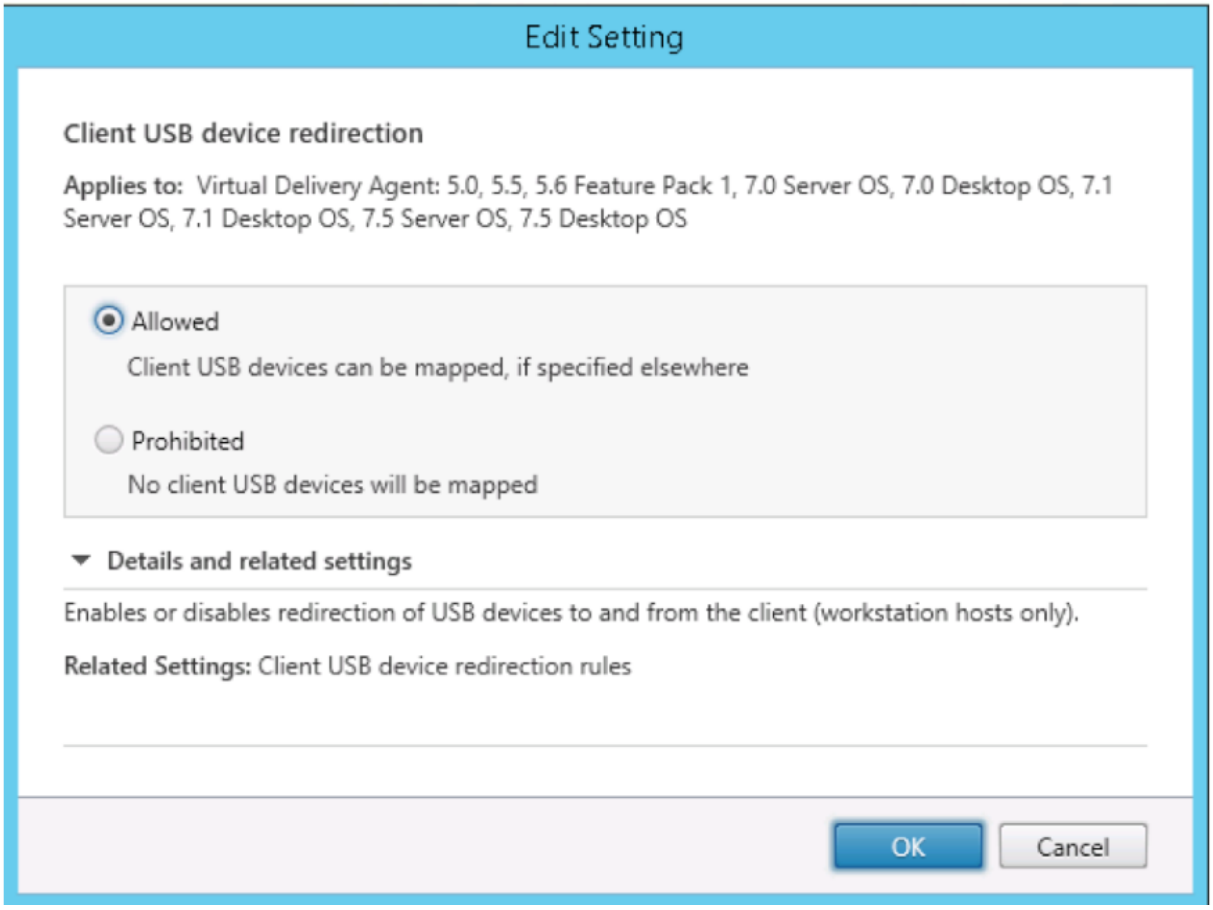
- Client USB device redirection policy
- Client USB device redirection rules

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**.

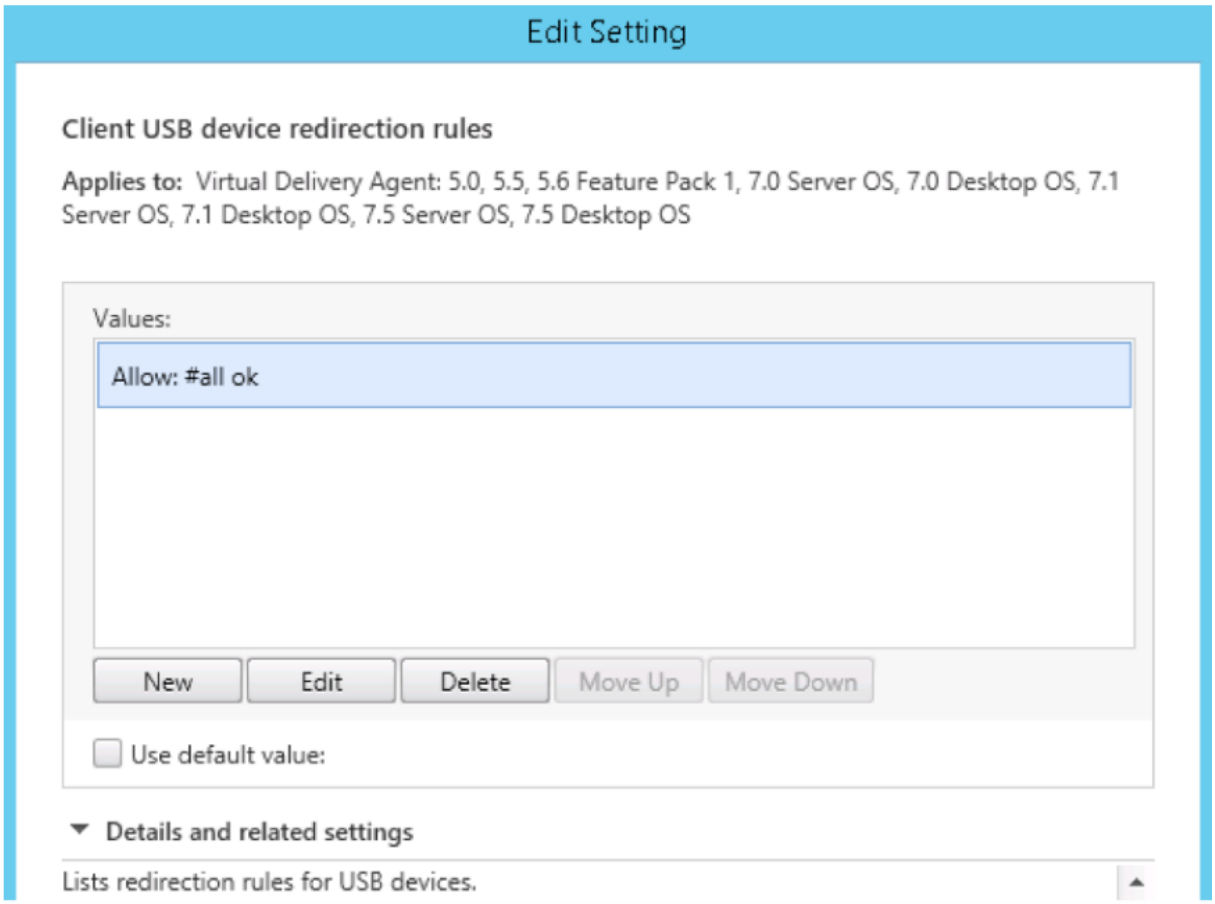


Set USB redirection rules

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.

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:

Supported Linux distribution	Kernel version
RHEL 7.3	3.10.0-514.el7.x86_64
RHEL 6.6	2.6.32-504.el6.x86_64
SUSE 12.2	4.4.49-92.11-default
SUSE 11.4	3.0.101-0.47.55-default
Ubuntu 16.04	4.4.0-45-generic

> **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, run the following command to verify whether the kernel is consistent:

```
insmod /opt/Citrix/VDA/lib64/usb-vhci-hcd.ko
<!--NeedCopy-->
```

If the command runs successfully, the kernel module has loaded successfully and the version is consistent with the one installed by Citrix.

If the command runs 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-souces/vhci-hcd-1.15.tar.bz2**; you can restore VHCI files using **tar xvf vhci-hcd-1.15.tar.bz2**.
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 7.3/RHEL 6.9/RHEL 6.6:

   ```
   yum install kernel-devel
   <!--NeedCopy-->
   ```

   SUSE 12.2:

   ```
   zypper install kernel-devel

   zypper install kernel-source
   <!--NeedCopy-->
   ```

   SUSE 11.4:


```
zypper install kernel-source
<!--NeedCopy-->
```

Ubuntu 16.04:

```
apt-get install linux-headers
<!--NeedCopy-->
```

> **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:

```
#KDIR = $(BUILD_PREFIX)/lib/modules/$(KVERSION)/build

KDIR = /usr/src/kernels/3.10.0-327.10.1.el7.x86_64
<!--NeedCopy-->
```

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.

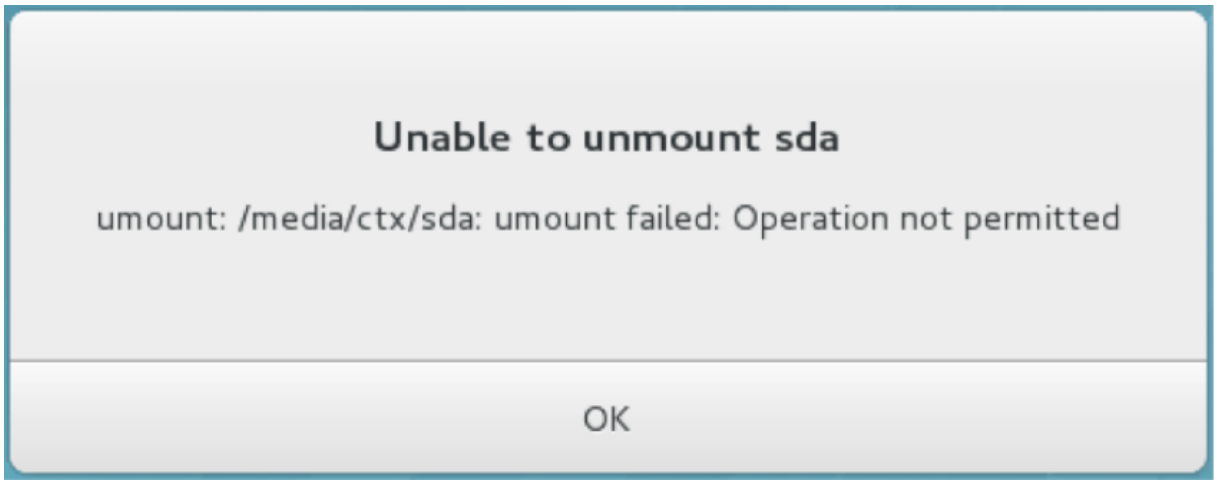
7. Replace the kernel module with the newly built one: **cp -f usb-vhci-*.ko /opt/Citrix/VDA/lib64/**
8. Restart the USB service: **service ctxusbsd restart**
9. Log off and back on to the session again. Check whether USB redirection is functioning.

Troubleshoot USB redirection issues

Use the information in this section to troubleshoot various issues that you might encounter when using the Linux VDA.

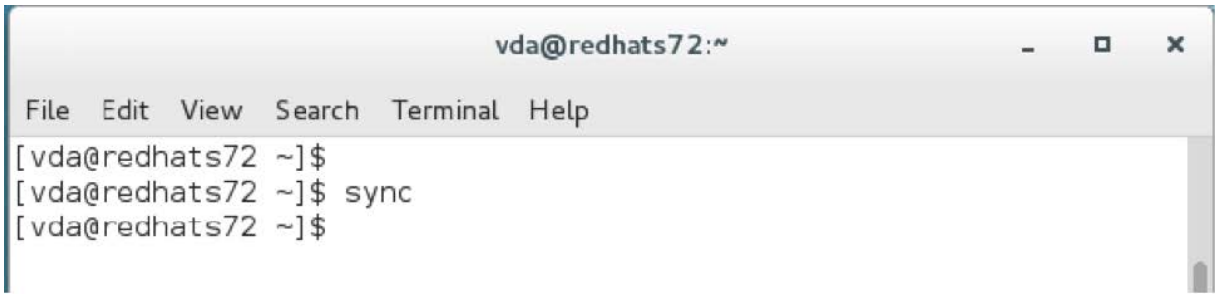
Unable to unmount the redirected USB disk

For the access control of all USB disks redirected from Citrix Receiver, 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

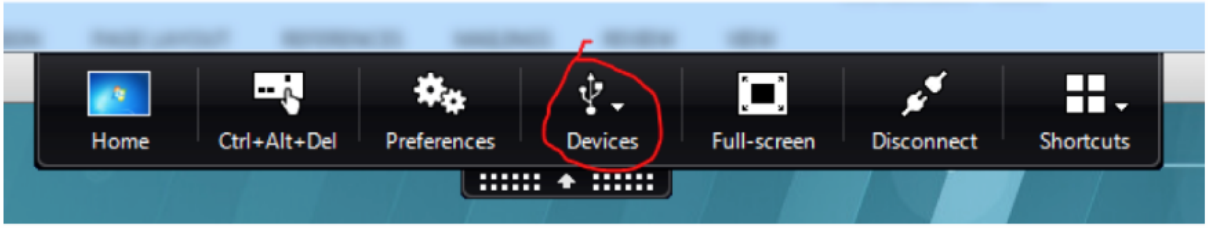
If you redirect a USB disk to a session and try to modify it (for example, create some files on the disk), then stop redirecting it immediately using the toolbar of Citrix Receiver, the file you modified or created can be lost. This issue occurs because when you write data to a file system, the system 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 Receiver, 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.



No devices in the toolbar of Citrix Receiver

Sometimes, you might not be able to see devices listed in the toolbar of Citrix Receiver, 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 Receiver for Linux.

Failed redirection when USB devices can be seen in the toolbar of Citrix Receiver, 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 Receiver. A device might be blocked by the registry setting of Citrix Receiver. 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.

Client Input Method Editor (IME)

June 18, 2020

Overview

Double-byte characters such as Chinese, Japanese, and Korean characters must be typed through an IME. Type such characters with 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 XenDesktop or XenApp session as per usual.

Change your input method as required on the client side to start using the Client IME.

Known issues

- Double-clicking a cell in a Google spreadsheet is a must before you can use the client IME feature 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 the input area.
- Client IME is not supported in a SUSE 11 distribution.

HDX Insight

August 20, 2022

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 NetScaler or Citrix SD-WAN 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 NetScaler between Citrix Workspace app and the Linux VDA.

You must set up a NetScaler Insight Center deployment with the Linux VDA and enable the HDX Insight feature. You can migrate your NetScaler Insight Center deployment to Citrix ADM without losing the existing configuration, settings, or data. For more information, see Migrate from NetScaler Insight Center to Citrix ADM.

Troubleshooting

No data points are displayed

There might be two causes:

- HDX Insight is not configured correctly.

  For example, AppFlow is not enabled on NetScaler or an incorrect instance of NetScaler is configured on the Insight Center.

- 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.

Tracing On

June 18, 2020

Overview

Collecting logs and reproducing issues slow down the diagnostics and degrade the user experience. The Tracing On feature eases 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:

```
setlog help
<!--NeedCopy-->
```


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:

```
setlog values

log_path (Log Output Path) = /var/log/xdl/hdx.log

log_size (Max Log Size (MiB)) = 200

log_count (Max Old Log Files) = 2
<!--NeedCopy-->
```

View or set a single `setlog` value:

```
setlog value <name> [<value>]
<!--NeedCopy-->
```

For example:

```
setlog value log_size 100
<!--NeedCopy-->
```

Levels By default, the log level is set to **Warnings**.

View the log levels set for different components:

```
setlog levels
<!--NeedCopy-->
```

You can set all log levels (including Disable, Inherited, Verbose, Information, Warnings, Errors, and Fatal Errors) by using the following command:

```
setlog level <class> [<level>]
<!--NeedCopy-->
```

The **<class>** variable specifies one component of the Linux VDA. To cover all components, set it to all:

```
setlog level all error

Setting log class ALL to ERROR.
<!--NeedCopy-->
```

Flags By default, the flags are set as follows:

```
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
<!--NeedCopy-->
```

View the current flags:

```
setlog flags
<!--NeedCopy-->
```

View or set a single log flag:

```
setlog flag <flag> [<state>]
<!--NeedCopy-->
```

Restore Defaults Revert all levels, flags, and values to the default settings:

```
setlog default
<!--NeedCopy-->
```

> **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:

```
Apr 1 02:28:21 RH72 citrix-ctxlogd[17881]: Failed to open logging
   configuration file.

Apr 1 02:28:21 RH72 systemd: ctxlogd.service: main process exited, code
   =exited, status=1/FAILURE

Apr 1 02:28:21 RH72 systemd: Unit ctxlogd.service entered failed state.

Apr 1 02:28:21 RH72 systemd: ctxlogd.service failed.
<!--NeedCopy-->
```

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.

Configure unauthenticated sessions

June 18, 2020

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 Receiver for Android.

Create an unauthenticated store

To support an unauthenticated session on the Linux VDA, create an unauthenticated store using StoreFront.

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 XenApp and XenDesktop 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 0
    x00000005
<!--NeedCopy-->
```

Set the maximum number of unauthenticated users

To set the maximum number of unauthenticated users, use the registry key **MaxAnonymousUser-Number**. 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 0
    x00000020
<!--NeedCopy-->
```

> **Important:**
>
> 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):

```
sudo /opt/Citrix/VDA/bin/ctxreg read –k "HKLM\System\CurrentControlSet
    \Control\Citrix" –v MaxAnonymousUserNumber
<!--NeedCopy-->
```

Verify that the **ncsd** service is running and configured to enable **passwd** cache:

```
ps uax | grep nscd
cat /etc/nscd.conf | grep 'passwd' | grep 'enable-cache'
<!--NeedCopy-->
```


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
<!--NeedCopy-->
```

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, modify the system-wide **kdeglobals** file to remove the **[$i]** 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

October 7, 2021

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:


```
/opt/Citrix/VDA/sbin/enable_ldaps.sh -Enable pathToRootCA
<!--NeedCopy-->
```

- Fall back to LDAP without SSL/TLS

```
/opt/Citrix/VDA/sbin/enable_ldaps.sh -Disable
<!--NeedCopy-->
```

The Java keystore dedicated for LDAPS is located in **/etc/xdl/.keystore**. Affected registry keys include:

```
HKLM\Software\Citrix\VirtualDesktopAgent\ListOfLDAPServers

HKLM\Software\Citrix\VirtualDesktopAgent\ListOfLDAPServersForPolicy

HKLM\Software\Citrix\VirtualDesktopAgent\UseLDAPS

HKLM\Software\Policies\Citrix\VirtualDesktopAgent\Keystore
<!--NeedCopy-->
```

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:

```
ldap.conf:

TLS_REQCERT never
```



```
4
5  smb.conf:
6
7  ldap ssl = start tls
8
9  ldap ssl ads = yes
10
11 client ldap sasl wrapping = plain
12 <!--NeedCopy-->
```

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:

```
1  smb.conf:
2
3  ldap ssl = off
4
5  ldap ssl ads = no
6
7  client ldap sasl wrapping = seal
8  <!--NeedCopy-->
```

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 port 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

June 18, 2020

The Linux VDA supports environments that use X11 display functionality (including `xterm` and `gvim`) for interactive remoting. 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 run 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:

```
# xauth -f ~/.Xauthority

# > list

# > us01msip06:107  MIT-MAGIC-COOKIE-1
      fb228d1b695729242616c5908f11624b
<!--NeedCopy-->
```

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:

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\
   CurrentControlSet\Control\Citrix\Xorg" -t "REG_DWORD" -v "
   XauthEnabled" -d "0x00000001" --force

sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\
   CurrentControlSet\Control\Citrix\Xorg" -t "REG_DWORD" -v "ListenTCP"
    -d "0x00000001" --force
<!--NeedCopy-->
```

After enabling Xauthority, pass the `.Xauthority` file to XClient manually or by mounting a shared home directory:

- 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:

```
export DISPLAY={
Display number stored in the Xauthority file }


export XAUTHORITY={
the file path of .Xauthority }

<!--NeedCopy-->
```

> **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:

   ```
   ps aux | grep -i xorg
   <!--NeedCopy-->
   ```

   This command displays the Xorg process and the parameters passed to Xorg while starting. Another parameter displays which `.Xauthority` file is used. For example:

   ```
   /var/xdl/xauth/.Xauthority110
   <!--NeedCopy-->
   ```

   Display the cookies using the **Xauth** command:

   ```
   Xauth -f /var/xdl/xauth/.Xauthority110
   <!--NeedCopy-->
   ```


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:

   ```
   telnet 10.158.11.11 6160
   <!--NeedCopy-->
   ```

   The port number is the sum of 6000 + <display number>.

   If this telnet operation fails, the firewall might be blocking the request.