# Linux Virtual Desktop

## Installation Guide for Red Hat Enterprise Linux

Technology Preview Release

CITRIX

# Glossary

| | |
|---|---|
| **Broker** | XenDesktop component responsible for brokering HDX sessions to the different VDAs within a XenDesktop deployment. Also known as the DDC or XenDesktop Delivery Controller. |
| **Broker Agent** | Component on the Linux VDA machine providing the desktop to be delivered. The Broker Agent communicates with the Broker to enable the brokering of sessions. It is composed of two key components, the VDA Service and the HDX Service. |
| **Citrix Director** | Citrix helpdesk/support console for monitoring and controlling XenDesktop VDAs. |
| **Citrix Studio** | Citrix administration console used to configure XenDesktop. |
| **DDC** | XenDesktop Desktop Delivery Controller. Also known as the Broker or Delivery Controller. |
| **HDX** | High Definition Experience protocol. Formerly known as the Citrix ICA protocol. |
| **HDX Service** | The Linux daemon that remotes the desktop being delivered. It communicates with the VDA service to enable the brokering of sessions. |
| **RHEL** | Red Hat Enterprise Linux. A commercial Linux distribution provided by Red Hat. |
| **VDA** | Virtual Desktop Agent. Also referred to as the Broker Agent. |
| **VDA Service** | The Linux daemon that communicates with the Broker to enable the brokering of sessions. It also communicates with the HDX Service for remote session delivery. |

# Table of Contents

# 1 Introduction

This document is a guide for installing the Linux Virtual Desktop Technology Preview Release product on Red Hat Enterprise Linux Workstation and Server 6.6. Please follow each section in order to ensure a successful installation.

The Linux shell commands used in this document have been verified to work with the GNU Bash shell only.

# 2 System Requirements

## 2.1 Linux Distributions

The following are the supported Linux distributions for the Linux Virtual Desktop product.

- Red Hat Enterprise Linux Workstation 6.6
- Red Hat Enterprise Linux Server 6.6
- SUSE Linux Enterprise Desktop 11 Service Pack 3
- SUSE Linux Enterprise Server 11 Service Pack 3

This document only describes the installation of the VDA product on Red Hat Enterprise Linux. A separate guide is provided for installation on SUSE Linux Enterprise.

In all cases, the only processor architecture supported is x86-64. 32-bit Linux distributions are not supported.

## 2.2 XenDesktop

The following lists the versions of XenDesktop supported by the Linux VDA.

- XenDesktop 7.1
- XenDesktop 7.5
- XenDesktop 7.6

The configuration process for Linux VDAs differs slightly than for Windows VDAs. However, any Delivery Controller farm is capable of brokering both Windows and Linux desktops.

The Linux VDA is incompatible with XenDesktop version 7.0 or earlier.

## 2.3 Citrix Receiver

The following lists the versions of Citrix Receiver supported by the Linux VDA.

- Windows Receiver version v4.2 or newer (This equates to v14.0 of wfica32.exe)
- Linux Receiver version v13.0 or newer
- Mac OSX Receiver v11.8.2 - see note (1) below
- Android Receiver available from the Google Play store
- iOS Receiver 5.9.4 or newer
- HTML5 Receiver 16.0 (only via Access Gateway)

(1) Note that the Citrix Mac Receiver on the App Store is out of date and not supported. More information about the latest Mac OSX Receiver is covered in the Known Issues section.

## 2.4 Hypervisors

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

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

Bare metal hosting is also supported.

## 2.5 Active Directory Integration Packages

The following lists the Active Directory integration packages or products supported by the Linux VDA.

- Samba Winbind
- Quest Authentication Services v4.1 or newer

# 3 Configure Delivery Controllers

## 3.1 Update Delivery Controller Configuration

A PowerShell script named **Update-BrokerServiceConfig.ps1** is provided which will update the Broker service configuration to support Linux VDA session brokering. This script is available within the installation package.

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

1. Copy the **Update-BrokerServiceConfig.ps1** script to the Delivery Controller machine.
2. Open a Windows PowerShell console in the context of the local Administrator.
3. Locate and change to the folder containing the script.
4. Execute the script:

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

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

```
Set-ExecutionPolicy Unrestricted
```

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

These changes will have no impact on the brokering of Windows VDA's configured to use the same Delivery Controller farm. This allows for a single Controller farm to manage and broker sessions to both Windows and Linux VDAs seamlessly.

## 3.2 Verify Delivery Controller Configuration

To verify whether the required configuration changes have been applied to a Delivery Controller, confirm the string **EndpointLinux** appears 5 times in the file:

```
%PROGRAMFILES%\Citrix\Broker\Service\BrokerService.exe.config
```

From the Windows command prompt, logged on as a local administrator:

```
cd "%PROGRAMFILES%"\Citrix\Broker\Service
findstr EndpointLinux BrokerService.exe.config
```

# 4 Prepare Linux Machine for VDA Installation

## 4.1 Verify Network Configuration

It is recommended that the network is connected and properly configured correctly before proceeding.

### 4.1.1 Assign Loopback Address to Hostname

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

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

For example:

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

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

### 4.1.2 Check Hostname

Check the hostname is set correctly:

```
hostname -f
```

This should return the machine's fully qualified domain name (FQDN).

### 4.1.3 Check Name Resolution and Service Reachability

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

```
nslookup domain-controller-fqdn
ping domain-controller-fqdn
nslookup delivery-controller-fqdn
ping delivery-controller-fqdn
```

## 4.2 Configure NTP Service

Maintaining accurate clock synchronization between the VDAs, XenDesktop Controllers and domain controllers is crucial. 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 **System** > **Administration** > **Date & Time**.
2. In the **Date and Time** tab, ensure **Synchronize date and time over the network** is checked.
3. In the **NTP Servers** list, click **Add**.
4. Enter the hostname or IP address of a local NTP Server. This is typically the hostname of the AD Domain Controller. The configuration tool will test for service reachability.
5. Optionally delete the other default NTP servers listed.
6. Click **OK**. This will restart the NTP daemon.

## 4.3 Disable Network Proxy Authentication Popup

There is a specific RHEL 6 issue that causes users to receive a popup asking for the root password after logging on.

To workaround this issue, as root, create the file **/etc/polkit-1/localauthority/30-site.d/20-no-show-proxy-dialog.pkla** in a text editor and add the following content:

```
[No Show Proxy Dialog]
Identity=unix-user:*
Action=org.freedesktop.packagekit.system-network-proxy-configure
ResultAny=no
ResultInactive=no
ResultActive=no
```

For more information on this issue, see https://access.redhat.com/solutions/195833#md25. The correct workaround is described in the comments section.

## 4.4 Install Linux VDA Dependent Packages

### 4.4.1 Install OpenJDK

The Linux VDA has dependencies on OpenJDK 1.7.0. The runtime environment should have been installed as part of the operating system installation; confirm this with:

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

On RHEL 6.6, the pre-packaged OpenJDK is about 10 patch versions out-of-date. Update to the latest version as required:

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

This will install java under **/usr/lib/jvm**. Set the **JAVA_HOME** environment variable by adding the following line to **~/.bashrc** file:

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

Open a new shell and verify the version of Java:

```
java -version
```

To avoid problems, do not install Oracle Java, IBM Java or multiple versions of OpenJDK.

### 4.4.2 Install PostgreSQL

The Linux VDA requires PostgreSQL 8.4 or later. On RHEL 6.6, the latest version is 8.4.20.

Install the following packages:

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

The following post-installation step is required to initialize the database and ensure service starts on boot. This will create database files under **/var/lib/pgsql/data**.

```
sudo service postgresql initdb
```

To ensure the **postgresql** service starts on boot and to start the service now:

```
sudo chkconfig postgresql on
sudo service postgresql start
```

Check the version of PostgreSQL using:

```
psql --version
```

Check the data directory is set using the **psql** command-line utility:

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

### 4.4.3 Install Other Packages

Install the other required packages:

```
sudo yum -y install redhat-lsb-core
sudo yum -y install ImageMagick
sudo yum –y install openmotif
```

## 4.5 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 running the Linux machine on bare metal hardware.

### 4.5.1 Citrix XenServer

#### *4.5.1.1 Fix Time Synchronization*

If the XenServer Time Sync feature is enabled, within each paravirtualized Linux VM you will 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 only. This requires disabling host time synchronization. No changes are required in HVM mode.

If running a paravirtualized Linux kernel with XenTools installed, you can check whether the XenServer Time Sync feature is enabled from within the Linux VM:

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

This will return either:

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

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

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

To make this change permanent and persist after reboot, edit the **/etc/sysctl.conf** file and add the line:

```
xen.independent_wallclock = 1
```

To verify these changes, reboot the system:

```
reboot
```

After reboot, check that this has been set correctly:

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

This should return the value **1**.

### 4.5.2 Microsoft Hyper-V

#### *4.5.2.1 Fix Time Synchronization*

Linux VMs with Hyper-V Linux Integration Services installed can leverage the Hyper-V time synchronization feature to use the host operating system's time. To ensure the system clock remains accurate, this feature should 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 **Time synchronization** is checked.

Note that this approach is different from VMware and XenServer, where host time synchronization is disabled to avoid conflicts with NTP. Hyper-V time synchronization can co-exist and supplement NTP time synchronization.

### 4.5.3 VMware ESX and ESXi

#### *4.5.3.1 Fix Time Synchronization*

If the VMware Time Synchronization feature is enabled, within each paravirtualized Linux VM you will 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 only. This requires disabling host time synchronization.

If 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, uncheck **Synchronize guest time with host**.

## 4.6 Add Machine to Windows Domain

There are currently two methods for adding Linux machines to the Windows domain that are supported by the Linux VDA:

1. Samba Winbind
2. Quest Authentication Service

Follow the instructions below for your chosen method.

### 4.6.1 Samba Winbind

#### *4.6.1.1 Install or Update Required Packages*

##### *4.6.1.1.1 Samba/Winbind*

The standard RHEL installation process will install the Samba/Winbind v3.x packages required for the Linux VDA, and enable the required winbindd service. Install the required packages if not already installed:

```
sudo yum -y install samba-winbind
```

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

```
sudo yum -y update samba-winbind
```

The RHEL repositories also provide Samba v4 packages but are not installed by default. These are not required and should not be installed.

##### *4.6.1.1.2 Kerberos*

The standard RHEL installation process will install the Kerberos v5 packages required for the Linux VDA. Install the required packages if not already installed:

```
sudo yum -y install krb5-workstation
```

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

```
sudo yum -y update krb5-workstation
```

#### *4.6.1.2 Enable Winbind Daemon to Start on Boot*

The Winbind Daemon must be configured to start on boot:

```
sudo chkconfig winbind on
```

#### *4.6.1.3 Configure Kerberos for Winbind*

By default, Winbind will not create the system keytab file **/etc/krb5.keytab** when joining the domain, which implies the Kerberos tools and libraries won't be able to authenticate the machine account.

To force Winbind to create and maintain the system keytab file and automatically renew tickets, open **/etc/samba/smb.conf** and add the following entries under the **[Global]** section:

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

The first setting will create the system keytab when the machine is joined to the domain. If the machine is already domain joined, you can force the creation of the keytab file by changing machine keys:

```
sudo net ads changetrustpw
```

#### *4.6.1.4 Join Windows Domain*

This requires that your domain controller is reachable and you have a Windows account with permissions to add machines to the domain.

1. Open **System** > **Administration** > **Authentication**.
2. On **Identity & Authentication** tab, change **User Account Database** to **Winbind**.
3. Set the **Security Model** to **ads**.
4. Enter values for each of the following fields:

    - **Winbind Domain** - Enter the NetBIOS name of the AD domain, which may be different from the Windows ADS Realm name.
    - **Winbind ADS Realm** - Enter the Kerberos realm name for the domain. This must be specified in uppercase.
    - **Windows Domain Controllers** - Enter the FQDN of the AD domain controller.

5. Change Template Shell to **/bin/bash**.
6. Click **Join Domain...**.
7. Save configuration when prompted.
8. When prompted, enter the credentials of a domain user with permission to add machines to the domain. If successful, control will return the **Authentication Configuration** window.
9. Change to the **Advanced Options** tab.
10. Under **Other Authentication Options**, check **Create home directories on the first login**.
11. Click **Apply**.

#### *4.6.1.5 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
```

These changes require restarting the Winbind daemon:

```
sudo service winbind restart
```

Note that the winbind process will only continue to run if the machine is joined to a domain.

#### *4.6.1.6 Verify Domain Membership*

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

Verify the machine is joined to a domain using Samba's net ads command:

```
sudo net ads testjoin
```

Additional domain and computer object information can be verified with:

```
sudo net ads info
```

#### *4.6.1.7 Verify Kerberos Configuration*

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

```
sudo klist -ke
```

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

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

The machine and realm names must be specified in uppercase, and 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 the realm name is used. If this command is successful, no output will be displayed.

Check the TGT ticket for the machine account has been cached using:

```
sudo klist
```

Examine the machine account details using:

```
net ads status
```

#### *4.6.1.8 Verify User Authentication*

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

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

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

To verify that the Winbind PAM module is configured correctly, logon locally with a domain user account that has not logged onto the machine previously.

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

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

```
ls /tmp/krb5cc_uid
```

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

```
klist
```

Exit the session:

```
exit
```

A similar test can be performed by logging on via Gnome Display Manager.

### 4.6.2 Quest Authentication Service

#### *4.6.2.1 Configure Quest on Domain Controller*

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

##### *4.6.2.1.1 Enable Domain Users to Logon to Linux VDA Machines*

For each domain user that needs to establish HDX sessions on a Linux VDA machine:

1. Open AD 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 that these instructions are equivalent for setting up domain users for logon using the console, RDP, SSH or any other remoting protocol.

#### *4.6.2.2 Configure Quest on Linux VDA*

#### *4.6.2.3 Workaround SELinux Policy Enforcement*

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

There are a few ways to workaround this as outlined at:
https://support.software.dell.com/authentication-services/kb/70022.

The easiest method is to disable SELinux:

As root, edit **/etc/selinux/config** and change the **SELinux** setting:

```
SELINUX=disabled
```

This change requires a reboot:

```
reboot
```

Note: Take care with this setting. Re-enabling SELinux policy enforcement after disabling can cause a complete lockout, even for the root user and other local users.

##### *4.6.2.3.1 Configure Auto Ticket Renewal*

Auto-renewal of Kerberos tickets needs to be enabled:

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

This sets the renewal interval to 9 hours (32400 seconds) which is an hour less than the default 10 hour ticket lifetime. This value will need to be set to a lower value on systems with a shorter ticket lifetime.

##### *4.6.2.3.2 Configure PAM and NSS*

Quest requires that PAM and NSS be manually configured to enable domain user login via HDX and other services such as su, ssh, and RDP. To configure PAM and NSS:

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

##### *4.6.2.3.3 Join Windows Domain*

Join the Linux machine to the AD domain using the Quest **vastool** command:

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

The **user** is any domain user with permissions to join machines to the Windows domain. The **domain-name** is the DNS name of the domain; for example, **example.com**.

#### *4.6.2.3.4 Verify Domain Membership*

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

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

If the machine is joined to a domain this will return the domain name. If not joined, you will see the following error:

```
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
```

#### *4.6.2.3.5 Verify User Authentication*

To verify that the Quest can authenticate domain users using PAM, logon with a domain user account that has not logged onto the machine previously.

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

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

```
ls /tmp/krb5cc_uid
```

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

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

Exit the session:

```
exit
```

A similar test can be performed by logging on via Gnome Display Manager.

## 4.7 Add Linux Machine to Machine Catalog

The process for creating machine catalogs and adding Linux VDA machines is very similar to the traditional Windows VDA approach. Refer to the online Citrix Product documentation for a more complete description of how to complete these tasks.

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

- For operating system, select **Window Server OS**. Citrix Studio does not yet support the notion of a "Linux OS"; however, selecting **Windows Server OS** here implies an equivalent shared desktops delivery model. The **Windows Desktop OS** option implies a single user per machine delivery model, which is not applicable to Linux VDAs. Even for "Desktop" Linux distributions, **Windows Server OS** must be selected.
- Ensure machines are set as not power managed.
- As PVS and MCS are not supported for Linux VDAs, choose the **Another service or technology** (existing images) deployment method.

- For XenDesktop version 7.6 and newer: When adding machines, set the VDA version installed as **7.0 (or newer)**. The Linux VDA does not yet support XenDesktop 7.6 VDA functionality.
- Do not mix Linux and Windows VDA machines in the same machine catalog.

The Citrix documentation for creating machine catalogs is referenced below:

- **XenDesktop 7.1:** http://support.citrix.com/proddocs/topic/xendesktop-71/cds-create-new-scheme-rho.html
- **XenDesktop 7.5:** http://support.citrix.com/proddocs/topic/xenapp-xendesktop-75/cds-create-new-scheme-rho.html
- **XenDesktop 7.6:** http://support.citrix.com/proddocs/topic/xenapp-xendesktop-76/xad-mach-cat-create.html

Earlier versions of XenDesktop are not supported.

Note that if a Linux machine leaves and is re-joined to the Active Directory domain, the machine will need to be added again to the machine catalog.

## 4.8 Add Delivery Group

The process for creating a delivery group and adding machine catalogs containing Linux VDAs machine is almost identical for Windows VDA machines. Refer to the online Citrix Product documentation for a more complete description of how to complete these tasks.

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

- For delivery type, select **Desktops**. Linux VDA machines do not support application delivery.
- Ensure the AD users and groups you select have been properly configured to logon 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.

The Citrix documentation for creating delivery groups is referenced below:

- **XenDesktop 7.1:** http://support.citrix.com/proddocs/topic/xendesktop-71/cds-create-desktops-t-rho.html
- **XenDesktop 7.5:** http://support.citrix.com/proddocs/topic/xenapp-xendesktop-75/cds-create-desktops-t-rho.html
- **XenDesktop 7.6:** http://support.citrix.com/proddocs/topic/xenapp-xendesktop-76/xad-dg-create.html

Earlier versions of XenDesktop are not supported.

# 5 Install Linux VDA Software

## 5.1 Uninstall Old Version

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

Stop the Linux VDA services:

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

Uninstall the package:

```
sudo rpm -e XenDesktopVDA
```

## 5.2 Install Linux VDA

Install the Linux VDA software using the RPM package manager:

```
sudo rpm -i XenDesktopVDA-0.9.3.106-0.x86_64.rpm
```

## 5.3 Configure Linux VDA

After the installation of the package you will need to configure the VDA by running the **ctxsetup.sh** script. Before making any changes, this script will verify the environment and ensure all dependencies are installed. If required, this script can be re-run at any time to change settings.

This can either be run manually with prompting or automatically with pre-configured responses. Review help about this script before proceeding:

```
sudo /usr/local/sbin/ctxsetup.sh --help
```

### 5.3.1 Prompted Configuration

Run a manual configuration with prompted questions:

```
sudo /usr/local/sbin/ctxsetup.sh
```

### 5.3.2 Automated Configuration

For an automated install, the options required by the setup script can be provided with environment variables. If all of the required variables are present then the script will not prompt the user for any information, allowing the installation process to be scripted.

The supported environment variables are:

- **CTX_XDL_DDC_LIST = list-ddc-fqdns** - A space-separated list of FQDNs for your Delivery Controllers. At least one FQDN must be specified.
- **CTX_XDL_REGISTER_SERIVCE = Y | N** - Whether or not the Linux VDA services should start on boot. This is typically Y.
- **CTX_XDL_ADD_FIREWALL_RULES = Y | N** - Whether or not the firewall exception rules for the Linux VDA should be added to the system. This is typically Y.
- **CTX_XDL_AD_INTEGRATION = 1 | 2** - Specifies which Active Directory integration method to use:

**1** - Samba Winbind
**2** - Quest Authentication Service

- **CTX_XDL_USER_FORMAT = 1 | 2 | 3** - Specifies which user name format to use with the PAM modules:
  **1** - domain\user
  **2** - user@realm
  **3** - user
- **CTX_XDL_START_SERVICE = Y | N** - Whether or not the Linux VDA services are to be started when finished configuring the Linux VDA. This is typically Y.

Set the environment variable and run the configure script:

```
export CTX_XDL_DDC_LIST=list-ddc-fqdns
export CTX_XDL_REGISTER_SERVICE=Y|N
export CTX_XDL_ADD_FIREWALL_RULES=Y|N
export CTX_XDL_AD_INTEGRATION=1|2
export CTX_XDL_USER_FORMAT=1|2|3
export CTX_XDL_START_SERVICE=Y|N
sudo -E /usr/local/sbin/ctxsetup.sh
```

You must provide the **-E** option with **sudo** to the pass the existing environment variables to the new shell it creates. It is recommended that you create a shell script file from the commands above with **#!/bin/bash** on the first line.

Alternatively, all parameters can be specified with a single command:

```
sudo CTX_XDL_DDC_LIST=list-ddc-fqdns \
     CTX_XDL_REGISTER_SERVICE=Y|N \
     CTX_XDL_ADD_FIREWALL_RULES=Y|N \
     CTX_XDL_AD_INTEGRATION=1|2 \
     CTX_XDL_USER_FORMAT=1|2|3 \
     CTX_XDL_START_SERVICE=Y|N \
     /usr/local/sbin/ctxsetup.sh
```

### 5.3.3 Remove Configuration Changes

In some scenarios it may be necessary 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
```

To remove configuration changes:

```
sudo /usr/local/sbin/ctxcleanup.sh
```

This script will delete all configuration data from the database and will make the Linux VDA inoperable.

### 5.3.4 Configuration Logs

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

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

# 6 Run VDA Software

## 6.1 Start Linux VDA

To start the Linux VDA services:

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

## 6.2 Stop Linux VDA

To stop the Linux VDA services:

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

## 6.3 Restart Linux VDA

To restart the Linux VDA services:

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

## 6.4 Check Linux VDA Status

To check the running state of the Linux VDA services:

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

# 7 Uninstall Linux VDA Software

## 7.1 Query Linux VDA Installation Status

To check whether the Linux VDA is installed and view the version of the package installed:

```
rpm -q XenDesktopVDA
```

To view more detailed information:

```
rpm –qi XenDesktopVDA
```

## 7.2 Uninstall Linux VDA

To uninstall the Linux VDA package:

```
sudo rpm -e XenDesktopVDA
```

Uninstalling the Linux VDA software will delete the associated PostgreSQL and other configuration data. However, the PostgreSQL package and other dependent packages that were setup prior to the installation of the Linux VDA will not be removed.

## 7.3 Remove Dependent Packages

This guide does not cover the removal of dependent packages including PostgreSQL.

# 8 Troubleshooting

## 8.1 Check the Linux machine has been prepared correctly

The most common issues are a direct result of Linux machine misconfiguration, mainly around networking, NTP timeserver configuration or Windows domain membership. Fixing the Linux machine's configuration will often resolve issues with the VDA software.

## 8.2 Configure logging and tracing

The method for enabling logging (and tracing) differs between the Broker Agent and the HDX Service.

### 8.2.1 Broker Agent logging

The Broker Agent (also known as the **ctxvda** service) writes log data to:

```
/var/log/xdl/vda.log
```

Tail the content of this file using:

```
tail -f /var/log/xdl/vda.log
```

The Broker Agent will roll the log file over at 10MB, archiving it to a gzip compressed file in the same directory. These files have the format:

```
vda.log.N.gz
```

Where **N** is 1 to 10. To conserve disk space, only the last 10 archived log files are retained. To view an archived log file decompress the file using **gunzip** or display directly with **zcat** or **zmore**.

### 8.2.2 HDX Service tracing

Unlike the Broker Agent, the HDX Service does not trace or log anything by default. If tracing is required, a separate script provided in the installation package will self-extract the tools required.

Copy **xdl-trace-install.sh** from the installation package onto the Linux VDA machine, preferably in your home directory. Make it executable and run it:

```
chmod 755 xdl-trace-install.sh
./xdl-trace-install.sh
```

This extracts the **setdbg** UI application and configuration file into your **~/bin** directory. Run the application to set logging and tracing options:

```
~/bin/setdbg
```

The HDX service writes trace log data to:

```
/var/log/xdl/hdx.log
```

Tail the content of this file using:

```
tail -f /var/log/xdl/hdx.log
```

The **setdbg** application allows the tracing to be configured for many aspects of the HDX service. The top left drop-down contains the tracing categories, and selecting a particular component provides tracing options for individual components. Any changes made will take immediate effect.

## 8.3 What to try if HDX sessions won't start

Ensure you have no orphaned processes that might be preventing new sessions from starting:

```
sudo pkill -9 ctxfm
sudo pkill -9 ctxgfx
sudo pkill -9 ctxlogin
sudo pkill -9 ctxvfb
```

Restart the Linux VDA services and retry connection.

## 8.4 Verify ownership and permissions of key directories and files

Check the file ownership and permission of the following directories and files:

- **/var** - Owner: root, Group: root, Permissions: 0755
- **/var/xdl** - Owner: ctxsrvr, Group: ctxadm, Permissions: 0755
- **/var/xdl/.isacagent** - Owner: root, Group: root, Permissions: 0666
- **/var/xdl/.winsta** - Owner: ctxsrvr, Group: ctxadm, Permissions: 0777
- **/var/xdl/vda** - Owner: root, Group: root, Permissions: 0755

## 8.5 Try a direct HDX connection

Try to make a direct HDX connection using an ICA file. This bypasses the XenDesktop Controller and makes the Citrix Receiver connect directly to the Linux VDA. This is also referred to as an unbrokered connection.

Below is a sample ICA file created with the **.ica** extension. Be sure to change the **Address**, **Username**, and **ClearPassword** entries.

```
[WFClient]
Version=2

[ApplicationServers]
XDL=

[XDL]
; Change the following to match your environment
Address=ip-address-of-linux-vda-machine
Username=domain\user
ClearPassword=password
Compress=On
DesiredHRES=1024
DesiredVRES=768
DesiredColor=8
TransportDriver=TCP/IP
WinStationDriver=ICA 3.0
ConnectionBar=1
```

Ensure the **Username** field includes the domain name in uppercase. The username can also be provided in UPN format:

```
user@domain.net
```

Another approach is to try making a direct HDX connection with local Linux user credentials instead of domain user credentials. This helps isolate whether there are domain authentication issues, which are often caused by misconfigured networking or Active Directory integration.

# 9 Known Issues

## 9.1 General issues

- Linux VDA only supports connection from the HTML5 Receiver through Citrix Access Gateway. HDX sessions connecting to the Linux VDA in this way will not appear in Citrix Studio or Citrix Director.

## 9.2 HDX issues

- Linux VDA has issues with some keyboard shortcut combinations involving the Windows key (on Windows) or the Command key (on Mac), including those available in the HDX Connection Bar.
- Keyboard input may be lost when reconnecting to a HDX session on Linux VDA with the screen locked using an Android or iOS Receiver. Disconnecting and reconnecting to the HDX session should resolve the issue.
- Session roaming from an Android Receiver to a Windows Receiver or Mac OSX Receiver may result in a black screen. Disconnecting and reconnecting to the HDX session should resolve the issue.
- The time zone of the client is not mapped into the Linux desktop in a HDX session.
- The keyboard layout is not mapped into the Linux desktop in a HDX session from a non-Windows Receiver. Refer to http://support.citrix.com/article/CTX129166 for how to setting the **KeyboardLayout** configuration parameter in the Receiver's **wfclient.ini** file.
- Linux VDA HDX sessions from the Android or iOS Receiver may freeze or disconnect when the session is rotated continuously.
- Linux VDA clipboard virtual channel only supports plain text and images.
- HDX pass-through does not work. The Citrix Receiver installed in the remote Linux HDX session cannot connect to another machine, Linux or Windows.
- The Linux VDA only supports HDX sessions with a single monitor and maximum resolution of 3072 x 3072.

## 9.3 Linux Broker Agent issues

- Linux Broker Agent does not support DNS CNAME lookups for XenDesktop Delivery Controller addresses.
- Linux Broker Agent does not support DNS-based lookup of Kerberos realm names; it currently only supports realm mapping in the krb5.conf file.
- Linux VDA does not auto-refresh Kerberos TGTs for the domain users with HDX sessions; this is the responsibility of the PAM module of the Active Directory integration tool used. An expired TGT may cause issues when trying to reconnect to a disconnected HDX session.
- Linux Broker Agent does not support multi-forest Active Directory environments.
- Linux Broker Agent uses message-based encryption when connecting to the XenDesktop Delivery Controller; it does not support SSL/TLS connections to the XenDesktop Delivery Controller.
- Linux Broker Agent does not currently support NetBIOS name truncation. The workaround is to limit the hostname to 15 characters.

- Linux Broker Agent does not handle power state changes in the same way as the Windows VDA – it does not unregister with the XenDesktop Delivery Controller when the machine or VM is suspended.
- Linux Broker Agent does not auto-detect when its IP address changes. A service restart is required to pick up the new IP address.
- VDA registration fails after leaving and re-joining an Active Directory domain. This is due to Winbind removing the computer object from Active Directory on leaving the domain. This means Linux machines get new computer SIDs when they re-join the domain and, as a result, must be re-added to the Machine Catalog and Delivery Group. Windows disables the computer object on leaving the domain and re-activates it when re-joining the domain. This means Windows machines have the same computer SID and, as a result, do not need to be re-added to the Machine Catalog and Delivery Group.