Linux Virtual Delivery Agent 7.16

Nov 27, 2017

This pdf file includes the Linux Virtual Delivery Agent 7.16 documentation. You can save a local copy of this file and use it offline. Use the built-in Search and Bookmark features to find what you need.

The Linux Virtual Delivery Agent (VDA) enables access to Linux virtual desktops and applications, anywhere, from any device where Citrix Receiver is installed.

You can create Linux virtual desktops and applications based on an RHEL, CentOS, SUSE, or Ubuntu distribution. Prepare your Linux virtual machines, install the new software on them, configure your Delivery Controller, and then use Citrix Studio to make the desktops and applications available to users.

Current release: 7.16

For information about earlier Linux Virtual Delivery Agent releases, see the following sections:

- Linux Virtual Delivery Agent 7.15
- Linux Virtual Delivery Agent 7.14
- Linux Virtual Delivery Agent 7.13
- Linux Virtual Delivery Agent 7.12
- Linux Virtual Delivery Agent 7.12
- Linux Virtual Delivery Agent 1.4
What's new

Nov 27, 2017

What's new in 7.16

Version 7.16 of the Linux VDA includes the following new features:

Session shadowing

Enabled by default as of this release, the session shadowing feature allows domain administrators to view users' ICA sessions in an intranet. The feature uses noVNC to connect to the ICA sessions and is currently supported only with RHEL 7.3 and Ubuntu 16.04. For more information, see Shadow sessions.

SSL encryption for secure user sessions

The Linux VDA supports SSL encryption for secure user sessions. For more information, see Secure user sessions using SSL.

Session reliability

Session reliability reconnects ICA sessions seamlessly across network interruptions. With this release, Citrix is introducing the session reliability feature to all supported Linux platforms. For more information, see Configure session reliability.

Support for multiple language inputs

This feature provides a language bar for published applications that accept text input. The language bar enables users to select a server-side IME and to switch between keyboard layouts during an application session. For more information, see Support for multiple language inputs.

Adaptive transport

With this release, Citrix is introducing adaptive transport to all supported Linux platforms as an experimental feature. For more information, see Adaptive transport.

Dependency change

Two new dependencies, python-websockify and x11vnc, are required for session shadowing. The python-websockify and x11vnc dependencies are automatically installed when you install the Linux VDA on Ubuntu 16.04. On RHEL 7.3, you must manually install python-websockify and x11vnc after you install the Linux VDA. For more information, see the dependency list in the respective installation guide of the Installation overview section.

https://docs.citrix.com © 1999-2017 Citrix Systems, Inc. All rights reserved. p.2
System behavior changes

- User-level language settings are supported. Different users can choose their own UI languages in a session.
- To change language settings through the Region & Language dialog box of the language bar in a published desktop or application session, ensure that the accounts-service package has been upgraded to Version 0.6.37 or later.
Fixed issues

Nov 27, 2017

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

- Sessions cannot be launched in leased connection mode. [#LNXVDA-1198]

- Sessions cannot be launched after the color depth for simple graphics is set to 8 bits per pixel. [#LNXVDA-2738]

- On a SLES 12.2 VDA, 20 concurrent sessions can cause the `ctxhd` service to exit unexpectedly. [#LNXVDA-3292]

- The **Devices** menu does not appear on the toolbar at the top of the desktop view, and attempts to start the `ctxusbsd` service fail. [#LNXVDA-3412]
Known issues

Nov 27, 2017
The following issues have been identified in this release:

- Sessions cannot be launched in Citrix Receiver for Linux when SSL encryption is enabled.
  [RFLNX-1557]

- The indicator-date-time-service process does not consume the $TZ environment variable. As a result, the unity panel on Ubuntu 16.04 Unity Desktop does not display the time zone of the user device when it is different from the time zone of the user session.
  [LNXVDA-2128]

- Ubuntu graphics: In HDX 3D Pro, a black frame might appear around applications after resizing the desktop viewer, or in some cases, the background can appear black.

- Printers created by the Linux VDA printing redirection might not be deleted 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 can be reversed when session roaming. The CAPS LOCK state can be lost when roaming an existing connection to the 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 connecting 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. It is possible to change this within the Citrix Receiver settings but the results vary with different applications.

- Slow session launches might occur when using the Linux VDA with a Delivery Controller from XenDesktop v7.1. The slow launch is caused by the presence of CGP settings in the ICA file generated by the v7.1 Delivery Controller. When these settings are present, Citrix Receiver attempts to establish a connection on TCP port 2598. The default firewall settings on some Linux distributions, such as SLED 12, is to drop the TCP SYN packets, resulting in a timeout and hence a slow session launch. As a workaround, configure the firewall on the Linux VDA to reject the TCP SYN on port 2598. This issue has been addressed in later versions of the Delivery Controller.

- Registration fails when the Linux VDA is rejoined to the domain. Under certain circumstances, when the Linux VDA is rejoined to the domain and a fresh set of Kerberos keys are generated, the Broker fails to establish a security context with the VDA. This is often caused by the Broker using a cached out-of-date VDA service ticket based on the previous set of Kerberos keys. This won't stop the VDA from connecting to the Broker, but the Broker will not be able to establish a return security context to the VDA. The usual symptom is that the VDA registration fails.

  This problem will eventually resolve itself when the VDA service ticket eventually expires and is renewed, but service tickets are usually long-lived. This could potentially take a long time.

  As a workaround, clear the Broker's ticket cache. You can reboot the Broker or run the following command on the Broker from a command prompt as Administrator:

```bash
# For Citrix XenDesktop 7.13.193190 (or newer)
rm -rf /var/citrix/TicketCache
```
This will purge all service tickets in the LSA cache held by the Network Service principal under which the Citrix Broker Service runs. This will remove service tickets for other VDAs and potentially other services. However, this is harmless – these service tickets will be reacquired from the KDC when needed again.

- Audio plug-n-play is not supported. Citrix recommends that any audio capture device be connected to the client machine before starting to record audio in the ICA session. If a device is attached after the audio recording application has started, the application might become unresponsive. If this issue occurs, restart the application. A similar issue can occur if a capture device is unplugged while recording.

- Citrix Receiver for Windows 10 might experience audio distortion during audio recording.
Third party notices

Nov 27, 2017

Linux Virtual Desktop Version 7.16

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

Nov 27, 2017

Linux distributions

The following Linux distributions are supported by the Linux VDA:

- SUSE Linux Enterprise:
  - Desktop 12 Service Pack 2
  - Server 12 Service Pack 2
- Red Hat Enterprise Linux
  - Workstation 7.3
  - Workstation 6.9
  - Workstation 6.8
  - Server 7.3
  - Server 6.9
  - Server 6.8
- CentOS Linux
  - CentOS 7.3
  - CentOS 6.9
  - CentOS 6.8
- Ubuntu Linux
  - Ubuntu Desktop 16.04 (with the 4.4.x kernel)
  - Ubuntu Server 16.04 (with the 4.4.x kernel)

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

Note

Citrix support for a particular Linux OS platform and version will expire if the support from the OS vendor expires for that Linux OS platform and version.

Important

Gnome and KDE desktops are supported in SUSE, RedHat and CentOS. Unity desktop is supported in Ubuntu only. At least one desktop must be installed.

Tip

CentOS Linux is supported as of Version 1.3.
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 than for Windows VDAs. However, any Delivery Controller farm is capable of brokering 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 4.9
- Citrix Receiver for Linux Version 13.7
- Citrix Receiver for Mac OSX Version 12.7
- Citrix Receiver for Android Version 3.13
- Citrix Receiver for iOS Version 7.3
- Citrix Receiver for Chrome Version 2.5
- Citrix Receiver for HTML5 Version 2.5 (only through NetScaler Gateway)

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.

Tip

Refer to the hypervisor vendor’s documentation for the list of supported platforms.
Active Directory integration packages

The following Active Directory integration packages or products are supported by the Linux VDA:

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

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

HDX 3D Pro

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

Hypervisors

- XenServer
- VMware ESX and ESXi

Note: The hypervisors are compatible with certain Linux distributions.

GPU

The following GPUs are supported for GPU pass-through:

- NVIDIA GTX750Ti
- NVIDIA GRID™ 3.0 - Tesla M60
- NVIDIA GRID™ - K2

The following GPUs are supported for vGPU:

- NVIDIA GRID™ 3.0 - Tesla M60
- NVIDIA GRID™ 3.0 - Tesla M10
Configure Delivery Controllers

Aug 14, 2017

XenDesktop 7.6 or an earlier version requires the necessary changes to support the Linux VDA. Thus, for these versions of XenDesktop, a hotfix or update script is required. The installation and verification are provided in this article.

Update Delivery Controller configuration

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

- CTX142438: Hotfix Update 2 - For Delivery Controller 7.6 (32-bit) – English
- CTX142439: Hotfix Update 2 - For Delivery Controller 7.6 (64-bit) – English

For earlier versions of XenDesktop, you can use the PowerShell script named Update-BrokerServiceConfig.ps1 to update the Broker Service configuration. This script is available in the following package:

- citrix-linuxvda-scripts.zip

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

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

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

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

   ```powershell
   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 VDAs 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.

### Verify Delivery Controller configuration

To verify whether the required configuration changes have been applied to a Delivery Controller, confirm that the `stringEndpointLinux` appears five times in the file:

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

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

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

Aug 14, 2017
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

Nov 27, 2017
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

<table>
<thead>
<tr>
<th></th>
<th>Winbind</th>
<th>SSSD</th>
<th>Centrify</th>
</tr>
</thead>
<tbody>
<tr>
<td>RHEL</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.3</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>6.9</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>6.8</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>CentOS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.3</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>6.9</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>6.8</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Ubuntu</td>
<td>16.04</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
| 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.
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:

- Hostname - Linux VDA server name
- IP address of Domain Name Server
Considerations

1. 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 on the Linux VDA server if it does not exist.
   b. Add the line "workgroup=<your workgroup> to the file.

2. Because 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.

3. For Centrify on CentOS, easy install can fail at “adcheck,” the Centrify environment check tool, and report the following error:

   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.

   This is due to the special configuration of Centrify. Follow the steps below 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.

4. Easy install supports pure IPv6 as of Linux VDA 7.16. This enhancement has the following precondition and limitation:
   • You must configure your Linux repository to make sure that your machine can download required packages in pure IPv6 environments.
   • Centrify is not supported in pure IPv6 environments.
Note: If your environment is pure IPv6 and all your input is in proper IPv6 format, the VDA registers with the Delivery Controller using IPv6. If your environment has a hybrid stack with both IPv4 and IPv6, the type of the first DNS IP address determines whether IPv4 or IPv6 is used for registration.

5. The ctxinstall.sh script needs the Centrify package if you choose Centrify as the method to join a domain. There are two ways for ctxinstall.sh to get the Centrify package:

- **Easy install** helps download the Centrify package from the Internet automatically. Currently the given URLs for each distribution are as follows:
  
  **RHEL:**
  ```
  ```
  
  **CentOS:**
  ```
  ```
  
  **SUSE:**
  ```
  ```
  
  **Ubuntu:**
  ```
  ```

- **Fetch the Centrify package from a local directory:** You must follow the steps below 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:
Step 2: Install the Linux VDA package

Execute the following commands to set up the environment for the Linux VDA.
For RHEL and CentOS distributions:

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

For Ubuntu distributions:

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

For SUSE distributions:

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

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

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

**Interactive mode**

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

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

**Silent mode**

To use easy install in silent mode, you must set the following environment variables before running ctxinstall.sh.
The following variables are used by ctxsetup.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 using a TCP/IP port.
- **CTX_XDL_REGISTER_SERVICE**=Y | N – The Linux Virtual Desktop services support starting during boot.
- **CTX_XDL_ADD_FIREWALL_RULES**=Y | N – The Linux Virtual Desktop services require incoming network connections to be allowed through the system firewall. You can automatically open the required ports (by default ports 80 and 1494) in the system firewall for the Linux Virtual Desktop.
- **CTX_XDL_HDX_3D_PRO**=Y | N – Linux Virtual Desktop supports HDX 3D Pro, a set of graphics acceleration technologies designed to optimize the virtualization of rich graphics applications. HDX 3D Pro requires a compatible NVIDIA GRID graphics card to be installed. If HDX 3D Pro is selected, the VDA is configured for VDI desktops (single-session) mode - (i.e. CTX_XDL_VDI_MODE=Y). This is not supported on SUSE. Ensure this value is set to N.
- **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 to Y.
- **CTX_XDL_SITE_NAME**=dns-name – The Linux VDA discovers LDAP servers using DNS, querying for LDAP service records. 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 by default queries DNS to discover LDAP servers. However, if DNS cannot provide LDAP service records, you can provide a space-separated list of LDAP Fully Qualified Domain Names (FQDNs) with LDAP port (e.g. ad1.mycompany.com:389). If unnecessary, it can be set to ‘<none>’.
- **CTX_XDL_SEARCH_BASE**=search-base – The Linux VDA by default queries LDAP using a search base set to the root of the Active Directory Domain (e.g. DC=mycompany,DC=com). However, to improve search performance, you can specify a search base (e.g. 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 provided that all parameters can be provided by environment variables.

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

### command

| export CTX_EASYINSTALL_HOSTNAME=host-name |

https://docs.citrix.com  © 1999-2017 Citrix Systems, Inc. All rights reserved.  p.19
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_FIREFALL_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=seach-base-set | '<none>'

export CTX_XDL_START_SERVICE=Y | N

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

When running the `sudo` command, enter 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 commands above with `#!/bin/bash` as the first line.

Alternatively, you can specify all parameters with a single command:

```
command
```

https://docs.citrix.com © 1999-2017 Citrix Systems, Inc. All rights reserved. p.21
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=seach-base-set \\

CTX_XDL_START_SERVICE=Y \\

/opt/Citrix/VDA/sbin/ctxinstall.sh

Troubleshooting

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

Failure when joining a domain with SSSD

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

log
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)'.


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 (8))' of type 'class javax.xml.ws.soap.SOAPFaultException'.

2016-11-04 02:11:52 [INFO ] - 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.
/var/log/messages:

To resolve this issue:

1. `rm -f /etc/krb5.keytab`
2. `net ads leave $REALM -U $domain-administrator`
3. Delete the machine catalog and delivery group on the Delivery Controller
4. Execute `/opt/Citrix/VDA/sbin/ctxinstall.sh`
5. Create the machine catalog and delivery group on the Delivery Controller

Ubuntu can launch a session, but is blocked in an empty desktop

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 appears in the same state when you log on with domain administrator credentials.

To resolve this issue:

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

Ubuntu launches a session, but cannot log on due to missing home directory

/var/log/xdl/hdx.log:

Tip

The root cause of this issue is that the home directory is not created for the domain administrator.
To resolve this issue:

1. From a command line, enter **pam-auth-update**.

2. In the resulting popup window, verify that **Create home directory login** is selected.

![Screenshot of the 'Create home directory login' option in a popup window](image)

Session cannot be launched or is quickly terminated with dbus/message bus 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:7fff8c3469d0 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.
```

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

```
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:7fff8c3469d0 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.
```
Some groups or modules do not take effect until after a reboot. When *dbus* or *message bus* error messages appear in the log, Citrix recommends that you reboot the system and then retry.

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

The user can launch a session but cannot log on.

/var/log/ctxinstall.log:

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

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

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

```
If you want to allow polyinstantiation to enabled

Then you must tell SELinux about this by enabling the 'polyinstantiation_enabled' boolean.
```
You can read 'None' man page for more details.

Do

`setsebool -P polyinstantiation_enabled 1`

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

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

Then you should report this as a bug.

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

Do

allow this access for now by executing:

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

# semodule -i mypol.pp
```

To resolve this issue:

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

   SELINUX=disabled

2. Reboot
Install Linux Virtual Delivery Agent for RHEL/CentOS

Nov 27, 2017

You can choose to follow the steps below 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 hostname**

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

```
HOSTNAME=hostname
```

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

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

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

For example:

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

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

---

**Note**

The Linux VDA currently does not support NetBIOS name truncation. Therefore, the hostname 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 hostname with a number and do not end with a hyphen. This rule also applies to Delivery Controller hostnames.
Step 1d: Check the hostname

Verify that the hostname is set correctly:

```
hostname
```

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

Verify that the FQDN is set correctly:

```
hostname -f
```

This command returns the FQDN of the machine.

Step 1e: Check name resolution and service reachability

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

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

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

Step 1f: Configure clock synchronization

Maintaining accurate clock synchronization between the VDAs, Delivery Controllers, and domain controllers is crucial. Hosting the Linux VDA as a virtual machine can cause clock skew problems. For this reason, synchronizing time with a remote time service is preferred.
RHEL 6.x and earlier releases use the NTP daemon (ntpd) for clock synchronization, whereas an RHEL 7.x default environment uses the newer Chrony daemon (chronyd) instead. The configuration and operational process between the two services is similar.

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

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

```
config

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

Save changes and restart the NTP daemon:

```
command

sudo /sbin/service ntpd restart
```

**Configure the Chrony service (RHEL 7/CentOS 7 only)**

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

```
config

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

Save changes and restart the Chrony daemon:

```
command

sudo /sbin/service chronyd restart
```
Step 1g: Install OpenJDK

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

Confirm the correct version with:

- RHEL 7/CentOS 7:
  ```
  sudo yum info java-1.8.0-openjdk
  ```

- RHEL 6/CentOS 6:
  ```
  sudo yum info java-1.7.0-openjdk
  ```

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

- RHEL 7/CentOS 7:
  ```
  sudo yum -y update java-1.8.0-openjdk
  ```

- RHEL 6/CentOS 6:
sudo yum -y update java-1.7.0-openjdk

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

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

Open a new shell and verify the version of Java:

```bash
java --version
```

Tip
To avoid problems, make sure 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:

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

The following post-installation step is required to initialize the database and ensure that the service starts on boot. This will create database files under /var/lib/pgsql/data. The command differs between PostgreSQL 8 and 9:

- **RHEL 7 only**: PostgreSQL 9

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

- RHEL 6 only: PostgreSQL 8

  command

  sudo /sbin/service postgresql initdb

Step 1: Start PostgreSQL

Configure the service to start on boot, and to start now:

- RHEL 7 only: PostgreSQL 9

  command

  sudo systemctl start postgresql

  sudo systemctl enable postgresql

- RHEL 6 only: PostgreSQL 8

  command

  sudo /sbin/chkconfig postgresql on

  sudo /sbin/service postgresql start

Check the version of PostgreSQL using:

command
psql --version

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

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

**Important**

In this release, we add a new dependency for gperftools-libs, but it doesn't exist in the original repository. You need to add a new repository with the following command:

```bash
```

This will affect RHEL 6/ CentOS 6 only and you need to run the command before installing the Linux VDA package.

---

**Step 2: Prepare the hypervisor**

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

**Fix time synchronization on Citrix XenServer**

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

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

```bash
command
```
su -

cat /proc/sys/xen/independent_wallclock

This command returns 0 or 1:

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

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

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

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

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

```bash
xen.independent_wallclock = 1
```

To verify these changes, reboot the system:

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

This command returns the value 1.

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

From the management operating system:

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

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

Fix time synchronization on ESX and ESXi

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

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

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

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

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

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

Follow the instructions below for your chosen method.

Samba Winbind

Install or update the required packages:
sudo yum -y install samba-winbind samba-winbind-clients krb5-workstation authconfig oddjob-mkhomedir

**Enable Winbind daemon to start on boot**

The Winbind daemon must be configured to start on boot:

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

**Configure Winbind Authentication**

Configure the machine for Kerberos authentication using Winbind:

```bash
sudo authconfig --disablecache --disablesssd --disablesssdauth --enablewinbind --enablewinbindauth --disablewinbindoffline --smbsecurity=ads --smbworkgroup=REALM
domain
```

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

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

```bash
--enablekrb5kdcdns --enablekrb5realmdns
```

Ignore any errors returned from the authconfig command about the winbind service failing to start. These are due to authconfig trying 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:
The system keytab file /etc/krb5.keytab is required by the Linux VDA to authenticate and register with the Delivery Controller. The kerberos method setting above will force 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
```

Where REALM is the Kerberos realm name in uppercase, and user is a domain user with 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:
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:

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

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:

```bash
command
sudo net ads testjoin
```

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

```bash
command
sudo net ads info
```

Verify Kerberos configuration

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

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 that the realm name is used. If this command is successful, no output is displayed.

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

```
sudo klist
```

Examine the account details of the machine using:

```
sudo net ads status
```

**Verify user authentication**

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

```
sudo wbinfo
```
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, log on locally with a domain user account that has not logged on to the machine previously:

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

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

```bash
klist
```

Exit the session:

```bash
exit
```

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

**Quest authentication service**

**Configure Quest on domain controller**

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

**Enable domain users to log on to Linux VDA machines**
To enable domain users to establish HDX sessions on a Linux VDA machine:

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

**Note**

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

**Configure Quest on Linux VDA**

**Work around SELinux policy enforcement**

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

**Tip**

There are several ways to work around this issue, as described here.

The easiest way is to disable SELinux. As root, edit `/etc/selinux/config` and change the SELinux setting:

```
SELINUX=permissive
```

This change requires a reboot:

```
reboot
```

**Important**

Use this setting carefully. Reenabling SELinux policy enforcement after disabling can cause a complete lockout, even for the root
Configure VAS daemon

Auto-renewal of Kerberos tickets needs to be enabled and disconnected; authentication (offline logon) needs to 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
```

This sets the renewal interval to 9 hours (32,400 seconds) which is an 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

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
```

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
```

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:

```
command

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

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

```
error

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 using PAM, log on with a domain user account that has not logged on to the machine previously:

```
command

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:

```
command
```
ls /tmp/krb5cc_uid

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

/command

/opt/quest/bin/vastool klist

Exit the session:

/command

exit

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

Centrify DirectControl

Join Windows domain

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

/command

su –

adjoin -w -V -u user domain-name

The user parameter is any Active Directory domain user with 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, whether Windows or Linux, have a computer object in Active Directory. To verify that a Centrify-joined Linux machine is on the domain:
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:

```
su -
adinfo
```

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

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

SSSD

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

**Note**

If you are using SSSD, follow the instructions contained in this section instead of the information provided by the Add Linux machine to Windows domain section.

**What is SSSD?**

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

Setting up SSSD on RHEL and CentOS involves 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 older 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. There are a few methods for achieving this, including:

- adcli
- realmd
- winbind
- samba

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

On the Linux client with properly configured files:

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

Configure the machine for Samba and Kerberos authentication:

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

Where **REALM** is the Kerberos realm name in uppercase and **domain** is the short NetBIOS name of the Active Directory domain.
If DNS-based lookups of the KDC server and realm name is required, add the following two options to the above command:

```
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:

```
config

kerberos method = secrets and keytab
```

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

```
command

sudo net ads join REALM -U user
```

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

Set up SSSD

Setting up SSSD consists of the following steps:

- install the `sssd-ad` package on the Linux client machine
- make configuration changes to various files (for example, `sssd.conf`)
- start the `sssd` service:

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

```
config

[sssd]

config_file_version = 2
```
domains = ad.example.com

services = nss, pam

[domain/ad.example.com]

# Uncomment if you need offline logins
#
# cache_credentials = true

id_provider = ad

auth_provider = ad

access_provider = ad

ldap_id_mapping = true

ldap_schema = ad

# Should be specified as the lower-case version of the long version of the Active Directory domain.

ad_domain = ad.example.com

# Kerberos settings

krb5_ccachedir = /tmp

krb5_ccname_template = FILE:%d/krb5cc_%U
# Uncomment if service discovery is not working

# ad_server = server.ad.example.com

# Comment out if the users have the shell and home dir set on the AD side

default_shell = /bin/bash

fallback_homedir = /home/%d/%u

# Uncomment and adjust if the default principal SHORTNAME$@REALM is not available

# ldap_sasl_authid = host/client.ad.example.com@AD.EXAMPLE.COM

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

Set the file ownership and permissions on sssd.conf:

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

Configure NSS/PAM

**RHEL/CentOS**

Use authconfig to enable SSSD, install oddjob-mkhomedir to make sure home directory creation works with SELinux:

```bash
```

https://docs.citrix.com © 1999-2017 Citrix Systems, Inc. All rights reserved.
authconfig --enablesssd --enablesssdauth --enablemkhomedir --update

sudo service sssd start

sudo chkconfig sssd on

Verify Kerberos configuration

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

```
sudo klist -ke
```

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

```
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 that the realm name is used. If this command is successful, no output is displayed.

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

```
sudo klist
```

Verify user authentication

Use the getent command to verify that the logon format it supported and whether the NSS works.
The `DOMAIN` parameter indicates the short version domain name. If another logon format from Citrix Receiver 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, log on locally with a domain user account that has not logged on to the machine previously.

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

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

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

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

```
klist
```

---

Install NVIDIA GRID drivers

---
To enable HDX 3D Pro, additional installation steps are required to install the requisite graphics drivers on the hypervisor as well as 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 Passthrough mode to the VM.
3. Start the RHEL VM.

To prepare the machine for the NVIDIA GRID drivers, the following steps are required:

```command
yum install gcc
yum install "kernel-devel-uname-r == $(uname -r)"
systemctl set-default multi-user.target
```

Once complete, follow the steps in the [Red Hat Enterprise Linux document](https://docs.citrix.com) to install the NVIDIA GRID driver.

**Note**

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

**Important**

Once GPU Pass through has been enabled, the Linux VM is no longer accessible via XenCenter so you need to use SSH to connect.
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 the license, follow the product documentation from “GRID Licensing Guide.pdf - DU-07757-001 September 2015”.

## Step 4: Install the Linux VDA

### Step 4a: Uninstall the old version

If you have previously installed a version of the Linux VDA earlier than v1.0, uninstall it before installing the new version.

(a) Stop the Linux VDA services:
sudo /sbin/service ctxvda stop

sudo /sbin/service ctxhdx stop

(b) Uninstall the package:

```bash
sudo rpm -e XenDesktopVDA
```

**Note**

Upgrading from the latest 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 execute 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:**

```bash
command
```
sudo yum install -y XenDesktopVDA-7.16.0.412-1.el7_x.x86_64.rpm

For RHEL 6/CentOS 6:

command

```
sudo yum install -y XenDesktopVDA-7.16.0.412-1.el6_x.x86_64.rpm
```

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

For RHEL 7/CentOS 7:

```
sudo rpm -i XenDesktopVDA-7.16.0.412-1.el7_x.x86_64.rpm
```

For RHEL 6/CentOS 6:

```
sudo rpm -i XenDesktopVDA-7.16.0.412-1.el6_x.x86_64.rpm
```

RPM dependency list for RHEL 7/CentOS 7:

```
postgresql-server >= 9.2
postgresql-jdbc >= 9.2
java-1.8.0-openjdk >= 1.8.0
```
ImageMagick >= 6.7.8.9
firewalld >= 0.3.9
policycoreutils-python >= 2.0.83
dbus >= 1.6.12
dbus-x11 >= 1.6.12
xorg-x11-server-utils >= 7.7
xorg-x11-xinit >= 1.3.2
libXpm >= 3.5.10
libXrandr >= 1.4.1
libXtst >= 1.2.2
motif >= 2.3.4
pam >= 1.1.8
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
### RPM dependency list for RHEL 6/CentOS 6:

<table>
<thead>
<tr>
<th>Dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>postgresql-jdbc &gt;= 8.4</td>
</tr>
<tr>
<td>postgresql-server &gt;= 8.4</td>
</tr>
<tr>
<td>java-1.7.0-openjdk &gt;= 1.7.0</td>
</tr>
<tr>
<td>ImageMagick &gt;= 6.5.4.7</td>
</tr>
<tr>
<td>GConf2 &gt;= 2.28.0</td>
</tr>
<tr>
<td>system-config-firewall-base &gt;= 1.2.27</td>
</tr>
<tr>
<td>policycoreutils-python &gt;= 2.0.83</td>
</tr>
<tr>
<td>xorg-x11-server-utils &gt;= 7.7</td>
</tr>
<tr>
<td>xorg-x11-xinit &gt;= 1.0.9</td>
</tr>
<tr>
<td>ConsoleKit &gt;= 0.4.1</td>
</tr>
<tr>
<td>dbus &gt;= 1.2.24</td>
</tr>
<tr>
<td>dbus-x11 &gt;= 1.2.24</td>
</tr>
<tr>
<td>libXpm &gt;= 3.5.10</td>
</tr>
<tr>
<td>libXrandr &gt;= 1.4.1</td>
</tr>
<tr>
<td>libXtst &gt;= 1.2.2</td>
</tr>
<tr>
<td>openmotif &gt;= 2.3.3</td>
</tr>
<tr>
<td>pam &gt;= 1.1.1</td>
</tr>
<tr>
<td>util-linux-ng &gt;= 2.17.2</td>
</tr>
</tbody>
</table>
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

Note
After you install the Linux VDA on RHEL 7.3, you must run the `sudo yum install -y python-websockify x11vnc` command to manually install python-websockify and x11vnc for using the session shadowing feature. For more information, see Shadow sessions.

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:

```bash
sudo yum install -y XenDesktopVDA-7.16.0.412-1.el7_x86_64.rpm
```

For RHEL 6/CentOS 6:

```bash
sudo yum install -y XenDesktopVDA-7.16.0.412-1.el6_x86_64.rpm
```

Upgrade the Linux VDA software using the RPM package manager:

For RHEL 7/CentOS 7:

```bash
sudo rpm -U XenDesktopVDA-7.16.0.412-1.el7_x86_64.rpm
```

For RHEL 6/CentOS 6:

```bash
sudo rpm -U XenDesktopVDA-7.16.0.412-1.el6_x86_64.rpm
```

Important

You must reboot 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
```

Prompted configuration

Run a manual configuration with prompted questions:

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

Automated configuration

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

Supported environment variables include:

- **CTX_XDL_SUPPORT_DDC_AS_CNAME** = Y | N – The Linux VDA supports specifying a Delivery Controller name using a DNS CNAME record. Set to N by default.
- **CTX_XDL_DDC_LIST** = list-ddc-fqdns – The Linux VDA requires a space-separated list of Delivery Controller Fully Qualified Domain Names (FQDNs) to use for registering with a Delivery Controller. At least one FQDN or CNAME alias must be specified.
- **CTX_XDL_VDA_PORT** = port-number – The Linux VDA communicates with Delivery Controllers using a TCP/IP port, which is port 80 by default.
- **CTX_XDL_REGISTER_SERVICE** = Y | N - The Linux Virtual Desktop services support starting during boot. Set to Y by default.
- **CTX_XDL_ADD_FIREWALL_RULES** = Y | N – The Linux Virtual Desktop services require incoming network connections to be allowed through the system firewall. You can automatically open the required ports (ports 80 and 1494 by default) in the system firewall for the Linux Virtual Desktop. Set to Y by default.
- **CTX_XDL_AD_INTEGRATION** = 1 | 2 | 3 | 4 – The Linux VDA requires Kerberos configuration settings to authenticate...
with the Delivery Controllers. The Kerberos configuration is determined from the installed and configured Active Directory integration tool on the system. Specify the supported Active Directory integration method to use:

- 1 – Samba Winbind
- 2 – Quest Authentication Service
- 3 – Centrify DirectControl
- 4 – SSSD

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

**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 using DNS, querying for LDAP service records. To limit the DNS search results to a local site, specify a DNS site name. This variable is empty [none] by default.

**CTX_XDL_LDAP_LIST** = list-ldap-servers – The Linux VDA by default queries DNS to discover LDAP servers. However, if DNS cannot provide LDAP service records, you can provide a space-separated list of LDAP Fully Qualified Domain Names (FQDNs) with LDAP port (for example, ad1.mycompany.com:389). This variable is empty [none] by default.

**CTX_XDL_SEARCH_BASE** = search-base – The Linux VDA by default queries LDAP using 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 empty [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:

```
command
```
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
export CTX_XDL_START_SERVICE=Y|N

sudo -E /opt/Citrix/VDA/sbin/ctxsetup.sh

When running the sudo command, enter 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 commands above with #!/bin/bash as the first line.

Alternatively, you can specify all parameters with a single command:
sudo CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N \n
CTX_XDL_DDC_LIST=list-ddc-fqdns \n
CTX_XDL_VDA_PORT=port-number \n
CTX_XDL_REGISTER_SERVICE=Y|N \n
CTX_XDL_ADD_FIREWALL_RULES=Y|N \n
CTX_XDL_AD_INTEGRATION=1|2|3|4 \n
CTX_XDL_HDX_3D_PRO=Y|N \n
CTX_XDL_VDI_MODE=Y|N \n
CTX_XDL_SITE_NAME=dns-name \n
CTX_XDL_LDAP_LIST=list-ldap-servers \n
CTX_XDL_SEARCH_BASE=search-base \n
CTX_XDL_START_SERVICE=Y|N \n
/opt/Citrix/VDA/sbin/ctxsetup.sh

Remove configuration changes

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

Review Help about this script before proceeding:

```
sudo /opt/Citrix/VDA/sbin/ctxcleanup.sh --help
```
To remove configuration changes:

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

**Important**

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

**Configuration logs**

The `ctxsetup.sh` and `ctxcleanup.sh` scripts display errors on the console, with additional information written to the configuration log file `/tmp/xdl.configure.log`.

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

**Step 6: Run the Linux VDA**

Once you have configured the Linux VDA 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
```

**Stop the Linux VDA**

To stop the Linux VDA services:

```
sudo /sbin/service ctxhdx stop
sudo /sbin/service ctxvda stop
```
Restart the Linux VDA

To restart the Linux VDA services:

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

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
```

Step 7: Create the machine catalog in XenApp or XenDesktop

The process for creating machine catalogs and adding Linux VDA machines is very 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 XenDesktop 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 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.

The Citrix documentation for creating machine catalogs and delivery groups is referenced below:

https://docs.citrix.com
- XenDesktop 7.16
- XenDesktop 7.15
- XenDesktop 7.14
- XenDesktop 7.13
- XenDesktop 7.12
- XenDesktop 7.11
- XenDesktop 7.9
- XenDesktop 7.8
- XenDesktop 7.7
- XenDesktop 7.6
- XenDesktop 7.5
- XenDesktop 7.1

Earlier versions of XenDesktop are not supported.
Install Linux Virtual Delivery Agent for SUSE

Nov 27, 2017

You can choose to follow the steps below 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:

```bash
su -
yast
```

Alternatively, launch the UI-based YaST tool:

```bash
su -
yast2 &
```

#### Step 1b: Configure networking

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

**Configure hostname 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:

   - Hostname – Add the DNS hostname of the machine.
   - Domain Name – Add the DNS domain name of the machine.
   - Name Server – Add the IP address of the DNS server. This 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 hostname 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 hostname with a number and do not end with a hyphen. This rule also applies to Delivery Controller hostnames.

### 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 hostname
Verify that the hostname is set correctly:

```
hostname
```

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

Verify that the FQDN is set correctly:

```
hostname -f
```

This returns the machine's FQDN.

Check name resolution and service reachability

Verify 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
```

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

Step 1c: Configure the NTP service

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, 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. Enter 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, if the NTP daemon fails to start, this might be due to a known SUSE issue with AppArmor policies. Follow the [resolution](https://docs.citrix.com) for additional information.

### Step 1d: Install Linux VDA dependent packages

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

- **PostgreSQL**
  - SLED/SLES 12: Version 9.3 or later
- **OpenJDK 1.7.0**
- **OpenMotif Runtime Environment 2.3.1 or later**
- **Cups**
  - SLED/SLES 12: Version 1.6.0 or later
- **Foomatic filters**
  - SLED/SLES 12: Version 1.0.0 or later
- **ImageMagick**
  - SLED/SLES 12: Version 6.8 or later

#### Add repositories

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

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

To resolve this, the recommended approach is to obtain missing packages from the media for the alternate 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 approach described below mounts both SLED and SLES ISO media files and adds repositories.

```bash
sudo mkdir -p /mnt/sles
```
Install the Kerberos client

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

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

The Kerberos client configuration is dependent on which Active Directory integration approach is used, which is described later.

Install OpenJDK

The Linux VDA depends on OpenJDK 1.7.0.

Tip

To avoid problems, make sure that you installed only OpenJDK Version 1.7.0. Remove all other versions of Java on your system.

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
```

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

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

Check the Java version:
| SLES | On SLES, install the Java runtime environment:  
|      | sudo zypper install java-1_7_0-openjdk  
|      | Check the Java version:  
|      | java -version |

### Install PostgreSQL

| SLED/SLES 12 | Install the packages:  
|              | sudo zypper install postgresql-init  
|              | sudo zypper install postgresql-server  
|              | sudo zypper install postgresql-jdbc  
|              | Post-installation steps are required to initialize the database service and ensure PostgreSQL starts on boot:  
|              | sudo systemctl enable postgresql  
|              | sudo systemctl restart postgresql  
|              | Database files will reside under /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:

| SLED 12 | Remove the following packages:  
|         | sudo zypper rr sles  
|         | sudo umount /mnt/sles  
|         | sudo rmdir /mnt/sles |

| SLED/SLES 12 | Remove the following packages:  
|              | sudo zypper rr sdk  
|              | sudo umount /mnt/sdk |
Step 2: Prepare Linux VM for Hypervisor

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

Fix time synchronization on Citrix XenServer

If the XenServer Time Sync feature is enabled, within each paravirtualized Linux VM you 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. This 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
```

This command returns 0 or 1:

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

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

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

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

To make this change permanent and persistent 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 command returns the value 1.

Fix time synchronization on Microsoft Hyper-V

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

From the management operating system:

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

Note

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

Fix time synchronization on ESX and ESXi
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. This 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 domain:

- Samba Winbind
- Quest Authentication Service
- Centrify DirectControl

Follow the instructions below for 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 is entered 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 will ask whether you want to join the domain. Click Yes.
5. When prompted, enter 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 to be restarted or the machine to be rebooted. It is advisable to reboot:

```
su -
reboot
```

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

SLED/SLES 12 has changed the default Kerberos credential cache name specification from the usual `FILE:/tmp/krb5cc_%{uid}` to `DIR:/run/user/%{uid}/krb5cc`. This new DIR caching method is not compatible with the Linux VDA and must be manually changed. As root, 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, whether Windows or Linux, have a computer object in Active Directory.

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

```
sudo net ads testjoin
```

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

```
```
Verify Kerberos configuration

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

```
sudo klist –ke
```

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

```
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 that the realm name is used. If this command is successful, no output is displayed.

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

```
sudo klist
```

Examine the machine account details using:
Verify user authentication

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

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

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

To verify that the Winbind PAM module is configured correctly, log on locally with a domain user account that has not logged on to 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 the user's Kerberos credential cache are valid and not expired:

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

**Quest authentication service**

**Configure Quest on domain controller**

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

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

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

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

**Note**

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

**Configure Quest on Linux VDA**

**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

This command sets the renewal interval to 9 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 Kerberos ticket lifetime.

**Configure PAM and NSS**

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

```
command

  sudo /opt/quest/bin/vastool configure pam

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

**Join Windows domain**

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

```
command

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

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

**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:
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 using PAM, log on with a domain user account that has not logged on to 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 the Kerberos credential cache are valid and not expired:
Exit the session:

```
exit
```

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

**Centrify DirectControl**

**Join Windows domain**

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

```
su -

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

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

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:

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

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

```command
adinfo --test
```

### Step 4: Install the Linux VDA

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

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

(a) Stop the Linux VDA services:

```command
```
sudo /sbin/service ctxvda stop

(b) Uninstall the package:

```
sudo rpm -e XenDesktopVDA
```

---

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

---

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

To execute a command, the full path is needed; 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 `zypper`:

**For SUSE 12:**

```
command
```

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

For SUSE 12:

```
sudo zypper install XenDesktopVDA-7.16.0.412-1.sle12_x.x86_64.rpm
```

Step 4d: Upgrade the Linux VDA (optional)

You can upgrade the Linux VDA software from Versions 7.13 and 7.12 using the RPM package manager.

For SUSE 12:

```
sudo rpm -i XenDesktopVDA-7.16.0.412-1.sle12_x.x86_64.rpm
```

RPM Dependency list for SUSE 12:

```
postgresql-server >= 9.3
postgresql-jdbc >= 9.2
java-1.7.0-openjdk >= 1.7.0
ImageMagick >= 6.8
dbus-1 >= 1.8.8
dbus-1-x11 >= 1.8.8
```
Important

You must reboot 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
```

Prompted configuration

Run a manual configuration with prompted questions:

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

Automated configuration

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

Supported environment variables include:

- **CTX_XDL_SUPPORT_DDC_AS_CNAME = Y | N** – The Linux VDA supports specifying a Delivery Controller name using a DNS CNAME record. Set to N by default.
- **CTX_XDL_DDC_LIST = list-ddc-fqdns** – The Linux VDA requires a space-separated list of Delivery Controller Fully Qualified Domain Names (FQDNs) to use for registering with a Delivery Controller. At least one FQDN or CNAME alias must be specified.
- **CTX_XDL_VDA_PORT = port-number** – The Linux VDA communicates with Delivery Controllers using a TCP/IP port, which is port 80 by default.
- **CTX_XDL_REGISTER_SERVICE = Y | N** - The Linux Virtual Desktop services support starting during boot. Set to Y by default.
- **CTX_XDL_ADD_FIREWALL_RULES = Y | N** – The Linux Virtual Desktop services require incoming network connections to be allowed through the system firewall. You can automatically open the required ports (ports 80 and 1494 by default) in the system firewall for the Linux Virtual Desktop. Set to Y by default.
- **CTX_XDL_AD_INTEGRATION = 1 | 2 | 3 | 4** – The Linux VDA requires Kerberos configuration settings to authenticate with the Delivery Controllers. The Kerberos configuration is determined from the installed and configured Active Directory integration tool on the system. Specify the supported Active Directory integration method to use:
  - 1 – Samba Winbind
  - 2 – Quest Authentication Service
3 – Centrify DirectControl
4 – SSSD

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

**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 using DNS, querying for LDAP service records. To limit the DNS search results to a local site, specify a DNS site name. This variable is empty [none] by default.

**CTX_XDL_LDAP_LIST** = list-ldap-servers – The Linux VDA by default queries DNS to discover LDAP servers. However, if DNS cannot provide LDAP service records, you can provide a space-separated list of LDAP Fully Qualified Domain Names (FQDNs) with LDAP port (for example, ad1.mycompany.com:389). This variable is empty [none] by default.

**CTX_XDL_SEARCH_BASE** = search-base – The Linux VDA by default queries LDAP using 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 empty [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.

**Note**

HDX 3D Pro is not currently available on SUSE.

Set the environment variable and run the configure script:

```shell
command
```
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
export CTX_XDL_START_SERVICE=Y|N

sudo -E /opt/Citrix/VDA/sbin/ctxsetup.sh

When running the sudo command, enter 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 commands above with #!/bin/bash as the first line.

Alternatively, you can specify all parameters with 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 \ 

CTX_XDL_START_SERVICE=Y|N \ 

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

Remove configuration changes

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

Review Help about this script before proceeding:

command

sudo /usr/local/sbin/ctxcleanup.sh --help
To remove configuration changes:

```
command

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

**Important**

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

Configuration logs

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

```
command

/tmp/xdl.configure.log
```

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

**Step 6: Run the Linux VDA**

Once you have configured the Linux VDA 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:

```
command

sudo /sbin/service ctxhdx start

sudo /sbin/service ctxvda start
```
Stop the Linux VDA

To stop the Linux VDA services:

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

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
```

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
```

Step 7: Create the machine catalog in XenApp or XenDesktop
The process for creating machine catalogs and adding Linux VDA machines is very 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 XenDesktop 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 complete these tasks, see Create Delivery Groups.

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

- For delivery type, select Desktops or Applications.
- Ensure 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.

The Citrix documentation for creating machine catalogs and delivery groups is referenced below:

- XenDesktop 7.16
- XenDesktop 7.15
- XenDesktop 7.14
- XenDesktop 7.13
- XenDesktop 7.12
- XenDesktop 7.11
- XenDesktop 7.9
- XenDesktop 7.8
- XenDesktop 7.7
- XenDesktop 7.6
- XenDesktop 7.5
- XenDesktop 7.1

Earlier versions of XenDesktop are not supported.
Install Linux Virtual Delivery Agent for Ubuntu

Nov 27, 2017
You can choose to follow the steps below for manual installation or use easy install for automatic installation and configuration. Easy install saves time and labor and is less error-prone than the manual installation.

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

Step 1: Prepare Ubuntu for VDA installation

Step 1a: Verify the network configuration
Citrix recommends that the network is connected and configured correctly before proceeding.

Step 1b: Set the hostname
To ensure that the hostname of the machine is reported correctly, change the /etc/hostname file to contain only the hostname of the machine.

hostname

Step 1c: Assign a loopback address to the hostname
To ensure that the DNS domain name and FQDN of the machine are reported back correctly, change the following line of the /etc/hosts file to include the FQDN and hostname 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 hostname 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 hostname with a number and do not end with a hyphen. This rule also applies to Delivery Controller hostnames.

Step 1d: Check the hostname
Verify that the hostname is set correctly:
This command returns only the host name of the machine and not its FQDN.

Verify that the FQDN is set correctly:

```
hostname
```

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 XenDesktop Delivery Controller:
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:

```
command
apt-get install chrony
```

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

```
config
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:
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
```

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
```

Step 1j: Install Motif

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

Step 1k: Install other packages

```
sudo systemctl restart chrony
```

```
sudo apt-get install -y postgresql
```

```
sudo apt-get install -y postgresql
```

```
sudo apt-get install -y postgresql
```

```
sudo systemctl restart chrony
```

```
sudo systemctl restart chrony
```
sudo apt-get install -y libtasl2-2

sudo apt-get install -y libtasl2-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

---

**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:

```
command

su -

cat /proc/sys/xen/independent_wallclock
```

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

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

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

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

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

```
xen.independent_wallclock = 1
```

To verify these changes, reboot the system:

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

This command returns the value 1.

Fix time synchronization on Microsoft Hyper-V

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

From the management operating system:

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

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

Follow the instructions below for 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
```

Enable Winbind daemon to start on boot

The Winbind daemon must be configured to start on boot:
Configure Kerberos

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

```config
[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
```

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:
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:
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:

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

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

**Restart winbind:**

```bash
sudo systemctl restart winbind
```

Configure PAM for Winbind

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

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

**Tip**

The winbind daemon stays running only if the machine is joined to a domain.
Verify Domain Membership

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

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

```
sudo net ads testjoin
```

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

```
sudo net ads info
```

Verify Kerberos Configuration

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

```
sudo klist -ke
```

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

```
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 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:

```
command

sudo klist
```

Examine the account details of the machine using:

```
command

sudo net ads status
```

**Verify user authentication**

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

```
command

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

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

To verify that the Winbind PAM module is configured correctly, log on locally with a domain user account that has not logged on to the machine previously:

```
command

ssh localhost -l domain\username

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

```
ls /tmp/krb5cc_uid
```

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

```
klist
```

Exit the session:

```
exit
```

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

**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

**Workaround SELinux policy enforcement**

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

**Tip**
There are several ways to work around this issue, as described [here](https://docs.citrix.com).

The easiest is to disable SELinux. As root, edit `/etc/selinux/config` and change the **SELinux** setting:

```
SELINUX=disabled
```

This change requires a reboot:

```
reboot
```

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

**Configure VAS daemon**

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

Quest requires that PAM and NSS are 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
```

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
```

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:

```

```
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 using PAM, log on with a domain user account that has not logged on to 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 the Kerberos credential cache are valid and not expired:
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
```

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
```
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:

```
command

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

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

```
command

adinfo --test
```

**SSSD**

**Configure Kerberos**

**Install kerberos**

```
command

sudo apt-get install krb5-user
```

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

```
Setting
```

https://docs.citrix.com  © 1999-2017 Citrix Systems, Inc. All rights reserved.
The domain-dns-name property in this context is the DNS domain name, such as example.com. The REALM is the Kerberos realm name in uppercase, such as EXAMPLE.COM.

### Join the domain

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

- `adcli`
- `samba`
- `realmd`

#### Note

This section only provides information for `adcli` and `samba`. 

```ini
[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
```
Use adcli to join the domain

Install adcli

Install the required package:

```
sudo apt-get install adcli
```

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
```

The **user** is a domain user with permissions to add machines to the domain. The **hostname-fqdn** is the hostname 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
```

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
```

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
```

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

**Set up SSSD**

**Install or update required packages**

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

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

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

```
sudo apt-get update sssd
```

**Note**

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

**Configure SSSD**

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

```
[sssd]
```
services = nss, pam

config_file_version = 2

domains = domain-dns-name

[domain/domain-dns-name]

id_provider = ad

access_provider = ad

auth_provider = krb5

krb5_realm = REALM

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

krb5_renewable_lifetime = 14d

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

krb5_renew_interval = 1h

krb5_ccachedir = /tmp

krb5_ccname_template = FILE:%d/krb5cc_%U

# This ldap_id_mapping setting is also the default value

ldap_id_mapping = true

override_homedir = /home/%d/%u

default_shell = /bin/bash

ad_gpo_map_remote_interactive = +ctxhdx
Note
ldap_id_mapping is set to true so that SSSD itself takes care of mapping Windows SIDs to Unix UIDs. Otherwise, the Active Directory must be able to provide POSIX extensions (for more information, see the RHEL site). PAM service ctxhdx is added into ad_gpo_map_remote_interactive. For more information, see SSSD GPO-Based Access Control.

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:

```bash
sudo chmod 0600 /etc/sssd/sssd.conf
```

Start SSSD daemon

Start the SSSD daemon and enable it to start on boot:

```bash
sudo systemctl start sssd

sudo systemctl enable sssd
```

PAM configuration

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

```bash
```

https://docs.citrix.com  © 1999-2017 Citrix Systems, Inc. All rights reserved.  p.120
Verify domain membership

The Delivery Controller requires that all VDA machines, whether Windows or Linux, 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
```

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
```

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

```
sudo net ads info
```

Verify Kerberos configuration

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

```
```

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 klist -ke
```

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 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
```

**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, log on locally with a domain user account that has not logged on to the machine previously:
Check that the Kerberos tickets returned by the `klist` command are correct for that user and have not expired.

As root, check that a corresponding ticket cache file was created for the uid returned by the previous `id -u` command:

```
ls /tmp/krb5cc_uid
```

A similar test can be performed by logging on to KDE or Gnome Display Manager.

**Step 4: Install the Linux VDA**

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

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

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

Install the Linux VDA software using the Debian package manager:

```
sudo dpkg -i xendesktopvda_7.16.0.412-1.ubuntu16.04_amd64.deb
```

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

libxst6 >= 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
Step 4c: Configure the Linux VDA

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

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

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

Prompted configuration

Run a manual configuration with prompted questions:
sudo /opt/Citrix/VDA/sbin/ctxsetup.sh

Automated configuration

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 using a TCP/IP port, which is port 80 by default.
- **CTX_XDL_REGISTER_SERVICE** = Y | N – The Linux Virtual Desktop services support starting during boot. Set to Y by default.
- **CTX_XDL_ADD_FIREWALL_RULES** = Y | N – The Linux Virtual Desktop services require incoming network connections to be allowed through the system firewall. You can automatically open the required ports (ports 80 and 1494 by default) in the system firewall for the Linux Virtual Desktop. Set to Y by default.
- **CTX_XDL_AD_INTEGRATION** = 1 | 2 | 3 | 4 – The Linux VDA requires Kerberos configuration settings to authenticate with the Delivery Controllers. The Kerberos configuration is determined from the installed and configured Active Directory integration tool on the system. Specify the supported Active Directory integration method to use:
  - 1 – Samba Winbind
  - 2 – Quest Authentication Service
  - 3 – Centrify DirectControl
  - 4 – SSSD
- **CTX_XDL_HDX_3D_PRO** = Y | N – Linux Virtual Desktop supports HDX 3D Pro, a set of graphics acceleration technologies designed to optimize the virtualization of rich graphics applications. HDX 3D Pro requires a compatible NVIDIA GRID graphics card to be installed. If HDX 3D Pro is selected, the Virtual Delivery Agent is configured for VDI desktops (single-session) mode – (that is, CTX_XDL_VDI_MODE=Y).
- **CTX_XDL_VDI_MODE** = Y | N – Whether to configure the machine as a dedicated desktop delivery model (VDI) or hosted shared desktop delivery model. For HDX 3D Pro environments, set this variable to Y. This variable is set to N by default.
- **CTX_XDL_SITE_NAME** = dns-name – The Linux VDA discovers LDAP servers using DNS, querying for LDAP service records. To limit the DNS search results to a local site, specify a DNS site name. This variable is empty [none] by default.
- **CTX_XDL_LDAP_LIST** = list-ldap-servers – The Linux VDA by default queries DNS to discover LDAP servers. However, if DNS cannot provide LDAP service records, you can provide a space-separated list of LDAP Fully Qualified Domain Names (FQDNs) with LDAP port (for example, ad1.mycompany.com:389). This variable is empty [none] by default.
- **CTX_XDL_SEARCH_BASE** = search-base – The Linux VDA by default queries LDAP using 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 empty [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
export CTX_XDL_START_SERVICE=Y|N

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

When running the sudo command, enter 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 commands above with `#!/bin/bash` as the first line.

Alternatively, you can specify all parameters with a single command:

```
```
sudo CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N \nCTX_XDL_DDC_LIST=list-ddc-fqdns \nCTX_XDL_VDA_PORT=port-number \nCTX_XDL_REGISTER_SERVICE=Y|N \nCTX_XDL_ADD_FIREWALL_RULES=Y|N \nCTX_XDL_AD_INTEGRATION=1|2|3|4 \nCTX_XDL_HDX_3D_PRO=Y|N \nCTX_XDL_VDI_MODE=Y|N \nCTX_XDL_SITE_NAME=dns-name \nCTX_XDL_LDAP_LIST=list-ldap-servers \nCTX_XDL_SEARCH_BASE=search-base \nCTX_XDL_START_SERVICE=Y|N \n
/opt/Citrix/VDA/sbin/ctxsetup.sh

Remove configuration changes

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

Review Help about this script before proceeding:

```
sudo /opt/Citrix/VDA/sbin/ctxcleanup.sh --help
```
To remove configuration changes:

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

**Important**

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

Configuration logs

The `ctxsetup.sh` and `ctxcleanup.sh` scripts display errors on the console, with additional information written to the configuration log file `/tmp/xdl.configure.log`.

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

Uninstall the Linux VDA software

**Query Linux VDA installation status**

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

```
dpkg -l xendesktopvda
```

To view more detailed information:

```
apt-cache show xendesktopvda
```

**Note**

Uninstalling the Linux VDA software deletes the associated PostgreSQL and other configuration data. However, the PostgreSQL package and other dependent packages that were set up before the installation of the Linux VDA are not removed.
**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
```

#### Stop the Linux VDA

To stop the Linux VDA services:

```
sudo systemctl stop ctxvda
sudo systemctl stop ctxhdx
```

#### Restart the Linux VDA

To restart the Linux VDA services:

```
sudo systemctl restart ctxhdx
sudo systemctl restart ctxvda
```
```bash
sudo systemctl stop ctxvda

sudo systemctl restart ctxhdx

sudo systemctl restart ctxvda
```

### Check the Linux VDA status

To check the running state of the Linux VDA services:

<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>sudo systemctl status ctxvda</td>
</tr>
<tr>
<td>sudo systemctl status ctxhdx</td>
</tr>
</tbody>
</table>

### Step 6: Create the machine catalog in XenApp or XenDesktop

The process for creating machine catalogs and adding Linux VDA machines is very 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 XenDesktop 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.

The Citrix documentation for creating machine catalogs and delivery groups is referenced below:

- XenDesktop 7.16
- XenDesktop 7.15
- XenDesktop 7.14
- XenDesktop 7.13
- XenDesktop 7.12
- XenDesktop 7.11
- XenDesktop 7.9
- XenDesktop 7.8
- XenDesktop 7.7
- XenDesktop 7.6
- XenDesktop 7.5
- XenDesktop 7.1

Earlier versions of XenDesktop are not supported.
Configure the Linux VDA

This chapter details features of the Linux VDA, including feature description, configuration, and troubleshooting.
Integrate NIS with Active Directory

Aug 14, 2017
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 (username 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 here. For an Ubuntu distribution, follow the instructions here.

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 to the system. It provides both PAM and NSS modules, and in the future will 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. If you are using an earlier version, follow the instructions provided in Configuring the LDAP provider with AD.

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, you must:

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
```

Set the NIS domain:

```
ypdomainname nis.domain

echo "NISDOMAIN=nis.domain" >> /etc/sysconfig/network
```

Add the IP address for the NIS server and client in `/etc/hosts`:

```
[NIS server IP address]    server.nis.domain nis.domain
```

Configure NIS by authconfig:

```
sudo authconfig --enablenis --nisdomain=nis.domain --nisserver=server.nis.domain --enablemkhomedir --update
```

`nis.domain` represents the domain name of the NIS server, and `server.nis.domain` is the hostname 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
```

Ensure that the NIS configuration is correct:

```
ypwhich
```

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

```
getent passwd nisaccount
```

**Note**

`nisaccount` represents the real NIS account on the NIS server. Make sure that the UID, GID, home directory, and login shell are configured correctly.

Join the domain and create a host keytab with 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 this, 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 with 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
```

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

If DNS-based lookups of the KDC server and realm name is required, add the following two options to the above 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 your domain controller to be reachable and that you have an AD user account with permissions to add computers to the domain:

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

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

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 (additional options can be added as needed):

```conf
[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
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`:

```
command
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 make sure home directory creation works with SELinux:

```
command
authconfig --enablesssd --enablesssdauth --enablemkhomedir --update
sudo systemctl start sssd
sudo systemctl enable sssd
```

Tip

When configuring Linux VDA settings, consider that for SSSD, there has no special settings for the Linux VDA client. For additional solutions in the `ctxsetup.sh` script, use the default value.

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

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

The machine and realm names must be specified in uppercase, 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 is displayed.

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

```
sudo klist -ke
```

Verify user authentication

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

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

The `DOMAIN` parameter indicates the short version domain name. If another logon format from Citrix Receiver 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, log on locally with a domain user account that has not previously logged on to the machine.

```
sudo localhost -l DOMAIN\username
id -u
```

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

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

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

```
klist
```
Publish applications

Nov 27, 2017

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. For more information, see publish applications.

Publish applications using Citrix Studio

You can publish applications installed on a Linux VDA machine when you create a delivery group or add applications to an existing delivery group. This process is similar to publishing applications installed on the Windows VDA. Refer to the following pages for additional information (based on the version of XenDesktop being used):

- XenDesktop 7.16
- XenDesktop 7.15
- XenDesktop 7.14
- XenDesktop 7.13
- XenDesktop 7.12
- XenDesktop 7.11
- XenDesktop 7.9
- XenDesktop 7.8
- XenDesktop 7.7
- XenDesktop 7.6
- XenDesktop 7.5
- XenDesktop 7.1

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've 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 fails to show windows in seamless mode. If this occurs, verify that seamless mode has been enabled on both Linux VDA and StoreFront, and check whether you have completed all the configurations mentioned above.

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"
```

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 "0x00000001"
```

Known issues

The following issues are known when publishing applications:

- Non-seamless published applications fail to launch when the seamless mode is disabled on StoreFront but still enabled on Linux VDA. It is recommended to enable/disable the seamless mode on both 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 (CentOS6.9\RHEL6.9\CentOS6.8\RHEL6.8), and Compiz (Ubuntu 16.04). Kwin and other window managers are

https://docs.citrix.com  © 1999-2017 Citrix Systems, Inc. All rights reserved.  p.144
not supported. Make sure 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 same process.
- Published Qt5-based applications like “Dolphin” might not show icons. To resolve the issue, see the article at https://wiki.archlinux.org/index.php/Qt#Configuration_of_Qt5_apps_under_environments_other_than_KDE_Plasma.
- All the taskbar buttons of published applications running in the same ICA session are combined in the same group. To resolve this issue, set the taskbar property to not combine taskbar buttons.
Print

Aug 14, 2017
This article provides information about printing best practices.

Installation

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

**RHEL 7 printing support**

```bash
sudo yum -y install cups
sudo yum -y install foomatic-filters
```

**RHEL 6 printing support**

```bash
sudo yum -y install cups
sudo yum -y install foomatic
```

Usage

You can print from both published desktops and published applications. Only the client-side default printer is mapped into a Linux VDA session. The printer name must be different for desktops and applications. Consider the following:

- For published desktops:
  `CitrixUniversalPrinter:SCLIENT_NAME:dsk$SESSION_ID`

- For published applications:
  `CitrixUniversalPrinter:SCLIENT_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 a number of 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 whether the printing daemon is running.

```
ps –ef | grep ctxlpmngt
```

If the `ctxlpmngt` process is not running, manually start `ctxlpmngt` from a command line. If printing is still not working, check CUPS framework. The `ctxcups` service is for printer management and communicates with the Linux CUPS framework. This is a single process per machine and can be checked by:

```
service ctxcups status
```

Extra log when printing CUPS

As one of the components of the Linux VDA, the method of how to get the log of a printing component is similar to other components.

For RHEL, some extra steps are necessary to configure the CUPS service file. Otherwise some logs cannot get logged in `hdx.lo`:

```
COPY
```
sudo service cups stop

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

PrivateTmp=false

sudo service cups start

sudo systemctl daemon-reload

Note
This configuration is only for collecting the full printing log when an issue arises. Normally this configuration is not recommended because it breaks CUPS security.

Print output is garbled
Garbled output can be caused by an incompatible printer driver. A per-user driver configuration is available and can be configured by editing the ~/.CtxlpProfileSCLIENT_NAME configuration file:

```ini
[DEFAULT_PRINTER]

printername=

model=

ppdpath=

drivertype=
```

Important
The **printername** is a field containing the name of the current client side default printer. This is a read-only value and must not be edited.
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 with the `lpinfo` command:

```
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
```

You can then set the model to match the printer:

```
config
Model=xerox/ph3115.ppd.gz
```

If the Universal Printer driver is not compatible with the client printer, configure the ppd file path of the native printer driver. The value of `ppdpath` is the absolute path of the native printer driver file.

For example, there is a `ppd` driver under `/home/tester/NATIVE_PRINTER_DRIVER.ppd`:

```
config
ppdpath=/home/tester/NATIVE_PRINTER_DRIVER.ppd
```

There are three types of Universal Printer Driver supplied by Citrix (postscript, pcl5 and pcl6). You can configure those in the driver type if no native printer driver is available.

For example, if the client default printer driver type is PCL5:
Output size is zero

Try different types of printers. And try with 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, this will cause an issue.

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
```

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, this represents an issue. Contact Citrix Support (and provide the printing log) for additional 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 with the client-side printer. After completing this step, test it with 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:
b. Choose the mapped printer `CitrixUniversalPrinter:$ClientName:app/dek$SESSION_ID` and Modify Printer. This operation requires administrator privileges.

c. Retain the cups-ctx connection, then click Continue to modify the printer driver.

d. In the Make and Model page, choose some other postcript 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.

e. 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 when printing using 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 print file is transferred over the server connection. On slow connections, this 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 modifying 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 into the default policy XML block:

```
<configuration>
  
  ...

  <policy name="default">
    ...  
  
  </policy>

  ...

</configuration>
```
<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>
PDF printing

Aug 14, 2017

Using a version of Citrix Receiver that supports PDF printing, you can print PDFs converted from within Linux VDA sessions. Session print jobs are sent to the endpoint, where you can open them using your PDF viewer of choice, and print them on your printers of choice.

To use the PDF printing feature, you must be using a version of Citrix Receiver that supports PDF printing. The following versions of Citrix Receiver support PDF printing:

- Citrix Receiver for HTML5 Version 2.4
- Citrix Receiver for Chrome Version 2.4

Configuration

Apart from using a version of Citrix Receiver mentioned above, you must also 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 Receiver documentation for information about setting default printers.
Configure graphics

Aug 14, 2017
As of Version 7.12, Linux VDA 3D Pro supports vGPU for Nvidia Tesla M60 and M10. This article serves as a guide for 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\CurrentControlSet\Control\Citrix\Thinwire that you can tune with the ctxreg utility under /opt/Citrix/VDA/bin.

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 Receiver version. If the client does not meet the requirements, it falls back to Thinwire Plus.

<table>
<thead>
<tr>
<th>OS</th>
<th>Windows</th>
<th>Mac OS X</th>
<th>Linux</th>
<th>Android</th>
<th>iOS</th>
<th>Chrome OS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum requirement for H.264</td>
<td>3.4 or later</td>
<td>11.8 or later</td>
<td>13.0 or later</td>
<td>3.5</td>
<td>5.9</td>
<td>1.4</td>
</tr>
</tbody>
</table>

The latest feature matrix for Citrix Receiver is available here.

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

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "AdvertiseH264"
```

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:

```
```
Parameters to tune Thinwire Plus for lower bandwidth

- **MaxColorDepth**

  Location: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\Thinwire
  Type: DWORD
  Default: 0x20

  This parameter 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)

  Location: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\Thinwire
  Type: DWORD
  Default: 0x1 (medium), valid values: 0x0 (low), 0x1 (medium), 0x2 (high), 0x3 (build to lossless), 0x4 always lossless.

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

Other parameters

- **TargetFPS** (target frame rate)

  Location: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\Thinwire
  Type: DWORD
  Default: 0x1e (30)

- **MinFPS** (target minimum frame rate)

  Location: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\Thinwire
  Type: DWORD
  Default: 0xa (10)
• **MaxScreenNum** (the maximum number of monitors the client can have)

<table>
<thead>
<tr>
<th>Location: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\Thinwire</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: DWORD</td>
</tr>
<tr>
<td>Default: 0x2</td>
</tr>
</tbody>
</table>

For a standard VDA, you can set a value 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 and 0 means TW+):

```
sudo /opt/Citrix/VDA/bin/ctxreg dump | grep H264
```

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
```

The results resemble:
Another way is to use the `nvidia-smi` command. When using this command, the output resembles the following if hardware encoding is in use:

```plaintext
create -k "HKLM\Software\Citrix\ICA\Session\1\Graphics" -t "REG_DWORD" -v "HardwareEncoding" -d "0x00000001" --force
```
Verify whether the NVIDIA GRID graphics driver is installed correctly

<table>
<thead>
<tr>
<th>GPU Name</th>
<th>Persistence-M</th>
<th>Bus-Id</th>
<th>Disp.A</th>
<th>Volatile Uncorr. ECC</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 GRID K1</td>
<td>Off</td>
<td>0000:00:05.0</td>
<td>Off</td>
<td>N/A</td>
</tr>
<tr>
<td>N/A 42C</td>
<td>P0</td>
<td>14W / 31W</td>
<td>207MiB / 4095MiB</td>
<td>8% Default</td>
</tr>
<tr>
<td>Processes:</td>
<td>GPU Memory</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPU</td>
<td>PID</td>
<td>Type</td>
<td>Process name</td>
<td>Usage</td>
</tr>
</tbody>
</table>
To verify whether the NVIDIA GRID graphics driver is installed correctly, run `nvidia-smi`. The results resemble:

```
<table>
<thead>
<tr>
<th>NVIDIA-SMI 352.70</th>
<th>Driver Version: 352.70</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>GPU Name</th>
<th>Persistence-M</th>
<th>Bus-Id</th>
<th>Disp.A</th>
<th>Volatile Uncorr. ECC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fan</td>
<td>Temp</td>
<td>Perf</td>
<td>Pwr:Usage/Cap</td>
<td>Memory-Usage</td>
</tr>
<tr>
<td>---------</td>
<td>--------</td>
<td>--------</td>
<td>------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>Tesla M60</td>
<td>Off</td>
<td>0000:00:05.0</td>
<td>Off</td>
<td>Off</td>
</tr>
<tr>
<td>N/A</td>
<td>20C</td>
<td>P0</td>
<td>37W / 150W</td>
<td>19MiB / 8191MiB</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Processes:</th>
<th>GPU Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPU</td>
<td>PID</td>
</tr>
<tr>
<td>---------</td>
<td>------</td>
</tr>
<tr>
<td>No running processes found</td>
<td></td>
</tr>
</tbody>
</table>
```

Set the correct configuration for the card:
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 \textit{Xorg\{DISPLAY\}.log} in the \texttt{/var/log/} folder.

Known issues and limitations

For vGPU, the XenServer local console shows the ICA desktop session screen

\textbf{Workaround}: Disable the VM's local VGA console by running the following command:

\begin{verbatim}
xe vm-param-set uuid=<vm-uuid> platform:vgpu_extra_args="disable_vnc=1"
\end{verbatim}

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 passthrough mode

This is a limitation of Nvidia K2 graphics cards.

Gnome 3 desktop popups slow when logging on

This is a limitation of Gnome 3 desktop session startup.

Some OpenGL/WebGL applications do not render well upon resizing the Receiver window

Resizing the Citrix Receiver window will change screen resolution. The Nvidia proprietary driver changes some internal states and might require applications to respond accordingly. For example, the WebGL library element \texttt{lightgl.js} could spawn an error saying that 'Rendering to this texture is not supported (incomplete framebuffer)'.
Non-GRID 3D graphics

Aug 14, 2017

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 already exists as an extension of XServer.

Configuration

Xorg configuration files

If your 3D card driver is NVIDIA, the configuration files are installed and set automatically.

Other types of 3D cards

If your 3D card driver is NOT NVIDIA, you must modify the four template configuration files installed under /etc/X11/:

- ctx-driver_name-1.conf
- ctx-driver_name-2.conf
- ctx-driver_name-3.conf
- ctx-driver_name-4.conf

Using ctx-driver_name-1.conf as an example, follow the steps below 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 modify 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. You must enable this section before adding your video driver information. To enable this section:

   a. Refer to the 3D card guide provided by the manufacturer for configuration information. A native configuration file will be generated. Verify that your 3D card works 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 to let the Linux VDA recognize the configuration file name modified in Step 1.

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Citrix\XDamage" -t "REG_SZ" -v "DriverName"
```

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"
```

Troubleshooting

No or garbled graphic output

If you can run 3D applications locally and all configurations are done correctly, missing or garbled graphic output is the result of a bug. Use /opt/Citrix/VDA/bing/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

Aug 14, 2017

Installation

Follow the installation articles to prepare the Linux VDA.

Dependencies

Make sure that you install these dependencies before installing the Linux VDA package.

RHEL/CentOS

<table>
<thead>
<tr>
<th>command</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>sudo yum -y install openldap</code></td>
</tr>
<tr>
<td><code>sudo yum -y install libxml2</code></td>
</tr>
<tr>
<td><code>sudo yum -y install cyrus-sasl</code></td>
</tr>
<tr>
<td><code>sudo yum -y install cyrus-sasl-gssapi</code></td>
</tr>
</tbody>
</table>

SLES/SELD

<table>
<thead>
<tr>
<th>command</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>sudo zypper install openldap2</code></td>
</tr>
<tr>
<td><code>sudo zypper install libxml2</code></td>
</tr>
<tr>
<td><code>sudo zypper install cyrus-sasl</code></td>
</tr>
<tr>
<td><code>sudo zypper install cyrus-sasl-gssapi</code></td>
</tr>
</tbody>
</table>

Ubuntu

<table>
<thead>
<tr>
<th>command</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>command</code></td>
</tr>
<tr>
<td><code>COPY</code></td>
</tr>
<tr>
<td><code>COPY</code></td>
</tr>
<tr>
<td><code>COPY</code></td>
</tr>
</tbody>
</table>
.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 required by the policy service to perform an LDAP search in these environments.

After installing the Linux VDA package, run the command:

```
/command

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

Enter 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:

```
/command

ctxreg
```

https://docs.citrix.com  © 1999-2017 Citrix Systems, Inc. All rights reserved.  p.164
New policy settings for the Linux VDA

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

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\VirtualDesktopAgent" -t "REG_SZ" -v "ListOfLDAPServers" -d "ad1.mycompany.com:389 ad2.mycompany.com:389" --force
```

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\VirtualDesktopAgent\PolicySupportList\ClipboardSelectionUpdateMode" -t "REG_DWORD" -v "Flags" -d "your value" --force
```

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\VirtualDesktopAgent\PolicySupportList\PrimarySelectionUpdateMode" -t "REG_DWORD" -v "Flags" -d "your value" --force
```

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\VirtualDesktopAgent\PolicySupportList\MaxSpeexQuality" -t "REG_DWORD" -v "MaxSpeexQuality" -d "your value" --force
```

**Note**

The values are restricted to a certain range. For detailed descriptions, see the Policy support list.
### Linux VDA policy support list

<table>
<thead>
<tr>
<th>Studio Policy</th>
<th>Key Name</th>
<th>Type</th>
<th>Module</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICA keep alives</td>
<td>SendICAKeepAlives</td>
<td>computer</td>
<td>ICA keep alive</td>
<td>Do not send ICA keep alive messages (0)</td>
</tr>
<tr>
<td>ICA keep alive timeout</td>
<td>ICAKeepAliveTimeout</td>
<td>computer</td>
<td>ICA keep alive</td>
<td>60 seconds</td>
</tr>
<tr>
<td>ICA listener port number</td>
<td>IcaListenerPortNumber</td>
<td>computer</td>
<td>ICA</td>
<td>1494</td>
</tr>
<tr>
<td>HDX Enlightened Data Transport (For Evaluation Only)</td>
<td>HDXoverUDP</td>
<td>computer</td>
<td>ICA</td>
<td>Off(0)</td>
</tr>
<tr>
<td>Session reliability connections</td>
<td>AcceptSessionReliabilityConnections</td>
<td>computer</td>
<td>ICA\Session Reliability</td>
<td>Allowed(1)</td>
</tr>
<tr>
<td>Reconnection UI transparency level</td>
<td>ReconnectionUiTransparencyLevel</td>
<td>computer</td>
<td>ICA\Auto Clinet Reconnect</td>
<td>80%</td>
</tr>
<tr>
<td>Session reliability port number</td>
<td>SessionReliabilityPort</td>
<td>computer</td>
<td>ICA\Session Reliability</td>
<td>2598</td>
</tr>
<tr>
<td>Session reliability timeout</td>
<td>SessionReliabilityTimeout</td>
<td>computer</td>
<td>ICA\Session Reliability</td>
<td>180s</td>
</tr>
<tr>
<td>Feature</td>
<td>Setting</td>
<td>Type</td>
<td>Protocol</td>
<td>State</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>----------------------------------------------</td>
<td>--------</td>
<td>----------</td>
<td>--------</td>
</tr>
<tr>
<td>Auto Client Reconnect</td>
<td>AllowAutoClientReconnect</td>
<td>user</td>
<td>ICA</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Audio redirection bandwidth limit</td>
<td>LimitAudioBw</td>
<td>user</td>
<td>Audio</td>
<td>0 Kbps</td>
</tr>
<tr>
<td>Client audio redirection</td>
<td>AllowAudioRedirection</td>
<td>user</td>
<td>Audio</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Client printer redirection</td>
<td>AllowPrinterRedir</td>
<td>user</td>
<td>Printing</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Auto-create PDF Universal Printer</td>
<td>AutoCreatePDFPrinter</td>
<td>user</td>
<td>Printing</td>
<td>Disabled (0)</td>
</tr>
<tr>
<td>Client clipboard redirection</td>
<td>AllowClipboardRedir</td>
<td>user</td>
<td>Clipboard</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Client USB device redirection</td>
<td>AllowUSBRedir</td>
<td>user</td>
<td>USB</td>
<td>Prohibited (0)</td>
</tr>
<tr>
<td>Client USB device redirection rules</td>
<td>USBDeviceRules</td>
<td>user</td>
<td>USB</td>
<td>“0”</td>
</tr>
<tr>
<td>Moving image compression</td>
<td>MovingImageCompressionConfiguration</td>
<td>user</td>
<td>ThinWire</td>
<td>Enabled (1)</td>
</tr>
<tr>
<td>Target minimum frame rate</td>
<td>TargetedMinimumFramesPerSecond</td>
<td>user</td>
<td>ThinWire</td>
<td>10 fps</td>
</tr>
<tr>
<td>Target frame rate</td>
<td>FramesPerSecond</td>
<td>user</td>
<td>ThinWire</td>
<td>30 fps</td>
</tr>
<tr>
<td>Visual quality</td>
<td>VisualQuality</td>
<td>user</td>
<td>ThinWire</td>
<td>Medium (3)</td>
</tr>
<tr>
<td>Use video codec for compression</td>
<td>VideoCodec</td>
<td>user</td>
<td>ThinWire</td>
<td>Use when preferred (3)</td>
</tr>
<tr>
<td>Configuration</td>
<td>Setting Name</td>
<td>Setting Type</td>
<td>Setting Context</td>
<td>Value</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------</td>
<td>--------------</td>
<td>----------------</td>
<td>-------</td>
</tr>
<tr>
<td>Use hardware encoding for video codec</td>
<td>UseHardwareEncodingForVideoCodec</td>
<td>user</td>
<td>ThinWire</td>
<td>Enabled (1)</td>
</tr>
<tr>
<td>Preferred color depth for simple graphics</td>
<td>PreferredColorDepth</td>
<td>user</td>
<td>ThinWire</td>
<td>24 bits per pixel(1)</td>
</tr>
<tr>
<td>Audio quality</td>
<td>SoundQuality</td>
<td>user</td>
<td>Audio</td>
<td>High – high definition audio (2)</td>
</tr>
<tr>
<td>Client microphone redirection</td>
<td>AllowMicrophoneRedir</td>
<td>user</td>
<td>Audio</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Maximum number of sessions</td>
<td>MaximumNumberOfSessions</td>
<td>computer</td>
<td>Load Management</td>
<td>250</td>
</tr>
<tr>
<td>Concurrent logons tolerance</td>
<td>ConcurrentLogonsTolerance</td>
<td>computer</td>
<td>Load Management</td>
<td>2</td>
</tr>
<tr>
<td>Enable auto update of Controllers</td>
<td>EnableAutoUpdateOfControllers</td>
<td>computer</td>
<td>Virtual Delivery Agent Settings</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Clipboard selection update mode</td>
<td>ClipboardSelectionUpdateMode</td>
<td>user</td>
<td>Clipboard</td>
<td>3</td>
</tr>
<tr>
<td>Primary selection update mode</td>
<td>PrimarySelectionUpdateMode</td>
<td>user</td>
<td>Clipboard</td>
<td>3</td>
</tr>
<tr>
<td>Max speex quality</td>
<td>MaxSpeexQuality</td>
<td>user</td>
<td>Audio</td>
<td>5</td>
</tr>
<tr>
<td>Auto connect client drives</td>
<td>AutoConnectDrives</td>
<td>user</td>
<td>File redirection/CDM</td>
<td>Enabled (1)</td>
</tr>
<tr>
<td>Client optical drives</td>
<td>AllowCdromDrives</td>
<td>user</td>
<td>File redirection/CDM</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>-----------------------</td>
<td>------------------</td>
<td>------</td>
<td>----------------------</td>
<td>------------</td>
</tr>
<tr>
<td>Client fixed drives</td>
<td>AllowFixedDrives</td>
<td>user</td>
<td>File redirection/CDM</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Client floppy drives</td>
<td>AllowFloppyDrives</td>
<td>user</td>
<td>File redirection/CDM</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Client network drives</td>
<td>AllowNetworkDrives</td>
<td>user</td>
<td>File redirection/CDM</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Client removable drives</td>
<td>AllowRemoveableDrives</td>
<td>user</td>
<td>File redirection/CDM</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Client drive redirection</td>
<td>AllowDriveRedir</td>
<td>user</td>
<td>File redirection/CDM</td>
<td>Allowed (1)</td>
</tr>
<tr>
<td>Read-only client drive access</td>
<td>ReadOnlyMappedDrive</td>
<td>user</td>
<td>File redirection/CDM</td>
<td>Disabled (0)</td>
</tr>
</tbody>
</table>

**New policies for Linux VDA**

The following policies can be configured in Version 7.12 of Citrix Studio.

**Applies to:** Linux VDA 1.4 and later

**Scope:**

**Value (integer):** [0 - 10]

**Use 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

**Applies to:** Linux VDA 1.4 and later

**Scope:**

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

**Use default value:** 3

**Details:**

PRIMARY selection is used for explicit copy/paste actions such as mouse selection and middle mouse button paste. This setting controls whether PRIMARY selection changes on the Linux VDA can be updated on the client’s clipboard (and vice versa). It can include one of the following selection changes:

- **Selection changes are not updated on the client or the host.**
  PRIMARY selection changes do not update a client’s clipboard. Client clipboard changes do not update PRIMARY selection.

- **Host selection changes are not updated on the client.**
  PRIMARY selection changes do not update a client’s clipboard. Client clipboard changes update the PRIMARY selection.

- **Client selection changes are not updated on the host.**
  PRIMARY selection changes update the client’s clipboard. Client clipboard changes do not update the PRIMARY selection.

- **Selection changes are updated on both the client and host.**
  PRIMARY selection change updates the client’s clipboard. Client clipboard changes update the PRIMARY selection.

**Related Settings:** Clipboard Selection Update Mode

**Applies to:** Linux VDA 1.4 and later

**Scope:**

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

**Use default value:** 3

**Details:**

This setting controls whether CLIPBOARD selection changes on the Linux VDA are updated on the client’s clipboard (and
vice versa. It can include one of the following selection changes:

- **Selection changes are not updated on the client or the host.**
  CLIPBOARD selection changes do not update a client’s clipboard. Client clipboard changes do not update CLIPBOARD selection.

- **Host selection changes are not updated on the client.**
  CLIPBOARD selection changes do not update a client’s clipboard. Client clipboard changes update the CLIPBOARD selection.

- **Client selection changes are not updated on the host.**
  CLIPBOARD selection changes update the client’s clipboard. Client clipboard changes do not update the CLIPBOARD selection.

- **Selection changes are updated on both the client and host.**
  CLIPBOARD selection change updates the client’s clipboard. Client clipboard changes update the CLIPBOARD selection.

**Related Settings:** Primary Selection Update Mode
Configure IPv6

Aug 14, 2017
This release of the Linux VDA provides support for IPv6 to align with functionality previously provided by XenApp and XenDesktop. When using this feature, consider the following:

- For dual stack environments, IPv4 will be 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 needs to be IPv6, not only for the Linux VDA.
- Centrify does not support pure IPv6.

No special setup tasks are required for IPv6 when installing 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:

```
"HKLM\Software\Policies\Citrix\VirtualDesktopAgent" -t "REG_DWORD" -v "OnlyUseIPv6ControllerRegistration"

"HKLM\Software\Policies\Citrix\VirtualDesktopAgent" -t "REG_DWORD" -v "ControllerRegistrationIPv6Netmask"
```

`OnlyUseIPv6ControllerRegistration` must be set to 1 to enable IPv6 on the Linux VDA:

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Policies\Citrix\VirtualDesktopAgent" -t "REG_DWORD" -v "OnlyUseIPv6ControllerRegistration" 0x00000001 --force
```

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:
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Policies\Citrix\VirtualDesktopAgent" -t "REG_SZ" -v "ControllerRegistrationIPv6Netmask" -d "\{IPv6 netmask\}" --force

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 DDC are reachable.
Configure CEIP

When you participate in the Citrix Customer Experience Improvement Program (CEIP), anonymous statistics and usage information are sent to Citrix to help improve the quality and performance of Citrix products.

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

By default, the **CEIPSwitch** property is hidden in the registry. When it remains unspecified, the CEIP is enabled.

You can run the following command on a client to disable the CEIP:

```bash
regedit
```
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:

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\CEIP" -v "DataPersistPath" -d "your_path"
```

If the path you configured does not exist or cannot be accessed, data is saved in the default path.
If the path you configured does not exist or cannot be accessed, data is saved in the default path.

## CEIP data collected from the Linux VDA

The following table gives an example of the types of anonymous information collected. The data does not contain any details that identify you as a customer.

<table>
<thead>
<tr>
<th>Data Point</th>
<th>Key Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Machine GUID</td>
<td>machine_guid</td>
<td>Identifying the machine where the data originates</td>
</tr>
<tr>
<td>AD solution</td>
<td>ad_solution</td>
<td>String denoting the machine's domain joining method</td>
</tr>
<tr>
<td>Linux kernel version</td>
<td>kernel_version</td>
<td>String denoting the machine's kernel version</td>
</tr>
<tr>
<td>LVDA version</td>
<td>vda_version</td>
<td>String denoting the installed version of the Linux VDA.</td>
</tr>
<tr>
<td>LVDA update or fresh install</td>
<td>update_or_fresh_install</td>
<td>String denoting the current Linux VDA package is being freshly installed or updated</td>
</tr>
<tr>
<td>HDX 3D pro enabled or not</td>
<td>hdx_3d_pro</td>
<td>String denoting whether HDX 3D Pro is enabled on the machine</td>
</tr>
<tr>
<td>VDI mode enabled or not</td>
<td>vdi_mode</td>
<td>String denoting whether VDI mode is enabled</td>
</tr>
<tr>
<td>System Locale</td>
<td>system_locale</td>
<td>String denoting the locale of this machine</td>
</tr>
<tr>
<td>LVDA key services last restart time</td>
<td>ctxhdx ctxvda</td>
<td>The last restart time of the ctxhdx and ctxvda services, in the format of dd-hh:mm:ss, for example, 10-17:22:19</td>
</tr>
<tr>
<td>GPU type</td>
<td>gpu_type</td>
<td>Denoting the GPU type of the machine</td>
</tr>
<tr>
<td>CPU cores</td>
<td>cpu_cores</td>
<td>Integer denoting the number of CPU cores of the machine</td>
</tr>
<tr>
<td>CPU frequency</td>
<td>cpu_frequency</td>
<td>Float denoting the CPU frequency in MHz</td>
</tr>
<tr>
<td>Physical memory size</td>
<td>memory_size</td>
<td>Integer denoting the physical memory size in KB</td>
</tr>
<tr>
<td>Active session number</td>
<td>active_session_number</td>
<td>Integer denoting the number of active sessions on the machine at the time we collect this data point</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------</td>
<td>---------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Linux OS name and version</td>
<td>os_name_version</td>
<td>String denoting the Linux OS name and version of the machine</td>
</tr>
<tr>
<td>Session key</td>
<td>session_key</td>
<td>Identifying the session where the data originates</td>
</tr>
<tr>
<td>Reconnect time cost</td>
<td>reconnect_time_cost</td>
<td>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.</td>
</tr>
<tr>
<td>Active session time</td>
<td>active_session_time</td>
<td>Used to save the session’s active times. One session could have multiple active times because the session can disconnect/reconnect.</td>
</tr>
<tr>
<td>Session duration time</td>
<td>session_duration_time</td>
<td>Used to save the session’s duration from logon to logoff</td>
</tr>
<tr>
<td>Receiver client type</td>
<td>receiver_type</td>
<td>Integer denoting the Citrix Receiver type used to launch the session</td>
</tr>
<tr>
<td>Receiver client version</td>
<td>receiver_version</td>
<td>String denoting the version of Citrix Receiver used to launch the session</td>
</tr>
<tr>
<td>Printing count</td>
<td>printing_count</td>
<td>Integer denoting the number of times the session uses the printing function</td>
</tr>
<tr>
<td>USB redirection count</td>
<td>usb_redirecting_count</td>
<td>Integer denoting the number of times the session uses a USB device</td>
</tr>
<tr>
<td>Gfx Provider type</td>
<td>gfx_provider_type</td>
<td>String denoting the graphics provider type of the session</td>
</tr>
<tr>
<td>Shadowing count</td>
<td>shadow_count</td>
<td>Integer denoting the number of times the session has been shadowed</td>
</tr>
<tr>
<td>Language list on Language bar</td>
<td>ctxism_open</td>
<td>Composed long string that contains all languages in the drop-down list of the language bar</td>
</tr>
<tr>
<td>User selected</td>
<td>ctxism_select</td>
<td>Composed long string that contains all languages that</td>
</tr>
</tbody>
</table>
Configure USB redirection

Nov 27, 2017

USB devices are shared among Citrix Receiver and the Linux VDA desktop. Once 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 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. However, in some cases the redirected device cannot be used by an active Linux VDA session. USB devices rely on drivers to function properly (and some devices require special drivers), and in some cases drivers are not provided, which results in some redirected USB devices being inaccessible to an active Linux VDA session. When situations like these occur, the drivers must be installed and the system must be configured properly to ensure USB device connectivity.

The Linux VDA supports a list of USB devices that are successfully redirected to and from the client. In addition, the device is properly mounted, especially the USB disk, allowing the user to access the disk without any additional configuration.

Configure USB redirection

A Citrix policy controls whether USB device redirection is enabled or disabled. In addition, the type of device can also be specified using a Delivery Controller policy. When configuring USB redirection for the Linux VDA, the following policy and rules must be configured:

- Client USB device redirection policy
- Client USB device redirection rules
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**.

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 modifying) a rule, click **OK**.
For more information about configuring generic USB redirection, see Citrix Generic USB Redirection Configuration Guide.

### Troubleshoot USB redirection issues

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

**No devices on the Receiver toolbar**

In some cases, you might not be able to see devices listed on the Citrix Receiver toolbar, which indicates that no USB redirection is taking place. If this occurs, verify the following:

- the policy is configured to allow USB redirection
- the Kernel module is compatible with your kernel
USB devices can be seen in the Receiver toolbar, but are labeled *policy restricted*, which results in failed redirection.

This issue occurs because of the device's policy configuration. In such cases:

- configure the Linux VDA policy to enable redirection
- Verify whether any additional policy restrictions are configured in the Receiver registry; a device might be blocked by the Receiver registry setting. 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](https://docs.citrix.com) on the Citrix Support site.

### A USB device is redirected successfully, but I cannot use it in my session

In most cases, only USB devices in the **supported devices list** can be redirected. However, in some cases other kinds of devices might be redirected into 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 the case of USB drives, the Linux VDA configures and mounts the disk. The user (and only the owner who installed it) can access the disk without any additional configuration. This might not be the case for devices that are not in the supported device list.

### Build the VHCI kernel module

USB redirection depends on the VHCI kernel modules (**usb-vhci-hcd.ko** and **usb-vhci-iocif.ko**). These modules are part of the Linux VDA distribution (as part of the RPM package). They are compiled based on the official Linux distribution kernels and are noted in the table below:

<table>
<thead>
<tr>
<th>Supported Linux distribution</th>
<th>Kernel version</th>
</tr>
</thead>
<tbody>
<tr>
<td>RHEL 6.9</td>
<td>2.6.32-696.10.3.el6.x86_64</td>
</tr>
</tbody>
</table>
Important

If the kernel of your machine is not compatible with the driver built by Citrix for the Linux VDA, the USB service might fail to start. In this case you cannot use the USB redirection feature unless you build your own VHCI kernel modules.

On the command line, execute the following command to verify whether the kernel is consistent:

```
insmod /opt/Citrix/VDA/lib64/usb-vhci-hcd.ko
```

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

If the command executes with errors, the kernel is inconsistent with the Citrix module and must be rebuilt.

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/CentOS**
yum install kernel-devel

tip

zypper install kernel-devel
zypper install kernel-source

Ubuntu 16.04

apt-get install linux-headers

Tip
If the installation is successful, there will be 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, you must 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

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 log on to the session again. Check to see if USB redirection is functioning.

### Supported USB devices

The following devices have been verified to work with this version of the Linux VDA. Other devices might be freely used, with unexpected results:

<table>
<thead>
<tr>
<th>USB mass storage device</th>
<th>VID:PID</th>
<th>File system</th>
</tr>
</thead>
<tbody>
<tr>
<td>Netac Technology Co., Ltd</td>
<td>0dd8:173c</td>
<td>FAT32</td>
</tr>
<tr>
<td>Kingston Datatraveler 101 II</td>
<td>0951:1625</td>
<td>FAT32</td>
</tr>
<tr>
<td>Kingston Datatraveler GT101 G2</td>
<td>1567:8902</td>
<td>FAT32</td>
</tr>
<tr>
<td>SanDisk SDCZ80 flash drive</td>
<td>0781:5580</td>
<td>FAT32</td>
</tr>
<tr>
<td>SanDisk Cruzer 16GB</td>
<td>1058:1088</td>
<td>FAT32</td>
</tr>
<tr>
<td>WD HDD</td>
<td>0781:5567</td>
<td>FAT32</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>USB 3D mouse</th>
<th>VID:PID</th>
</tr>
</thead>
</table>
Known issues

Unable to unmount the redirected USB disk. For the access control of all USB disks redirected from Citrix Receiver, the Linux VDA manages all of these devices under administrative privilege to ensure that only the owner can access the redirected device. As a result, the user is not permitted to unmount the device without the administrative privilege.

![Unable to unmount sda](image)

File lost when you stop redirecting a USB disk. If you redirect a USB disk into a session and try to modify it (for example, create some files on the disk), then stop redirecting it immediately using the Receiver toolbar, 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 actually written to the disk itself. If you stop redirecting using the Receiver toolbar, there is no time remaining for the data being flushed into the disk, which results in lost data.

To resolve this issue, whenever you write data into the disk, use the sync command in a terminal to flush data into the disk, then stop USB redirection.
[vda@redhats72 ~]$ sync
[vda@redhats72 ~]$
Configure session reliability

Nov 27, 2017

With this release, Citrix is introducing the session reliability feature to all supported Linux platforms. Session reliability is enabled by default.

Session reliability reconnects ICA sessions seamlessly across network interruptions. For more information about session reliability, see Auto client reconnect and session reliability.

Note: Data transmitted through a session reliability connection is in plain text by default. For security purposes, Citrix recommends that you enable SSL encryption. For more information about SSL encryption, see Secure user sessions using SSL.

Configuration

You can set the following policies for session reliability in Citrix Studio:

- Session reliability connections
- Session reliability timeout
- Session reliability port number
- Reconnection UI transparency level

For more information, see Session reliability policy settings and Auto client reconnect policy settings.

Note: After you set the Session reliability connections or Session reliability port number policy, restart the VDA service and the HDX service, in this order, for your settings to take effect.

Enable/disable the session reliability TCP listener

By default, the session reliability TCP listener is enabled and listening on port 2598. To disable the listener, run the following command.

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\WinStations\cgp" -v "fEnableWinStation" -d "0x00000000"
```

Note: Restart the HDX service for your settings to take effect. Disabling the TCP listener does not disable session reliability. Session reliability is still available through other listeners (for example, SSL) if the feature is enabled through the Session reliability connections policy.
- **Session reliability port number**

You can also set the session reliability port number with the following command (using port number 2599 as an example).

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\WinStations\cgp" -v "PortNumber" -d 2599
```

**Note**: Restart the HDX service for your setting to take effect. If the port number has been set through the policy setting in Citrix Studio, your setting on the Linux VDA is ignored. Ensure that the firewall on the VDA is configured not to prohibit network traffic through the set port.

- **Server-to-client keep-alive interval**

Session reliability keep-alive messages are sent between the Linux VDA and the ICA client when there is no activity in the session (for example, no mouse movement, no screen update). The keep-alive messages are used to detect whether the client is still responsive. If there is no response from the client, the session is suspended until the client reconnects. This setting specifies the number of seconds between successive keep-alive messages. By default, this setting is not configured. To configure it, run the following command (using 10 seconds as an example).

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\XTEConfig" -t "REG_DWORD" -v "CgpServerToClientKeepAlive" -d 10 --force
```

- **Client-to-server keep-alive interval**

This setting specifies the number of seconds between successive keep-alive messages sent from the ICA client to the Linux VDA. By default, this setting is not configured. To configure it, run the following command (using 10 seconds as an example).

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\XTEConfig" -t "REG_DWORD" -v "CgpClientToServerKeepAlive" -d 10 --force
```

**Troubleshooting**
Unable to launch sessions after enabling session reliability through the policy setting

To work around this issue, do the following:

1. Ensure that the VDA service and HDX service are restarted, in this order, after you enable session reliability through the policy setting in Citrix Studio.
2. Use the following command to verify that the session reliability TCP listener is running (using port 2598 as an example).

```bash
netstat -an | grep 2598
```

If there is no TCP listener on the session reliability port, enable the listener with the following command.

```bash
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\WinStations\cgp" -v "fEnableWinStation" -d "0x00000001"
```
Client IME

Aug 14, 2017

Overview

Double-byte characters (such as Chinese, Japanese, and Korean characters) must be typed through an Input Method Editor (IME). The Client IME provides a way to type such characters by means of any IME that works with Citrix Receiver 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

- You must double-click a cell in a Google spreadsheet before you can use the Client IME to type characters in the cell.
- The Client IME is not automatically disabled in Password fields.
- IME compose panel does not follow the cursor in input area.
- Client IME is not supported in a SUSE 11 distribution.
Support for multiple language inputs

Nov 27, 2017

As of Linux VDA Version 1.4, Citrix has added support for published applications so that users can access a desired Linux application without the Linux desktop environment.

However, the native language bar on the Linux VDA server was unavailable to the published application because the language bar is highly integrated with the Linux desktop environment. As a result, users could not input text in a language that requires IME such as Chinese, Japanese, or Korean. Further, it was also not possible for users to switch between keyboard layouts during an application session.

To address those issues, this feature provides a language bar for published applications that accept text input. The language bar enables users to select a server-side IME and to switch between keyboard layouts during an application session.

Configuration

You can use the `ctxreg` utility to enable or disable this feature (disabled by default). The feature configuration on a given Linux VDA server applies to all applications published on that VDA.

The configuration key is “HKEY_LOCAL_MACHINE \SYSTEM\CurrentControlSet\Control\Citrix\LanguageBar” and the type is DWORD.

To enable this feature, run the command:

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE \SYSTEM\CurrentControlSet\Control\Citrix\LanguageBar" -v "Enabled" -d "0x00000001"
```

To disable this feature, run the command:

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE \SYSTEM\CurrentControlSet\Control\Citrix\LanguageBar" -v "Enabled" -d "0x00000000"
```

Usage
The usage is quite straightforward.

1. Enable the feature.
2. Access a published application that can accept text input. A language bar appears in the session, alongside the application.
3. From the drop-down menu, select **Region & Language** to add the desired language (input source).

4. Select the IME or keyboard layout from the drop-down menu.
5. Type a language using the selected IME or keyboard layout.

**Note**
- When you change a keyboard layout on the VDA-side language bar, ensure that the same keyboard layout is used on the client side (running Citrix Receiver).
- The `accountsservice` package must be upgraded to Version 0.6.37 or later before you can perform settings in the **Region & Language** dialog box.
HDX Insight

Aug 14, 2017

Overview

HDX Insight is part of the NetScaler Insight Center 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 CloudBridge application networking fabric.

In this release, the Linux VDA partially supports the HDX Insight feature. Because the End User Experience Management (EUEM) feature is not implemented, the data points related to time duration are not available.

Installation

No dependent packages need installation.

Usage

HDX Insight analyzes the ICA messages passed through NetScaler between the Citrix Receiver and the Linux VDA.

You must set up an Insight Center environment with the Linux VDA and enable the HDX Insight feature. For the information on using the HDX Insight feature, see Use Cases: HDX Insight.

Troubleshooting

No data points are displayed

There might be two causes:

- HDX Insight is not configured correctly.
  
  For example, AppFlow is not enabled on 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.
Adaptive transport

Nov 27, 2017

With this release, Citrix is introducing adaptive transport to all supported Linux platforms as an experimental feature. This feature is disabled by default.

Adaptive transport is a data transport mechanism for XenApp and XenDesktop. It is faster, more scalable, improves application interactivity, and is more interactive on challenging long-haul WAN and internet connections. For more information about adaptive transport, see Adaptive transport.

Note: Data transmitted through adaptive transport is in plain text. Datagram Transport Layer Security (DTLS) is not available with the Linux VDA.

Enable adaptive transport

In Citrix Studio, verify that the HDX Adaptive Transport policy is set to Preferred or Diagnostic mode.

Preferred: Adaptive transport over Enlightened Data Transport (EDT) is used when possible, with fallback to TCP. Diagnostic mode: EDT is forced on and fallback to TCP is disabled.

In StoreFront, add the “HDXoverUDP=Preferred” or “HDXoverUDP= On” (Diagnostic mode) entry to the “Application” section of the default.ica configuration file. Ensure that the set mode is consistent with the policy setting in Citrix Studio.

Run the following command as a root user on the Linux VDA to enable adaptive transport.

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server\Wds\icawd\Tds\udp" -v "fEnabled" -d "0x00000001"
```

Adaptive transport reuses the ICA (1494) or session reliability (2598) port. Ensure that your firewall rules are set to allow UDP traffic through either of the ports.

To make your firewall allow UDP traffic through port 1494 or 2598, do the following based on the Linux distribution:

RHEL 7/CentOS 7

Run the following command as a root user.

```bash
firewall-cmd --permanent --zone=public --add-port=1494/udp
firewall-cmd --permanent --zone=public --add-port=2598/udp
```

**RHEL 6/CentOS 6**

1. Add the following lines to the `ctxhdx` file at `/etc/xdl/firewall`.

```bash
-I INPUT 1 -p udp -m udp --dport 1494 -j ACCEPT
-I INPUT 1 -p udp -m udp --dport 2598 -j ACCEPT
```

2. Run the following command as a root user.

```bash
lokkit --custom-rules=ipv4:filter:/etc/xdl/firewall/ctxhdx
```

**SUSE 12**

1. Add the following line to the `ctxhdx` file at `/etc/sysconfig/SuSEfirewall2.d/services/`.

```bash
```

https://docs.citrix.com  © 1999-2017 Citrix Systems, Inc. All rights reserved.  p.197
2. Run the following command as a root user.

```
yast2 firewall services add zone=EXT service=service:ctxhdx
```

Ubuntu 16

Run the following commands as a root user.

```
ufw allow proto udp from any to any port 1494
ufw allow proto udp from any to any port 2598
```

**Note:** After enabling adaptive transport, restart the VDA service and the HDX service, in this order, for your settings to take effect.

**Troubleshooting**

Run the following command to check whether UDP listeners are running.
In normal circumstances, the output is similar to the following.

```
udp    0       0 0.0.0.0:2598            0.0.0.0:*                 
udp    0       0 ::::1494               :::*                 
```
Tracing On

Aug 14, 2017

Overview

Collecting logs and reproducing issues slow down the diagnostics and degrade the user experience. This release provides the Tracing On feature to ease such efforts. Tracing is enabled for the Linux VDA by default.

Configuration

The ctxlogd daemon and the setlog utility are now included in the Linux VDA release package. By default, the ctxlogd daemon starts after you install and configure the Linux VDA.

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.

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
```

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

View or set a single setlog value:

$ setlog value <name> [value]

For example:

$ setlog value log_size 100

Levels

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

View the log levels set for different components:
You can set all log levels (including Disable, Inherited, Verbose, Information, Warnings, Errors, and Fatal Errors) with the following command:

```
$ setlog levels
```

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.

**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

View the current flags:
$ setlog flags

View or set a single log flag:

$ setlog flag <flag> [state]

Restore Defaults

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

$ setlog default

Important

The ctxlogd service is configured using the /var/xdl/.ctxlog file, which only root users can create. Other users do not have write permission to this file. Citrix recommends that root users not give write permission to other users. Failure to comply can cause the arbitrary or malicious configuration to ctxlogd, which can affect server performance and consequently 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.

To solve this issue, run setlog as a root user to recreate the /var/xdl/.ctxlog file. Then restart the ctxlogd service on which other services depend.
Shadow sessions

Nov 27, 2017
Enabled by default as of this release, the session shadowing feature allows domain administrators to view users' ICA sessions in an intranet. The feature uses noVNC to connect to the ICA sessions and is currently supported only with RHEL 7.3 and Ubuntu 16.04.

Note
To use the session shadowing feature, the version of Citrix Director must be 7.16 or later.

Installation and configuration

Two new dependencies, python-websockify and x11vnc, are required for session shadowing. The python-websockify and x11vnc dependencies are automatically installed when you install the Linux VDA on Ubuntu 16.04. On RHEL 7.3, you must manually install python-websockify and x11vnc after you install the Linux VDA.

Run the following command on RHEL 7.3 to install python-websockify and x11vnc (x11vnc version 0.9.13 or later).

```bash
sudo yum install -y python-websockify x11vnc
```

To resolve python-websockify and x11vnc, enable the following repositories on RHEL 7.3:

- EPEL

The Extra Packages for Enterprise Linux (EPEL) repository is required for both python-websockify and x11vnc. Run the following command to enable the EPEL repository:

```bash
sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-$(rpm -E '%{rhel}').noarch.rpm
```

Optional RPMs
Run either of the following commands to enable the optional RPMs repository for installing some dependency packages of x11vnc:

For workstation:

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

For server:

```
subscription-manager repos --enable=rhel-7-server-optional-rpms
```

The session shadowing feature automatically selects available ports from within 6001-6099 to build up connections from the Linux VDA to Citrix Director. Therefore, the number of ICA sessions that you can shadow concurrently is limited to 99. Make sure that enough ports are available to meet your requirements, especially for multi-session shadowing.

The table below lists related registries:

<table>
<thead>
<tr>
<th>Registry</th>
<th>Description</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>EnableSessionShadowing</td>
<td>Enables or disables the session shadowing feature</td>
<td>1 (Enabled)</td>
</tr>
<tr>
<td>ShadowingUseSSL</td>
<td>Determines whether to encrypt the connection between the Linux VDA and Citrix Director</td>
<td>0 (Disabled)</td>
</tr>
</tbody>
</table>

Run the ctxreg command on the Linux VDA to change the registry values. For example, to disable session shadowing, run the following command:

```
```
The noVNC connection between the Linux VDA and Citrix Director uses the WebSocket protocol. For session shadowing, whether ws:// or wss:// is chosen is determined by the aforementioned "ShadowingUseSSL" registry. By default, ws:// is chosen. However, for security reasons, Citrix recommends that you use wss:// and install certificates on each Citrix Director client and on each Linux VDA server. Citrix disclaims any security responsibility for the Linux VDA session shadowing with ws://.

Obtain server and root SSL certificates

Certificates must be signed by a trusted Certificate Authority (CA).

A separate server certificate (including the key) is required for each Linux VDA server on which you want to configure SSL. A server certificate identifies a specific computer, so you must know the Fully Qualified Domain Name (FQDN) of each server. For convenience, you can use a wildcard certificate for the whole domain instead. In this case, you must know at least the domain name.

In addition to installing a server certificate on each server, you must install a root certificate from the same CA on each Citrix Director client that communicates with the Linux VDA server. Root certificates are available from the same CAs that issue the server certificates. You can install server and client certificates from a CA that is bundled with your operating system, from an enterprise CA (a CA that your organization makes accessible to you), or from a CA not bundled with your operating system. Consult the security team of your organization to find out which of the methods they require for obtaining certificates.

Important:

- The Common Name for a server certificate must be the exact FQDN of the Linux VDA server or at least the correct wildcard plus domain characters (for example, vda1.basedomain.com or *.basedomain.com).
- Hashing algorithms including the SHA1 and MD5 are too weak for signatures in digital certificates for some browsers to support. So SHA-256 is specified as the minimum standard.

Install a root certificate on each Citrix Director client

Session shadowing uses the same registry-based certificate store as IIS, so you can install root certificates using IIS or the Microsoft Management Console (MMC) Certificates snap-in. When you receive a certificate from a CA, you can restart the Web Server Certificate Wizard in IIS and the wizard installs the certificate. Alternatively, you can view and import certificates on the computer using the MMC and add the certificate as a standalone snap-in. Internet Explorer and Google Chrome import the certificates installed on your operation system by default. For Mozilla Firefox, you must import your root SSL certificates on the Authorities tab of Certificate Manager.

Install a server certificate and its key on each Linux VDA server

Name the server certificates "shadowingcert.*" and the key file "shadowingkey.*" (* can indicate the format, for example,
shadowingcert.csr and shadowingkey.key). Put server certificates and key files under the path `/etc/xdl/shadowingssl` and protect them properly with restricted permissions. An incorrect name or path makes the Linux VDA unable to find a specific certificate or key file and consequently causes connection failure with Citrix Director.

Usage

From Citrix Director, find the target session and click Shadow in the Session Details view to send a shadowing request to the Linux VDA.

![Session Details](image)

After the connection initializes, a confirmation appears on the ICA session client (not the Citrix Director client) to ask the user for permission to shadow the session.

![Confirmation](image)

If the user clicks Yes, a window appears on the Citrix Director side, indicating that the ICA session is being shadowed.

For more information about the usage, see the Citrix Director Documentation.

Limitations

- Session shadowing is designed for use in an Intranet only. It does not work for external networks even connecting
Through NetScaler, Citrix disclaims any responsibility for the Linux VDA session shadowing in an external network.

- With session shadowing enabled, a domain administrator can only view the ICA sessions, but has no permission to write or control it.
- After an administrator clicks Shadow from Citrix Director, a confirmation appears to ask the user for permission to shadow the session. A session can be shadowed only when the session user gives the permission.
- The aforementioned confirmation has a timeout limitation, which is 20s. A shadowing request fails when the time runs out.
- One ICA session can be shadowed by only one administrator in one Citrix Director window. If an ICA session has been shadowed by administrator A and meanwhile, administrator B sends a shadowing request, the confirmation for getting the user permission reappears on the user device. If the user agrees, the shadowing connection for administrator A discontinues and a new shadowing connection is built for administrator B. It is the same if another shadowing request for the same ICA session is sent by the same administrator.
- To use session shadowing, install Citrix Director 7.16 or later.
- A Citrix Director client uses a FQDN rather than an IP address to connect to the target Linux VDA server. Therefore, the Citrix Director client must be able to resolve the FQDN of the Linux VDA server.

**Troubleshooting**

If session shadowing fails, perform debugging on both the Citrix Director client and the Linux VDA.

Through the developer tools of the browser, check the output logs on the Console tab, or check the response of the ShadowLinuxSession API on the Network tab. If the confirmation for getting the user permission appears but the connection fails to be built, manually ping the FQDN of the Linux VDA to verify that Citrix Director can resolve the FQDN. In case of an issue with the wss:// connection, check your certificates.

Verify that the confirmation for getting the user permission appears in response to a shadowing request. If it does not, check the vda.log and hdx.log files for clues. To obtain the vda.log file, do the following:

1. Find the `/etc/xdl/ctx-vda.conf` file. Uncomment the following line to enable the vda.log configuration:

   ```
   Log4jConfig="/etc/xdl/log4j.xml"
   ```

2. Open `/etc/xdl/log4j.xml`, locate the `com.citrix.dmc` part, and change “info” to “trace” as follows:

   ```
   Log4jConfig="/etc/xdl/log4j.xml"
   ```
3. Run the `service ctxvda restart` command to restart the ctxvda service.

In case of an error during connection build-up:

1. Check for any firewall limitation that stops session shadowing from opening the port.
2. Verify that the certificates and key files are named properly and put under the right path in case of the SSL scenario.
3. Verify that there are enough ports left between 6001-6099 for new shadowing requests.
Secure user sessions using SSL

Nov 27, 2017

As of Version 7.16, the Linux VDA supports SSL encryption for secure user sessions. SSL encryption is disabled by default.

Enable SSL encryption

To enable SSL encryption for secure user sessions, obtain certificates and enable SSL encryption on both the Linux VDA and the Delivery Controller (the Controller).

Obtain server certificates in PEM format and root certificates in CRT format from a trusted Certificate Authority (CA). A server certificate contains the following sections:

- certificate
- password-decrypted private key
- intermediate certificates (optional)

An example of a server certificate:

```
-----BEGIN CERTIFICATE-----
MIIDTTCCArAgAwIBAgIJALluncpliqGXCMA0GCSqGSIb3DQEBBQUAMGcxCzAJBgNVB
BAcTCUNhbWJvdXJuZTEUMBIGA1UEChMLQ2l0cml4IFRlc3RvXzAZBgNVBAsTElNl
```

https://docs.citrix.com © 1999-2017 Citrix Systems, Inc. All rights reserved.
-----END RSA PRIVATE KEY-----

-----BEGIN CERTIFICATE-----

MIIDGTCCAoAwIBAgIuAdhMA0GCSqGSIb3DQEBBQUAMGcxCzAJBgNV

-----END CERTIFICATE-----
Enable SSL encryption on the Linux VDA

On the Linux VDA, use the `enable_vdassl.sh` tool to enable (or disable) SSL encryption. The tool is located in the `/opt/Citrix/VDA/sbin` directory. For information about options available in the tool, run the `/opt/Citrix/VDA/sbin/enable_vdassl.sh –h` command.

Tip: A server certificate must be installed on each Linux VDA server and root certificates must be installed on each Linux VDA server and client.

Enable SSL encryption on the Controller

Note

- The Controller must use the Fully Qualified Domain Name (FQDN) of the Linux VDA – it cannot use the IP address, which is used by default, to connect to the target Linux VDA.
- You can enable SSL encryption only for entire delivery groups. You cannot enable SSL encryption for specific applications.

In a PowerShell window on the Controller, run the following commands in sequence to enable SSL encryption for the target delivery group and to make the Controller use the FQDN of the Linux VDA.

1. Asnp citrix.*
3. Set-BrokerSite –DnsResolutionEnabled True

To disable SSL encryption on the Controller, run the following commands in sequence:

1. Asnp citrix.*
3. Set-BrokerSite –DnsResolutionEnabled False

Troubleshooting
The following “Can't assign requested address” error might occur in Citrix Receiver for Windows when you try to access a published desktop session:

As a workaround, add an entry to the **hosts** file, which is similar to:

```
10.108.13.180   rhvm72work.citrixlab.local
```

where

- **10.108.13.180** is the IP address of the Linux VDA.
- **rhvm72work.citrixlab.local** is the FQDN of the Linux VDA.

On Windows machines, the **hosts** file is typically located at `C:\Windows\System32\drivers\etc\hosts`. 
Configure unauthenticated sessions

Nov 27, 2017
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. This functionality is also not supported on Citrix Receiver for Android.

Create an unauthenticated store

You must create an unauthenticated store using StoreFront to support an unauthenticated session on the Linux VDA.

Enable unauthenticated users in a Delivery Group

After creating an unauthenticated store, enable unauthenticated users in a Delivery Group to support an unauthenticated session. To enable unauthenticated users in a Delivery Group, follow the directions based on the version of XenDesktop:

- XenDesktop 7.16
- XenDesktop 7.15
- XenDesktop 7.14
- XenDesktop 7.13
- XenDesktop 7.12
- XenDesktop 7.11
- XenDesktop 7.9
- XenDesktop 7.8
- XenDesktop 7.7
- XenDesktop 7.6

Note
Unauthenticated users are supported beginning with XenApp and XenDesktop 7.6.

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 5
minutes:

```
sudo /opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\CurrentControlSet\Control\Citrix" -v AnonymousUserIdleTime -d 0x00000005
```

Set the maximum number of unauthenticated users

To set the maximum number of unauthenticated users, use the registry key `MaxAnonymousUserNumber`. This setting limits the number of unauthenticated sessions running on a single Linux VDA concurrently. Use the `ctxreg` tool to configure this registry setting. For example, to set the value to 32:

```
sudo /opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\CurrentControlSet\Control\Citrix" -v MaxAnonymousUserNumber -d 0x00000020
```

Important

It is essential that you limit the number of unauthenticated sessions. Too many sessions being launched concurrently can cause problems on the VDA, including running out of available memory.

Troubleshooting

Consider the following when configuring unauthenticated sessions:

- **Failed to log on to an unauthenticated session.**

Verify that the registry was updated to include the following (set to 0):
sudo /opt/Citrix/VDA/bin/ctxreg read –k "HKLM\System\CurrentControlSet\Control\Citrix" –v MaxAnonymousUserNumber

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'
```

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, you need to add the following lines to the configuration file `$Home/.kde/share/config/kdeglobals`. For example:

  ```
  [KDE Action Restrictions]
  action/lock_screen=false
  
  However, if the KDE Action Restrictions parameter is configured as immutable in a global wide `kdeglobals` file such as `/usr/share/kde-settings/kde-profile/default/share/config/kdeglobals`, the user configuration has no effect.

  To resolve this issue, try to modify the system-wide `kdeglobals` file to remove the `[si]` tag at the [KDE Action Restrictions] section or directly use the system-wide configuration to disable the lock screen button and menu. For details about the KDE configuration, see the KDE System Administration/Kiosk/Keys page.
Configure LDAPS

Aug 14, 2017

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 will 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 in 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`

- fallback to LDAP without SSL/TLS

  `/opt/Citrix/VDA/sbin/enable_ldaps.sh -Disable`

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
Configure LDAPS for third-party platform

Besides the Linux VDA components, there are several third-party software components that adhere to the Linux VDA that 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.

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](https://docs.citrix.com).

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`. Make changes to the files as shown below:

**ldap.conf:**

```
TLS_REQCERT never
```

**smb.conf:**

```
ldap ssl = start tls
ldap ssl ads = yes
client ldap sasl wrapping = plain
```

Alternately, secure LDAP can be configured by SASL GSSAPI sign and seal, but it cannot coexist with TLS/SSL. To use SASL encryption, change the `smb.conf` configuration:
smb.conf:

ldap ssl = off

ldap ssl ads = no

client ldap sasl wrapping = seal

Centrify does not support LDAPS on port 636. However, it does provide secure encryption on port 389. For more information, see the Centrify site.

Quest Authentication Service does not support LDAPS on port 636, but it provides secure encryption on 389 using a different method. For more information, see the Quest authentication article.

Troubleshooting

The following issues might arise when using 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 port(s) 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 (listed above) 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 occurs.
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 Certification is the correct type. Generally speaking, it’s supposed to work with most of the Java Keytool types supported. If it’s 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 new keystore.
Configure Xauthority

Aug 14, 2017

Environments that use interactive remoting using X11 display functionality (including xterm, gvim, etc.) are supported by the Linux VDA. This feature provides a security mechanism necessary to ensure secure communication between XClient and XServer.

There are two methods to secure permission for this secure communication:

- **Xhost.** By default, Xhost allows only the localhost XClient to communicate with XServer. If you choose to allow a remote XClient to access XServer, the Xhost command must be executed to grant permission on the specific machine. Or, you can alternately use `xhost +` to allow any XClient to connect to XServer.
- **Xauthority.** The Xauthority file can be found in each user's home directory and is used to store credentials in cookies used by xauth for authentication of XServer. Once an XServer instance (Xorg) is started, the cookie is used to authenticate connections to that specific display.

How it works

When Xorg starts up, an 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
```

If XClient connects to the Xorg remotely, two prerequisites must be met:

- Set the `DISPLAY` environment variable to the remote XServer.
- Get the Xauthority file which contains one of the cookie numbers in Xorg.

Configure Xauthority
To enable Xauthority on the Linux VDA for remote X11 display, you must create the following two registry keys:

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

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 (the filename is .Xauthority) 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:

```bash
set DISPLAY={Display number stored in the Xauthority file}
set XAUTHORITY={the file path of .Xauthority}
```

**Note**

If the XAUTHORITY environment variable is not set, the ~/.Xauthority file will be 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 manually transmit this Xauthority file to XClient. After the DISPLAY and XAUTHORITY environment variables are set correctly, the GUI is displayed in XServer desktop automatically.

**Troubleshooting**

If Xauthority fails to work, follow the troubleshooting steps below:

1. As an administrator with root privilege, retrieve all of Xorg cookies:
This command displays the Xorg process and the parameters passed to Xorg while starting. Another parameter displays which Xauthority file is used. For example:

```
ps aux | grep -i xorg
```

```
/var/xdl/xauth/.Xauthority110
```

Display the cookies using the `Xauth` command:

```
Xauth -f /var/xdl/xauth/.Xauthority110
```

2. Use the `Xauth` command to show the cookies contained in `~/.Xauthority`. For the same display number, the displayed cookies must be the same in the Xauthority files of Xorg and XClient.

3. If the cookies are the same, check the remote display port accessibility with 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
```

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

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