Linux Virtual Delivery Agent 2407

Contents

Linux Virtual Delivery Agent 2407 6

What's new 6

Fixed issues 10

Known issues 10

Third party notices 14

Deprecation 14

System requirements 16

Installation overview 21

Prepare to install 21

Create domain-joined VDAs using easy install 23

Create non-domain-joined Linux VDAs using MCS 46

Create a non-domain-joined Linux VDA using easy install (preview) 59

Create Linux VDAs using Machine Creation Services (MCS) 69

Create Linux VDAs using Citrix Provisioning 93

Create Linux VDAs in Citrix DaaS Standard for Azure 94

Install the Linux VDA manually 99

Install the Linux VDA on Amazon Linux 2, RHEL, and Rocky Linux manually 99

Install the Linux VDA on SUSE manually 134

Install the Linux VDA on Ubuntu manually 161

Install the Linux VDA on Debian manually 192

Configure 221

Administration 222

Linux VDA data collection program 222

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

HDX Insight **226**

Integration with the Citrix Telemetry Service **227**

Linux VDA self-update through Azure **231**

Linux VM and Linux session metrics **234**

Log collection **237**

Session shadowing **242**

The monitor service daemon **250**

Troubleshooting **252**

Others **261**

Citrix Workspace app for HTML5 support **261**

Create a Python3 virtual environment **262**

Integrate NIS with Active Directory **264**

IPv6 **269**

LDAPS **271**

Xauthority **275**

Authentication **278**

Authentication with Azure Active Directory **278**

Double-hop single sign-on authentication **282**

Federated Authentication Service **284**

FIDO2 (preview) **293**

Non-SSO authentication **295**

Smart cards **295**

Access by unauthenticated (anonymous) users **308**

File **310**

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

File copy and paste 310

File transfer 311

Graphics 314

Automatic DPI scaling 315

Client battery status display 316

Graphics configuration and fine-tuning 318

HDX screen sharing 330

Multi-monitor support 337

Non-virtualized GPUs 343

Session watermark 346

Shared GPU acceleration on a multi-session Linux VDA 350

System tray 352

Thinwire progressive display 357

General content redirection 359

Client drive mapping 360

USB device redirection 361

Clipboard redirection 369

Keyboard 371

X Keyboard Extension (XKB) configuration 371

Client Input Method Editor (IME) 372

Client IME user interface synchronization 373

Dynamic keyboard layout synchronization 377

Soft keyboard 380

Support for multiple language inputs 383

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Multimedia 385

Audio features 385

Browser content redirection 389

HDX webcam video compression 399

Non-domain-joined Linux VDAs 404

Policy support list 407

Printing 421

Printing best practices 421

PDF printing 428

Remote PC Access 429

Session 441

Adaptive transport 442

HDX adaptive throughput 445

Custom backgrounds and banner messages on session logon screens 445

Custom desktop environments by session users 450

Logon with a temp home directory 451

Publish applications 453

Rendezvous V1 455

Rendezvous V2 459

Secure HDX (preview) 464

Secure user sessions using DTLS 465

Secure user sessions using TLS 466

Session reliability 469

Session recording (preview) 472

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Virtual Channel SDK (preview) **474**

Wayland (preview) **475**

Best practices **476**

Create Linux VDAs on Google Cloud Platform (GCP) using Machine Creation Services (MCS) **476**

Manage your deployment using Ansible **497**

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Linux Virtual Delivery Agent 2407

August 7, 2024

> **Important:**
>
> The product lifecycle strategy for Current Releases (CR) and Long Term Service Releases (LTSR) is described in Lifecycle Milestones.

The Linux Virtual Delivery Agent (VDA) enables access to the Linux virtual apps and desktops anywhere from any device where Citrix Workspace app is installed.

You can deliver virtual apps and desktops based on supported Linux distributions. Install the VDA software on your Linux virtual machines (VMs), configure the Delivery Controller, and then use Citrix Studio to make the apps and desktops available to users.

What's new

August 1, 2024

What's new in 2407

Support for RHEL 9.4 / 8.10, Rocky Linux 9.4 / 8.10, and Debian 12.5 / 11.9

The Linux VDA now supports the following Linux distributions:

- RHEL 9.4
- Rocky Linux 9.4
- RHEL 8.10
- Rocky Linux 8.10
- Debian 12.5
- Debian 11.9

For more information about supported Linux distributions, see System requirements.

New .NET runtime required

In addition to the .NET Runtime, you must install .ASP.NET Core Runtime on all supported Linux distributions before you install or upgrade the Linux VDA. Version 6 is required for Amazon Linux 2. Version 8 is required for other distributions.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

If your Linux distribution contains the .NET version that you require, install it from the built-in feed. Otherwise, install .NET from the Microsoft package feed. For more information, see https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers.

Token-based enrollment extended to on-premises environments (preview)

This release extends the token-based enrollment feature to on-premises environments. When creating non-domain-joined VDAs using easy install, you can now:

- Enroll non-domain-joined VDAs with a machine catalog of both cloud and on-premises delivery controllers.
- Use the Citrix Web Studio to create an empty machine catalog and generate an enrollment token.

For more information, see Create a non-domain-joined Linux VDA using easy install (preview).

SSSD designated as the default domain joining method

SSSD is now the default method for joining domains, replacing Winbind. This change impacts the configuration behavior when you run ctxsetup.sh or the easy install script (ctxinstall.sh) in interactive mode. For more information, see Create domain-joined VDAs using easy install.

Delegating package managers to handle configuration files during upgrades

This release delegates package managers **rpm** or **dpkg** to handle configuration files during upgrades. Furthermore, this release addresses some issues flagged by RPMlint and rebuilds the RPM package to better comply with RPMlint requirements. For more information, see Step 13: Upgrade the Linux VDA (optional).

Support for smart card removal settings

You can now determine what happens when the smart card for a signed-in user is removed from the smart card reader during a session by editing the registry on the Linux VDA. For more information, see Smart card removal settings.

Ability to prevent specific client printers from redirecting to HDX sessions

You can now prevent specific client printers from redirecting to HDX sessions by configuring the Citrix policy setting **Printer driver mapping and compatibility**. For more information, see Printer driver mapping and compatibility.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Loss tolerant mode for audio (preview)

Audio is now supported over the Enlightened Data Transport (EDT) loss tolerant protocol. This feature increases the user experience for real-time streaming when users are connecting through networks with high latency and packet loss. When this feature is enabled, Adaptive Transport in Citrix Virtual Apps and Desktops uses the EDT loss tolerant transport protocol for a better audio experience. This feature is disabled by default. For more information, see Loss tolerant mode for audio in the Audio features article.

Packet Loss Concealment implemented for improving audio performance (preview)

Starting with the 2407 release, the jitter buffer mechanism is improved, and the Packet Loss Concealment (PLC) is added for the adaptive audio codec. PLC helps to reconstruct the lost data packets. This enhancement helps to improve the packet loss tolerance and jitter tolerance and thus improves audio performance for loss tolerant mode (EDT lossy) for audio.

To enable this feature, you also need to enable the Loss tolerant mode for audio feature. For more information, see Packet Loss Concealment for improving audio performance (preview) in the Audio features article.

Support for PipeWire in Debian 12.5, RHEL 9.x, and Rocky Linux 9.x

The Linux VDA now supports PipeWire in Debian 12.5, RHEL 9.x, and Rocky Linux 9.x, where it is the default audio service. For more information, see Audio features.

Enhanced graphics performance

This release enhances graphics performance by upgrading JPEG Turbo to Version 3 (V3) on the Linux VDA.

Login process details shown on the login UI

Session users can now see the following login process details on the login UI when they open a Linux VDA session:

- **Evaluating Citrix group policy settings**
- **Authenticating user access**
- **Launching desktop environment** (in case of a virtual desktop session) or **Launching application window** (in case of a virtual application session)

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Keyboard layout synchronization support for KDE and XFCE desktop environments

This release extends the dynamic keyboard layout synchronization feature to the XFCE and KDE Plasma 5 desktop environments. For more information, see Dynamic keyboard layout synchronization.

Making evdev the default X Keyboard Extension (XKB) rule

To create a better user experience with more responsive and predictable keyboard behavior, the Linux VDA now specifies evdev as the default XKB rule for keyboard configuration.

For more information, see X Keyboard Extension (XKB) configuration.

LDAP query performance optimization

With the new release, unnecessary communication to the LDAP server is removed and indexed query key is used to achieve better LDAP query performance and reduce cost.

Improved VDA scalability for Cloud Connectors

This release implements improvements in the VDA scalability so that each Citrix Cloud Connector (4 vCPU, 10 GB memory) can now support 6,000 Linux VDAs. For more information, see Size and scale considerations for Cloud Connectors.

HDX adaptive throughput is now enabled by default

For more information, see HDX adaptive throughput.

> **Note:**
>
> Starting with this release, we offer a comprehensive best practice article for customers interested in using Ansible for deployment management. For more information, see Manage your deployment using Ansible.

What's new in earlier releases

For new features included in the releases that shipped after the 1912 LTSR through the 2402 LTSR, see What's new history.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Fixed issues

July 11, 2024

The Linux Virtual Delivery Agent 2407 includes the following fixed issues:

- Root users are unable to copy read-only files to client devices mapped using CDM. [CVADHELP-25124]
- The group policy parser might consume 100% CPU when handling huge group policy sets from a GPF file. [CVADHELP-25323]
- The ctxnsap process exits unexpectedly when sessions disconnect, preventing the Citrix Application Delivery Management (ADM) from displaying HDX insight information. [CVADHELP-25493]
- Incorrect ad_join.sh execution on the template machine can generate a flag called mcs_setting_flag that prevents newly MCS-created machines from joining a domain. This fix introduces a check in the deploymcs.sh script to identify and remove the flag when detected. [CVADHELP-25117]

Known issues

August 8, 2024

The following issues have been identified in this release:

- The smart card service leaks file descriptors during smart card authentication, leading to a blockage of new smart card access. This issue occurs because, by default, most Linux distributions limit the maximum number of open files to 1,024 for each process. When the smart card service exhausts this limit, it can no longer establish new connections, effectively blocking subsequent smart card access.

  This issue affects VDAs with smart card logon enabled. Symptoms include numerous **Failed to accept new connection: Too many open files** errors in **/var/log/xdl/hdx.log** and an accumulation of file descriptors in **/proc/${pid}/fd/**, where **${pid}** represents the process ID of **ctxscardsd**. To determine the PID, use the command **systemctl status ctxscardsd|grep PID**.

  To mitigate this issue, you can either increase the maximum open files limit for the smart card service or restart the smart card service. Ensure that there are no active sessions before attempting to restart the service. Use the following commands to increase the limit or restart the service:

  - To restart the smart card service:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
systemctl restart ctxscardsd
```

- To query the current service max open files:

```
cat /proc/${
PID }
/limits
```

- To set the maximum open files for the smart card service:

  1. Open the **ctxscardsd.service** file in read-only mode to check the current settings:

     ```
     vim -R /lib/systemd/system/ctxscardsd.service
     ```

  2. Add the following line to the **Service** section in **ctxscardsd.service** to increase the limit:

     ```
     LimitNOFILE=65536
     ```

  3. Reload the **systemd** daemon and restart the **ctxscardsd** service:

     ```
     systemctl daemon-reload
     systemctl restart ctxscardsd
     ```

  4. Verify the new limit:

     ```
     cat /proc/${
      PID }
      /limits
     ```

> **Note:**
>
> Increasing the max open files can extend the time before running out of file descriptors, but a restart of **ctxscardsd** might still be necessary eventually.

[LNXVDA-17784]

- If you leave the CTX_XDL_DESKTOP _ENVIRONMENT variable unspecified, there is no way to know which desktop is to be used in the subsequent process. As a result, desktop-related environment variables are not configured and apps or plug-ins that depend on these variables might not work as expected. [LNXVDA-16212]
- Due to an issue with GNOME, the Linux VDA does not work as expected after you upgrade **samba-winbind** to version 4.18.6 on RHEL 8.X, Rocky Linux 8.x, RHEL 9.x, and Rocky Linux 9.x. For more information, see https://issues.redhat.com/browse/RHEL-17122.
- Session launch failures occur when the maximum connections set in PostgreSQL are insufficient to handle concurrent sessions. To work around the issue, increase the maximum connections by modifying the **max_connections** setting in the **postgresql.conf** file.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- VDA registration might fail due to the following LDAP exception thrown in **/var/log/xdl/jproxy.log**:

```
javax.naming.NamingException: LDAP response read timed out,
   timeout used: 10000 ms.
```

To work around the issue, do the following:

- Change the LDAP timeout value. For example, change the LDAP timeout value to 60 s using the following command:

```
ctxreg create -k "HKLM\Software\Citrix\GroupPolicy\Defaults"
   -t "REG_DWORD" -v "LDAPTimeout" -d "0x000EA60" --force
```

- Speed up LDAP queries by setting a search base. You can set a search base using the CTX_XDL_SEARCH_BASE variable in ctxsetup.sh or using the following command:

```
ctxreg create -k "HKLM\Software\Citrix\VirtualDesktopAgent" -
   t "REG_SZ" -v "LDAPComputerSearchBase" -d "<specify a
   search base instead of the root of the domain to improve
   search performance>" --force
```

[CVADHELP-20895]

- Microsoft released cumulative updates KB5019966 and KB5019964 for Windows 10 in November 2022. The updates introduce failures in domain joining and registration. To work around the issue, see Knowledge center article CTX474888.

- With the **RC4_HMAC_MD5** encryption type allowed for Kerberos, the Linux VDA might fail to register with the Controller and the following error message appears:

  Error: Failure unspecified at GSS-API level (Mechanism level: Encryption type RC4 with HMAC is not supported/enabled)

  To address this issue, disable **RC4_HMAC_MD5** globally in your Active Directory domain (*or specifically on an OU*) or allow weak encryption types on the Linux VDA. After that, clear the cached Kerberos tickets on the Controller and Citrix Cloud Connector by using the **klist -li 0x3e4 purge** command and restart the Linux VDA.

  To disable **RC4_HMAC_MD5** globally in your Active Directory domain, complete the following steps:

  1. Open the Group Policy Management Console.
  2. Locate the target domain, and then select **Default Domain Policy**.
  3. Right-click **Default Domain Policy** and select **Edit**. The Group Policy Management Editor opens.
  4. Select **Computer Configuration > Policies > Windows Settings > Security Settings > Local Policies > Security Options**.
  5. Double-click **Network security: Configure encryption types allowed for Kerberos**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

6. Clear the **DES_CBC_CRC**, **DES_CBC_MD5**, and **RC4_HMAC_MD5** check boxes and select **AES128_HMAC_SHA1**, **AES256_HMAC_SHA1**, and **Future encryption types**.

To allow weak encryption types on the Linux VDA, complete the following steps:

> **Note:**
>
> Weak encryption types make your deployment vulnerable to attacks.

1. Open the /etc/krb5.conf file on the Linux VDA.
2. Add the following entry under the **[libdefaults]** section:

   ```
   allow_weak_crypto= TRUE
   ```

- The Linux VDA does not support SecureICA for encryption. Enabling SecureICA on the Linux VDA causes session launch failure.
- In a GNOME desktop session, attempts to change the keyboard layout might fail. [CVADHELP-15639]
- Ubuntu graphics: In HDX 3D Pro, a black frame might appear around applications after resizing the Desktop Viewer, or sometimes, the background can appear black.
- Printers created by the Linux VDA printing redirection might not be removed after logging out of a session.
- CDM files are missing when a directory contains numerous files and subdirectories. This issue might occur if the client side has too many files or directories.
- In this release, only UTF-8 encoding is supported for non-English languages.
- Citrix Workspace app for Android CAPS LOCK state might be reversed during session roaming. The CAPS LOCK state can be lost when roaming an existing connection to Citrix Workspace app for Android. As a workaround, use the Shift key on the extended keyboard to switch between upper case and lower case.
- Shortcut keys with ALT do not always work when you connect to the Linux VDA using Citrix Workspace app for Mac. Citrix Workspace app for Mac sends AltGr for both left and right Options/Alt keys by default. You can modify this behavior within the Citrix Workspace app settings but the results vary with different applications.
- Registration fails when the Linux VDA is rejoined to the domain. The rejoining generates a fresh set of Kerberos keys. But, the Broker might use a cached out-of-date VDA service ticket based on the previous set of Kerberos keys. When the VDA tries to connect to the Broker, the Broker might not be able to establish a return security context to the VDA. The usual symptom is that the VDA registration fails.

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

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

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

- Audio plug-n-play is not supported. You can connect an audio capture device to the client machine before starting to record audio in the ICA session. If a capture device is attached after the audio recording application has started, the application might become unresponsive and you must restart it. If a capture device is unplugged while recording, a similar issue might occur.
- Citrix Workspace app for Windows might experience audio distortion during audio recording.

Third party notices

June 20, 2024

Linux Virtual Delivery Agent Version 2407 (PDF Download)

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

Deprecation

July 15, 2024

The announcements in this article give you advanced notice of platforms, Citrix products, and features that are being phased out so that you can make timely business decisions. Citrix monitors customer use and feedback to determine when they are withdrawn. Announcements can change in subsequent releases and might not include every deprecated feature or functionality.
For details about product lifecycle support, see the Product Lifecycle Support Policy article.

Deprecations and removals

The following table shows the platforms, Citrix products, and features that are deprecated or removed.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Deprecated items are not removed immediately. Citrix continues to support them in this release but they will be removed in a future Current Release.
Removed items are either removed, or are no longer supported, in the Linux VDA.

Item	Deprecation announced in	Removed in
Support for RHEL 9.3, RHEL 8.9	2407	2411
Support for Rocky Linux 9.3, Rocky Linux 8.9	2407	2411
Support for Ubuntu 20.04	2407	2502
Support for SUSE 15.5	2407	2411
Support for Debian 11.7/11.3	2407	2407
Support for RHEL 9.0, RHEL 8.6	2402	2407
Support for Rocky Linux 9.0, Rocky Linux 8.6	2402	2407
Support for RHEL 7.9, CentOS 7.9	2402	2407
Support for SUSE 15.4	2308	2311
Support for Rocky Linux 9.1, Rocky Linux 8.7	2305	2308
Support for RHEL 9.1, RHEL 8.7	2305	2308
Support for RHEL 8.4	2303	2308
Support for Ubuntu 18.04	2212	2305
Support for SUSE 15.3	2210	2301
Support for Debian 10.9	2206	2210
Support for SUSE 15.2	2206	2209
Support for RHEL 8.2	2206	2209
Support for RHEL 8.1, RHEL 8.3	2203	2206
Support for RHEL 7.8, CentOS 7.8	2203	2204
Support for CentOS 8.x	2110	2201
Support for SUSE 12.5	2109	2204
Support for Ubuntu 16.04	2109	2203
Support for RHEL 7.7, CentOS 7.7	2006	2009

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Item	Deprecation announced in	Removed in
Support for SUSE 12.3	2006	2006
Support for RHEL 6.10, CentOS 6.10	2003	2003
Support for RHEL 6.9, CentOS 6.9	1909	1909
Support for RHEL 7.5, CentOS 7.5	1903	1903
Support for RHEL 7.4, CentOS 7.4	1811	1811
Support for RHEL 6.8, CentOS 6.8	1811	1811
Support for RHEL 7.3, CentOS 7.3	7.18	7.18
Support for RHEL 6.6, CentOS 6.6	7.16	7.16
SUSE 11.4	7.16	7.16

System requirements

August 5, 2024

The Current Release of the Linux VDA is aligned with Citrix Virtual Apps and Desktops. It is also backward compatible with earlier versions of Citrix Virtual Apps and Desktops that haven't yet reached the end of their lifecycle. For information about the Citrix product lifecycle, and to find out when Citrix stops supporting specific versions of products, see the Citrix Product Lifecycle Matrix.

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

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

For information about using a Current Release (CR) in a Long Term Service (LTSR) environment and other FAQs, see the Knowledge Center article.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Supported Linux distributions, Xorg versions, and desktop environments

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

Linux distribution	Xorg version	Supported desktop
Amazon Linux 2	1.20	MATE
Debian 12.5	1.20	GNOME, GNOME Classic, KDE, MATE, **Xfce***
Debian 11.9	1.20	GNOME, GNOME Classic, KDE, MATE
RHEL 9.4/9.3/9.2	1.20	GNOME, **Xfce**
RHEL 8.10/8.9/8.8	1.20	GNOME, GNOME Classic, MATE, **Xfce**
Rocky Linux 9.4/9.3/9.2	1.20	GNOME, **Xfce***
Rocky Linux 8.10/8.9/8.8	1.20	GNOME, GNOME Classic, KDE, MATE, **Xfce**
SUSE 15.5	1.20	GNOME, GNOME Classic, MATE
Ubuntu 22.04	1.21	GNOME, GNOME Classic, KDE, MATE, **Xfce**
Ubuntu 20.04	1.20	GNOME, GNOME Classic, KDE, MATE, **Xfce**

Note:

- When the support from your OS vendor expires, Citrix might be limited in its ability to remediate problems. For deprecated or removed platforms, see Deprecation.
- At least one desktop must be installed. You can specify through the ctxinstall.sh or ctxsetup.sh script the GNOME, GNOME Classic, or MATE desktop environment to use in sessions.
- According to the Red Hat Enterprise Linux documentation, GNOME is the only desktop environment available in RHEL 9. However, you can also use the **Xfce** desktop environment in RHEL 9 and Rocky Linux 9 if you have the EPEL repository installed.
- Do not use `HWE kernel` or `HWE Xorg` on Ubuntu.
- Your user name format must comply with the `systemd` syntax rules for your current display manager. For more information about the `systemd` user name syntax, see User/Group

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> Name Syntax.

.Net requirements

In addition to the .NET Runtime, you must install .ASP.NET Core Runtime on all supported Linux distributions before you install or upgrade the Linux VDA. Version 6 is required for Amazon Linux 2. Version 8 is required for other distributions.

If your Linux distribution contains the .NET version that you require, install it from the built-in feed. Otherwise, install .NET from the Microsoft package feed. For more information, see https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers.

Supported host platforms and virtualization environments

- Bare metal servers
- Amazon Web Services (AWS)
- XenServer (formerly Citrix Hypervisor)
- Google Cloud Platform (GCP)
- Kernel-based Virtual Machine (KVM)
- Microsoft Azure
- Microsoft Hyper-V
- VMware vSphere Hypervisor
- Nutanix AHV

> **Note:**
>
> In all cases, the supported processor architecture is x86-64.
>
> Starting with the 2203 release, you can host the Linux VDA on Microsoft Azure, AWS, and GCP for Citrix Virtual Apps and Desktops as well as Citrix DaaS (formerly Citrix Virtual Apps and Desktops service). To add these public cloud host connections to your Citrix Virtual Apps and Desktops deployment, you need the Citrix Universal Hybrid Multi-Cloud (HMC) license.

Active Directory integration packages

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

	Winbind	SSSD	Centrify	PBIS	Quest
Amazon Linux 2	Yes	Yes	Yes	Yes	No

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

	Winbind	SSSD	Centrify	PBIS	Quest
Debian 12.5/11.9	Yes	Yes	Yes	Yes	Yes
RHEL 9.4/9.3/9.2, Rocky Linux 9.4/9.3/9.2/8.10/8.9/8.8	Yes	Yes	No	No	Yes (Quest v4.1 and later)
RHEL 8.10/8.9/8.8	Yes	Yes	Yes	Yes	Yes (Quest v4.1 and later)
SUSE 15.5	Yes	Yes	Yes	Yes	Yes
Ubuntu 22.04/20.04	Yes	Yes	Yes	Yes	Yes (Quest v4.1 and later)

Size and scale considerations for Cloud Connectors

When you connect Linux VDAs to the control plane with Citrix Cloud Connectors, consider the following based on Citrix internal testing:

- Each Citrix Cloud Connector (4 vCPU, 10 GB memory) can support 6,000 Linux VDAs.
- Deploy two Cloud Connectors in each resource location for high availability and also deploy a maximum of 6,000 Linux VDAs in each resource location.

Database considerations

- We recommend you use SQLite for VDI mode only and use PostgreSQL for a hosted shared desktops delivery model.
- For easy install and MCS, you can specify SQLite or PostgreSQL to use without having to install them manually. Unless otherwise specified through **/etc/xdl/db.conf**, the Linux VDA uses PostgreSQL by default.
- For manual installations, you must install SQLite, PostgreSQL, or both manually. If you install both SQLite and PostgreSQL, you can specify one of them to use by editing **/etc/xdl/db.conf** after installing the Linux VDA package.

HDX 3D Pro

HDX 3D Pro of Citrix Virtual Apps and Desktops lets you deliver desktops and applications that perform best using a Graphics Processing Unit (GPU) for hardware acceleration.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Hypervisors

For the Linux VDA, HDX 3D Pro is compatible with the following hypervisors:

- XenServer (formerly Citrix Hypervisor)
- VMware vSphere Hypervisor
- Nutanix AHV
- Microsoft Azure
- Amazon Web Services (AWS)
- Google Cloud Platform (GCP)

> **Note:**
>
> The hypervisors are compatible with certain Linux distributions.
> To use HDX 3D Pro for Amazon Linux 2, we recommend you install NVIDIA driver 470.

GPUs

For the Linux VDA, HDX 3D Pro supports the following types of GPUs:

NVIDIA vGPUs To learn which NVIDIA GPU cards your Linux distribution supports, go to the NVIDIA product support matrix and check the **Hypervisor or Bare-Metal OS**, **Software Product Deployment**, **Hardware Supported**, and **Guest OS Support** columns.

Ensure that you install the latest vGPU driver for your GPU card. Currently, the Linux VDA supports up to vGPU 17. For more information, see NVIDIA Virtual GPU Software Supported GPUs.

Non-virtualized GPUs In the Linux VDA documentation, non-virtualized GPUs refer to:

- GPUs used in Remote PC Access scenarios
- GPUs passed through from a hypervisor

NVIDIA GPUs that support the NVIDIA Capture SDK for Linux For NVIDIA GPUs that support the NVIDIA Capture SDK for Linux, enable HDX 3D Pro by setting **CTX_XDL_HDX_3D_PRO** to **Y** when installing the Linux VDA. No additional configuration is required. Hardware acceleration is enabled by default after you enable HDX 3D Pro.

> **Tip:**
>
> To use HDX 3D Pro with an NVIDIA GPU, you must install an NVIDIA graphics driver version that

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> supports NVIDIA Capture SDK version 8. For more information, see the NVIDIA Capture SDK documentation.

Installation overview

June 20, 2024

This section guides you through the following procedures:

- Prepare to install
- Create domain-joined VDAs using easy install
- Create non-domain-joined Linux VDAs using MCS
- Create a non-domain-joined Linux VDA using easy install (preview)
- Create Linux VDAs using MCS
- Create Linux VDAs using Citrix Provisioning
- Create Linux VDAs in Citrix DaaS Standard for Azure
- Install the Linux VDA manually
  - Install the Linux VDA on Amazon Linux 2, CentOS, RHEL, and Rocky Linux manually
  - Install the Linux VDA on SUSE manually
  - Install the Linux VDA on Ubuntu manually
  - Install the Linux VDA on Debian manually

Prepare to install

June 20, 2024

This article presents a method for selecting a suitable installation method based on your specific circumstances and directs you to the relevant installation guide. For information on planning the Citrix Virtual Apps and Desktops and Citrix DaaS deployments, see Prepare to install and Get started: Plan and build a deployment respectively.

Before you start installing the VDA, note the following:

- Verify that your target deployment is officially supported by checking the system requirements for the Linux VDA.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- Decide whether to connect your VDAs to a domain or not.
- Check a suitable link from the following table based on your provisioning method and domain joining decision:

		Easy install script	Machine Creation Services (MCS)	Citrix Provisioning	Pure manual	Note
	Domain-joined	Create domain-joined VDAs using easy install	Create Linux VDAs using Machine Creation Services (MCS)	Create Linux VDAs using Citrix Provisioning	Install the Linux VDA manually	N/A
	Non-domain-joined	Create a non-domain-joined Linux VDA using easy install (preview)	Create non-domain-joined Linux VDAs using MCS	Not supported	Not supported	DaaS only

> **Note:**
>
> - MCS is NOT supported for Remote PC Access use cases.
> - For bulk provisioning, you can use MCS, Citrix Provisioning, or third-party automation on your own efforts. For third-party automation, you have the option to incorporate the easy install script (recommended) or automate the pure manual installation steps within your script.
> - For a quick Proof of Concept (POC) on a single VM, we recommend you use easy install.
> - You can also run the easy install script (ctxinstall.sh -s) to quickly update your environment variables such as CTX_XDL_DDC_LIST and CTX_XDL_LDAP_LIST.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Create domain-joined VDAs using easy install

August 2, 2024

> **Important:**
>
> - For fresh installations, we recommend you refer to this article for a quick installation. This article steps through how to install and configure the Linux VDA by using easy install. Easy install saves time and labor and is less error-prone than manual installation. It helps you set up a running environment of the Linux VDA by installing the necessary packages and customizing the configuration files automatically.
> - To create non-domain joined VDAs, you can use both Machine Creation Services (MCS) and easy install. For more information, see Create non-domain-joined Linux VDAs using MCS and Create a non-domain-joined Linux VDA using easy install (preview).
> - To learn about the features available for non-domain-joined VDAs, go to Non-domain-joined VDAs.

Step 1: Prepare configuration information and the Linux machine

Collect the following configuration information needed for easy install:

- Host name –Host name of the machine on which the Linux VDA is to be installed.
- IP address of Domain Name Server.
- IP address or string name of NTP Server.
- Domain name –The NetBIOS name of the domain.
- Realm name –The Kerberos realm name.
- Fully Qualified Domain Name (FQDN) of the domain.
- Active Directory (AD) integration method - Currently, easy install supports SSSD, Winbind, Centrify, PBIS, and Quest. Easy install supports Quest only for RHEL and Rocky Linux.
- User name –The name of the user who joins the machine to the domain.
- Password - The password of the user who joins the machine to the domain.
- OU –The organization unit. Optional.

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> - For security, easy install does not save the domain joining password. Every time you run the easy install script (ctxinstall.sh) in interactive mode, you must enter the domain joining password manually. In silent mode, you must set the domain joining password in **/Citrix/VDA/sbin/ctxinstall.conf** or export the password. We recommend you not use the administrator account for domain joining. Instead, delegate domain joining permissions to an Active Directory user other than the administrator account. To do so, delegate control on the domain controller using the **Delegation of Control Wizard**.

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 based on the hypervisor platform in use. No changes are required if you are running the Linux machine on bare metal hardware.

Fix time synchronization on XenServer (formerly Citrix Hypervisor)

When the XenServer Time Sync feature is enabled, within each paravirtualized Linux VM you experience issues with NTP and XenServer. Both try to manage the system clock. To avoid the clock becoming out of sync with other servers, make sure 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.

If you are running a paravirtualized Linux kernel with XenServer VM 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 restart, edit the **/etc/sysctl.conf** file and add the line:

```
xen.independent_wallclock = 1
```

To verify these changes, restart the system:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
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 (formerly Citrix Hypervisor), 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 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: Install .NET

In addition to the .NET Runtime, you must install .ASP.NET Core Runtime on all supported Linux distributions before you install or upgrade the Linux VDA. Version 6 is required for Amazon Linux 2. Version

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

8 is required for other distributions.

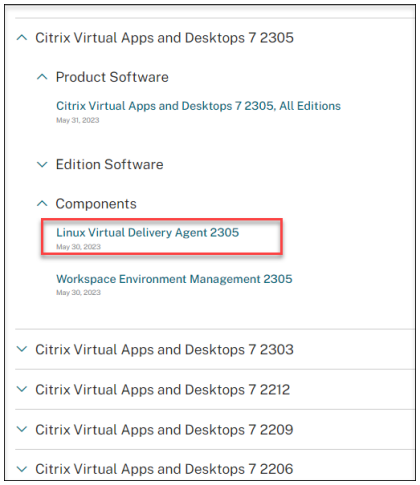
If your Linux distribution contains the .NET version that you require, install it from the built-in feed. Otherwise, install .NET from the Microsoft package feed. For more information, see https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers.

After installing .NET, run the **which dotnet** command to find your runtime path.

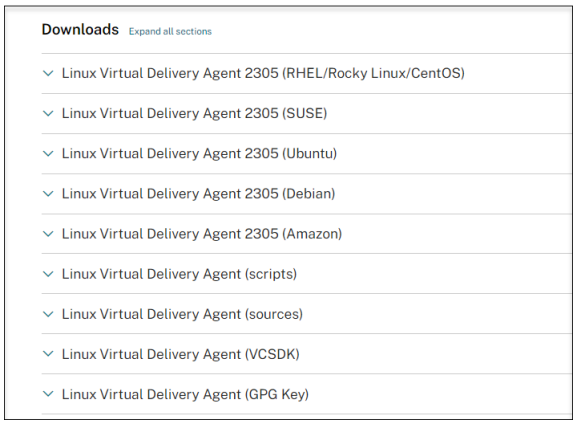
Based on the command output, set the .NET runtime binary path. For example, if the command output is /aa/bb/dotnet, use /aa/bb as the .NET binary path.

Step 4: Download the Linux VDA package

1. Go to the Citrix Virtual Apps and Desktops download page.
2. Expand the appropriate version of Citrix Virtual Apps and Desktops.
3. Expand **Components** to find the Linux VDA. For example:



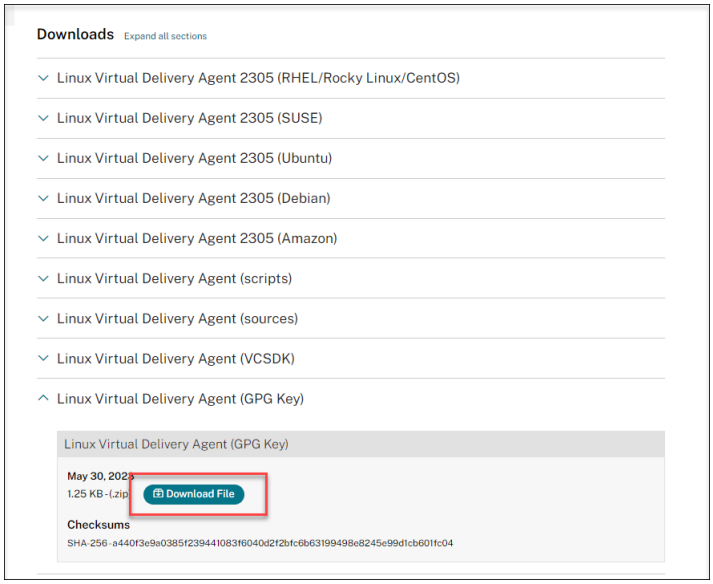
4. Click the Linux VDA link to access the Linux VDA downloads.



5. Download the Linux VDA package that matches your Linux distribution.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

6. Download the GPG public key that you can use to verify the integrity of the Linux VDA package. For example:



   To verify the integrity of the Linux VDA package by using the public key:

   - For an RPM package, run the following commands to import the public key into the RPM database and to check the package integrity:

     ```
     rpmkeys --import <path to the public key>
     rpm --checksig --verbose <path to the Linux VDA package>
     ```

   - For a DEB package, run the following commands to import the public key into the DEB database and to check the package integrity:

     ```
     sudo apt-get install dpkg-sig
     gpg --import <path to the public key>
     dpkg-sig --verify <path to the Linux VDA package>
     ```

Step 5: Install the Linux VDA package

To set up the environment for the Linux VDA, run the following commands.

For Amazon Linux 2, CentOS, RHEL, and Rocky Linux distributions:

> **Note:**
>
> - For RHEL and CentOS, install the EPEL repository before you can install the Linux VDA successfully. For information on how to install EPEL, see the instructions at https://docs.fedoraproject.org/en-US/epel/.
> - Before installing the Linux VDA on RHEL 9.4/9.3/9.2 and Rocky Linux 9.4/9.3/9.2, update the

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **libsepol** package to version 3.4 or later.

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

> **Note:**
>
> After you install the Linux VDA on RHEL 8.x/9.x and Rocky Linux 8.x/9.x hosted on GCP, the Ethernet connection might be lost and the Linux VDA might be unreachable after a VM restart. To work around the issue, set a root password when logging on to the VM for the first time and make sure that you can log on to the VM as root. Then, run the following commands in the console after restarting the VM:
>
> ```
> nmcli dev connect eth0
> systemctl restart NetworkManager
> ```

For Ubuntu/Debian distributions:

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

> **Note:**
>
> - To install the necessary dependencies for a Debian 11 distribution, add the `deb http://deb.debian.org/debian/ bullseye main` line to the /etc/apt/sources.list file.
> - For Ubuntu 20.04 on GCP, disable RDNS. To do so, add the **rdns = false** line under **[libdefaults]** in /etc/krb5.conf.

For SUSE distributions:

1. For SUSE 15.5 on AWS, Azure, and GCP, ensure that:
   - You are using **libstdc++6** version 12 or later.
   - The **Default_WM** parameter in **/etc/sysconfig/windowmanager** is set to "**gnome**".
2. Run the following command to install the Linux VDA:

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

Step 6: Install NVIDIA GRID drivers

Enabling HDX 3D Pro requires you to install the NVIDIA GRID drivers on your hypervisor and on the VDA machines.

To install and configure the NVIDIA GRID Virtual GPU Manager (the host driver) on the specific hypervisors, see the following guides:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- XenServer
- VMware ESX
- Nutanix AHV

To install and configure the NVIDIA GRID guest VM drivers, perform the following general steps:

1. Ensure that the guest VM is shut down.
2. In the hypervisor control panel, allocate a GPU to the VM.
3. Start the VM.
4. Install the guest VM driver on the VM.

Step 7: Specify a database to use

You can switch between SQLite and PostgreSQL after installing the Linux VDA package. To do so, complete the following steps:

> **Note:**
>
> - We recommend you use SQLite for VDI mode only and use PostgreSQL for a hosted shared desktops delivery model.
> - For easy install and MCS, you can specify SQLite or PostgreSQL to use without having to install them manually. Unless otherwise specified through **/etc/xdl/db.conf**, the Linux VDA uses PostgreSQL by default.
> - You can also use **/etc/xdl/db.conf** to configure the port number for PostgreSQL.

1. Edit **/etc/xdl/db.conf** to specify a database to use.
2. Run **sudo /opt/Citrix/VDA/sbin/ctxinstall.sh** or **/opt/Citrix/VDA/bin/easyinstall**.

Step 8: Run easy install to configure the environment and VDA to complete the installation

After installing the Linux VDA package, configure the running environment by using the ctxinstall.sh script or GUI.

> **Note:**
>
> Before setting up the runtime environment, ensure that the **en_US.UTF-8** locale is installed in your OS. If the locale is not available in your OS, run the **sudo locale-gen en_US.UTF-8** command. For Debian, edit the **/etc/locale.gen** file by uncommenting the **# en_US.UTF-8 UTF-8** line and then run the **sudo locale-gen** command.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

ctxinstall.sh

ctxinstall.sh is the easy install script for doing some pre-configuration and setting up the VDA running environment variables.

- Only root can run this script.
- Easy install uses /opt/Citrix/VDA/sbin/ctxinstall.conf as its configuration file to set, save, and synchronize the values of all environment variables used. We recommend you read the template (ctxinstall.conf.tmpl) carefully and then customize your own ctxinstall.conf. When you first create the configuration file, use either of the following ways:
  - By copying the /opt/Citrix/VDA/sbin/ctxinstall.conf.tmpl template file and saving it as /opt/Citrix/VDA/sbin/ctxinstall.conf.
  - By running ctxinstall.sh. Each time you run ctxinstall.sh, your input is saved in /opt/Citrix/VDA/sbin/ctxinstall.conf.
- Easy install supports modular running. Modules include pre-check, installation, domain-configuration, setup, and verification.
- Debugging details for this script can be found in /var/log/xdl/ctxinstall.log.

For more information, use the help command **ctxinstall.sh -h**.

> **Note:**
>
> - Following the principle of least privilege, ensure that only the root user can read **/opt/Citrix/VDA/sbin/ctxinstall.conf** because the domain joining password might be set in the file.
> - Uninstalling the Linux VDA removes files under **/opt/Citrix/VDA**. We recommend you back up **/opt/Citrix/VDA/sbin/ctxinstall.conf** before uninstalling the VDA.

You can run ctxinstall.sh in interactive mode or silent mode. Before you run the script, set the following environment variables:

- **CTX_XDL_NON_DOMAIN_JOINED='y|n'**–Whether to join the machine to a domain. The default value is 'n'. For domain-joined scenarios, set it to 'n'.
- **CTX_XDL_AD_INTEGRATION='sssd|winbind|centrify|pbis|quest'** –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.
- **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.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- **CTX_XDL_VDI_MODE='y|n'**–Whether to configure the machine as a dedicated desktop delivery model (VDI) or hosted shared desktop delivery model. For HDX 3D Pro environments, set the value to **'y'**.
- **CTX_XDL_HDX_3D_PRO='y|n'**–The Linux VDA supports HDX 3D Pro, a set of GPU acceleration technologies designed to optimize the virtualization of rich graphics applications. If HDX 3D Pro is selected, the VDA is configured for VDI desktops (single-session) mode - (that is, CTX_XDL_VDI_MODE=**'y'**).
- **CTX_XDL_START_SERVICE='y|n'**–Determines whether the Linux VDA services are started when the configuration is complete.
- **CTX_XDL_REGISTER_SERVICE='y|n'**–The Linux Virtual Desktop services are started after machine startup.
- **CTX_XDL_ADD_FIREWALL_RULES='y|n'**–The Linux VDA services require incoming network connections to be allowed through the system firewall. You can open the required ports (by default ports 80 and 1494) automatically in the system firewall for the Linux Virtual Desktop.
- **CTX_XDL_DESKTOP_ENVIRONMENT=gnome/gnome-classic/kde/mate/xfce/'<none>'** – Specifies the GNOME, GNOME Classic, KDE, MATE, or **Xfce** desktop environment to use in sessions. If you set it to **'<none>'**, the default desktop configured on the VDA is used.
- **CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime** –The path to install .NET for supporting the new broker agent service (**ctxvda**). The default path is **'/usr/bin'**.
- **CTX_XDL_VDA_PORT=port-number** –The Linux VDA communicates with Delivery Controllers through a TCP/IP port.
- **CTX_XDL_SITE_NAME=<dns-name>** –The Linux VDA discovers LDAP servers through DNS. To limit the DNS search results to a local Site, specify a DNS Site name. If unnecessary, set to **'<none>'**.
- **CTX_XDL_LDAP_LIST='<list-ldap-servers>'** –The Linux VDA queries DNS to discover LDAP servers. If DNS cannot provide LDAP service records, you can provide a space-separated list of LDAP FQDNs with LDAP ports. For example, ad1.mycompany.com:389 ad2.mycompany.com:3268 ad3.mycompany.com:3268. To enable faster LDAP queries within an Active Directory forest, enable Global Catalog on a domain controller and specify the relevant LDAP port number as 3268. This variable is set to **'<none>'** by default.
- **CTX_XDL_SEARCH_BASE=search-base-set** –The Linux VDA queries LDAP through a search base set to the root of the Active Directory Domain (for example, DC=mycompany,DC=com). To improve search performance, you can specify a search base (for example, OU=VDI,DC=mycompany,DC=com). If unnecessary, set to **'<none>'**.
- **CTX_XDL_SUPPORT_DDC_AS_CNAME='y|n'**–The Linux VDA supports specifying a Delivery Controller name using a DNS CNAME record.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- **CTX_EASYINSTALL_DNS='<ip-address-of-dns>'**–The IP address of DNS.
- **CTX_EASYINSTALL_HOSTNAME=host-name** –The host name of the Linux VDA server.
- **CTX_EASYINSTALL_NTPS=address-of-ntps** –The IP address or string name of the NTP server.
- **CTX_EASYINSTALL_REALM=realm-name** –The Kerberos realm name.
- **CTX_EASYINSTALL_FQDN=ad-fqdn-name**
- **CTX_EASYINSTALL_USERNAME=domain-user-name** –The name of the user who joins the machine to the domain.
- **CTX_EASYINSTALL_PASSWORD=password** –The password of the user who joins the machine to the domain.

> **Note:**
>
> We recommend you not use the administrator account for domain joining. Instead, delegate domain joining permissions to an Active Directory user other than the administrator account. To do so, delegate control on the domain controller using the **Delegation of Control Wizard**.

The following four variables are optional. Even if they are not set, ctxinstall.sh won't abort in silent mode and you won't be prompted for user input in interactive mode. You can set them only by exporting their values or editing **/Citrix/VDA/sbin/ctxinstall.conf**.

- **CTX_EASYINSTALL_NETBIOS_DOMAIN=netbios-domain-name** –The NetBIOS domain name is typically the first component of the DNS domain name separated by a dot (.). Otherwise, customize a different NetBIOS domain name. This variable is optional.
- **CTX_EASYINSTALL_OU=ou-value** –OU values vary with different **AD** integration methods. For an example of OU values, see the table in the Considerations section of this article. This variable is optional.
- **CTX_EASYINSTALL_CENTRIFY_LOCAL_PATH=centrify-local-path** –Easy install helps download the Centrify package from the Internet. However, if Centrify is already installed, you can fetch the Centrify package from a local directory defined by this variable. This variable is optional.
- **CTX_EASYINSTALL_PBIS_LOCAL_PATH= pbis-local-path** –Easy install helps download the PBIS package from the Internet. However, if PBIS is already installed, you can fetch the PBIS package from a local directory defined by this variable. This variable is optional.

Considerations

- The NetBIOS domain name is typically the first component of the DNS domain name separated by a dot (.). To customize a different NetBIOS domain name in your environment, set the

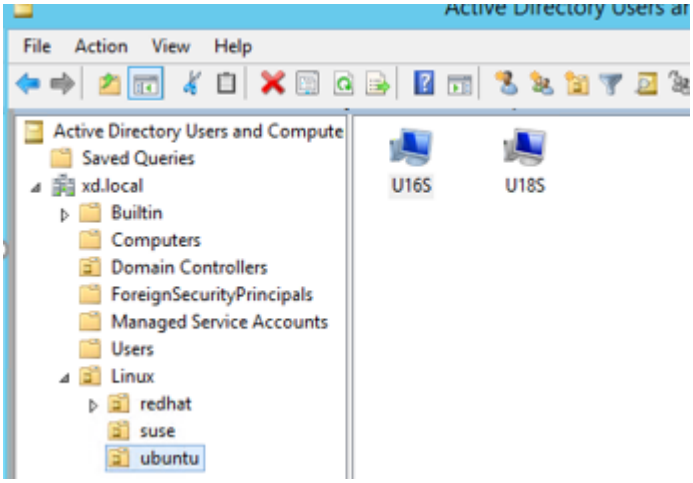
© 1999–2024 Cloud Software Group, Inc. All rights reserved.

environment variable **CTX_EASYINSTALL_NETBIOS_DOMAIN** in **/opt/Citrix/VDA/sbin/ctxinstall.conf**.

- To join your VDA to a specific OU, do the following:

  1. Ensure that the specific OU exists on the domain controller.

     For an example OU, see the following screen capture:

  2. Set the environment variable **CTX_EASYINSTALL_OU** in **/opt/Citrix/VDA/sbin/ctxinstall.conf**.

     OU values vary with different AD methods. The following table reflects the example OU names in the preceding screen capture. You can use any other OU names in your organization.

OS	Winbind	SSSD	Centrify	PBIS
Amazon Linux 2	`"Linux/amazon"`	`"Linux/amazon"`	`"XD.LOCAL/Linux/amazon"`	`"Linux/amazon"`
Debian	`"Linux/debian"`	`"Linux/debian"`	`"XD.LOCAL/Linux/debian"`	`"Linux/debian"`
RHEL 9.4/9.3/9.2, Rocky Linux 9.4/9.3/9.2	`"OU=redhat,OU=Linux"`	`"OU=redhat,OU=Linux"`	N/A	N/A
RHEL 8.x	`"OU=redhat,OU=Linux"`	`"OU=redhat,OU=Linux"`	`"XD.LOCAL/Linux/redhat"`	`"Linux/redhat"`

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

OS	Winbind	SSSD	Centrify	PBIS
Rocky Linux 8.x	`"OU=redhat, OU=Linux"`	`"OU=redhat, OU=Linux"`	N/A	N/A
SUSE	`"Linux/suse"`	`"Linux/suse"`	`"XD.LOCAL/Linux/suse"`	`"Linux/suse"`
Ubuntu	`"Linux/ubuntu"`	`"Linux/ubuntu"`	`"XD.LOCAL/Linux/ubuntu"`	`"Linux/ubuntu"`

- Centrify does not support pure **IPv6** DNS configuration. At least one DNS server using **IPv4** is required in /etc/resolv.conf for **adclient** to find AD services properly.

  Log:

  ```
  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 issue is unique to Centrify and its configuration. To resolve this issue, do the following:

  a. Open **Administrative Tools** on the domain controller.
  b. Select **Active Directory Sites and Services**.
  c. Add a proper subnet address for **Subnets**.

- Easy install supports pure **IPv6** starting with the Linux VDA 7.16. The following preconditions and limitations apply:

  - Your Linux repository must be configured to ensure that your machine can download the required packages over pure **IPv6** networks.
  - Centrify is not supported on pure **IPv6** networks.

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

- You can specify a desktop environment to use in sessions by using the CTX_XDL_DESKTOP_ENVIRONMENT variable as described earlier. You can also switch between desktop environments by running commands or using the system tray. For more information, see Desktop switching commands and System tray.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- If you choose Centrify as the method to join a domain, the ctxinstall.sh script requires the Centrify package. Ways for ctxinstall.sh to get the Centrify package:

  - Easy install helps download the Centrify package from the Internet automatically. The following are the URLs for each distribution:

    Amazon Linux 2/RHEL: wget https://downloads.centrify.com/products/server-suite/2022/component-update-1/delinea-server-suite-2022-rhel6-x86_64.tgz

    CentOS: wget https://downloads.centrify.com/products/server-suite/2022/component-update-1/delinea-server-suite-2022-rhel6-x86_64.tgz

    SUSE: wget https://downloads.centrify.com/products/server-suite/2022/component-update-1/delinea-server-suite-2022-suse12-x86_64.tgz

    Ubuntu/Debian: wget https://downloads.centrify.com/products/server-suite/2022/component-update-1/delinea-server-suite-2022-deb9-x86_64.tgz

  - Fetch the Centrify package from a local directory if Centrify is already installed. To designate the directory of the Centrify package, set **CTX_EASYINSTALL_CENTRIFY_LOCAL_PATH=/home/m** in **/opt/Citrix/VDA/sbin/ctxinstall.conf**. For example:

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- If you choose PBIS as the method to join a domain, the ctxinstall.sh script requires the PBIS package. Ways for ctxinstall.sh to get the PBIS package:

  - Easy install helps download the PBIS package from the Internet automatically. For example, the following are the URLs for each distribution:

    Amazon Linux 2, RHEL 8, SUSE 15.5: wget https://github.com/BeyondTrust/pbis-open/releases/download/9.1.0/pbis-open-9.1.0.551.linux.x86_64.rpm.sh

    Debian, Ubuntu: wget https://github.com/BeyondTrust/pbis-open/releases/download/9.1.0/pbis-open-9.1.0.551.linux.x86_64.deb.sh

  - Fetch a specific version of the PBIS package from the Internet. To do so, change the "pbisDownloadRelease"and "pbisDownloadExpectedSHA256"lines in the /opt/Citrix/VDA/sbin/ctxinstall.sh file.

    For an example, see the following screen capture:

    ```
    local pbisDownloadURL="https://github.com/BeyondTrust/pbis-open/releases/download"
    local pbisDownloadExpectedSHA256="f37555abf22f453c3865f06eba3c5f913605c300917be29663b23941087137f6"
    local pbisDownloadFMT="rpm"
    local pbisDownloadRelease="9.1.0"
    local pbisDonwloadBuild="551"
    ```

  - Fetch the PBIS package from a local directory if PBIS is already installed. To designate the directory of the PBIS package, set the environment variable **CTX_EASYINSTALL_PBIS_LOCAL_PATH** in **/opt/Citrix/VDA/sbin/ctxinstall.conf**.

Interactive mode To run the **ctxinstall.sh** script in interactive mode, use the **sudo /opt/Citrix/VDA/sbin/ctxinstall.sh** command without the **-S** option. Type the relevant variable value at each prompt in the command-line interface. If a variable is already set, ctxinstall.sh asks for confirmation in case you want to change it.

Silent mode In silent mode, you must set the preceding variables by using **/opt/Citrix/VDA/sbin/ctxinstall.conf** or the export command. After that, run **ctxinstall.sh -S** (note that the letter **S** here is in **uppercase**). If not all required variables are set or some value is invalid, **ctxinstall.sh** aborts execution, unless there are default values.

The exported value for each variable overwrites the value in **/Citrix/VDA/sbin/ctxinstall.conf**, unless it is not set. All updated values are saved in **/Citrix/VDA/sbin/ctxinstall.conf**, except the domain joining password. So in silent mode, you must set the domain joining password in **/Citrix/VDA/sbin/ctxinstall.conf** or export the password.

```
export CTX_XDL_NON_DOMAIN_JOINED='n'
export CTX_XDL_AD_INTEGRATION=sssd|winbind|centrify|pbis|quest
export CTX_XDL_DDC_LIST='<list-ddc-fqdns>'
export CTX_XDL_VDI_MODE='y|n'
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
export CTX_XDL_HDX_3D_PRO='y|n'
export CTX_XDL_START_SERVICE='y|n'
export CTX_XDL_REGISTER_SERVICE='y|n'
export CTX_XDL_ADD_FIREWALL_RULES='y|n'
export CTX_XDL_DESKTOP_ENVIRONMENT=gnome|gnome-classic|kde|mate|xfce|'<none>'
export CTX_XDL_DOTNET_RUNTIME_PATH='<path-to-install-dotnet-runtime>'
export CTX_XDL_VDA_PORT='<port-number>'
export CTX_XDL_SITE_NAME='<dns-site-name>'|'<none>'
export CTX_XDL_LDAP_LIST='<list-ldap-servers>'|'<none>'
export CTX_XDL_SEARCH_BASE='<search-base-set>'|'<none>'
export CTX_XDL_SUPPORT_DDC_AS_CNAME='y|n'
export CTX_EASYINSTALL_DNS='<ip-address-of-dns>'
export CTX_EASYINSTALL_HOSTNAME='<host-name>'
export CTX_EASYINSTALL_NTPS='<address-of-ntps>'
export CTX_EASYINSTALL_REALM='<realm-name>'
export CTX_EASYINSTALL_FQDN='<ad-fqdn-name>'
export CTX_EASYINSTALL_USERNAME='<domain-user-name>'
export CTX_EASYINSTALL_PASSWORD='<password>'
export CTX_EASYINSTALL_NETBIOS_DOMAIN='<netbios-domain>'
export CTX_EASYINSTALL_OU='<organization-unit>'
sudo -E /opt/Citrix/VDA/sbin/ctxinstall.sh -S
```

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

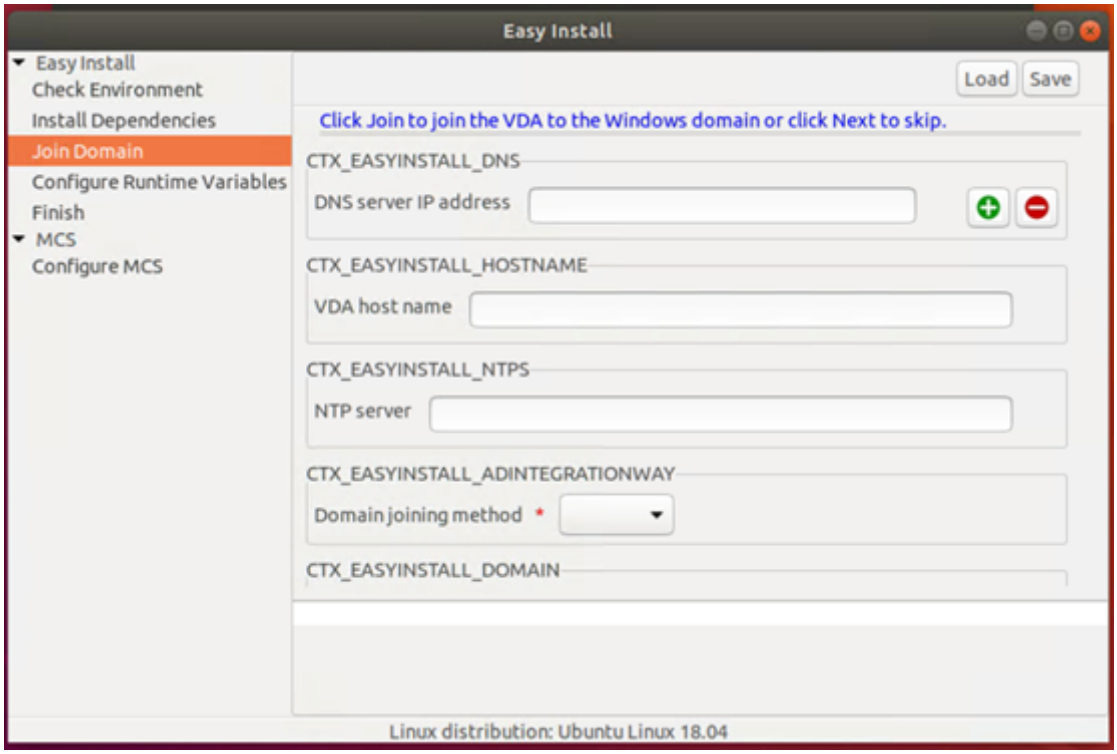
Alternatively, you can specify all variables by using a single command.

To set up the VDA running environment variables (those beginning with '**CTX_XDL_**'), you can run **ctxinstall.sh -s** (note that the letter **s** here is in **lowercase**).

GUI

You can use easy install through a GUI. Run the **/opt/Citrix/VDA/bin/easyinstall** command in the desktop environment of your VDA and then follow the instructions on the easy install GUI.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



The easy install GUI guides you through the following operations:

- Check the system environment
- Install dependencies
- Join the VDA to a specified domain
- Configure the runtime environment

> **Tip:**
>
> Click **Save** to save variable settings to a local file under the path you specify. Click **Load** to load variable settings from a file that you specify. For information on configuring MCS variables, see Step 3: Prepare a master image.

Step 9: Run XDPing

Run `sudo /opt/Citrix/VDA/bin/xdping` to check for common configuration issues with a Linux VDA environment. For more information, see **XDPing**.

Step 10: Run the Linux VDA

Start the Linux VDA:

To start the Linux VDA services:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
sudo systemctl start ctxhdx.service

sudo systemctl start ctxvda.service
```

Stop the Linux VDA:

To stop the Linux VDA services:

```
sudo systemctl stop ctxvda.service

sudo systemctl stop ctxhdx.service
```

> **Note:**
>
> Before you stop the **ctxvda** and **ctxhdx** services, run the **systemctl stop ctxmonitord** command to stop the monitor service daemon. Otherwise, the monitor service daemon restarts the services you stopped.

Restart the Linux VDA:

To restart the Linux VDA services:

```
sudo systemctl stop ctxvda.service

sudo systemctl restart ctxhdx.service

sudo systemctl start ctxvda.service
```

Check the status of the Linux VDA:

To check the running status of the Linux VDA services:

```
sudo systemctl status ctxvda.service

sudo systemctl status ctxhdx.service
```

Step 11: Create machine catalogs

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

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

- For the operating system, select:
  - The **Multi-session OS** option for a hosted shared desktops delivery model.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

  - The **Single-session OS** option for a VDI dedicated desktop delivery model.

- Do not mix Linux and Windows VDA machines in the same machine catalog.

> **Note:**
>
> Early versions of Citrix Studio did not support the notion of a "Linux OS."However, selecting the **Windows Server OS** or **Server OS** option implies an equivalent hosted shared desktops delivery model. Selecting the **Windows Desktop OS** or **Desktop OS** option implies a single user per machine delivery model.
>
> **Tip:**
>
> If you remove and rejoin a machine to the Active Directory domain, you must remove and add the machine to the machine catalog again.

Step 12: Create delivery groups

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:

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

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

For information about how to create machine catalogs and delivery groups, see Citrix Virtual Apps and Desktops 7 2407.

Step 13: Upgrade the Linux VDA (optional)

The Linux VDA supports upgrades from the most recent version. For example, you can upgrade the Linux VDA from 2308 to 2311 and from 1912 LTSR to 2203 LTSR.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> - Upgrading an existing installation overwrites the configuration files under /etc/xdl. Before you conduct an upgrade, make sure to back up the files.
> - Starting with the 2407 release, the Linux VDA delegates package managers **rpm** or **dpkg** to handle configuration files during upgrades. The following describes how rpm and dpkg interact with changes to configuration files:
>   - **rpm**: by default keeps the local version and saves the new version from the package with a .rpmnew extension.
>   - **dpkg**: interactively prompts you with a choice on how to proceed. To silently upgrade the Linux VDA while retaining your local configuration file and saving the new package version as **.dpkg-new** or **.dpkg-dist**, use the following command:
>
>     ```
>     dpkg --force-confold -i package.deb  # Always keep your version, then save new package's version as *.dpkg-new or *.dpkg-dist
>     ```

For Amazon Linux 2, RHEL, and Rocky Linux distributions:

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

> **Note:**
>
> Before upgrading the Linux VDA on RHEL 9.4/9.3/9.2 and Rocky Linux 9.4/9.3/9.2, update the **libsepol** package to version 3.4 or later.

For SUSE distributions:

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

For Ubuntu/Debian distributions:

```
sudo dpkg -i <PATH>/<Linux VDA deb>

sudo apt-get install -f
```

Troubleshooting

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Joining a domain by using SSSD fails

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

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

/var/log/xdl/vda.log:

```
2016-11-04 02:11:52.317 [INFO ] - The Citrix Desktop Service
    successfully obtained the following list of 1 delivery controller(s)
     with which to register: 'CTXDDC.citrixlab.local (10.158.139.214)'.
2016-11-04 02:11:52.362 [ERROR] - RegistrationManager.
    AttemptRegistrationWithSingleDdc: Failed to register with http://
    CTXDDC.citrixlab.local:80/Citrix/CdsController/IRegistrar. Error:
    General security error (An error occurred in trying to obtain a TGT:
     Client not found in Kerberos database (6))
2016-11-04 02:11:52.362 [ERROR] - The Citrix Desktop Service cannot
    connect to the delivery controller 'http://CTXDDC.citrixlab.local
    :80/Citrix/CdsController/IRegistrar' (IP Address '10.158.139.214')
Check the following:- The system clock is in sync between this machine
    and the delivery controller.
-  The Active Directory provider (e.g. winbind daemon) service is
    running and correctly configured.
-  Kerberos is correctly configured on this machine.
If the problem persists, please refer to Citrix Knowledge Base article
    CTX117248 for further information.
Error Details:
Exception 'General security error (An error occurred in trying to
    obtain a TGT: Client not found in Kerberos database (6))' of type '
    class javax.xml.ws.soap.SOAPFaultException'.
2016-11-04 02:11:52.362 [INFO ] - RegistrationManager.
    AttemptRegistrationWithSingleDdc: The current time for this VDA is
    Fri Nov 04 02:11:52 EDT 2016.
Ensure that the system clock is in sync between this machine and the
    delivery controller.
Verify the NTP daemon is running on this machine and is correctly
    configured.
2016-11-04 02:11:52.364 [ERROR] - Could not register with any
    controllers. Waiting to try again in 120000 ms. Multi-forest - false
2016-11-04 02:11:52.365 [INFO ] - The Citrix Desktop Service failed to
    register with any controllers in the last 470 minutes.
```

/var/log/messages:

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
ldap_child[14867]]]: Client 'RH-WS-68$@CITRIXLAB.LOCAL'not found
in Kerberos database
```

To resolve this issue:

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

Ubuntu desktop sessions show a gray screen

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

To resolve this issue:

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

Attempts to launch the Ubuntu desktop sessions fail due to a missing home directory

/var/log/xdl/hdx.log:

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

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

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

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

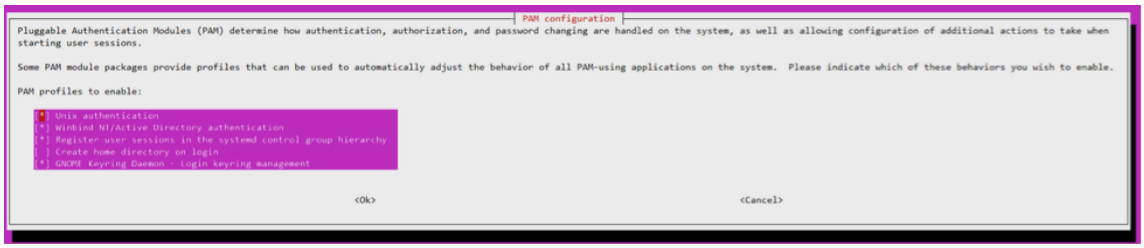
> **Tip:**
>
> The root cause of this issue is that the home directory is not created for the domain administrator.

To resolve this issue:

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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



Session does not launch or ends quickly with dbus error

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

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

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

Oct 27 04:17:18 CentOS7 gnome-session[19146]: ERROR: Failed to connect
    to system bus: Exhausted all available authentication mechanisms (
    tried: EXTERNAL, DBUS_COOKIE_SHA1, ANONYMOUS) (available: EXTERNAL,
    DBUS_COOKIE_SHA1, ANONYMOUS)#012aborting...

Oct 27 04:17:18 CentOS7 gnome-session: gnome-session[19146]: ERROR:
    Failed to connect to system bus: Exhausted all available
    authentication mechanisms (tried: EXTERNAL, DBUS_COOKIE_SHA1,
    ANONYMOUS) (available: EXTERNAL, DBUS_COOKIE_SHA1, ANONYMOUS)

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

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

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

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

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

Nov  3 11:03:52 user01-HVM-domU pulseaudio[25326]: [pulseaudio] bluez5-
    util.c: Failed to get D-Bus connection: Did not receive a reply.
    Possible causes include: the remote application did not send a reply
    , the message bus security policy blocked the reply, the reply
    timeout expired, or the network connection was broken.

Nov  3 11:03:52 user01-HVM-domU pulseaudio[25326]: [pulseaudio] hashmap
    .c: Assertion 'h' failed at pulsecore/hashmap.c:116, function
    pa_hashmap_free(). Aborting.

```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
7  Nov  3 11:03:52 user01-HVM-domU pulseaudio[25352]: [pulseaudio] core-
       util.c: Failed to connect to system bus: Did not receive a reply.
       Possible causes include: the remote application did not send a reply
       , the message bus security policy blocked the reply, the reply
       timeout expired, or the network connection was broken.
8
9  Nov  3 11:03:52 user01-HVM-domU pulseaudio[25352]: message repeated 10
       times: [ [pulseaudio] core-util.c: Failed to connect to system bus:
       Did not receive a reply. Possible causes include: the remote
       application did not send a reply, the message bus security policy
       blocked the reply, the reply timeout expired, or the network
       connection was broken.]
10
11 Nov  3 11:03:52 user01-HVM-domU pulseaudio[25352]: [pulseaudio] pid.c:
       Daemon already running.Nov  3 11:03:58 user01-HVM-domU citrix-ctxgfx
       [24693]: Exiting normally
```

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

SELinux prevents SSHD from accessing the home directory

The user can launch a session but cannot log on.

/var/log/xdl/ctxinstall.log:

```
1  Jan 25 23:30:31 yz-rhel72-1 setroubleshoot[3945]: SELinux is preventing
       /usr/sbin/sshd from setattr access on the directory /root. For
       complete SELinux messages. run sealert -l 32f52c1f-8ff9-4566-a698
       -963a79f16b81
2
3  Jan 25 23:30:31 yz-rhel72-1 python[3945]: SELinux is preventing /usr/
       sbin/sshd from setattr access on the directory /root.
4
5  *****  Plugin catchall_boolean (89.3 confidence) suggests
       ******************
6
7  If you want to allow polyinstantiation to enabled
8
9     Then you must tell SELinux about this by enabling the '
       polyinstantiation_enabled' boolean.
10
11 You can read 'None' man page for more details.
12
13     Do
14
15        setsebool -P polyinstantiation_enabled 1
16
17 *****  Plugin catchall (11.6 confidence) suggests
       **************************
18
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
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. Restart the VDA.

Create non-domain-joined Linux VDAs using MCS

July 12, 2024

This article walks you through using Machine Creation Services (MCS) to create non-domain-joined Linux VDAs in Citrix DaaS.

> **Important:**
>
> - Creating non-domain-joined VDAs using MCS is supported for Citrix DaaS only. For Citrix DaaS customers:
>   - Your control plane must be deployed over Citrix DaaS.
>   - You can deploy non-domain-joined VDAs in a public cloud or on-premises data center. Non-domain-joined VDAs are managed by the control plane in Citrix DaaS.
>   - To create non-domain-joined VDAs, customers utilizing the Citrix Gateway service must ensure that Rendezvous V2 is enabled. Cloud Connectors are required only if you plan to provision machines on on-premises hypervisors or if you want to use Active Directory as the identity provider in Workspace.
> - To create non-domain joined VDAs, you can use both MCS and easy install. For more information, see Create non-domain-joined Linux VDAs using MCS and Create a non-domain-joined Linux VDA using easy install (preview).

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

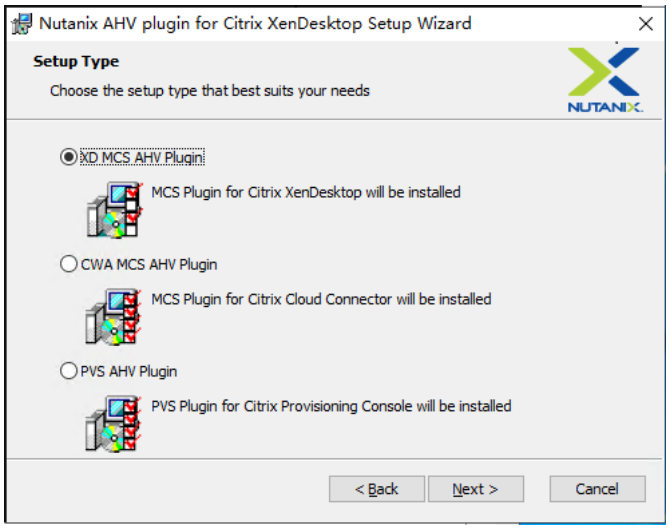
- MCS doesn't support bare metal servers.
- The following features are available for non-domain-joined Linux VDAs:
  - Create local users with specified attributes on non-domain-joined VDAs
  - Non-SSO authentication
  - Authentication with Azure Active Directory
  - Rendezvous V2

(For Nutanix only) Step 1: Install and register the Nutanix AHV plug-in

Obtain the Nutanix AHV plug-in package from Nutanix. Install and register the plug-in in your Citrix Virtual Apps and Desktops environment. For more information, see the Nutanix Acropolis MCS plug-in installation guide, available at the Nutanix Support Portal.

Step 1a: Install and register the Nutanix AHV plug-in for on-premises Delivery Controllers

After you install Citrix Virtual Apps and Desktops, select and install the **XD MCS AHV Plugin** on your Delivery Controllers.



Step 1b: Install and register the Nutanix AHV plug-in for cloud Delivery Controllers

Select and install the **CWA MCS AHV Plugin** for Citrix Cloud Connectors. Install the plug-in on all Citrix Cloud Connectors that are registered with the Citrix Cloud tenant. You must register Citrix Cloud Connectors even when they serve a resource location without the AHV.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 1c: Complete the following steps after installing the plug-in

- Verify that a Nutanix Acropolis folder has been created in `C:\Program Files\Common Files\Citrix\HCLPlugins\CitrixMachineCreation\v1.0.0.0`.
- Run the `"C:\Program Files\Common Files\Citrix\HCLPlugins\RegisterPlugins.exe" -PluginsRoot "C:\Program Files\Common Files\Citrix\HCLPlugins\CitrixMachineCreation\v1.0.0.0"` command.
- Restart the Citrix Host, Citrix Broker, and Citrix Machine Creation Services on your on-premises Delivery Controllers or restart the Citrix RemoteHCLServer Service on Citrix Cloud Connectors.

  > **Tip:**
  >
  > We recommend that you stop and then restart the Citrix Host, Citrix Broker, and Machine Creation Services when you install or update the Nutanix AHV plug-in.

Step 2: Create a host connection

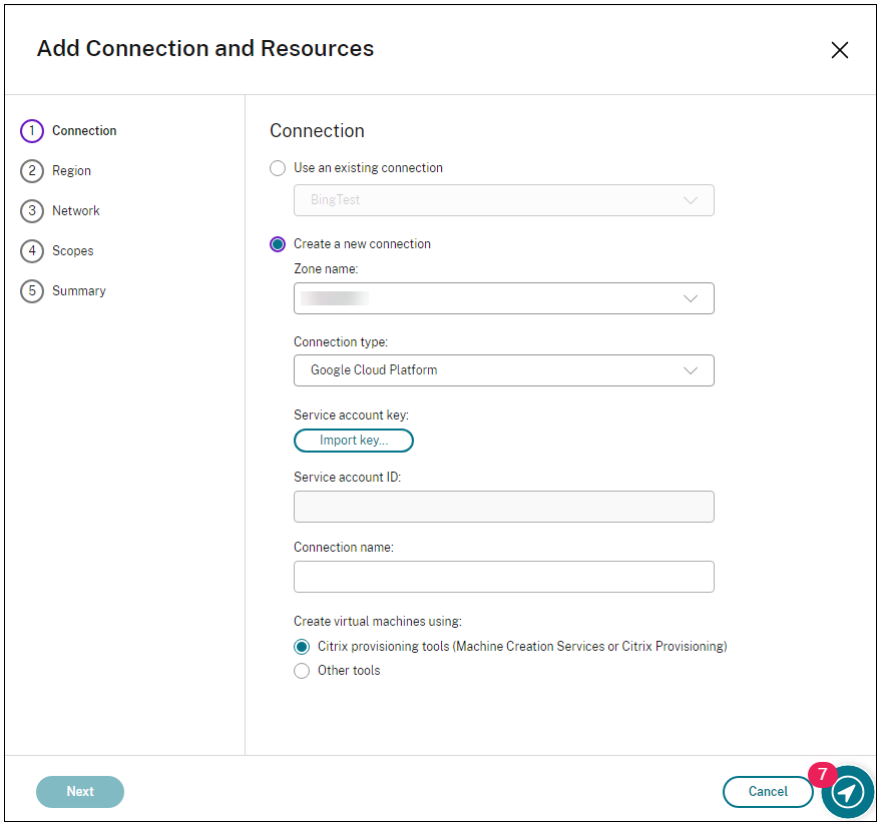
Hosts are hypervisors or cloud services that are in use in your resource locations. This step lets you specify information that DaaS uses to communicate with VMs on a host. Detailed information includes the resource location, host type, access credentials, storage method to use, and which networks the VMs on the host can use.

> **Important:**
>
> The host resources (storage and network) in your resource location must be available before you create a connection.

1. Sign in to Citrix Cloud.
2. In the upper left menu, select **My Services > DaaS**.
3. From **Manage > Full Configuration**, select **Hosting** in the left pane.
4. Select **Add Connections and Resources** in the action bar.
5. The wizard guides you through the following pages. Specific page content depends on the selected connection type. After completing each page, select **Next** until you reach the **Summary** page.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 2a: Connection



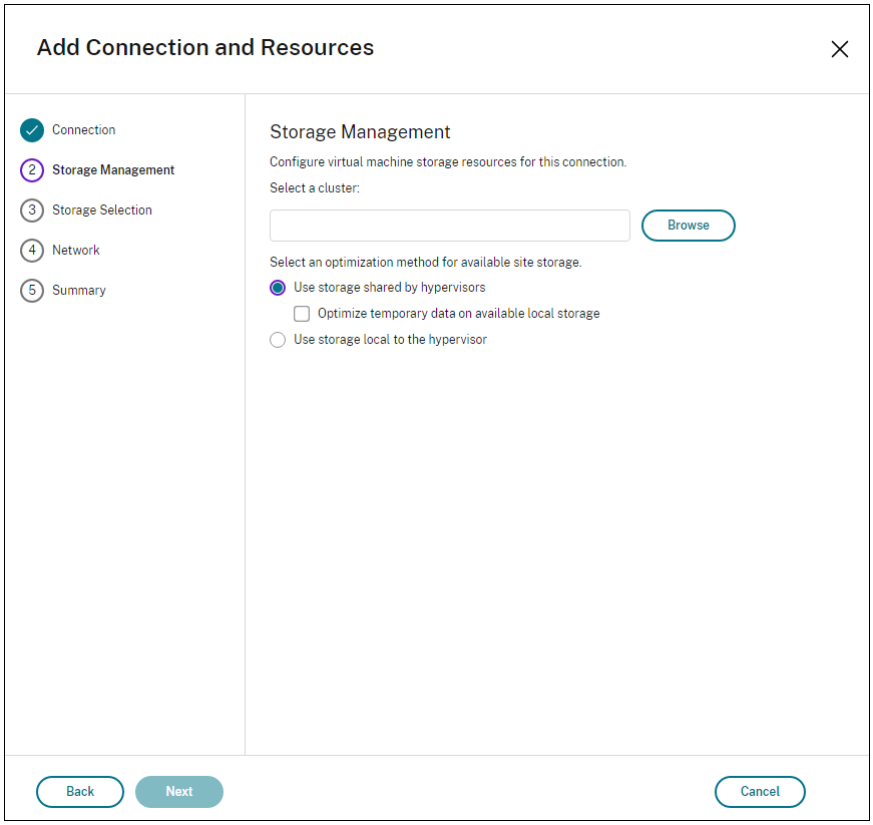
On the **Connection** page:

- To create a connection, select **Create a new Connection**. To create a connection based on the same host configuration as an existing connection, select **Use an existing Connection** and then choose the relevant connection.
- Select a zone in the **Zone name** field. The options are all resource locations you configured.
- Select a hypervisor or cloud service in the **Connection type** field. The options are hypervisors and cloud services that have their plug-ins installed properly in the zone.
  Alternatively, you can use the PowerShell command `Get-HypHypervisorPlugin -ZoneUid` to get the list of hypervisor plug-ins available with the selected zone.
- Enter a connection name. This name appears in the **Manage** display.
- Choose the tool to create virtual machines: Machine Creation Services or Citrix Provisioning.

Information on the **Connection** page differs depending on the host (connection type) you're using. For example, when using the Azure Resource Manager, you can use an existing service principal or create one.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 2b: Storage management



For information about storage management types and methods, see Host storage.

If you are configuring a connection to a Hyper-V or VMware host, browse to and then select a cluster name. Other connection types do not request a cluster name.

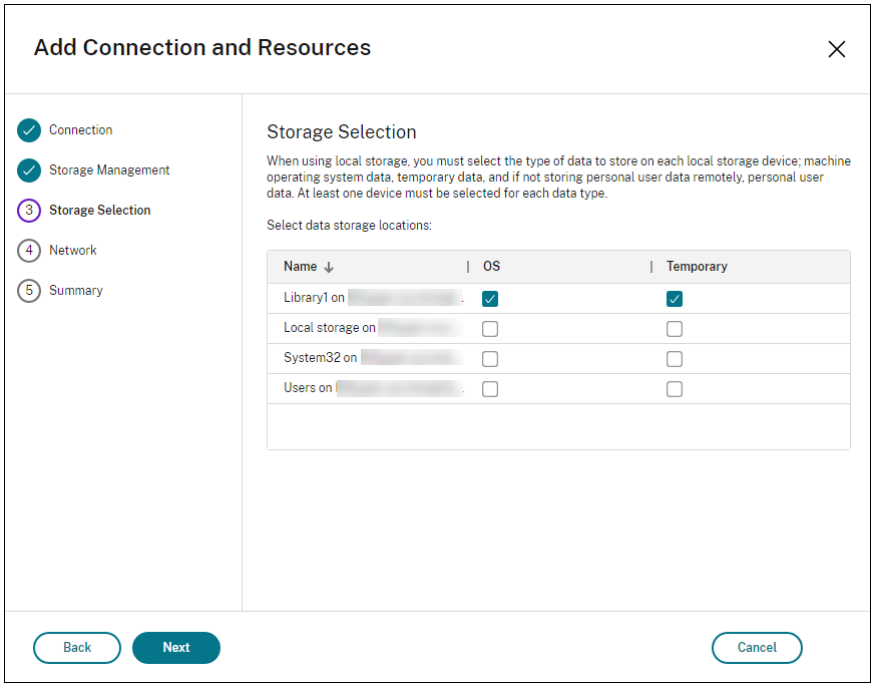
Select a storage management method: storage shared by hypervisors or storage local to the hypervisor.

- If you choose storage shared by hypervisors, indicate if you want to keep temporary data on the available local storage. (You can specify nondefault temporary storage sizes in the machine catalogs that use this connection.) **Exception:** When using Clustered Storage Volumes (CSV), Microsoft System Center Virtual Machine Manager does not allow temporary data cache disks to be created on local storage. Attempts to configure that storage management setup in the **Manage** console fails.

If you use shared storage in a Citrix Hypervisor pool, indicate if you want to use IntelliCache to reduce the load on the shared storage device. See Citrix Hypervisor virtualization environments.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 2c: Storage selection



For more information about storage selection, see Host storage.

Select at least one host storage device for each available data type. The storage management method that you selected on the previous page affects which data types are available for selection on this page. You must select at least one storage device for each supported data type before you can proceed to the next page in the wizard.

The lower portion of the **Storage Selection** page contains more configuration options if you chose storage shared by hypervisors and enabled **Optimize temporary data on available local storage**. You can select which local storage devices (in the same hypervisor pool) to use for temporary data.

The number of currently selected storage devices is shown (in the graphic, "1 storage device selected"). When you hover over that entry, the selected device names appear (unless no devices are configured).

1. Select **Select** to change the storage devices to use.
2. In the **Select Storage** dialog box, select or clear the storage device check boxes, and then select **OK**.

Step 2d: Region

(Appears only for some host types.) The region selection indicates where VMs will be deployed. Ideally, choose a region close to where users access their applications.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 2e: Network

Enter a name for the resources. This name appears in the **Manage** console to identify the storage and network combination associated with the connection.
Select one or more networks that the VMs use.

Some connection types (such as Azure Resource Manager) also list subnets that VMs use. Select one or more subnets.

Step 2f: Summary

Review your selections; if you want to make changes, use return to previous wizard pages. When you complete your review, select **Finish**.

Remember: If you store temporary data locally, you can configure nondefault values for temporary data storage when you create the catalog containing machines that use this connection.

> **Note:**
>
> A scope is not shown for Full access administrators. For more information, see Administrators, roles, and scopes.

For more information, see Create and manage connections.

Step 3: Prepare a master image

> **Tip:**
>
> You can use a single image for creating both domain-joined and non-domain-joined VDAs.

(For XenServer (formerly Citrix Hypervisor) only) Step 3a: Install XenServer VM Tools

Install XenServer VM Tools on the template VM for each VM to use the xe CLI or XenCenter. VM performance can be slow unless you install the tools. Without the tools, you can't do any of the following:

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

1. Download the XenServer VM Tools for Linux file from the XenServer Downloads page or the Citrix Hypervisor Downloads page based on the hypervisor version in use.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

2. Copy the `LinuxGuestTools-xxx.tar.gz` file to your Linux VM or to a shared drive that the Linux VM can access.
3. Extract the contents of the tar file: `tar -xzf LinuxGuestTools-xxx.tar.gz`
4. Run the following command to install the `xe-guest-utilities` package based on your Linux distribution.

   For RHEL/CentOS/Rocky Linux/SUSE:

   ```
   sudo rpm -i <extract-directory>/xe-guest-utilities_{
    package-version }
    _x86.64.rpm
   ```

   For Ubuntu/Debian:

   ```
   sudo dpkg -i <extract-directory>/xe-guest-utilities_{
    package-version }
    _amd64.deb
   ```

5. Check the virtualization state of the template VM on the **General** tab in XenCenter. If XenServer VM Tools are installed correctly, the virtualization state shows **Optimized**.

Step 3b: Install .NET and the Linux VDA package on the template VM

> **Note:**
>
> To use a currently running VDA as the template VM, skip this step.

Before installing the Linux VDA package, install .NET on the template VM and notice the following:

- In addition to the .NET Runtime, you must install .ASP.NET Core Runtime on all supported Linux distributions before you install or upgrade the Linux VDA. Version 6 is required for Amazon Linux 2. Version 8 is required for other distributions.
- If your Linux distribution contains the .NET version that you require, install it from the built-in feed. Otherwise, install .NET from the Microsoft package feed. For more information, see https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers.

After installing .NET, run the following commands based on your Linux distribution to install the Linux VDA:

For RHEL/CentOS/Rocky Linux:

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> After you install the Linux VDA on RHEL 8.x/9.x and Rocky Linux 8.x/9.x hosted on GCP, the Ethernet connection might be lost and the Linux VDA might be unreachable after a VM restart. To work around the issue, run the following commands before restarting the VM:
>
> ```
> nmcli dev connect eth0
> systemctl restart NetworkManager
> ```

For Ubuntu/Debian:

```
sudo dpkg –i  <PATH>/<Linux VDA DEB>

apt-get install -f
```

For SUSE:

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

Step 3c: (For RHEL only) Install the EPEL repository that can offer ntfs-3g Install the EPEL repository on RHEL 8. For information on how to install EPEL, see the instructions at https://docs.fedoraproject.org/en-US/epel/.

Step 3e: (For SUSE only) Manually install ntfs-3g

On the SUSE platform, no repository provides ntfs-3g. Download the source code, compile, and install ntfs-3g manually:

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

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

2. Download the ntfs-3g package.
3. Decompress the ntfs-3g package:

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

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

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

5. Install ntfs-3g:

   ```
   ./configure
   make
   make install
   ```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 3f: (For Ubuntu only) Edit the /etc/network/interfaces file

Add the `source /etc/network/interfaces.d/*` line to the `/etc/network/interfaces` file.

> **Tip**
>
> The `/etc/network/interfaces` file might not be available on your Ubuntu machine. If the file does not exist, you need to install the net-tools and ifupdown packages first.

Step 3f: (For Ubuntu only) Point /etc/resolv.conf

Point `/etc/resolv.conf` to `/run/systemd/resolve/resolv.conf` instead of pointing it to `/run/systemd/resolve/stub-resolv.conf`:

```
unlink /etc/resolv.conf

ln -s /run/systemd/resolve/resolv.conf /etc/resolv.conf
```

Step 3h: Specify a database to use

You can switch between SQLite and PostgreSQL after installing the Linux VDA package. To do so, complete the following steps:

> **Note:**
>
> - We recommend you use SQLite for VDI mode only and use PostgreSQL for a hosted shared desktops delivery model.
> - For easy install and MCS, you can specify SQLite or PostgreSQL to use without having to install them manually. Unless otherwise specified through **/etc/xdl/db.conf**, the Linux VDA uses PostgreSQL by default.
> - You can also use **/etc/xdl/db.conf** to configure the port number for PostgreSQL.

1. Run `/opt/Citrix/VDA/sbin/ctxcleanup.sh`. Omit this step if it is a fresh installation.
2. Edit `/etc/xdl/db.conf` before running `deploymcs.sh`.

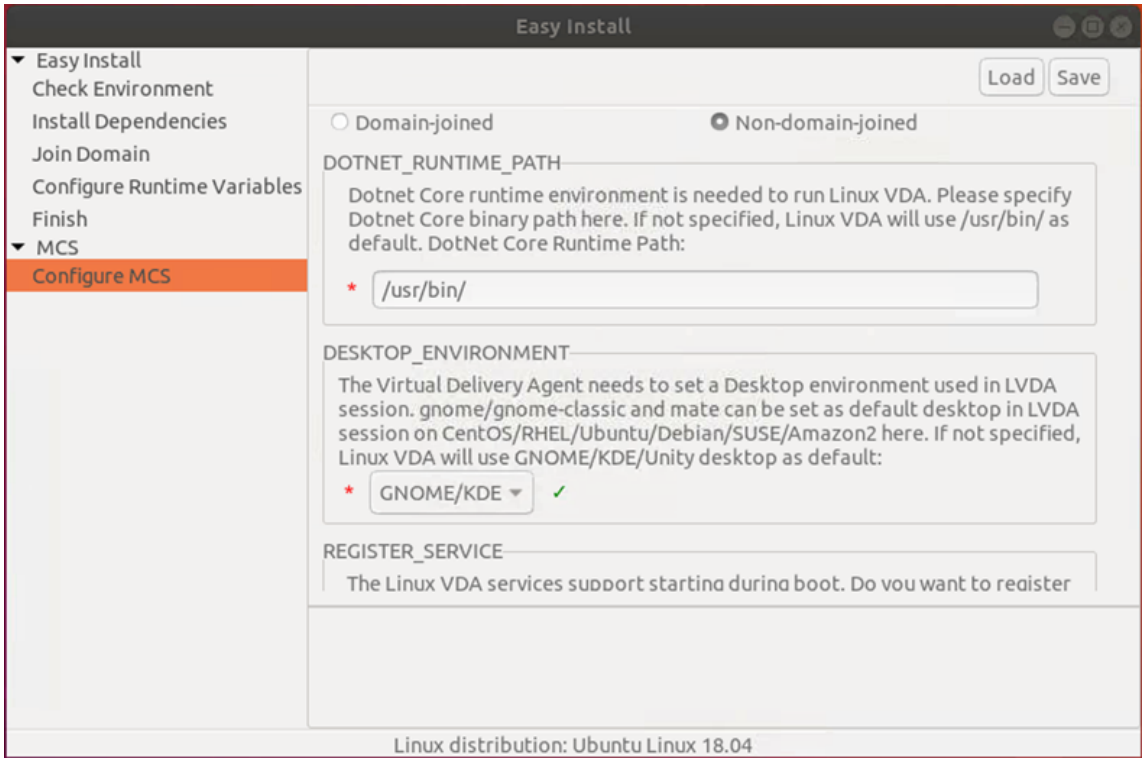
Step 3i: Configure MCS variables

There are two ways to configure MCS variables:

- Edit the `/etc/xdl/mcs/mcs.conf` file.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- Use the easy install GUI. To open the easy install GUI, run the `/opt/Citrix/VDA/bin/easyinstall` command in the desktop environment of your Linux VDA.



> **Tip:**
>
> Click **Save** to save variable settings to a local file under the path you specify. Click **Load** to load variable settings from a file that you specify.

The following are MCS variables that you can configure for non-domain-joined scenarios. You can use the default variable values or customize the variables as required (optional):

`DOTNET_RUNTIME_PATH`=path-to-install-dotnet-runtime
`DESKTOP_ENVIRONMENT`=gnome | mate
`REGISTER_SERVICE`=Y | N
`ADD_FIREWALL_RULES`=Y | N
`VDI_MODE`=Y | N
`START_SERVICE`=Y | N

(Optional) Step 3j: Write or update registry values for MCS

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

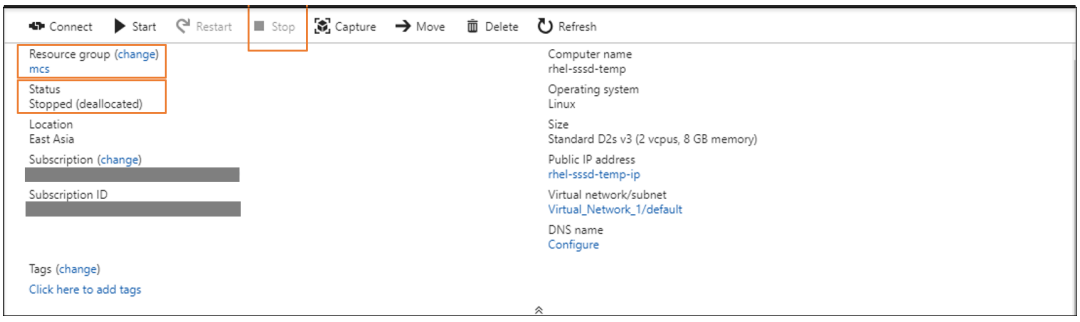
For example, you can add the following command lines to the /etc/xdl/mcs/mcs_local_setting.reg file to write or update a registry value respectively:

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

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

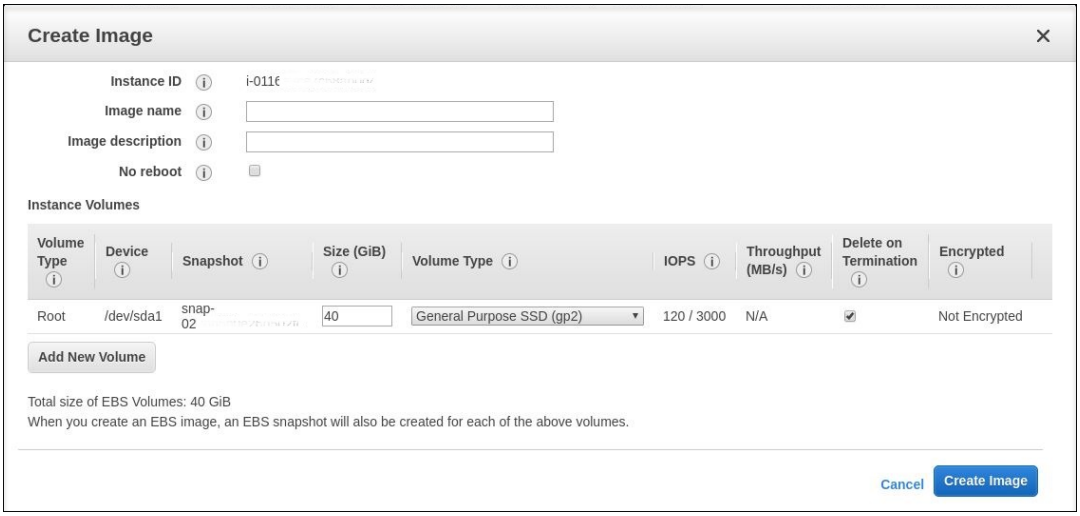
Step 3k: Create a master image

1. If you configure MCS variables by editing /etc/xdl/mcs/mcs.conf, run /opt/Citrix/VDA/sbin/deploymcs.sh. If you configure MCS variables by using the GUI, click **Deploy**. After you click **Deploy** on the GUI, the variables you set on the GUI override the variables you set in the /etc/xdl/mcs/mcs.conf file.
2. Create and name a snapshot of your master image based on the public cloud you use.
   - **(For XenServer (formerly Citrix Hypervisor), GCP, and VMware vSphere)** Install applications on the template VM and shut down the template VM. Create and name a snapshot of your master image.
   - **(For Azure)** Install applications on the template VM and shut down the template VM from the Azure portal. Ensure that the power status of the template VM shows **Stopped (deallocated)**. Remember the name of the resource group here. You need the name to locate your master image on Azure.



   - **(For AWS)** Install applications on the template VM and shut down the template VM from the AWS EC2 portal. Ensure that the instance state of the template VM shows **Stopped**. Right-click the template VM and select **Image > Create Image**. Type information and make settings as needed. Click **Create Image**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



- **(For Nutanix)** On Nutanix AHV, shut down the template VM. Create and name a snapshot of your master image.

  > **Note:**
  >
  > You must prefix Acropolis snapshot names with XD_ for use in Citrix Virtual Apps and Desktops. Use the Acropolis console to rename your snapshots when needed. After you rename a snapshot, restart the **Create Catalog** wizard to obtain a refreshed list.

Step 4: Create a machine catalog

1. Sign in to Citrix Cloud.
2. In the upper left menu, select **My Services > DaaS**.
3. From **Manage > Full Configuration**, select **Machine Catalogs**.
4. The wizard guides you to create a machine catalog.

   On the **Container** page that is unique to Nutanix, select the container that you specified for the template VM earlier.

   On the **Master Image** page, select the image snapshot.

   On the **Virtual Machines** page, check for the number of virtual CPUs and the number of cores per vCPU. Select MCS as the machine deployment method and select **Non-domain-joined** as the identity for machines to be created in the catalog.

   Do other configuration tasks as needed. For more information, see Create machine catalogs.

> **Note:**
>
> If your machine catalog creation process on the Delivery Controller takes a significant amount of

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> time, go to Nutanix Prism and power on the machine prefixed with **Preparation** manually. This approach helps to continue the creation process.

Step 5: Create a delivery group

A delivery group is a collection of machines selected from one or more machine catalogs. It specifies which users can use those machines, and the applications and desktops available to those users. For more information, see Create delivery groups.

Create a non-domain-joined Linux VDA using easy install (preview)

August 2, 2024

This article walks you through using the easy install method to create and enroll a non-domain-joined Linux VDA with a machine catalog in Citrix DaaS or Citrix Virtual Apps and Desktops 2407 and later.

> **Important:**
>
> - For Citrix DaaS customers:
>   - You can deploy non-domain-joined VDAs in a public cloud or in the on-premises data center. Non-domain-joined VDAs are managed by the control plane in Citrix DaaS.
>   - To create non-domain-joined VDAs, customers utilizing the Citrix Gateway service must ensure that Rendezvous V2 is enabled. Cloud Connectors are required only if you plan to provision machines on on-premises hypervisors or if you want to use Active Directory as the identity provider in Workspace.
> - To create non-domain joined VDAs, you can also use MCS. For more information, see Create non-domain-joined Linux VDAs using MCS.
>   - MCS doesn't support bare metal servers.
> - The following features are available for non-domain-joined Linux VDAs:
>   - Create local users with specified attributes on non-domain-joined VDAs
>   - Non-SSO authentication
>   - Authentication with Azure Active Directory
>   - Rendezvous V2

Step 1: Create an empty machine catalog

Sign in to the Citrix Web Studio and create an empty machine catalog with no machines in it. Note that the Linux VDA does not support the use of a token file for enrolling with a power-managed machine

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

catalog.

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

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

- For the operating system, select:
  - The **Multi-session OS** option for a hosted shared desktop delivery model.
  - The **Single-session OS** option for a VDI dedicated desktop delivery model.
- Do not mix Linux and Windows VDA machines in the same machine catalog.
- Do not mix domain-joined and non-domain-joined machines in the same machine catalog.
- Remote PC Access machine catalogs are supported for domain-joined machines only and are not supported for non-domain-joined machines.

> **Note:**
>
> Early versions of Citrix Studio don't support the notion of a "Linux OS."However, selecting the **Windows Server OS** or **Server OS** option implies an equivalent hosted shared desktops delivery model. Selecting the **Windows Desktop OS** or **Desktop OS** option implies a single user per machine delivery model.

Step 2: Create an enrollment token

To create a non-domain-joined VDA using easy install, you need a token file to enroll the VDA with a machine catalog and authenticate the VDA to the cloud or on-premises delivery controller. The Linux VDA does not support the use of a token file for enrolling with a power-managed machine catalog.

To create an enrollment token, complete the following steps in Citrix Web Studio:

> **Tip:**
>
> In Citrix DaaS, Web Studio is known as Full Configuration.

1. Select the empty machine catalog you created earlier and then select **Manage Enrollment Tokens** in the action bar.
2. On the **Manage Enrollment Tokens** page, click **Generate** to create an enrollment token. Alternatively, you can choose an existing token that is in a valid status.
3. Follow the wizard to complete the settings.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

4. Download the successfully created token and save it with least privilege in a secure location. Later when you run the easy install script, the CTX_XDL_NDJ_ENROLLMENT_TOKEN_FILE variable allows you to specify the path to the token file for enrolling the VDA.

Step 3: Install .NET

In addition to the .NET Runtime, you must install .ASP.NET Core Runtime on all supported Linux distributions before you install or upgrade the Linux VDA. Version 6 is required for Amazon Linux 2. Version 8 is required for other distributions.

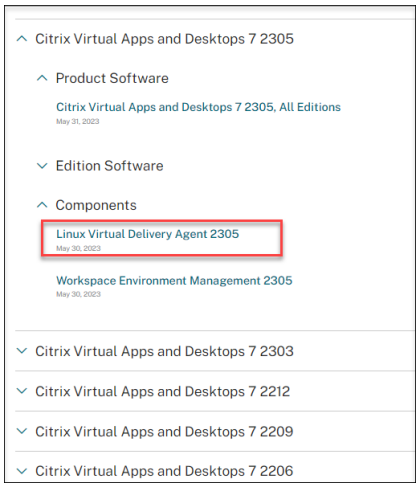
If your Linux distribution contains the .NET version that you require, install it from the built-in feed. Otherwise, install .NET from the Microsoft package feed. For more information, see https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers.

After installing .NET, run the **which dotnet** command to find your runtime path.

Based on the command output, set the .NET runtime binary path. For example, if the command output is /aa/bb/dotnet, use /aa/bb as the .NET binary path.

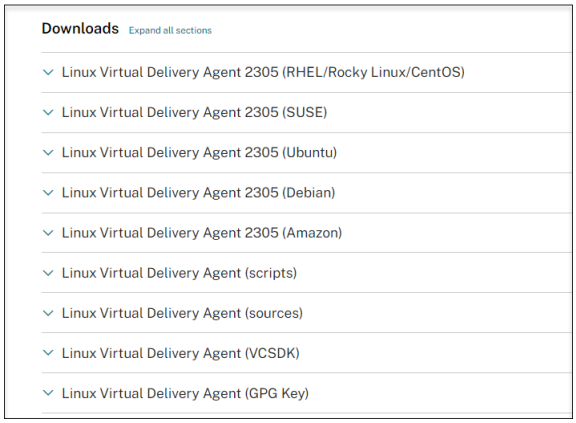
Step 4: Download the Linux VDA package

1. Go to the Citrix Virtual Apps and Desktops download page.
2. Expand the appropriate version of Citrix Virtual Apps and Desktops.
3. Expand **Components** to find the Linux VDA. For example:

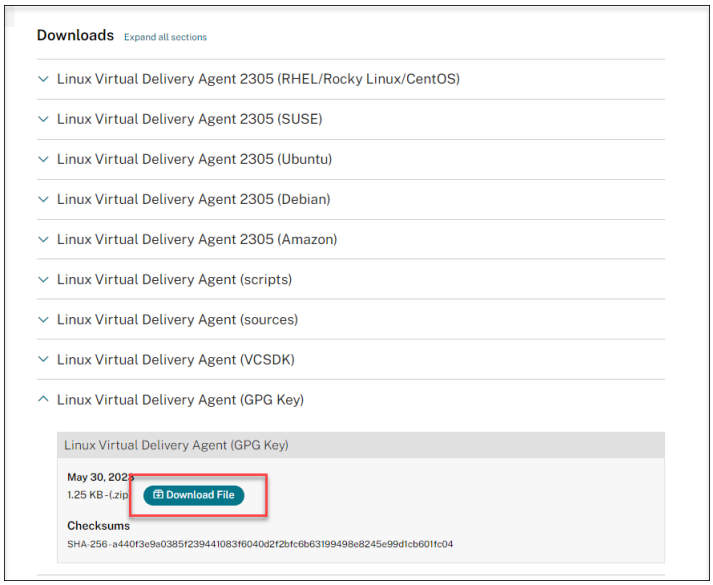


4. Click the Linux VDA link to access the Linux VDA downloads.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



5. Download the Linux VDA package that matches your Linux distribution.
6. Download the GPG public key that you can use to verify the integrity of the Linux VDA package. For example:



To verify the integrity of the Linux VDA package by using the public key:

- For an RPM package, run the following commands to import the public key into the RPM database and to check the package integrity:

```
rpmkeys --import <path to the public key>
rpm --checksig --verbose <path to the Linux VDA package>
```

- For a DEB package, run the following commands to import the public key into the DEB database and to check the package integrity:

```
sudo apt-get install dpkg-sig
gpg --import <path to the public key>
dpkg-sig --verify <path to the Linux VDA package>
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 5: Install the Linux VDA package

To set up the environment for the Linux VDA, run the following commands.

For Amazon Linux 2, CentOS, RHEL, and Rocky Linux distributions:

> **Note:**
>
> - For RHEL and CentOS, install the EPEL repository before you can install the Linux VDA successfully. For information on how to install EPEL, see the instructions at https://docs.fedoraproject.org/en-US/epel/.
> - Before installing the Linux VDA on RHEL 9.4/9.3/9.2 and Rocky Linux 9.4/9.3/9.2, update the **libsepol** package to version 3.4 or later.

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

> **Note:**
>
> After you install the Linux VDA on RHEL 8.x/9.x and Rocky Linux 8.x/9.x hosted on GCP, the Ethernet connection might be lost and the Linux VDA might be unreachable after a VM restart. To work around the issue, set a root password when logging on to the VM for the first time and make sure that you can log on to the VM as root. Then, run the following commands in the console after restarting the VM:
>
> ```
> nmcli dev connect eth0
> systemctl restart NetworkManager
> ```

For Ubuntu/Debian distributions:

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

> **Note:**
>
> - To install the necessary dependencies for a Debian 11 distribution, add the `deb http://deb.debian.org/debian/ bullseye main` line to the /etc/apt/sources.list file.
> - For Ubuntu 20.04 on GCP, disable RDNS. To do so, add the **rdns = false** line under **[libdefaults]** in /etc/krb5.conf.

For SUSE distributions:

1. For SUSE 15.5 on AWS, Azure, and GCP, ensure that:
   - You are using **libstdc++6** version 12 or later.
   - The **Default_WM** parameter in **/etc/sysconfig/windowmanager** is set to **"gnome"**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

2. Run the following command to install the Linux VDA:

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

Step 6: Install NVIDIA GRID drivers

Enabling HDX 3D Pro requires you to install the NVIDIA GRID drivers on your hypervisor and on the VDA machine.

To install and configure the NVIDIA GRID Virtual GPU Manager (the host driver) on the specific hypervisors, see the following guides:

- Citrix Hypervisor
- VMware ESX
- Nutanix AHV

To install and configure the NVIDIA GRID guest VM drivers, perform the following general steps:

1. Ensure that the guest VM is shut down.
2. In the hypervisor control panel, allocate a GPU to the VM.
3. Start the VM.
4. Install the guest VM driver on the VM.

Step 7: Specify a database to use

You can specify SQLite or PostgreSQL to use by editing **/etc/xdl/db.conf** after installing the Linux VDA package.

To do so, edit **/etc/xdl/db.conf** before running **sudo /opt/Citrix/VDA/sbin/ctxinstall.sh** or **/opt/Citrix/VDA/bin/easyinstall**.

> **Note:**
>
> - We recommend you use SQLite for VDI mode only.
> - For easy install and MCS, you can specify SQLite or PostgreSQL to use without having to install them manually. Unless otherwise specified through **/etc/xdl/db.conf**, the Linux VDA uses PostgreSQL by default.
> - You can also use **/etc/xdl/db.conf** to configure the port number for PostgreSQL.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 8: Run easy install to configure the environment and VDA to complete the installation

After installing the Linux VDA package, configure the running environment by using the ctxinstall.sh script.

> **Note:**
>
> Before setting up the runtime environment, ensure that the **en_US.UTF-8** locale is installed in your OS. If the locale is not available in your OS, run the **sudo locale-gen en_US.UTF-8** command. For Debian, edit the **/etc/locale.gen** file by uncommenting the **# en_US.UTF-8 UTF-8** line and then run the **sudo locale-gen** command.

ctxinstall.sh

ctxinstall.sh is the easy install script for doing some pre-configuration and setting up the VDA running environment variables.

- Only root can run this script.
- Easy install uses /opt/Citrix/VDA/sbin/ctxinstall.conf as its configuration file to set, save, and synchronize the values of all environment variables used. We recommend you read the template (ctxinstall.conf.tmpl) carefully and then customize your own ctxinstall.conf. When you first create the configuration file, use either of the following ways:
  - By copying the /opt/Citrix/VDA/sbin/ctxinstall.conf.tmpl template file and saving it as /opt/Citrix/VDA/sbin/ctxinstall.conf.
  - By running ctxinstall.sh. Each time you run ctxinstall.sh, your input is saved in /opt/Citrix/VDA/sbin/ctxinstall.conf.
- Easy install supports modular running. Modules include pre-check, installation, domain-configuration, setup, and verification.
- Debugging details for this script can be found in /var/log/xdl/ctxinstall.log.

For more information, use the help command **ctxinstall.sh -h**.

> **Note:**
>
> - Following the principle of least privilege, ensure that only the root user can read **/opt/Citrix/VDA/sbin/ctxinstall.conf** because the domain joining password might be set in the file.
> - Uninstalling the Linux VDA removes files under **/opt/Citrix/VDA**. We recommend you back up **/opt/Citrix/VDA/sbin/ctxinstall.conf** before uninstalling the VDA.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

You can run ctxinstall.sh in interactive mode or silent mode. Before you run the script, set the following environment variables:

- **CTX_XDL_NON_DOMAIN_JOINED='y|n'**–Whether to join the machine to a domain. The default value is 'n'. For non-domain-joined scenarios, set it to 'y'.
- **CTX_XDL_NDJ_ENROLLMENT_TOKEN_FILE='<path-to-token-file-on-vda-machine>'** –To create a non-domain-joined VDA using easy install, you need a token file to enroll the VDA in a machine catalog of the Delivery Controller. Save the token with least privilege in a secure location.
- **CTX_XDL_VDI_MODE='y|n'**–Whether to configure the machine as a dedicated desktop delivery model (VDI) or hosted shared desktop delivery model. For HDX 3D Pro environments, set the value to 'y'.
- **CTX_XDL_HDX_3D_PRO='y|n'**–The Linux VDA supports HDX 3D Pro, a set of GPU acceleration technologies designed to optimize the virtualization of rich graphics applications. If HDX 3D Pro is selected, the VDA is configured for VDI desktops (single-session) mode - (that is, CTX_XDL_VDI_MODE='y').
- **CTX_XDL_START_SERVICE='y|n'**–Determines whether the Linux VDA services are started when the configuration is complete.
- **CTX_XDL_REGISTER_SERVICE='y|n'**–The Linux Virtual Desktop services are started after machine startup.
- **CTX_XDL_ADD_FIREWALL_RULES='y|n'**–The Linux VDA services require incoming network connections to be allowed through the system firewall. You can open the required ports (by default ports 80 and 1494) automatically in the system firewall for the Linux Virtual Desktop.
- **CTX_XDL_DESKTOP_ENVIRONMENT=gnome/gnome-classic/mate** –Specifies the GNOME, GNOME Classic, or MATE desktop environment to use in sessions. If you leave the variable unspecified, the default desktop configured on the VDA is used.
- **CTX_XDL_DESKTOP_ENVIRONMENT=gnome/gnome-classic/kde/mate/xfce/'<none>'** –Specifies the GNOME, GNOME Classic, KDE, MATE, or Xfce desktop environment to use in sessions. If you set it to '<none>', the default desktop configured on the VDA is used. You can specify a desktop environment to use in sessions by using the CTX_XDL_DESKTOP_ENVIRONMENT variable. You can also switch between desktop environments by running commands or using the system tray. For more information, see Desktop switching commands and System tray.
- **CTX_XDL_DOTNET_RUNTIME_PATH='<path-to-install-dotnet-runtime>'**–The path to install .NET for supporting the new broker agent service (ctxvda). The default path is '/usr/bin'.
- **CTX_XDL_VDA_PORT='<port-number>'**–The Linux VDA communicates with Delivery Controllers through a TCP/IP port.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Interactive mode To run the **ctxinstall.sh** script in interactive mode, use the **sudo /opt/Citrix/VDA/sbin/ctxinstall.sh** command without the **-S** option. Type the relevant variable value at each prompt in the command-line interface. If a variable is already set, **ctxinstall.sh** asks for confirmation in case you want to change it.

Silent mode In silent mode, you must set the preceding variables by using **/opt/Citrix/VDA/sbin/ctxinstall.conf** or the export command. After that, run **ctxinstall.sh -S** (note that the letter **S** here is in **uppercase**). If not all required variables are set or some value is invalid, **ctxinstall.sh** aborts execution, unless there are default values.

If you set it, the exported value for each variable overwrites the value in **/Citrix/VDA/sbin/ctxinstall.conf**. All updated values are saved in **/Citrix/VDA/sbin/ctxinstall.conf**.

```
export CTX_XDL_NON_DOMAIN_JOINED='y'
export CTX_XDL_NDJ_ENROLLMENT_TOKEN_FILE='<token-file-path>'
export CTX_XDL_VDI_MODE='y|n'
export CTX_XDL_START_SERVICE='y|n'
export CTX_XDL_REGISTER_SERVICE='y|n'
export CTX_XDL_ADD_FIREWALL_RULES='y|n'
export CTX_XDL_HDX_3D_PRO='y|n'
export CTX_XDL_DESKTOP_ENVIRONMENT=gnome|gnome-classic|kde|mate|xfce|'<
    none>'
export CTX_XDL_DOTNET_RUNTIME_PATH='<path-to-install-dotnet-runtime>'
export CTX_XDL_VDA_PORT='<port-number>'
sudo -E /opt/Citrix/VDA/sbin/ctxinstall.sh -S
```

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

Alternatively, you can specify all variables by using a single command.

To set up the VDA running environment variables (those variables beginning with '**CTX_XDL_**'), you can run **ctxinstall.sh -s** (note that the letter **s** here is in **lowercase**).

Step 9: Run XDPing

Run `sudo /opt/Citrix/VDA/bin/xdping` to check for common configuration issues with a Linux VDA environment. For more information, see **XDPing**.

Step 10: Run the Linux VDA

Start the Linux VDA:

To start the Linux VDA services:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
sudo systemctl start ctxhdx.service

sudo systemctl start ctxvda.service
```

Stop the Linux VDA:

To stop the Linux VDA services:

```
sudo systemctl stop ctxvda.service

sudo systemctl stop ctxhdx.service
```

> **Note:**
>
> Before you stop the **ctxvda** and **ctxhdx** services, run the **systemctl stop ctxmonitord** command to stop the monitor service daemon. Otherwise, the monitor service daemon restarts the services you stopped.

Restart the Linux VDA:

To restart the Linux VDA services:

```
sudo systemctl stop ctxvda.service

sudo systemctl restart ctxhdx.service

sudo systemctl start ctxvda.service
```

Check the status of the Linux VDA:

To check the running status of the Linux VDA services:

```
sudo systemctl status ctxvda.service

sudo systemctl status ctxhdx.service
```

Step 11: Create delivery groups

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:

- Ensure that the AD users and groups that you select have been properly configured to log on to the Linux VDA machines.
- Do not allow logon of unauthenticated (anonymous) users.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- Do not mix the delivery group with machine catalogs that contain Windows machines.

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

For information about how to create machine catalogs and delivery groups, see Citrix Virtual Apps and Desktops 7 2407.

Create Linux VDAs using Machine Creation Services (MCS)

August 2, 2024

You can create domain-joined and non-domain-joined VDAs using MCS. If you want to create non-domain-joined Linux VDAs in Citrix DaaS, you can also refer to the dedicated article Create non-domain-joined Linux VDAs using MCS.

> **Important:**
>
> The following are important changes starting with the 2212 release:
>
> - This **AD_INTEGRATION** variable in the /etc/xdl/mcs/mcs.conf file or on the easy install GUI does not have a default value any longer. You must set a value as needed. For more information, see the Step 3h: Configure MCS variables section in this article.
> - The valid value of the **UPDATE_MACHINE_PW** entry in /etc/xdl/mcs/mcs.conf is no longer **enabled** or **disabled**, but **Y** or **N**. For more information, see the Automate machine account password updates section in this article.

Supported distributions

	Winbind	SSSD	Centrify	PBIS
Debian 12.5/11.9	Yes	Yes	No	Yes
RHEL 9.4/9.3/9.2	Yes	Yes	No	No
RHEL 8.10/8.9/8.8	Yes	Yes	Yes	Yes
Rocky Linux 9.4/9.3/9.2	Yes	Yes	No	No

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

	Winbind	SSSD	Centrify	PBIS
Rocky Linux 8.10/8.9/8.8	Yes	Yes	No	No
SUSE 15.5	Yes	Yes	No	Yes
Ubuntu 22.04, Ubuntu 20.04	Yes	Yes	No	Yes

- Citrix uses the following Centrify versions for initial feature validation on the relevant Linux distributions:

Linux distribution	Centrify version
RHEL 7/8	5.8.0
SUSE	5.7.1
Debian, Ubuntu	5.6.1

  Using other versions of Centrify might cause errors. Do not use Centrify to join a template machine to a domain.

- If you are using PBIS or Centrify for joining MCS-created machines to Windows domains, complete the following tasks:
  - On the template machine, configure the PBIS or Centrify package download path in the `/etc/xdl/mcs/mcs.conf` file or install the PBIS or Centrify package directly.
  - Before you run `/opt/Citrix/VDA/sbin/deploymcs.sh`, create an Organizational Unit (OU) that has write and password reset permissions to all its subordinate, MCS-created machines.
  - Before you restart MCS-created machines after `/opt/Citrix/VDA/sbin/deploymcs.sh` finishes running, run `klist -li 0x3e4 purge` on your Delivery Controller or on your Citrix Cloud Connector based on your deployment.
- To use a currently running RHEL 8.x/9.x or Rocky Linux 8.x/9.x VDA that is connected to the domain using SSSD as the template VM for MCS, ensure that:
  - The VDA is installed manually and not by using easy install. Easy install uses **Adcli** for RHEL 8.x/9.x and Rocky Linux 8.x/9.x and the combination of SSSD and **Adcli** is not supported by MCS.
  - A Samba server is configured to use SSSD for AD authentication. For more information, see the Red Hat article at https://access.redhat.com/solutions/3802321.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Supported hypervisors

- AWS
- XenServer (formerly Citrix Hypervisor)
- GCP
- Microsoft Azure
- Nutanix AHV
- VMware vSphere

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

Use MCS to create Linux VMs

Considerations

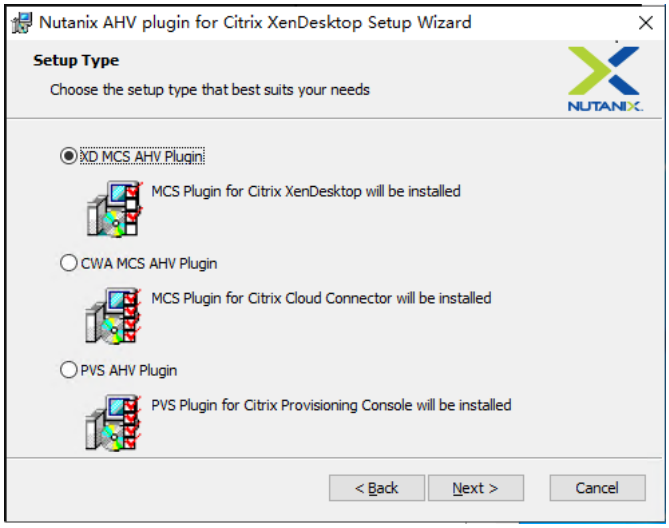
- Starting with the 2203 release, you can host the Linux VDA on Microsoft Azure, AWS, and GCP for Citrix Virtual Apps and Desktops as well as Citrix DaaS (formerly Citrix Virtual Apps and Desktops service). To add these public cloud host connections to your Citrix Virtual Apps and Desktops deployment, you need the Citrix Universal Hybrid Multi-Cloud (HMC) license.
- Bare metal servers are not supported for use with MCS to create virtual machines.

(For Nutanix only) Step 1: Install and register the Nutanix AHV plug-in

Obtain the Nutanix AHV plug-in package from Nutanix. Install and register the plug-in in your Citrix Virtual Apps and Desktops environment. For more information, see the Nutanix Acropolis MCS plug-in installation guide, available at the Nutanix Support Portal.

Step 1a: Install and register the Nutanix AHV plug-in for on-premises Delivery Controllers After you install Citrix Virtual Apps and Desktops, select and install the **XD MCS AHV Plugin** on your Delivery Controllers.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



Step 1b: Install and register the Nutanix AHV plug-in for cloud Delivery Controllers Select and install the **CWA MCS AHV Plugin** for Citrix Cloud Connectors. Install the plug-in on all Citrix Cloud Connectors that are registered with the Citrix Cloud tenant. You must register Citrix Cloud Connectors even when they serve a resource location without the AHV.

Step 1c: Complete the following steps after installing the plug-in

- Verify that a Nutanix Acropolis folder has been created in `C:\Program Files\Common Files\Citrix\HCLPlugins\CitrixMachineCreation\v1.0.0.0`.
- Run the `"C:\Program Files\Common Files\Citrix\HCLPlugins\RegisterPlugins.exe" -PluginsRoot "C:\Program Files\Common Files\Citrix\HCLPlugins\CitrixMachineCreation\v1.0.0.0"` command.
- Restart the Citrix Host, Citrix Broker, and Citrix Machine Creation Services on your on-premises Delivery Controllers or restart the Citrix RemoteHCLServer Service on Citrix Cloud Connectors.

  > **Tip:**
  >
  > We recommend that you stop and then restart the Citrix Host, Citrix Broker, and Machine Creation Services when you install or update the Nutanix AHV plug-in.

Step 2: Create a host connection

This section gives examples on how to create a host connection to Azure, AWS, XenServer (formerly Citrix Hypervisor), GCP, Nutanix AHV, and VMware vSphere.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> For on-premises Delivery Controllers, choose **Configuration > Hosting > Add Connection and Resources** in the on-premises Citrix Studio to create a host connection. For cloud Delivery Controllers, choose **Manage > Hosting > Add Connection and Resources** in the web-based Studio console on Citrix Cloud to create a host connection.

For more information, see Create and manage connections and resources in the Citrix Virtual Apps and Desktops documentation and Create and manage connections in the Citrix DaaS documentation.

- Create a host connection to Azure in Citrix Studio
- Create a host connection to AWS in Citrix Studio
- Create a host connection to XenServer in Citrix Studio
- Create a host connection to GCP in Citrix Studio
- Create a host connection to Nutanix in Citrix Studio
- Create a host connection to VMware in Citrix Studio

Create a host connection to Azure in Citrix Studio

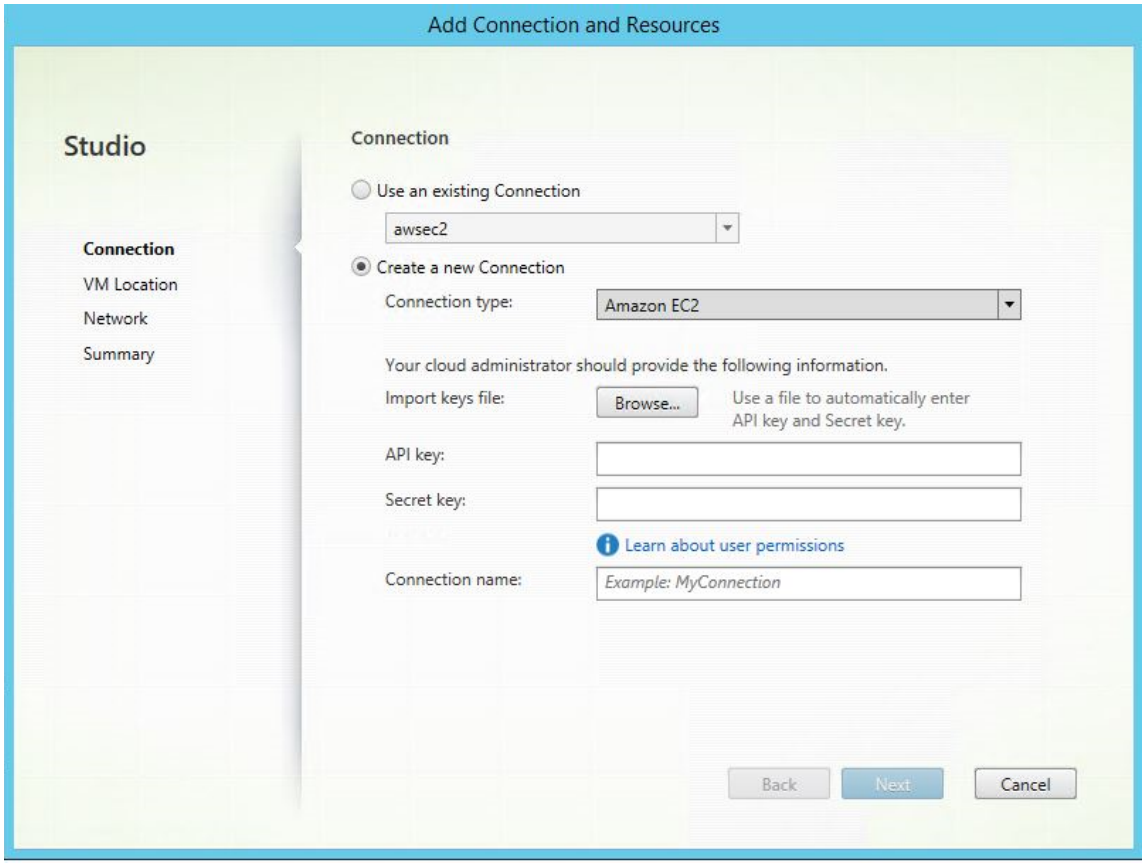
1. For on-premises Delivery Controllers, choose **Configuration > Hosting > Add Connection and Resources** in the on-premises Citrix Studio to create a host connection. For cloud Delivery Controllers, choose **Manage > Hosting > Add Connection and Resources** in the web-based Studio console on Citrix Cloud to create a host connection.
2. In the **Add Connection and Resources** wizard, select Microsoft Azure as the connection type.
3. Select Microsoft Azure as the connection type.
4. The wizard guides you through the pages. Specific page content depends on the selected connection type. After completing each page, select **Next** until you reach the **Summary** page. For more information, see **Step 2: Create a host connection** in the Create non-domain-joined Linux VDAs using MCS article.

Create a host connection to AWS in Citrix Studio

1. For on-premises Delivery Controllers, choose **Configuration > Hosting > Add Connection and Resources** in the on-premises Citrix Studio to create a host connection. For cloud Delivery Controllers, choose **Manage > Hosting > Add Connection and Resources** in the web-based Studio console on Citrix Cloud to create a host connection.
2. In the **Add Connection and Resources** wizard, select **Amazon EC2** as the connection type.

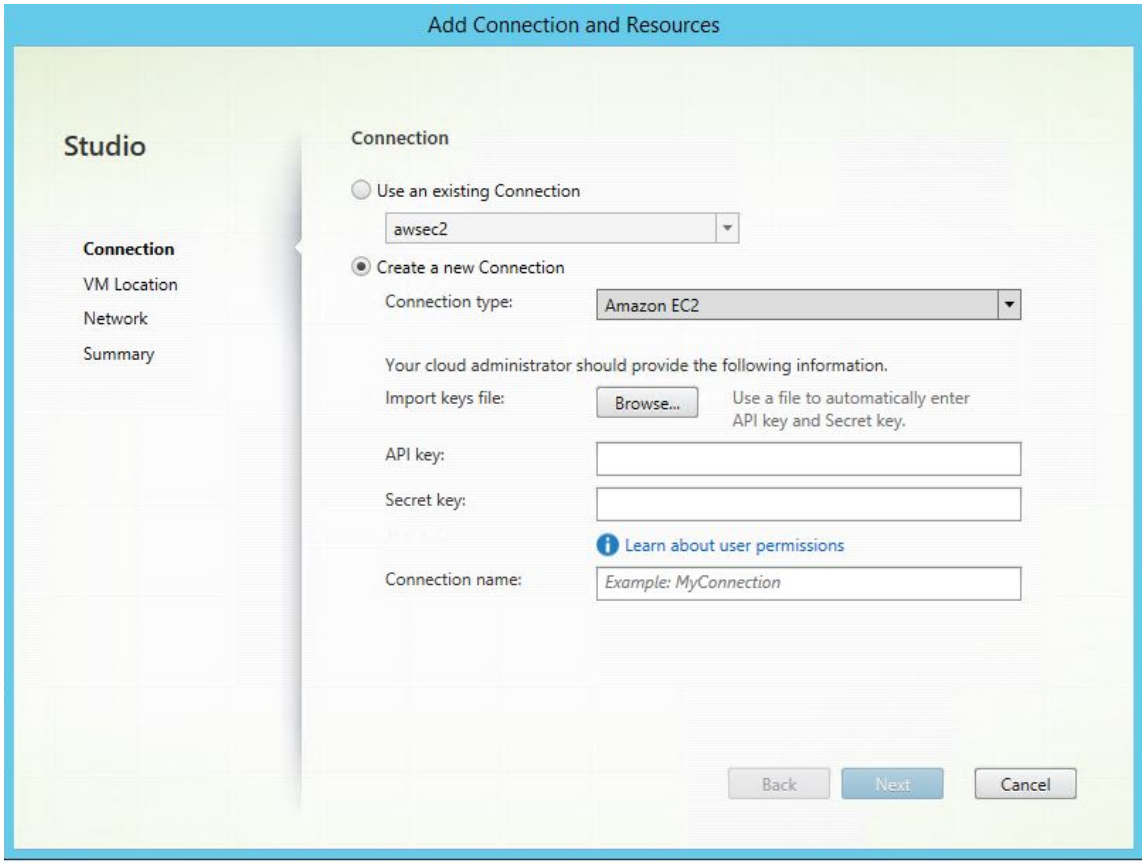
   For example, in the on-premises Citrix Studio:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



3. Type the API key and secret key of your AWS account and type your connection name.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



The **API key** is your access key ID and the **Secret key** is your secret access key. They are considered as an access key pair. If you lose your secret access key, you can delete the access key and create another one. To create an access key, do the following:

a) Sign in to the AWS services.
b) Navigate to the Identity and Access Management (IAM) console.
   1 On the left navigation pane, choose **Users**.
c) Select the target user and scroll down to select the **Security credentials** tab.
d) Scroll down and click **Create access key**. A new window appears.
e) Click **Download .csv file** and save the access key to a secure location.

4. The wizard guides you through the pages. Specific page content depends on the selected connection type. After completing each page, select **Next** until you reach the **Summary** page.

Create a host connection to XenServer in Citrix Studio

1. For on-premises Delivery Controllers, choose **Configuration > Hosting > Add Connection and Resources** in the on-premises Citrix Studio to create a host connection. For cloud Delivery Controllers, choose **Manage > Hosting > Add Connection and Resources** in the web-based Studio console on Citrix Cloud to create a host connection.

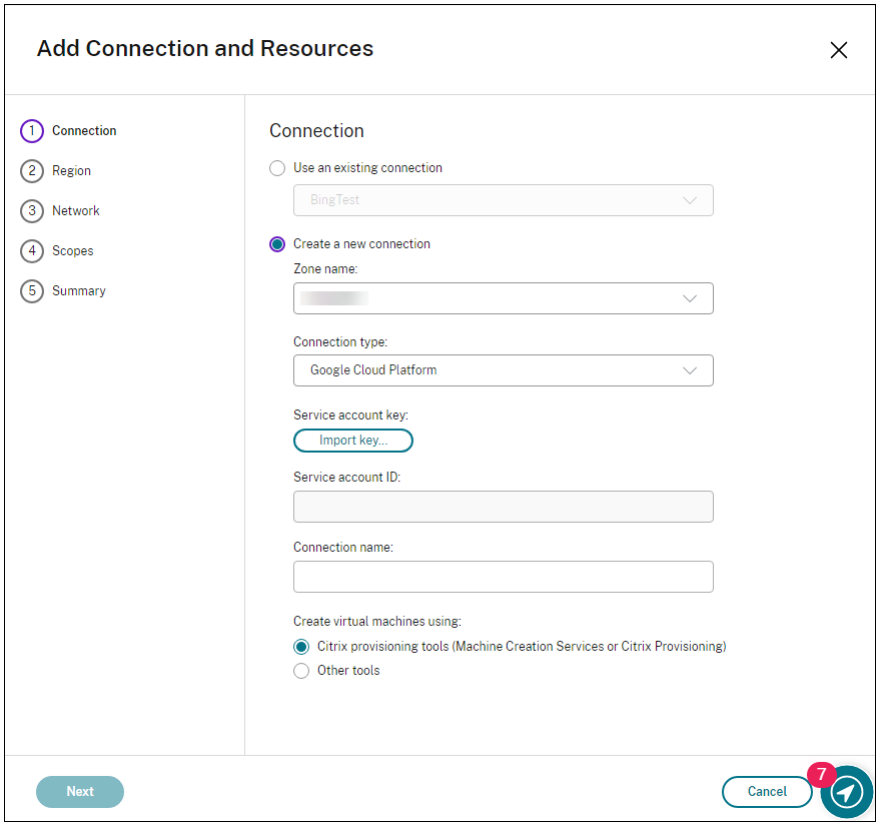
© 1999–2024 Cloud Software Group, Inc. All rights reserved.

2. In the **Add Connection and Resources** wizard, select XenServer (formerly Citrix Hypervisor) in the **Connection type** field.
3. Type the connection address (the XenServer URL) and credentials.
4. Enter a connection name.

Create a host connection to GCP in Citrix Studio Set up your GCP environment according to Google Cloud Platform virtualization environments and then complete the following steps to create a host connection to GCP.

1. For on-premises Delivery Controllers, choose **Configuration > Hosting > Add Connection and Resources** in the on-premises Citrix Studio to create a host connection. For cloud Delivery Controllers, choose **Manage > Hosting > Add Connection and Resources** in the web-based Studio console on Citrix Cloud to create a host connection.
2. In the **Add Connection and Resources** wizard, select **Google Cloud Platform** as the connection type.

   For example, in the web-based Studio console on Citrix Cloud:



3. Import the service account key of your GCP account and type your connection name.
4. The wizard guides you through the pages. Specific page content depends on the selected connection type. After completing each page, select **Next** until you reach the **Summary** page. For

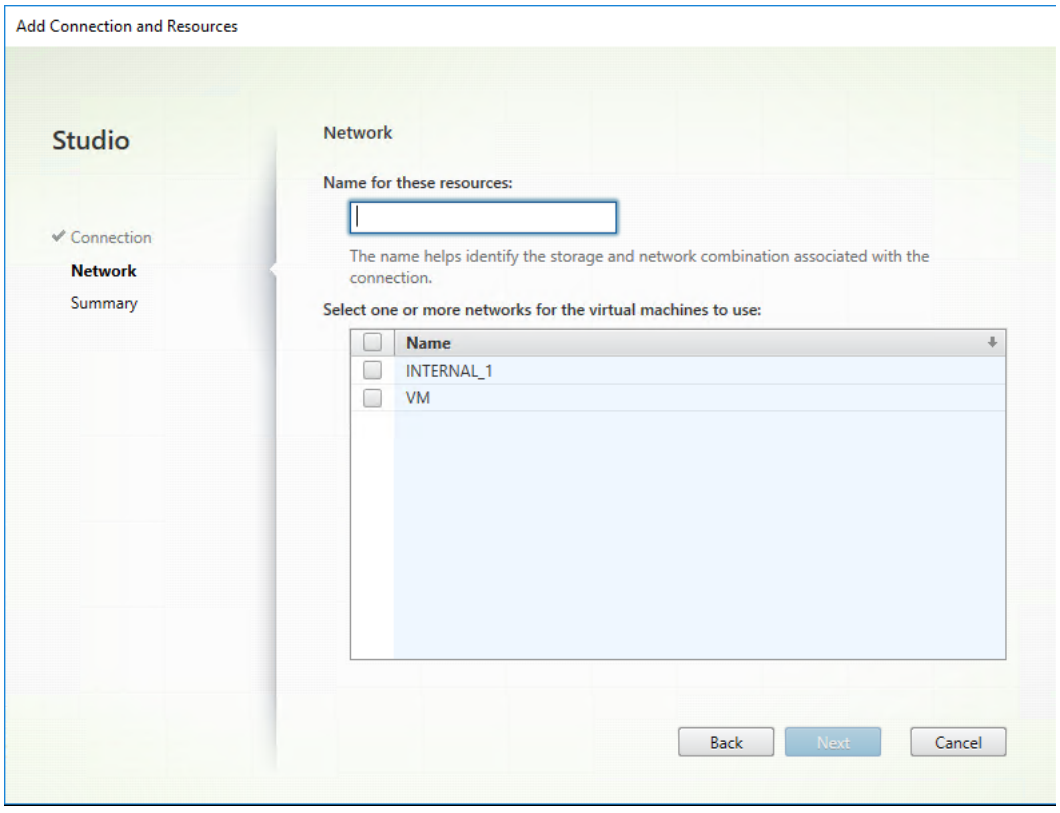
© 1999–2024 Cloud Software Group, Inc. All rights reserved.

more information, see **Step 2: Create a host connection** in the Create non-domain-joined Linux VDAs using MCS article.

Create a host connection to Nutanix in Citrix Studio

1. For on-premises Delivery Controllers, choose **Configuration > Hosting > Add Connection and Resources** in the on-premises Citrix Studio to create a host connection. For cloud Delivery Controllers, choose **Manage > Hosting > Add Connection and Resources** in the web-based Studio console on Citrix Cloud to create a host connection.
2. In the **Add Connection and Resources** wizard, select Nutanix AHV as the connection type on the **Connection** page, and then specify the hypervisor address, credentials, and your connection name. On the **Network** page, select a network for the unit.

   For example, in the on-premises Citrix Studio:



Create a host connection to VMware in Citrix Studio

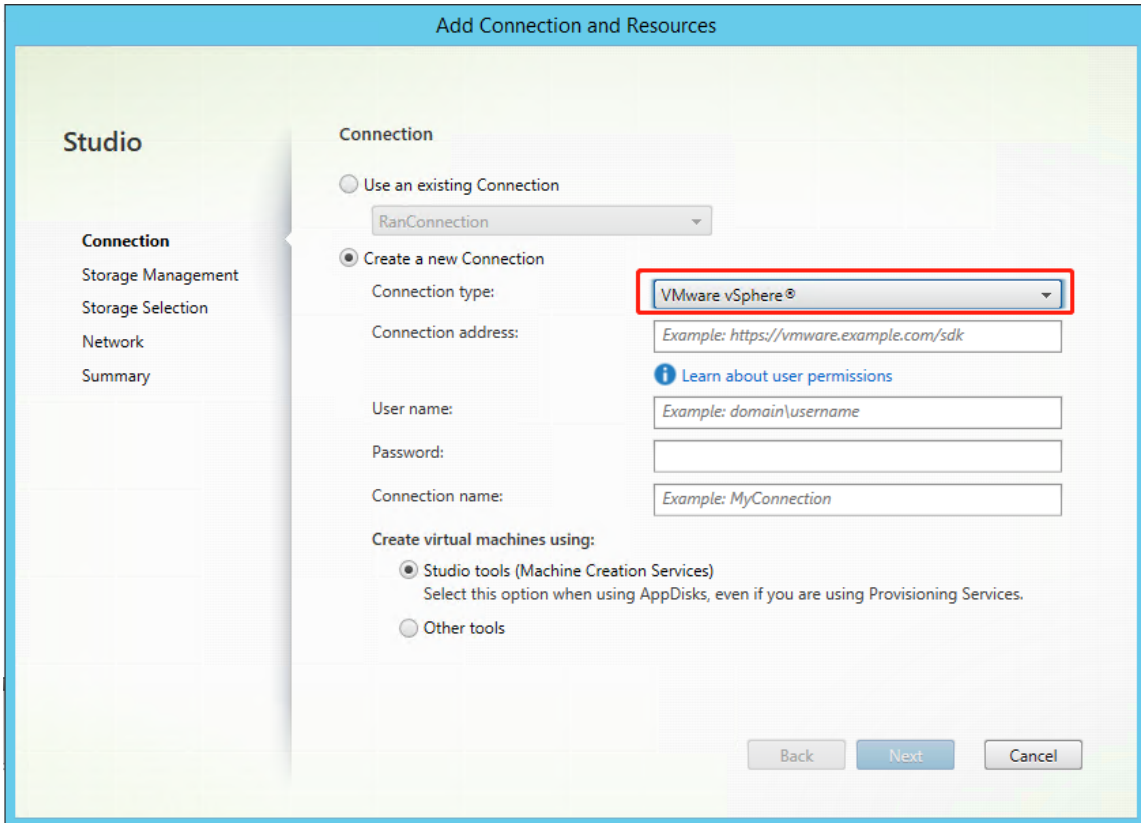
1. Install vCenter Server in the vSphere environment. For more information, see VMware vSphere.
2. For on-premises Delivery Controllers, choose **Configuration > Hosting > Add Connection and Resources** in the on-premises Citrix Studio to create a host connection. For cloud Delivery Con-

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

trollers, choose **Manage > Hosting > Add Connection and Resources** in the web-based Studio console on Citrix Cloud to create a host connection.

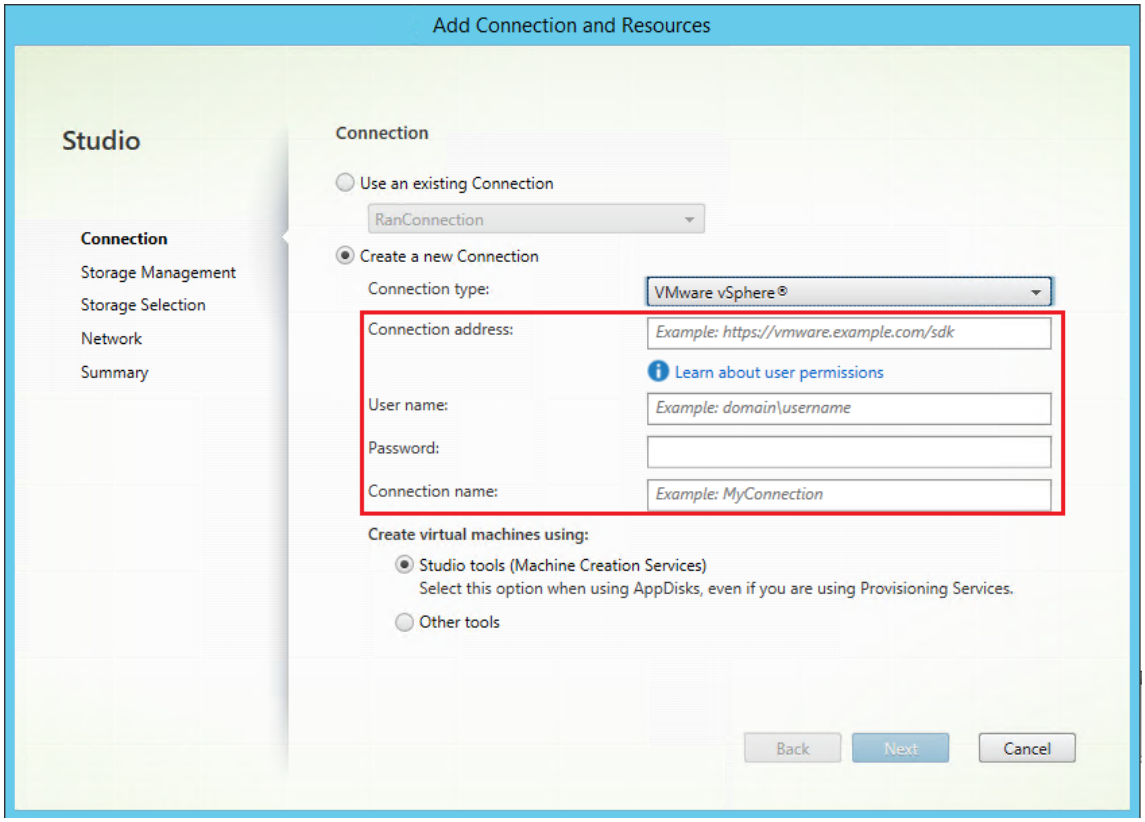
3. Choose VMware vSphere as the connection type.

   For example, in the on-premises Citrix Studio:



4. Type the connection address (the vCenter Server URL) of your VMware account, your credentials, and your connection name.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



Step 3: Prepare a master image

(For XenServer only) Step 3a: Install XenServer VM Tools Install XenServer VM Tools on the template VM for each VM to use the xe CLI or XenCenter. VM performance can be slow unless you install the tools. Without the tools, you can't do any of the following:

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

1. Download the XenServer VM Tools for Linux file from the XenServer Downloads page or the Citrix Hypervisor Downloads page based on the hypervisor version in use.
2. Copy the `LinuxGuestTools-xxx.tar.gz` file to your Linux VM or to a shared drive that the Linux VM can access.
3. Extract the contents of the tar file: `tar -xzf LinuxGuestTools-xxx.tar.gz`
4. Run the following command to install the `xe-guest-utilities` package based on your Linux distribution.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

For RHEL/CentOS/Rocky Linux/SUSE:

```
sudo rpm -i <extract-directory>/xe-guest-utilities_{
 package-version }
 _x86.64.rpm
```

For Ubuntu/Debian:

```
sudo dpkg -i <extract-directory>/xe-guest-utilities_{
 package-version }
 _amd64.deb
```

5. Check the virtualization state of the template VM on the **General** tab in XenCenter. If XenServer VM Tools are installed correctly, the virtualization state shows **Optimized**.

Step 3b: Verify configurations for SUSE 15.5 on AWS, Azure, and GCP For SUSE 15.5 on AWS, Azure, and GCP, ensure that:

- You are using **libstdc++6** version 12 or later.
- The **Default_WM** parameter in **/etc/sysconfig/windowmanager** is set to **"gnome"**.

Step 3c: Disable RDNS for Ubuntu 20.04 on GCP On the template VM, add the **rdns = false** line under **[libdefaults]** in /etc/krb5.conf.

Step 3d: Install .NET and the Linux VDA package on the template VM

> **Note:**
>
> To use a currently running VDA as the template VM, skip this step. To use a currently running RHEL 8.x/9.x or Rocky Linux 8.x/9.x VDA that is connected to the domain using SSSD as the template VM, ensure that:
>
> - The VDA is installed manually and not by using easy install. Easy install uses **Adcli** for RHEL 8.x/9.x and Rocky Linux 8.x/9.x and the combination of SSSD and **Adcli** is not supported by MCS.
> - A Samba server is configured to use SSSD for AD authentication. For more information, see the Red Hat article at https://access.redhat.com/solutions/3802321.

Before installing the Linux VDA package, install .NET on the template VM and notice the following:

- In addition to the .NET Runtime, you must install .ASP.NET Core Runtime on all supported Linux distributions before you install or upgrade the Linux VDA. Version 6 is required for Amazon Linux 2. Version 8 is required for other distributions.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- If your Linux distribution contains the .NET version that you require, install it from the built-in feed. Otherwise, install .NET from the Microsoft package feed. For more information, see https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers.

After installing .NET, run the following commands based on your Linux distribution to install the Linux VDA:

For RHEL/CentOS/Rocky Linux:

> **Note:**
>
> Before installing the Linux VDA on RHEL 9.4/9.3/9.2 and Rocky Linux 9.4/9.3/9.2, update the **libsepol** package to version 3.4 or later.

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

For Ubuntu/Debian:

```
sudo dpkg –i  <PATH>/<Linux VDA DEB>

apt-get install -f
```

For SUSE:

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

(For RHEL 7 only) Step 3e: Enable repositories to install the tdb-tools package **For RHEL 7 server:**

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

For RHEL 7 workstation:

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

Step 3f: (For RHEL only) Install the EPEL repository that can offer ntfs-3g Install the EPEL repository on RHEL 8. For information on how to install EPEL, see the instructions at https://docs.fedoraproject.org/en-US/epel/.

Step 3g: (For SUSE only) Manually install ntfs-3g On the SUSE platform, no repository provides ntfs-3g. Download the source code, compile, and install ntfs-3g manually:

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

2. Download the ntfs-3g package.
3. Decompress the ntfs-3g package:

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

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

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

5. Install ntfs-3g:

```
./configure
make
make install
```

Step 3h: Specify a database to use You can switch between SQLite and PostgreSQL after installing the Linux VDA package. To do so, complete the following steps:

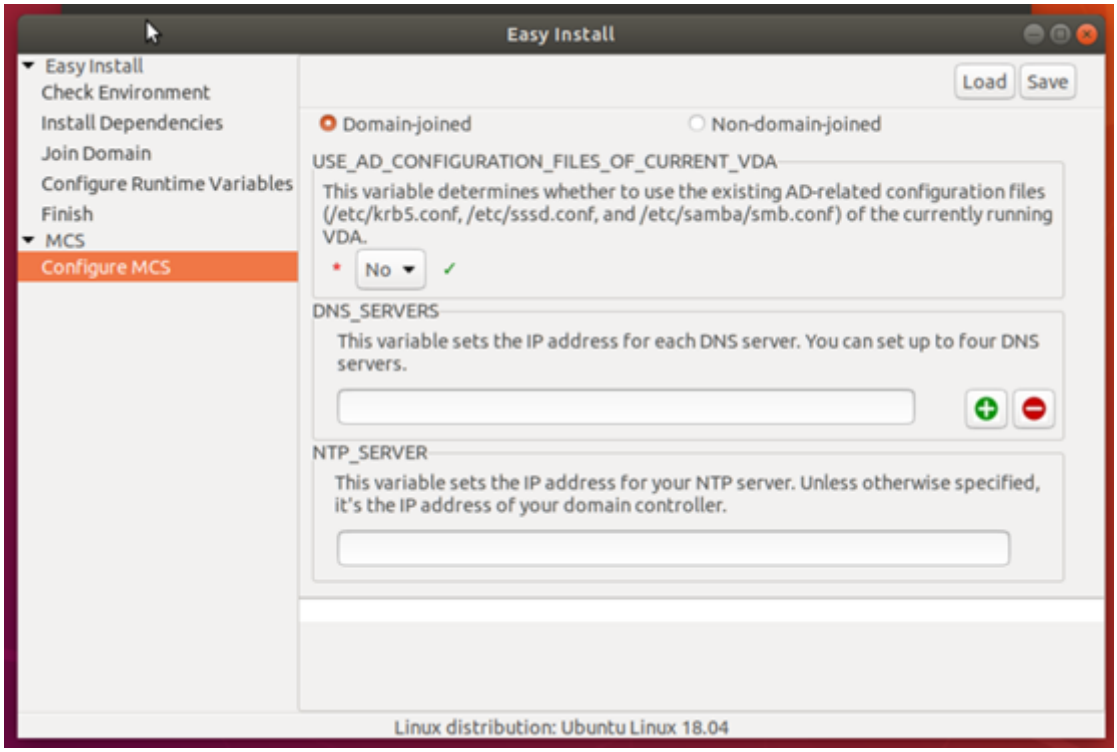
> **Note:**
>
> - We recommend you use SQLite for VDI mode only and use PostgreSQL for a hosted shared desktops delivery model.
> - For easy install and MCS, you can specify SQLite or PostgreSQL to use without having to install them manually. Unless otherwise specified through **/etc/xdl/db.conf**, the Linux VDA uses PostgreSQL by default.
> - You can also use **/etc/xdl/db.conf** to configure the port number for PostgreSQL.

1. Run `/opt/Citrix/VDA/sbin/ctxcleanup.sh`. Omit this step if it is a fresh installation.
2. Edit `/etc/xdl/db.conf` before running `deploymcs.sh`.

Step 3i: Configure MCS variables There are two ways to configure MCS variables:

- Edit the `/etc/xdl/mcs/mcs.conf` file.
- Use the easy install GUI. To open the easy install GUI, run the `/opt/Citrix/VDA/bin/easyinstall` command in the desktop environment of your Linux VDA.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



> **Tip:**
>
> Click **Save** to save variable settings to a local file under the path you specify. Click **Load** to load variable settings from a file that you specify.

The following are MCS variables that you can configure for non-domain-joined and domain-joined scenarios:

- **For non-domain-joined scenarios**

  You can use the default variable values or customize the variables as required (optional):

  `DOTNET_RUNTIME_PATH`=path-to-install-dotnet-runtime
  `DESKTOP_ENVIRONMENT`=gnome | mate
  `REGISTER_SERVICE`=Y | N
  `ADD_FIREWALL_RULES`=Y | N
  `VDI_MODE`=Y | N
  `START_SERVICE`=Y | N

- **For domain-joined scenarios**

  - `Use_AD_Configuration_Files_Of_Current_VDA`: Determines whether to use the existing AD-related configuration files (/etc/krb5.conf, /etc/sssd.conf, and /etc/samba/smb.conf) of the currently running VDA. If set to Y, the configuration files on MCS-created machines are the same as the equivalents on the currently running VDA. However, you still must configure the `dns` and `AD_INTEGRATION` variables. The default

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

value is N, which means the configuration templates on the master image determine the configuration files on MCS-created machines. To use a currently running VDA as the template VM, set the value to Y. Otherwise, set it to N.

- `dns`: Sets the IP address for each DNS server. You can set up to four DNS servers.
- `NTP_SERVER`: Sets the IP address for your NTP server. Unless otherwise specified, it's the IP address of your domain controller.
- `WORKGROUP`: Sets the workgroup name to the NetBIOS name (case-sensitive) that you configured in AD. Otherwise, MCS uses the part of the domain name that immediately follows the machine hostname as the workgroup name. For example, if the machine account is **user1.lvda.citrix.com**, MCS uses **lvda** as the workgroup name while **citrix** is the correct choice. Ensure that you set the workgroup name correctly.
- `AD_INTEGRATION`: Sets SSSD, Winbind, PBIS, or Centrify. For a matrix of the Linux distributions and domain joining methods that MSC supports, see Supported distributions in this article.
- `CENTRIFY_DOWNLOAD_PATH`: Sets the path for downloading the Server Suite Free (formerly Centrify Express) package. The value takes effect only when you set the `AD_INTEGRATION` variable to Centrify.
- `CENTRIFY_SAMBA_DOWNLOAD_PATH`: Sets the path for downloading the Centrify Samba package. The value takes effect only when you set the `AD_INTEGRATION` variable to Centrify.
- `PBIS_DOWNLOAD_PATH`: Sets the path for downloading the PBIS package. The value takes effect only when you set the `AD_INTEGRATION` variable to PBIS.
- `UPDATE_MACHINE_PW`: Enables or disables automating machine account password updates. For more information, see Automate machine account password updates.
- Linux VDA configuration variables:

  `DOTNET_RUNTIME_PATH`=path-to-install-dotnet-runtime
  `DESKTOP_ENVIRONMENT`=gnome | mate
  `SUPPORT_DDC_AS_CNAME`=Y | N
  `VDA_PORT`=port-number
  `REGISTER_SERVICE`=Y | N
  `ADD_FIREWALL_RULES`=Y | N
  `HDX_3D_PRO`=Y | N
  `VDI_MODE`=Y | N
  `SITE_NAME`=dns-site-name | '<none>'
  `LDAP_LIST`='list-ldap-servers'| '<none>'
  `SEARCH_BASE`=search-base-set | '<none>'

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
FAS_LIST='list-fas-servers'| '<none>'
START_SERVICE=Y | N
TELEMETRY_SOCKET_PORT=port-number
TELEMETRY_PORT=port-number
```

(Optional) Step 3j: Write or update registry values for MCS On the template machine, add command lines to the `/etc/xdl/mcs/mcs_local_setting.reg` file for writing or updating registry values as required. This action prevents the loss of data and settings every time an MCS-provisioned machine restarts.

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

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

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

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

> **Note**
>
> To modify settings for MCS, you are allowed to edit files under /etc/xdl/ad_join and /etc/xdl/mcs/, but editing any files under /var/xdl/mcs is prohibited.

Step 3j: Create a master image

1. (For SSSD + RHEL 8.x/9.x or Rocky Linux 8.x/9.x only) Run the `update-crypto-policies --set DEFAULT:AD-SUPPORT` command and then restart the template VM.
2. If you configure MCS variables by editing `/etc/xdl/mcs/mcs.conf`, run `/opt/Citrix/VDA/sbin/deploymcs.sh`. If you configure MCS variables by using the GUI, click **Deploy**. After you click **Deploy** on the GUI, the variables you set on the GUI override the variables you set in the `/etc/xdl/mcs/mcs.conf` file.
3. (If you are using a currently running VDA as the template VM or if it is a non-domain-joined scenario, skip this step.) On the template VM, update the configuration templates to customize the relevant `/etc/krb5.conf`, `/etc/samba/smb.conf`, and `/etc/sssd/sssd.conf` files on all created VMs.

   For Winbind users, update the `/etc/xdl/ad_join/winbind_krb5.conf.tmpl` and `/etc/xdl/ad_join/winbind_smb.conf.tmpl` templates.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

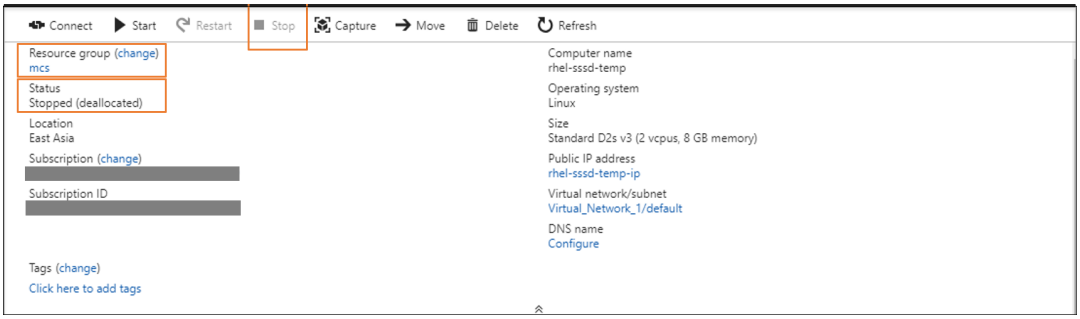
For SSSD users, update the `/etc/xdl/ad_join/sssd.conf.tmpl`, `/etc/xdl/ad_join/sssd_krb5.conf.tmpl`, and `/etc/xdl/ad_join/sssd_smb.conf.tmpl` templates.

For Centrify users, update the `/etc/xdl/ad_join/centrify_krb5.conf.tmpl` and `/etc/xdl/ad_join/centrify_smb.conf.tmpl` templates.

> **Note:**
>
> Keep the existing format used in the template files and use variables such as $WORKGROUP, $REALM, $realm, ${new_hostname}, and $AD_FQDN.

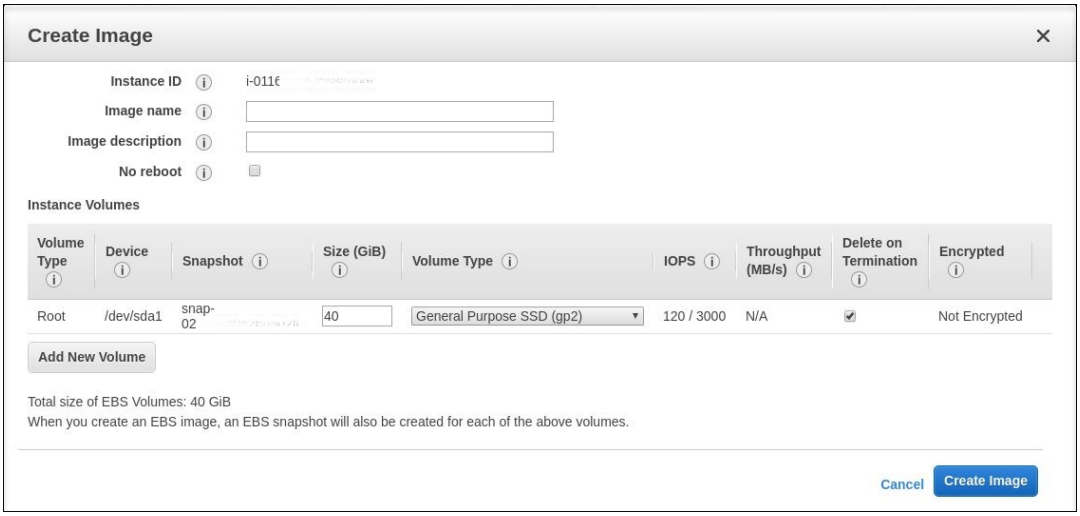
4. Create and name a snapshot of your master image based on the public cloud you use.

   - **(For XenServer, GCP, and VMware vSphere)** Install applications on the template VM and shut down the template VM. Create and name a snapshot of your master image.
   - **(For Azure)** Install applications on the template VM and shut down the template VM from the Azure portal. Ensure that the power status of the template VM is **Stopped (deallocated)**. Remember the name of the resource group here. You need the name to locate your master image on Azure.

   - **(For AWS)** Install applications on the template VM and shut down the template VM from the AWS EC2 portal. Ensure that the instance state of the template VM is **Stopped**. Right-click the template VM and select **Image > Create Image**. Type information and make settings as needed. Click **Create Image**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



- **(For Nutanix)** On Nutanix AHV, shut down the template VM. Create and name a snapshot of your master image.

  > **Note:**
  >
  > You must prefix Acropolis snapshot names with `XD_` for use in Citrix Virtual Apps and Desktops. Use the Acropolis console to rename your snapshots when needed. After you rename a snapshot, restart the **Create Catalog** wizard to obtain a refreshed list.

(For GCP) Step 3l: Configure Ethernet connection on RHEL 8.x/9.x and Rocky Linux 8.x/9.x After you install the Linux VDA on RHEL 8.x/9.x and Rocky Linux 8.x/9.x hosted on GCP, the Ethernet connection might be lost and the Linux VDA might be unreachable after a VM restart. To work around the issue, set a root password when logging on to the VM for the first time and make sure that you can log on to the VM as root. Then, run the following commands in the console after restarting the VM:

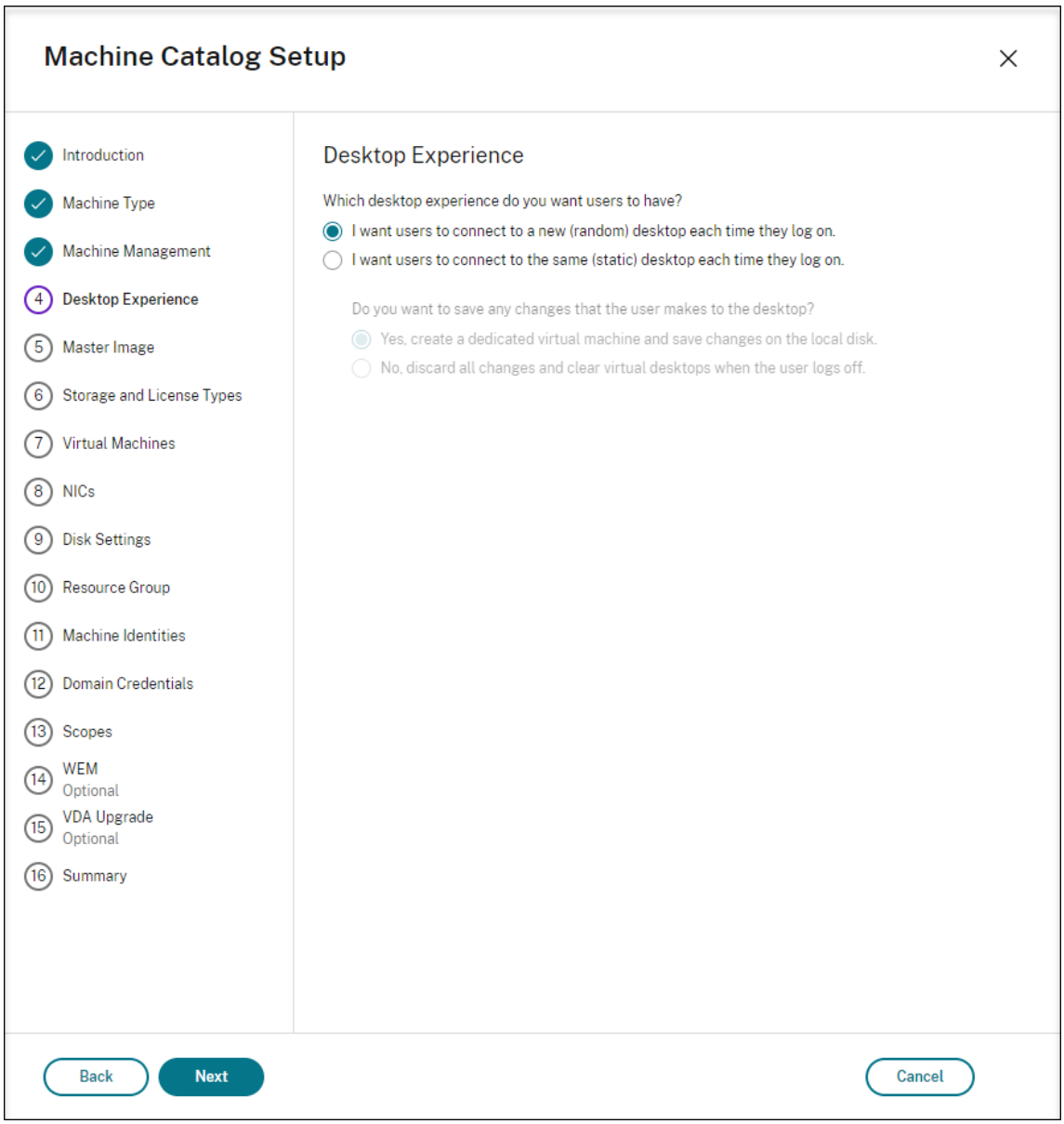
```
nmcli dev connect eth0
systemctl restart NetworkManager
```

Step 4: Create a machine catalog

In Citrix Studio or Web Studio, create a machine catalog and specify the number of VMs to create in the catalog. When creating the machine catalog, choose your master image and consider the following:

- On the **Container** page that is unique to Nutanix, select the container that you specified for the template VM earlier.
- When you create a catalog containing **single-session OS** machines, the **Desktop Experience** page appears and it lets you determine what occurs each time a user logs on.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



On the **Desktop Experience** page, select one of:

- Users connect to a new (random) desktop each time they log on.
- Users connect to the same (static) desktop each time they log on.

If you choose the first option, the changes that users make to the desktop will be discarded (non-persistent).

If you choose the second option and are using MCS to provision the machines, you can configure how user changes to the desktop are handled:

- Save user changes to the desktop on the local disk (persistent).
- Discard user changes and clear the virtual desktop when the user logs off (non-persistent). Select this option if you are using the user personalization layer.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- When updating the master image for an MCS catalog containing persistent machines, any new machines added to the catalog use the updated image. Pre-existing machines continue to use the original master image.

For more information, see machine catalog creation in the Citrix Virtual Apps and Desktops documentation and the Citrix DaaS documentation.

> **Note:**
>
> For Nutanix environments, if your machine catalog creation process on the Delivery Controller takes a significant amount of time, go to Nutanix Prism and power on the machine prefixed with **Preparation** manually. This approach helps to continue the creation process.

Step 5: Create a delivery group

A delivery group is a collection of machines selected from one or more machine catalogs. It specifies which users can use those machines, and the applications and desktops available to those users.

For more information, see delivery group creation in the Citrix Virtual Apps and Desktops documentation and the Citrix DaaS documentation.

> **Note:**
>
> The VMs you create using MCS might not be able to register with Citrix Cloud Connectors and show as **Unregistered**. The issue occurs when you host the VMs on Azure and join in the AD domain with Samba Winbind. To work around the issue, complete the following steps:
>
> 1. Go to the ADSI Edit console, select an unregistered VM, and edit the **msDS-SupportedEncryptionTypes** attribute of its machine account.
> 2. Restart the **ctxjproxy** and **ctxvda** services on the VM. If the VM's status changes to **Registered**, continue with steps 3 through 5.
> 3. Open the **/var/xdl/mcs/ad_join.sh** file on the template VM.
> 4. Add a line of **net ads enctypes set $NEW_HOSTNAME$ <Decimal value of encryption type attribute, for example, 28> -U $NEW_HOSTNAME$ -P password** after the following lines inside the **/var/xdl/mcs/ad_join.sh** file:
>
>    ```
>    if [ "$AD_INTEGRATION" == "winbind" ]; then
>                    join_domain_samba
>                    restart_service winbind /usr/bin/systemctl
>    ```
>
> 5. Take a new snapshot and create VMs using the new template.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Use MCS to upgrade your Linux VDA

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

1. Ensure that you installed .NET before you upgrade your Linux VDA to the current release.

   - Install .NET Runtime 8.0 on all supported Linux distributions except Amazon Linux 2.
   - For Amazon Linux 2, continue to install .NET Runtime 6.0.

   If your Linux distribution contains the .NET version that you require, install it from the built-in feed. Otherwise, install .NET from the Microsoft package feed. For more information, see https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers.

2. Upgrade your Linux VDA on the template machine:

   Note:

   - You can also use the Linux VDA self-update through Azure feature to schedule automatic software updates. To achieve this goal, add command lines to the etc/xdl/mcs/mcs_local_setting.reg file on the template machine. For example, you can add the following command lines:

     ```
     create -k "HKLM\System\CurrentControlSet\Control\Citrix\SelfUpdate" -t "REG_DWORD" -v "fEnabled" -d "0x00000001" --force

     create -k "HKLM\System\CurrentControlSet\Control\Citrix\SelfUpdate" -t "REG_SZ" -v "ScheduledTime" -d "Immediately" --force

     create -k "HKLM\System\CurrentControlSet\Control\Citrix\SelfUpdate" -t "REG_SZ" -v "Url" -d "<Your-Azure-Container-Url>" –force

     create -k "HKLM\System\CurrentControlSet\Control\Citrix\SelfUpdate" -t "REG_SZ" -v "CaCertificate" -d "<Local-Certificate-Path-of-PortalAzureCom>" --force
     ```

   - Starting with the 2407 release, the Linux VDA delegates package managers **rpm** or **dpkg** to handle configuration files during upgrades. The following describes how rpm and dpkg interact with changes to configuration files:

     – **rpm**: by default keeps the local version and saves the new version from the package with a .rpmnew extension.

     – **dpkg**: interactively prompts you with a choice on how to proceed. To silently upgrade the Linux VDA while retaining your local configuration file and saving the new package version as **.dpkg-new** or **.dpkg-dist**, use the following command:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
dpkg --force-confold -i package.deb  # Always keep your version, then save new package's version as *.dpkg-new or *.dpkg-dist
```

For RHEL and Rocky Linux distributions:

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

> **Note:**
>
> Before upgrading the Linux VDA on RHEL 9.4/9.3/9.2 and Rocky Linux 9.4/9.3/9.2, update the **libsepol** package to version 3.4 or later.

For SUSE distributions:

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

For Ubuntu/Debian distributions:

```
sudo dpkg -i <PATH>/<Linux VDA deb>

sudo apt-get install -f
```

3. Edit /etc/xdl/mcs/mcs.conf and /etc/xdl/mcs/mcs_local_setting.reg.
4. Take a new snapshot.
5. In Citrix Studio, select the new snapshot to update your machine catalog. Wait before each machine restarts. Do not restart a machine manually.

Automate machine account password updates

Machine account passwords, by default, expire 30 days after the machine catalog is created. To prevent password expiration and to automate machine account password updates, do the following:

1. Add the following entry to /etc/xdl/mcs/mcs.conf before running /opt/Citrix/VDA/sbin/deploymcs.sh.

   ```
   UPDATE_MACHINE_PW="Y"
   ```

2. After running /opt/Citrix/VDA/sbin/deploymcs.sh, open /etc/cron.d/mcs_update_password_cronjob to set the update time and frequency. The default setting updates machine account passwords weekly at 2:30AM, Sunday.

After each machine account password update, the ticket cache on the Delivery Controller becomes invalid and the following error might appear in /var/log/xdl/jproxy.log:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
[ERROR] - AgentKerberosServiceAction.Run: GSSException occurred.
 Error: Failure unspecified at GSS-API level (Mechanism level:
Checksum failed)
```

To eliminate the error, clear the ticket cache regularly. You can schedule a cache cleanup task on all Delivery Controllers or on the domain controller.

Enable FAS on MCS-created VMs

You can enable FAS on MCS-created VMs that run on the following distributions:

	Winbind	SSSD	Centrify	PBIS
RHEL 9.4/9.3/9.2	Yes	No	No	No
RHEL 8.x	Yes	No	No	Yes
Rocky Linux 9.4/9.3/9.2	Yes	No	No	No
Rocky Linux 8.x	Yes	No	No	No
Ubuntu 22.04, Ubuntu 20.04	Yes	No	No	No
Debian 12.5/11.9	Yes	Yes	No	Yes
SUSE 15.5	Yes	No	No	No

Enable FAS when you are preparing a master image on the template VM

1. Import the root CA certificate.

   ```
   1 sudo cp root.pem /etc/pki/CA/certs/
   ```

2. Run the easy install script (ctxinstall.sh) and make sure that the template VM is joined to a domain properly.
3. Run ctxfascfg.sh.
4. Set variables in `/etc/xdl/mcs/mcs.conf`.

   > **Note:**
   >
   > Set all necessary variables in `/etc/xdl/mcs/mcs.conf` because these variables are called upon VM startup.

   a) Set the value of `Use_AD_Configuration_Files_Of_Current_VDA` to Y.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

b) Set the other variables as required, such as `VDI_MODE`.

5. Add the following command line to the **/etc/xdl/mcs/mcs_local_setting.reg** file for setting your FAS server addresses:

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\
    VirtualDesktopAgent\Authentication\UserCredentialService" -t "
    REG_SZ" -v "Addresses" -d "<Your-FAS-Server-List>" --force
```

6. Run the script `/opt/Citrix/VDA/sbin/deploymcs.sh`.

Enable FAS on an MCS-created VM

If FAS is not enabled on the template machine as described earlier, you can enable FAS on each MCS-created VM.

To enable FAS on an MCS-created VM, do the following:

1. Set variables in /etc/xdl/mcs/mcs.conf.

> **Note:**
>
> Set all necessary variables in `/etc/xdl/mcs/mcs.conf` because these variables are called upon VM startup.

a) Set the value of `Use_AD_Configuration_Files_Of_Current_VDA` to Y.
b) Set the other variables as required, such as `VDI_MODE`.

2. Add the following command line to the **/etc/xdl/mcs/mcs_local_setting.reg** file for setting your FAS server addresses:

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\
    VirtualDesktopAgent\Authentication\UserCredentialService" -t "
    REG_SZ" -v "Addresses" -d "<Your-FAS-Server-List>" --force
```

3. Import the root CA certificate.

```
sudo cp root.pem /etc/pki/CA/certs/
```

4. Run the `/opt/Citrix/VDA/sbin/ctxfascfg.sh` script.

Create Linux VDAs using Citrix Provisioning

June 20, 2024

You can create domain-joined VDAs using Citrix Provisioning.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

This article provides information about streaming Linux target devices. Using this feature, you can provision Linux virtual desktops directly in the Citrix Virtual Apps and Desktops environment.

The following Linux distributions are supported:

- Ubuntu 22.04
- Ubuntu 20.04
- RHEL 9.2
- RHEL 8.8
- Rocky Linux 9.2
- Rocky Linux 8.8
- SUSE 15.5

> **Important:**
>
> - We recommend that you use the most recent installation package of Citrix Provisioning. Use the package based on your Linux distribution. Citrix Provisioning Server 2109 or later is required to use Linux streaming agent 2109 and later.
> - When using Citrix Provisioning to stream Linux target devices, create a separate boot partition on the single shared-disk image so that the provisioned devices can boot as expected.
> - Avoid formatting any partition with **btrfs**. GRUB2 has an intrinsic problem finding **btrfs** partitions. **GRUB** stands for **GRand Unified Bootloader**.

For more information, see Streaming Linux target devices in the Citrix Provisioning documentation.

Create Linux VDAs in Citrix DaaS Standard for Azure

June 20, 2024

You can create both domain-joined and non-domain-joined Linux VDAs in Citrix DaaS Standard for Azure (formerly Citrix Virtual Apps and Desktops Standard for Azure) to deliver virtual apps and desktops to any device from Microsoft Azure. For more information, see Citrix DaaS Standard for Azure.

Supported Linux distributions

The following Linux distributions support this feature:

- RHEL 9.4
- RHEL 9.3
- RHEL 9.2

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- RHEL 8.10
- RHEL 8.9
- RHEL 8.8
- Rocky Linux 9.4
- Rocky Linux 9.3
- Rocky Linux 9.2
- Rocky Linux 8.10
- Rocky Linux 8.9
- Rocky Linux 8.8
- SUSE 15.5
- Ubuntu 22.04
- Ubuntu 20.04

Step 1: Prepare a master image in Azure

> **Note:**
>
> You can also use the Linux VDA self-update feature to schedule automatic software updates. To achieve this goal, add command lines to the etc/xdl/mcs/mcs_local_setting.reg file on the master image.
> For example, you can add the following command lines:

```
create -k "HKLM\System\CurrentControlSet\Control\Citrix\SelfUpdate"
    -t "REG_DWORD" -v "fEnabled" -d "0x00000001" --force

create -k "HKLM\System\CurrentControlSet\Control\Citrix\SelfUpdate"
    -t "REG_SZ" -v "ScheduledTime" -d "Immediately" --force

create -k "HKLM\System\CurrentControlSet\Control\Citrix\SelfUpdate"
    -t "REG_SZ" -v "Url" -d "<Your-Azure-Container-Url>" --force

create -k "HKLM\System\CurrentControlSet\Control\Citrix\SelfUpdate"
    -t "REG_SZ" -v "CaCertificate" -d "<Local-Certificate-Path-of-
    PortalAzureCom>" --force
```

1. In Azure, create a Linux VM of a supported distribution.
2. Install a desktop environment on the Linux VM if necessary.
3. On the VM, install .NET based on your Linux distribution.

   In addition to the .NET Runtime, you must install .ASP.NET Core Runtime on all supported Linux distributions before you install or upgrade the Linux VDA. Version 6 is required for Amazon Linux 2. Version 8 is required for other distributions.

   If your Linux distribution contains the .NET version that you require, install it from the built-in feed. Otherwise, install .NET from the Microsoft package feed. For more information, see

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers.

4. (For Ubuntu only) Add the `source /etc/network/interfaces.d/*` line to the `/etc/network/interfaces` file.
5. (For Ubuntu only) Point `/etc/resolv.conf` to `/run/systemd/resolve/resolv.conf` instead of pointing it to `/run/systemd/resolve/stub-resolv.conf`:

```
unlink /etc/resolv.conf

ln -s /run/systemd/resolve/resolv.conf /etc/resolv.conf
```

6. Install the Linux VDA package.
7. Specify a database to use.

   As an experimental feature, you can use SQLite in addition to PostgreSQL. You can also switch between SQLite and PostgreSQL after installing the Linux VDA package. To do so, complete the following steps:

   a) Run `/opt/Citrix/VDA/sbin/ctxcleanup.sh`. Omit this step if it is a fresh installation.

   b) Edit `/etc/xdl/db.conf` before running `deploymcs.sh`.

   > **Note:**
   >
   > - We recommend you use SQLite for VDI mode only.
   > - For easy install and MCS, you can switch between SQLite and PostgreSQL without having to install them manually. Unless otherwise specified through `/etc/xdl/db.conf`, the Linux VDA uses PostgreSQL by default.
   > - You can also use `/etc/xdl/db.conf` to configure the port number for PostgreSQL.

8. Change MCS variables.

   There are two ways to configure MCS variables:

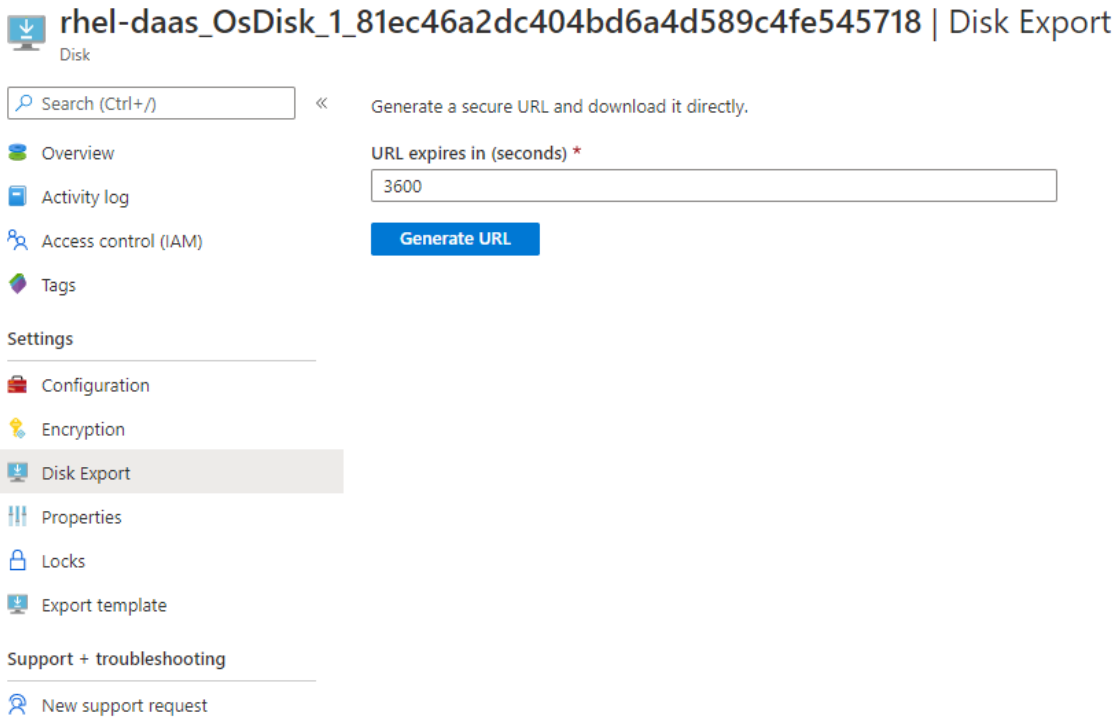
   - Edit the `/etc/xdl/mcs/mcs.conf` file.
   - Use the easy install GUI. To open the easy install GUI, run the `/opt/Citrix/VDA/bin/easyinstall` command in the desktop environment of your Linux VDA.

   > **Note:**
   >
   > Leave the `dns` variable unspecified.
   > If you select the **Static** or **Random** type when creating a machine catalog, set `VDI_MODE=Y`.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

If you configure MCS variables by editing `/etc/xdl/mcs/mcs.conf`, run `/opt/Citrix/VDA/sbin/deploymcs.sh`. If you configure MCS variables by using the GUI, click **Deploy**. After you click **Deploy** on the GUI, the variables you set on the GUI override the variables you set in the `/etc/xdl/mcs/mcs.conf` file.

9. In Azure, stop (or deallocate) the VM. Click **Disk Export** to generate a SAS URL for the Virtual Hard Disk (VHD) file that you can use as a master image to create other VMs.



10. (Optional) Make group policy settings on the master image. You can use the `ctxreg` tool to make group policy settings. For example, the following command enables the **Auto-create PDF Universal Printer** policy for PDF printing.

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\
   GroupPolicy\Defaults\PrintingPolicies" -t "REG_DWORD" -v "
   AutoCreatePDFPrinter" -d "0x00000001" --force
```

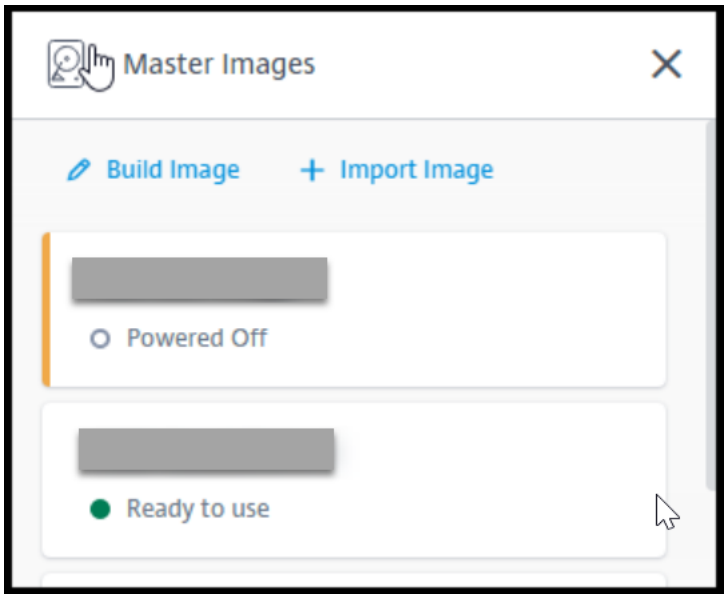
Step 2: Import the master image from Azure

1. From the Manage dashboard, expand **Master Images** on the right. The display lists the master images that Citrix provides, and images that you created and imported.

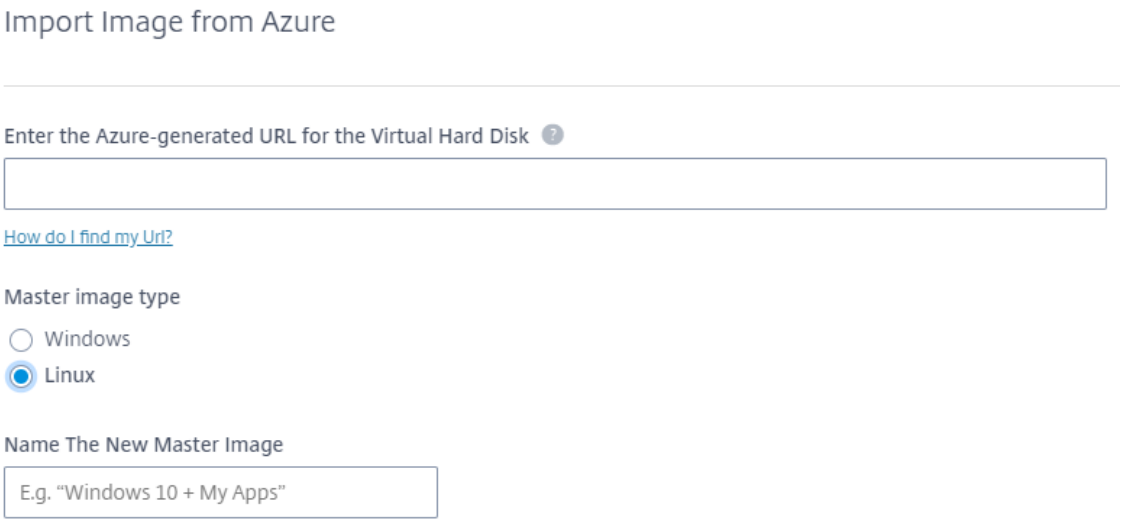
> **Tip:**
>
> Most of the administrator activities for this service are managed through the **Manage** and

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Monitor** dashboards. After you create your first catalog, the **Manage** dashboard launches automatically after you sign in to Citrix Cloud and select the **Managed Desktops** service.



2. Click **Import Image**.
3. Enter the SAS URL for the VHD file that you generated in Azure. Select **Linux** for the master image type.



4. Follow the instructions in the wizard to complete importing the master image.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 3: Create a machine catalog

Access the **Manage** dashboard and click **Create Catalog**. When creating the Machine Catalog, choose the master image you created earlier.

> **Note:**
>
> The VM used as a master image is not accessible through SSH or RDP. To access the VM, use the Serial Console in the Azure portal.

Install the Linux VDA manually

June 20, 2024

You can install the Linux VDA on the following Linux distributions manually:

- Amazon Linux 2, CentOS, RHEL, and Rocky Linux
- SUSE
- Ubuntu
- Debian

Install the Linux VDA on Amazon Linux 2, RHEL, and Rocky Linux manually

August 2, 2024

> **Important:**
>
> For fresh installations, we recommend you use easy install for a quick installation. Easy install saves time and labor and is less error-prone than the manual installation detailed in this article.

Step 1: Prepare configuration information and the Linux machine

Step 1a: Verify the network configuration

Make sure that the network is connected and configured correctly. For example, you must configure the DNS server on the Linux VDA.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 1b: Set the host name

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

```
hostname
```

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

To make sure that the DNS domain name and Fully Qualified Domain Name (FQDN) of the machine are reported back correctly, change the following line of the `/etc/hosts` file to include the FQDN and host name as the first two entries:

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

For example:

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

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

> **Note:**
>
> The Linux VDA currently does not support NetBIOS name truncation. The host name must not exceed 15 characters.
>
> **Tip:**
>
> Use a–z, A–Z, 0–9, and hyphen (-) characters only. Avoid underscores (_), spaces, and other symbols. Do not start a host name with a number and do not end with a hyphen. This rule also applies to Delivery Controller host names.

Step 1d: Check the host name

Verify that the host name is set correctly:

```
1 hostname
```

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

Verify that the FQDN is set correctly:

```
1 hostname -f
```

This command returns the FQDN of the machine.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 1e: Check name resolution and service reachability

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

```
nslookup domain-controller-fqdn

ping domain-controller-fqdn

nslookup delivery-controller-fqdn

ping delivery-controller-fqdn
```

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 (VM) can cause clock skew problems. For this reason, synchronizing time with a remote time service is preferred.

An RHEL default environment uses the Chrony daemon (`chronyd`) for clock synchronization.

Configure the Chrony service As a root user, edit **/etc/chrony.conf** and add a server entry for each remote time server:

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

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

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

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

Save changes and restart the Chrony daemon:

```
sudo systemctl restart chronyd
```

Step 1g: Install OpenJDK 11

The Linux VDA requires the presence of OpenJDK 11.

- If you are using RHEL, OpenJDK 11 is automatically installed as a dependency when you install the Linux VDA.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- If you are using Amazon Linux 2 or Rocky Linux, run the following command to enable and install OpenJDK 11:

```
amazon-linux-extras install java-openjdk11
```

Confirm the correct version:

```
sudo yum info java-11-openjdk
```

The prepackaged OpenJDK might be an earlier version. Update to OpenJDK 11:

```
sudo yum -y update java-11-openjdk
```

Step 1h: Install and specify a database to use

> **Note:**
>
> - We recommend you use SQLite for VDI mode only and use PostgreSQL for a hosted shared desktops delivery model.
> - For easy install and MCS, you can specify SQLite or PostgreSQL to use without having to install them manually. Unless otherwise specified through **/etc/xdl/db.conf**, the Linux VDA uses PostgreSQL by default.
> - For manual installations, you must install SQLite, PostgreSQL, or both manually. If you install both SQLite and PostgreSQL, you can specify one of them to use by editing **/etc/xdl/db.conf** after installing the Linux VDA package.

This section describes how to install PostgreSQL and SQLite and how to specify either of them to use.

Install PostgreSQL The Linux VDA requires PostgreSQL:

- PostgreSQL 9 for Amazon Linux 2
- PostgreSQL 10 for RHEL 8.x and Rocky Linux 8.x
- PostgreSQL 13 for RHEL 9.4/9.3/9.2 and Rocky Linux 9.4/9.3/9.2

Run the following commands to install PostgreSQL:

```
sudo yum -y install postgresql-server

sudo yum -y install postgresql-jdbc
```

For RHEL 8.x and RHEL 9.4/9.3/9.2, run the following command to install `libpq` for PostgreSQL:

```
sudo yum -y install libpq
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Run the following command to initialize the database. This action creates database files under **/var/lib/pgsql/data**.

```
sudo postgresql-setup initdb
```

Run the following commands to start PostgreSQL upon machine startup or immediately, respectively:

```
sudo systemctl enable postgresql

sudo systemctl start postgresql
```

Check the version of PostgreSQL by using:

```
psql --version
```

(For Amazon Linux 2 only) Verify that the data directory is set by using the **psql** command-line utility:

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

Install SQLite Run the following command to install SQLite:

```
sudo yum -y install sqlite
```

Specify a database to use If you install both SQLite and PostgreSQL, you can specify one of them to use by editing **/etc/xdl/db.conf** after installing the Linux VDA package.

1. Run **/opt/Citrix/VDA/sbin/ctxcleanup.sh**. Omit this step if it is a fresh installation.
2. Edit **/etc/xdl/db.conf** to specify a database to use.
3. Run **ctxsetup.sh**.

> **Note:**
>
> You can also use **/etc/xdl/db.conf** to configure the port number for PostgreSQL.

Step 2: Prepare the hypervisor

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

Fix time synchronization on XenServer (formerly Citrix Hypervisor)

When the XenServer Time Sync feature is enabled, within each paravirtualized Linux VM you experience issues with NTP and XenServer. Both try to manage the system clock. To avoid the clock becom-

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

ing 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.

If you are running a paravirtualized Linux kernel with XenServer VM 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 restart, edit the **/etc/sysctl.conf** file and add the line:

```
xen.independent_wallclock = 1
```

To verify these changes, restart the system:

```
su -

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

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. Make sure that **Time synchronization** is selected.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> This approach is different from VMware and XenServer (formerly Citrix Hypervisor), 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 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 VM to the Windows domain

The following methods are available for adding Linux machines to the Active Directory (AD) domain:

- Samba Winbind
- Quest Authentication Services
- Centrify DirectControl
- SSSD
- PBIS

Follow instructions based on your chosen method.

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

Samba Winbind

For RHEL 9.4/9.3/9.2 and Rocky Linux 9.4/9.3/9.2, run the following command to prevent **pam_winbind** from changing the ownership of the root directory:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
usermod -d /nonexistent nobody
```

Install or update the required packages:

For RHEL 9.4/9.3/9.2/8.x and Rocky Linux 9.4/9.3/9.2/8.x:

```
sudo yum -y install samba-winbind samba-winbind-clients krb5-
    workstation oddjob-mkhomedir realmd authselect
```

For Amazon Linux 2:

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

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

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

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

1. Run the following command.

   For RHEL 9.4/9.3/9.2/8.x and Rocky Linux 9.4/9.3/9.2/8.x:

   ```
   sudo authselect select winbind with-mkhomedir --force
   ```

   For Amazon Linux 2:

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

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

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

   `--enablekrb5kdcdns --enablekrb5realmdns`

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

   ```
   kerberos method = secrets and keytab
   winbind refresh tickets = true
   winbind offline logon = no
   ```

3. (For RHEL 9.4/9.3/9.2/8.x and Rocky Linux 9.4/9.3/9.2/8.x only) Open **/etc/krb5.conf** and add entries under the `[libdefaults]`, `[realms]`, and `[domain_realm]` sections:

   Under the `[libdefaults]` section:

   ```
   default_ccache_name = FILE:/tmp/krb5cc_%{ uid }
   default_realm = REALM
   dns_lookup_kdc = true
   ```

   Under the `[realms]` section:

   ```
   REALM = {
   kdc = fqdn-of-domain-controller
   }
   ```

   Under the `[domain_realm]` section:

   ```
   realm = REALM
   .realm = REALM
   ```

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

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

To add a Linux VM to the Windows domain, run the following command:

```
1 sudo realm join -U user --client-software=winbind REALM
```

> **Tip:**
>
> For the Linux VMs running on Amazon Linux 2, you can also use the following command to add them to the Windows domain:
>
> ```
> 1 sudo net ads join REALM -U user
> ```

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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
```

Make sure that any leading semicolons from each setting are removed. These changes require restarting the Winbind daemon:

```
1 sudo systemctl restart winbind
```

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

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

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

For RHEL 9.4/9.3/9.2 and Rocky Linux 9.4/9.3/9.2, run the following commands to resolve the SELinux issue with Winbind:

```
1 ausearch -c 'winbindd' --raw | audit2allow -M my-winbindd -p /etc/
     selinux/targeted/policy/policy.*
2
3 semodule -X 300 -i my-winbindd.pp
```

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

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

```
1 sudo net ads testjoin
```

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

```
1 sudo net ads info
```

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

```
1 sudo klist -ke
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

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

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

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

```
sudo klist
```

Examine the account details of the machine using:

```
sudo net ads status
```

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

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

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

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

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

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

```
klist
```

Exit the session.

```
exit
```

A similar test can be performed by logging on to the Gnome or KDE console directly. Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Quest Authentication Services

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

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

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

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

Configure Quest on Linux VDA

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

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

```
SELINUX=permissive
```

This change requires a machine restart:

```
1 reboot
```

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

Configure VAS daemon Autorenewal of Kerberos tickets must be enabled and disconnected. Authentication (offline logon) must be disabled.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

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

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

Restart the Linux machine after domain joining.

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

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

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

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

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

```
ls /tmp/krb5cc_uid
```

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

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

Exit the session.

```
exit
```

A similar test can be performed by logging on to the Gnome or KDE console directly. Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

Centrify DirectControl

Join a 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 who has permissions to join computers to the Active Directory domain. The **domain-name** is the name of the domain to join the Linux machine to.

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

```
su -
adinfo
```

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

More comprehensive system and diagnostic information is available using:

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Test connectivity to the various Active Directory and Kerberos services.

```
adinfo --test
```

Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

SSSD

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

To set up SSSD on RHEL, do the following:

1. Join the domain and create a host keytab
2. Set up SSSD
3. Enable SSSD
4. Verify the Kerberos configuration
5. Verify user authentication

Join the domain and create a host keytab SSSD does not provide Active Directory client functions for joining the domain and managing the system keytab file. You can use **adcli**, **realmd**, or **Samba** instead.

This section describes the **Samba** approach for Amazon Linux 2 and the `adcli` approach for RHEL 8.x/9.x and Rocky Linux 8.x/9.x. For **realmd**, see the RHEL documentation. These steps must be followed before configuring SSSD.

- **Samba (Amazon Linux 2):**

  Install or update the required packages:

  ```
  sudo yum -y install krb5-workstation authconfig oddjob-mkhomedir
     samba-common-tools
  ```

  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
  ```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

> **Note:**
>
> Settings in this article are meant for the single-domain, single-forest model. Configure Kerberos based on your AD infrastructure.

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

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

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

```
kerberos method = secrets and keytab
winbind offline logon = no
```

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

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

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

- **Adcli (RHEL 9.4/9.3/9.2/8.x and Rocky Linux 9.4/9.3/9.2/8.x):**

  Install or update the required packages:

  ```
  1  sudo yum -y install samba-common samba-common-tools krb5-
       workstation authconfig oddjob-mkhomedir realmd oddjob
       authselect
  ```

  Configure the machine for **Samba** and Kerberos authentication:

  ```
  1  sudo authselect select sssd with-mkhomedir --force
  ```

  Open **/etc/krb5.conf** and add the entries under the [realms] and [domain_realm] sections.

  Under the [realms] section:

  ```
  REALM = {
  kdc = fqdn-of-domain-controller
  }
  ```

  Under the [domain_realm] section:

  ```
  realm = REALM
  .realm = REALM
  ```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

```
sudo realm join REALM -U user
```

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

Set up SSSD Setting up SSSD consists of the following steps:

- Install the **sssd-ad** package on the Linux VDA by running the `sudo yum -y install sssd` command.
- Make configuration changes to various files (for example, sssd.conf).
- Start the **sssd** service.

(RHEL 9.4/9.3/9.2/8.x and Rocky Linux 9.4/9.3/9.2/8.x only)
Open **/etc/sssd/sssd.conf** and add the following entries under the [domain/ad.example.com] section:

```
ad_gpo_access_control = permissive
full_name_format = %2$s\\%1$s
fallback_homedir = /home/%d/%u
# Kerberos settings
krb5_ccachedir = /tmp
krb5_ccname_template = FILE:%d/krb5cc_%U
```

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

Set the file ownership and permissions on sssd.conf:

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

Enable SSSD **For RHEL 9.4/9.3/9.2/8.x and Rocky Linux 9.4/9.3/9.2/8.x:**

Run the following commands to enable SSSD:

```
sudo systemctl restart sssd
sudo systemctl enable sssd.service
sudo chkconfig sssd on
```

For Amazon Linux 2:

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

sudo systemctl start sssd

sudo chkconfig sssd on
```

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

```
sudo klist -ke
```

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

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

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

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

```
sudo klist
```

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

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

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

The supported logon formats are:

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

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

```
sudo ssh localhost – l DOMAIN\\username

id -u
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

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

Verify that the tickets in the user's Kerberos credential cache are valid and not expired.

```
klist
```

Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

PBIS

Download the required PBIS package

```
wget https://github.com/BeyondTrust/pbis-open/releases/download/9.1.0/
    pbis-open-9.1.0.551.linux.x86_64.rpm.sh
```

Make the PBIS installation script executable

```
chmod +x pbis-open-9.1.0.551.linux.x86_64.rpm.sh
```

Run the PBIS installation script

```
sh pbis-open-9.1.0.551.linux.x86_64.rpm.sh
```

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:

```
/opt/pbis/bin/domainjoin-cli join domain-name user
```

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

Note: To set Bash as the default shell, run the **/opt/pbis/bin/config LoginShellTemplate/bin/bash** command.

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

```
/opt/pbis/bin/domainjoin-cli query
```

If the machine is joined to a domain, this command returns the information about the currently joined AD domain and OU. Otherwise, only the host name appears.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

```
ssh localhost -l domain\\user

id -u
```

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

```
ls /tmp/krb5cc_uid
```

Exit the session.

```
exit
```

Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

Step 4: Install .NET

In addition to the .NET Runtime, you must install .ASP.NET Core Runtime on all supported Linux distributions before you install or upgrade the Linux VDA. Version 6 is required for Amazon Linux 2. Version 8 is required for other distributions.

If your Linux distribution contains the .NET version that you require, install it from the built-in feed. Otherwise, install .NET from the Microsoft package feed. For more information, see https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers.

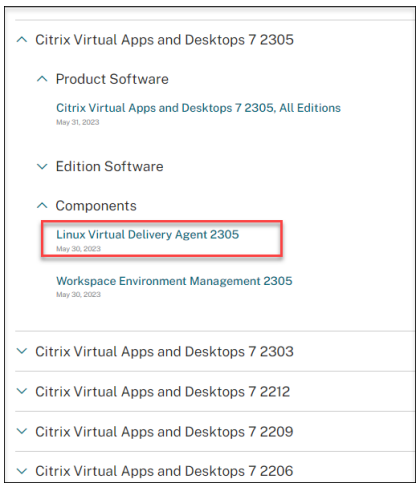
After installing .NET, run the **which dotnet** command to find your runtime path.

Based on the command output, set the .NET runtime binary path. For example, if the command output is /aa/bb/dotnet, use /aa/bb as the .NET binary path.

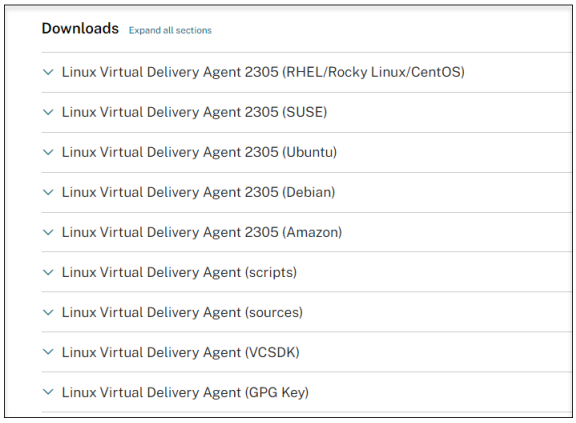
Step 5: Download the Linux VDA package

1. Go to the Citrix Virtual Apps and Desktops download page.
2. Expand the appropriate version of Citrix Virtual Apps and Desktops.
3. Expand **Components** to find the Linux VDA. For example:

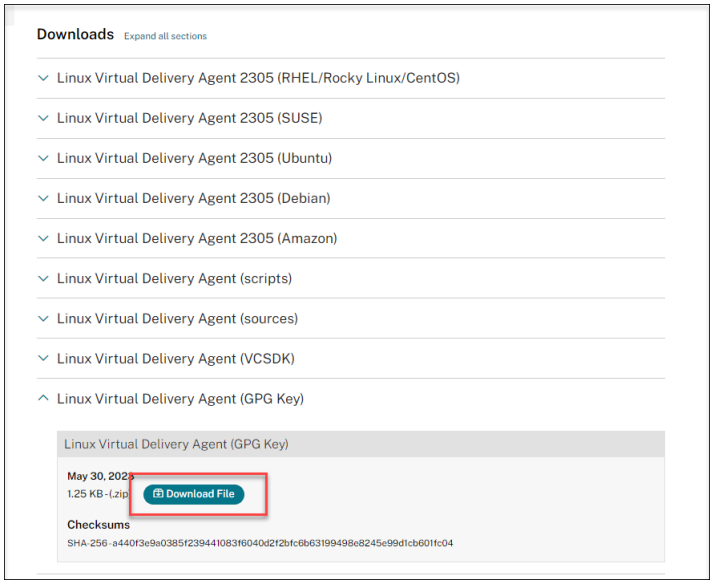
© 1999–2024 Cloud Software Group, Inc. All rights reserved.



4. Click the Linux VDA link to access the Linux VDA downloads.



5. Download the Linux VDA package that matches your Linux distribution.
6. Download the GPG public key that you can use to verify the integrity of the Linux VDA package. For example:



© 1999–2024 Cloud Software Group, Inc. All rights reserved.

To verify the integrity of the Linux VDA package, run the following commands to import the public key into the RPM database and to check the package integrity:

```
rpmkeys --import <path to the public key>
rpm --checksig --verbose <path to the Linux VDA package>
```

Step 6: Install the Linux VDA

You can do a fresh installation or upgrade an existing installation. The Linux VDA supports upgrades from the most recent version. For example, you can upgrade the Linux VDA from 2308 to 2311 and from 1912 LTSR to 2203 LTSR.

Step 6a: Do a fresh installation

1. (Optional) Uninstall the old version

   If you installed an earlier version other than the previous two and an LTSR release, uninstall it before installing the new version.

   a) Stop the Linux VDA services:

   ```
   sudo systemctl stop ctxvda

   sudo systemctl stop ctxhdx
   ```

   > **Note:**
   >
   > Before you stop the **ctxvda** and **ctxhdx** services, run the **systemctl stop ctxmonitord** command to stop the monitor service daemon. Otherwise, the monitor service daemon restarts the services you stopped.

   b) Uninstall the package:

   ```
   sudo rpm -e XenDesktopVDA
   ```

   > **Note:**
   >
   > To run a command, the full path is needed; alternately, you can add **/opt/Citrix/VDA/sbin** and **/opt/Citrix/VDA/bin** to the system path.

2. Download the Linux VDA package

   Go to the Citrix Virtual Apps and Desktops download page. Expand the appropriate version of Citrix Virtual Apps and Desktops and click **Components** to download the Linux VDA package that matches your Linux distribution.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

3. Install the Linux VDA

> **Note:**
>
> - For RHEL and Rocky Linux, install the EPEL repository before you can install the Linux VDA successfully. For information on how to install EPEL, see the instructions at https://docs.fedoraproject.org/en-US/epel/.
> - Before installing the Linux VDA on RHEL 9.4/9.3/9.2 and Rocky Linux 9.4/9.3/9.2, update the **libsepol** package to version 3.4 or later.

- Install the Linux VDA software using Yum:

  For Amazon Linux 2:

  ```
  sudo yum install -y XenDesktopVDA-<version>.amzn2.x86_64.rpm
  ```

  For RHEL 9.4/9.3/9.2 and Rocky Linux 9.4/9.3/9.2:

  ```
  sudo yum install -y XenDesktopVDA-<version>.el9_x.x86_64.rpm
  ```

  For RHEL 8.x and Rocky Linux 8.x:

  ```
  sudo yum install -y XenDesktopVDA-<version>.el8_x.x86_64.rpm
  ```

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

  For Amazon Linux 2:

  ```
  sudo rpm -i XenDesktopVDA-<version>.amzn2.x86_64.rpm
  ```

  RHEL 9.4/9.3/9.2 and Rocky Linux 9.4/9.3/9.2:

  ```
  sudo rpm -i XenDesktopVDA-<version>.el9_x.x86_64.rpm
  ```

  For RHEL 8.x and Rocky Linux 8.x:

  ```
  sudo rpm -i XenDesktopVDA-<version>.el8_x.x86_64.rpm
  ```

  RPM dependency list for RHEL 9.4/9.3/9.2 and Rocky Linux 9.4/9.3/9.2:

  ```
  gtk2 >= 2.24.33

  java-11-openjdk >= 11

  tzdata-java >= 2022

  ImageMagick >= 6.9

  firewalld >= 0.6.3

  policycoreutils-python-utils >= 2.8
  ```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
12
13  python3-policycoreutils >= 2.8
14
15  dbus >= 1.12.8
16
17  dbus-common >= 1.12.8
18
19  dbus-daemon >= 1.12.8
20
21  dbus-tools >= 1.12.8
22
23  dbus-x11 >= 1.12.8
24
25  xorg-x11-server-utils >= 7.7
26
27  xorg-x11-xinit >= 1.3.4
28
29  libXpm >= 3.5.12
30
31  libXrandr >= 1.5.1
32
33  libXtst >= 1.2.3
34
35  pam >= 1.3.1
36
37  util-linux >= 2.32.1
38
39  util-linux-user >= 2.32.1
40
41  xorg-x11-utils >= 7.5
42
43  bash >= 4.3
44
45  findutils >= 4.6
46
47  gawk >= 4.2
48
49  sed >= 4.5
50
51  cups >= 1.6.0
52
53  cups-filters >= 1.20.0
54
55  ghostscript >= 9.25
56
57  libxml2 >= 2.9
58
59  libmspack >= 0.7
60
61  ibus >= 1.5
62
63  nss-tools >= 3.44.0
64
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
chkconfig >= 1.20

cyrus-sasl-gssapi >= 2.1

python3 >= 3.6~

qt5-qtbase >= 5.5~

qt5-qtbase-gui >= 5.5~

qrencode-libs >= 3.4.4

imlib2 >= 1.4.9

fuse-libs >= 2.9

pulseaudio-utils >= 15.0
```

RPM dependency list for RHEL 8.x and Rocky Linux 8.x:

```
java-11-openjdk >= 11

icoutils >= 0.32

firewalld >= 0.6.3

policycoreutils-python >= 2.8.9

policycoreutils-python-utils >= 2.8

python3-policycoreutils >= 2.8

dbus >= 1.12.8

dbus-common >= 1.12.8

dbus-daemon >= 1.12.8

dbus-tools >= 1.12.8

dbus-x11 >= 1.12.8

xorg-x11-server-utils >= 7.7

xorg-x11-xinit >= 1.3.4

libXpm >= 3.5.12

libXrandr >= 1.5.1

libXtst >= 1.2.3

pam >= 1.3.1
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
util-linux >= 2.32.1

util-linux-user >= 2.32.1

xorg-x11-utils >= 7.5

bash >= 4.3

findutils >= 4.6

gawk >= 4.2

sed >= 4.5

cups >= 1.6.0

foomatic-filters >= 4.0.9

cups-filters >= 1.20.0

ghostscript >= 9.25

libxml2 >= 2.9

libmspack >= 0.7

krb5-workstation >= 1.13

ibus >= 1.5

nss-tools >= 3.44.0

gperftools-libs >= 2.4

cyrus-sasl-gssapi >= 2.1

python3 >= 3.6~

qt5-qtbase >= 5.5~

qt5-qtbase-gui >= 5.5~

qrencode-libs >= 3.4.4

imlib2 >= 1.4.9
```

RPM dependency list for Amazon Linux 2:

```
java-11-openjdk >= 11

ImageMagick >= 6.7.8.9

```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
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

xorg-x11-server-Xorg >= 1.20.4

libXpm >= 3.5.10

libXrandr >= 1.4.1

libXtst >= 1.2.2

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

libxml2 >= 2.9

libmspack >= 0.5

ibus >= 1.5

cyrus-sasl-gssapi >= 2.1

gperftools-libs >= 2.4

nss-tools >= 3.44.0

qt5-qtbase >= 5.5~

qrencode-libs >= 3.4.1

imlib2 >= 1.4.5
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> For a matrix of the Linux distributions and the Xorg versions that this version of the Linux VDA supports, see System requirements.

Step 6b: Upgrade an existing installation (optional)

The Linux VDA supports upgrades from the most recent version. For example, you can upgrade the Linux VDA from 2308 to 2311 and from 1912 LTSR to 2203 LTSR.

For Amazon Linux 2, RHEL, and Rocky Linux distributions:

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

> **Note:**
>
> - Upgrading an existing installation overwrites the configuration files under /etc/xdl. Before you conduct an upgrade, make sure to back up the files.
> - Before upgrading the Linux VDA on RHEL 9.4/9.3/9.2 and Rocky Linux 9.4/9.3/9.2, update the **libsepol** package to version 3.4 or later.
> - Starting with the 2407 release, the Linux VDA delegates package managers **rpm** or **dpkg** to handle configuration files during upgrades. The following describes how rpm and dpkg interact with changes to configuration files:
>   - **rpm**: by default keeps the local version and saves the new version from the package with a .rpmnew extension.
>   - **dpkg**: interactively prompts you with a choice on how to proceed. To silently upgrade the Linux VDA while retaining your local configuration file and saving the new package version as **.dpkg-new** or **.dpkg-dist**, use the following command:
>
>     ```
>     dpkg --force-confold -i package.deb  # Always keep your version, then save new package's version as *.dpkg-new or *.dpkg-dist
>     ```
>
> - Restart the Linux VDA machine after upgrading the software.

Step 7: Install NVIDIA GRID drivers

Enabling HDX 3D Pro requires you to install the NVIDIA GRID drivers on your hypervisor and on the VDA machines.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> To use HDX 3D Pro for Amazon Linux 2, we recommend you install NVIDIA driver 470. For more information, see System requirements.

To install and configure the NVIDIA GRID Virtual GPU Manager (the host driver) on the specific hypervisors, see the following guides:

- XenServer
- VMware ESX
- Nutanix AHV

To install and configure the NVIDIA GRID guest VM drivers, perform the following steps:

1. Ensure that the guest VM is shut down.
2. In XenCenter, allocate a GPU to the VM.
3. Start the VM.
4. Prepare the VM for the NVIDIA GRID driver:

```
yum install gcc

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

systemctl set-default multi-user.target
```

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

> **Note:**
>
> During the GPU driver install, select the default ('no') for each question.
>
> **Important:**
>
> After GPU pass-through is enabled, the Linux VM is no longer accessible through XenCenter. Use SSH to connect.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
nvidia-smi

+-----------------------------------------------------+
| NVIDIA-SMI 352.70     Driver Version: 352.70        |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|===============================+======================+======================|
|   0  Tesla M60           Off  | 0000:00:05.0     Off |                  Off |
| N/A   20C    P0    37W / 150W |     19MiB /  8191MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+

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

Set the correct configuration for the card:

`etc/X11/ctx-nvidia.sh`

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

Step 8: Configure the Linux VDA

> **Note:**
>
> Before setting up the runtime environment, ensure that the **en_US.UTF-8** locale is installed on your OS. If the locale is not available on your OS, run the **sudo locale-gen en_US.UTF-8** command. For Debian, edit the **/etc/locale.gen** file by uncommenting the **# en_US.UTF-8 UTF-8** line and then run the **sudo locale-gen** command.

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:

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

Prompted configuration

Run a manual configuration with prompted questions:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
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_NON_DOMAIN_JOINED='y|n'**–Whether to join the machine to a domain. The default value is 'n'. For domain-joined scenarios, set it to 'n'.
- **CTX_XDL_AD_INTEGRATION='winbind|sssd|centrify|pbis|quest'** –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.
- **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_VDI_MODE='y|n'**–Whether to configure the machine as a dedicated desktop delivery model (VDI) or hosted shared desktop delivery model. For HDX 3D Pro environments, set the value to **'y'**.
- **CTX_XDL_HDX_3D_PRO='y|n'**–The Linux VDA supports HDX 3D Pro, a set of GPU acceleration technologies designed to optimize the virtualization of rich graphics applications. If HDX 3D Pro is selected, the VDA is configured for VDI desktops (single-session) mode - (that is, CTX_XDL_VDI_MODE=**'y'**).
- **CTX_XDL_START_SERVICE='y|n'**–Determines whether the Linux VDA services are started when the configuration is complete.
- **CTX_XDL_REGISTER_SERVICE='y|n'**–The Linux Virtual Desktop services are started after machine startup.
- **CTX_XDL_ADD_FIREWALL_RULES='y|n'**–The Linux VDA services require incoming network connections to be allowed through the system firewall. You can open the required ports (by default ports 80 and 1494) automatically in the system firewall for the Linux Virtual Desktop.
- **CTX_XDL_DESKTOP_ENVIRONMENT=gnome/gnome-classic/kde/mate/xfce/'<none>'** – Specifies the GNOME, GNOME Classic, KDE, MATE, or **Xfce** desktop environment to use in sessions. If you set it to **'<none>'**, the default desktop configured on the VDA is used.

  You can also switch between desktop environments by running commands or using the system tray. For more information, see Desktop switching commands and System tray.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- **CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime** –The path to install .NET for supporting the new broker agent service (ctxvda). The default path is **'/usr/bin'**.
- **CTX_XDL_VDA_PORT=port-number** –The Linux VDA communicates with Delivery Controllers through a TCP/IP port.
- **CTX_XDL_SITE_NAME=<dns-name>** –The Linux VDA discovers LDAP servers through DNS. To limit the DNS search results to a local Site, specify a DNS Site name. If unnecessary, set to **'<none>'**.
- **CTX_XDL_LDAP_LIST='<list-ldap-servers>'** –The Linux VDA queries DNS to discover LDAP servers. If DNS cannot provide LDAP service records, you can provide a space-separated list of LDAP FQDNs with LDAP ports. For example, ad1.mycompany.com:389 ad2.mycompany.com:3268 ad3.mycompany.com:3268. To enable faster LDAP queries within an Active Directory forest, enable Global Catalog on a domain controller and specify the relevant LDAP port number as 3268. This variable is set to **'<none>'** by default.
- **CTX_XDL_SEARCH_BASE=search-base-set** –The Linux VDA queries LDAP through a search base set to the root of the Active Directory Domain (for example, DC=mycompany,DC=com). To improve search performance, you can specify a search base (for example, OU=VDI,DC=mycompany,DC=com). If unnecessary, set to **'<none>'**.
- **CTX_XDL_SUPPORT_DDC_AS_CNAME='y|n'**–The Linux VDA supports specifying a Delivery Controller name using a DNS CNAME record.

Set the environment variable and run the configure script:

```
export CTX_XDL_NON_DOMAIN_JOINED='n'
export CTX_XDL_AD_INTEGRATION=sssd|winbind|centrify|pbis|quest
export CTX_XDL_DDC_LIST='<list-ddc-fqdns>'
export CTX_XDL_VDI_MODE='y|n'
export CTX_XDL_HDX_3D_PRO='y|n'
export CTX_XDL_START_SERVICE='y|n'
export CTX_XDL_REGISTER_SERVICE='y|n'
export CTX_XDL_ADD_FIREWALL_RULES='y|n'
export CTX_XDL_DESKTOP_ENVIRONMENT=gnome|gnome-classic|kde|mate|xfce|'<
    none>'
export CTX_XDL_DOTNET_RUNTIME_PATH='<path-to-install-dotnet-runtime>'
export CTX_XDL_VDA_PORT='<port-number>'
export CTX_XDL_SITE_NAME='<dns-site-name>'|'<none>'
export CTX_XDL_LDAP_LIST='<list-ldap-servers>'|'<none>'
export CTX_XDL_SEARCH_BASE='<search-base-set>'|'<none>'
export CTX_XDL_SUPPORT_DDC_AS_CNAME='y|n'
sudo -E /opt/Citrix/VDA/sbin/ctxsetup.sh --silent
```

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

```
sudo CTX_XDL_NON_DOMAIN_JOINED='n' \
CTX_XDL_AD_INTEGRATION=winbind|centrify|sssd|pbis|quest \
CTX_XDL_DDC_LIST='<list-ddc-fqdns>' \
CTX_XDL_VDI_MODE='y|n' \
CTX_XDL_HDX_3D_PRO='y|n' \
CTX_XDL_START_SERVICE='y|n' \
CTX_XDL_REGISTER_SERVICE='y|n' \
CTX_XDL_ADD_FIREWALL_RULES='y|n' \
CTX_XDL_DESKTOP_ENVIRONMENT=gnome|gnome-classic|kde|mate|xfce|'<none>' \
CTX_XDL_DOTNET_RUNTIME_PATH='<path-to-install-dotnet-runtime>' \
CTX_XDL_VDA_PORT='<port-number>' \
CTX_XDL_SITE_NAME='<dns-site-name>'|'<none>' \
CTX_XDL_LDAP_LIST='<list-ldap-servers>'|'<none>' \
CTX_XDL_SEARCH_BASE='<search-base-set>'|'<none>' \
CTX_XDL_SUPPORT_DDC_AS_CNAME='y|n' \
/opt/Citrix/VDA/sbin/ctxsetup.sh --silent
```

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.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 9: Run XDPing

Run `sudo /opt/Citrix/VDA/bin/xdping` to check for common configuration issues with a Linux VDA environment. For more information, see **XDPing**.

Step 10: Run the Linux VDA

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

Start the Linux VDA:

To start the Linux VDA services:

```
sudo systemctl restart ctxhdx

sudo systemctl restart ctxvda
```

Stop the Linux VDA:

To stop the Linux VDA services:

```
sudo systemctl stop ctxvda

sudo systemctl stop ctxhdx
```

> **Note:**
>
> Before you stop the **ctxvda** and **ctxhdx** services, run the **systemctl stop ctxmonitord** command to stop the monitor service daemon. Otherwise, the monitor service daemon restarts the services you stopped.

Restart the Linux VDA:

To restart the Linux VDA services:

```
sudo systemctl stop ctxvda

sudo systemctl restart ctxhdx

sudo systemctl restart ctxvda
```

Check the status of the Linux VDA:

To check the running status of the Linux VDA services:

```
sudo systemctl status ctxvda

sudo systemctl status ctxhdx
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 11: Create machine catalogs

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

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

- For the operating system, select:
  - The **Multi-session OS** option for a hosted shared desktops delivery model.
  - The **Single-session OS** option for a VDI dedicated desktop delivery model.
- Do not mix Linux and Windows VDA machines in the same machine catalog.

> **Note:**
>
> Early versions of Citrix Studio did not support the notion of a "Linux OS."However, selecting the **Windows Server OS** or **Server OS** option implies an equivalent hosted shared desktops delivery model. Selecting the **Windows Desktop OS** or **Desktop OS** option implies a single user per machine delivery model.
>
> **Tip:**
>
> When you rejoin a removed machine to the Active Directory domain, remove the machine from and add it back to its machine catalog.

Step 12: Create delivery groups

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:

- Ensure that the AD users and groups that 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:**

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

For information about how to create machine catalogs and delivery groups, see Citrix Virtual Apps and Desktops 7 2407.

Install the Linux VDA on SUSE manually

August 2, 2024

> **Important:**
>
> For fresh installations, we recommend you use easy install for a quick installation. Easy install saves time and labor and is less error-prone than the manual installation detailed in this article.

Step 1: Prepare configuration information and the Linux machine

Step 1a: Launch the YaST tool

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

To launch the text-based YaST tool:

```
su -

yast
```

To launch the UI-based YaST tool:

```
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 that is not documented here.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Configure the host name and Domain Name System (DNS)

1. Launch the UI-based YaST tool.
2. Select **System** and then **Network Settings**.
3. Open the **Hostname/DNS** tab.
4. Select the **no** option for **Set Hostname via DHCP**.
5. Select the **Use Custom Policy** option for **Modify DNS Configuration**.
6. Edit the following to reflect your networking setup:
   - **Static Hostname** –Add the DNS host name of the machine.
   - **Name Server** –Add the IP address of the DNS server. It is typically the IP address of the AD Domain Controller.
   - **Domain Search List** –Add the DNS domain name.
7. Change the following line of the `/etc/hosts` file to include the FQDN and host name as the first two entries:

   ```
   127.0.0.1 <FQDN of the VDA> <hostname of the VDA> localhost
   ```

> **Note:**
>
> The Linux VDA currently does not support NetBIOS name truncation. Therefore, the host name must not exceed 15 characters.
>
> **Tip:**
>
> Use a–z, A–Z, 0–9, and hyphen (-) characters only. Avoid underscores (_), spaces, and other symbols. Do not start a host name with a number and do not end with a hyphen. This rule also applies to Delivery Controller host names.

Check the host name Verify that the host name is set correctly:

```
hostname
```

This command returns only the machine's host name and not its Fully Qualified Domain Name (FQDN).

Verify that the FQDN is set correctly:

```
hostname -f
```

This command returns the machine's FQDN.

Check name resolution and service reachability Verify that you can resolve the FQDN and ping the domain controller and Delivery Controller:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
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

It is crucial to maintain accurate clock synchronization between the VDAs, Delivery Controllers, and domain controllers. Hosting the Linux VDA as a virtual machine (VM) 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.

For SUSE 15.5:

1. Launch the UI-based YaST tool.
2. Select **Network Services** and then **NTP Configuration**.
3. In the **Start NTP Daemon** section, select **Now and on Boot**.
4. Select **Dynamic** for **Configuration Source**.
5. Add NTP servers as needed. The NTP service is normally hosted on the Active Directory domain controller.
6. Delete or comment the following line in /etc/chrony.conf if it exists.

   ```
   include /etc/chrony.d/*.conf
   ```

   After editing chrony.conf, restart the `chronyd` service.

   ```
   sudo systemctl restart chronyd.service
   ```

Step 1d: Install Linux VDA dependent packages

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

- OpenJDK 11
- Open Motif Runtime Environment 2.3.1 or later
- Cups 1.6.0 or later
- ImageMagick 6.8 or later

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Add repositories You can obtain most of the required packages except ImageMagick from official repositories. To obtain the ImageMagick packages, enable the `sle-module-desktop-applications` repository by using YaST or the following command:

```
SUSEConnect -p sle-module-desktop-applications/<version number>/
x86_64
```

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

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

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

Install OpenJDK 11 The Linux VDA requires the presence of OpenJDK 11.

To install OpenJDK 11, run the following command:

```
1 sudo zypper install java-11-openjdk
```

Install and specify a database to use

> **Note:**
>
> - We recommend you use SQLite for VDI mode only and use PostgreSQL for a hosted shared desktops delivery model.
> - For easy install and MCS, you can specify SQLite or PostgreSQL to use without having to install them manually. Unless otherwise specified through **/etc/xdl/db.conf**, the Linux VDA uses PostgreSQL by default.
> - For manual installations, you must install SQLite, PostgreSQL, or both manually. If you install both SQLite and PostgreSQL, you can specify one of them to use by editing **/etc/xdl/db.conf** after installing the Linux VDA package.

This section describes how to install PostgreSQL and SQLite and how to specify either of them to use.

Install PostgreSQL To install `Postgresql`, run the following commands:

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Run the following commands to start PostgreSQL upon machine startup or immediately, respectively:

```
sudo systemctl enable postgresql

sudo systemctl restart postgresql
```

Install SQLite For SUSE, run the following command to install SQLite:

```
sudo zypper install sqlite3
```

Specify a database to use If you install both SQLite and PostgreSQL, you can specify one of them to use by editing **/etc/xdl/db.conf** after installing the Linux VDA package.

1. Run **/opt/Citrix/VDA/sbin/ctxcleanup.sh**. Omit this step if it is a fresh installation.
2. Edit **/etc/xdl/db.conf** to specify a database to use.
3. Run **ctxsetup.sh**.

> **Note:**
>
> You can also use **/etc/xdl/db.conf** to configure the port number for PostgreSQL.

Step 2: Prepare the hypervisor

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

Fix time synchronization on XenServer (formerly Citrix Hypervisor)

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

If you are running a paravirtualized Linux kernel with XenServer VM 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:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- 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 restart, edit the **/etc/sysctl.conf** file and add the line:

```
xen.independent_wallclock = 1
```

To verify these changes, restart the system:

```
reboot
```

After restart, verify that the setting is correct:

```
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 apply the Hyper-V time synchronization feature to use the host operating system's time. To ensure that the system clock remains accurate, enable this feature alongside the NTP services.

From the management operating system:

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Fix time synchronization on ESX and ESXi

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

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

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

Step 3: Add the Linux VM to the Windows domain

The following methods are available for adding Linux machines to the Active Directory (AD) domain:

- Samba Winbind
- Centrify DirectControl
- SSSD
- PBIS

Follow instructions based on your chosen method.

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

Samba Winbind

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

1. Launch YaST, select **Network Services** and then **Windows Domain Membership**.
2. Make the following changes:
    - Set the **Domain or Workgroup** to the name of your Active Directory domain or the IP address of the domain controller. Ensure that the domain name is in uppercase.
    - Check **Use SMB information for Linux Authentication**.
        - Check **Create Home Directory on Login**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- 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 you are prompted to install some packages, click **Install**.
4. If a domain controller is found, it asks whether you want to join the domain. Click **Yes**.
5. When prompted, type the credentials of a domain user with permission to add machines to the domain and click **OK**.
6. Restart your services manually or restart the machine. We recommend you restart the machine:

```
su -
reboot
```

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

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

```
sudo net ads testjoin
```

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

```
sudo net ads info
```

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

```
sudo klist –ke
```

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

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

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

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

```
sudo klist
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Examine the machine account details using:

```
sudo net ads status
```

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

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

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

Verify that the Winbind PAM module is configured correctly. To do so, log on to the Linux VDA using a domain user account that has not been used before.

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

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

```
ls /tmp/krb5cc_uid
```

Verify 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. Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

Quest authentication service

Configure Quest on domain controller Assume that you have installed and configured the Quest software on the 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.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

Configure PAM and NSS To enable domain user logons through HDX and other services such as su, ssh, and RDP, configure PAM and NSS manually:

```
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 who has permission to join machines to the Active Directory domain. The **domain-name** is the DNS name of the domain, for example, example.com.

Restart the Linux machine after domain joining.

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

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

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

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

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

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

```
ls /tmp/krb5cc_uid
```

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

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

Exit the session.

```
exit
```

A similar test can be performed by logging on to the Gnome or KDE console directly. Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

Centrify DirectControl

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

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

The **user** is any Active Directory domain user who has permission to join machines to the Active Directory domain. The **domain-name** is the name of the domain to join the Linux machine to.

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
sudo adinfo
```

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

More comprehensive system and diagnostic information is available using:

```
adinfo --sysinfo all

adinfo – diag
```

Test connectivity to the various Active Directory and Kerberos services.

```
adinfo --test
```

Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

SSSD

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

To set up SSSD on SUSE, complete the following steps:

1. Join the domain and create a host keytab
2. Configure PAM for SSSD
3. Set up SSSD
4. Enable SSSD
5. Verify domain membership
6. Verify the Kerberos configuration
7. Verify user authentication

Join the domain and create a host keytab SSSD does not provide Active Directory client functions for joining the domain and managing the system keytab file. You can use the **Samba** approach instead. Complete the following steps before configuring SSSD.

1. Stop and disable the Name Service Cache Daemon (NSCD) daemon.

   ```
   sudo systemctl stop nscd
   sudo systemctl disable nscd
   ```

2. Check the host name and Chrony time synchronization.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
hostname
hostname -f
chronyc traking
```

3. Install or update the required packages:

```
sudo zypper install samba-client sssd-ad
```

4. Edit the `/etc/krb5.conf` file as a root user to permit the **kinit** utility to communicate with the target domain. Add the following entries under the **[libdefaults]**, **[realms]**, and **[domain_realm]** sections:

> **Note:**
>
> Configure Kerberos based on your AD infrastructure. The following settings are meant for the single-domain, single-forest model.

```
[libdefaults]

    dns_canonicalize_hostname = false

    rdns = false

    default_realm = REALM

    forwardable = true

[realms]

    REALM = {


        kdc = fqdn-of-domain-controller

        default_domain = realm

        admin_server = fqdn-of-domain-controller
     }

[domain_realm]

    .realm = REALM
```

realm is the Kerberos realm name, such as example.com. **REALM** is the Kerberos realm name in uppercase, such as EXAMPLE.COM.

5. Edit `/etc/samba/smb.conf` as a root user to permit the **net** utility to communicate with the target domain. Add the following entries under the **[global]** section:

```
[global]
    workgroup = domain
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
3
4       client signing = yes
5
6       client use spnego = yes
7
8       kerberos method = secrets and keytab
9
10      realm = REALM
11
12      security = ADS
```

domain is the short NetBIOS name of an Active Directory domain, such as EXAMPLE.

6. Modify the **passwd** and **group** entries in the **/etc/nsswitch.conf** file to reference SSSD when resolving users and groups.

```
1 passwd: compat sss
2
3 group: compat sss
```

7. Use the configured Kerberos client to authenticate to the target domain as Administrator.

```
1 kinit administrator
```

8. Use the **net** utility to join the system to the domain and generate a system keytab file.

```
1 net ads join osname="SUSE Linux Enterprise Server" osVersion=15 -U
      administrator
```

Configure PAM for SSSD Before configuring PAM for SSSD, install or update the required packages:

```
1 sudo zypper install sssd sssd-ad
```

Configure the PAM module for user authentication through SSSD and create home directories for user logons.

```
1 sudo pam-config --add  --sss
2 sudo pam-config --add --mkhomedir
```

Set up SSSD

1. Edit `/etc/sssd/sssd.conf` as a root user to permit the SSSD daemon to communicate with the target domain. An example `sssd.conf` configuration (extra options can be added as needed):

```
1 [sssd]
2       config_file_version = 2
3       services = nss,pam
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
    domains = domain-dns-name

[domain/domain-dns-name]
    id_provider = ad
    auth_provider = ad
    access_provider = ad
    ad_domain = domain-dns-name
    ad_server = fqdn-of-domain-controller
    ldap_id_mapping = true
    ldap_schema = ad

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

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

    fallback_homedir = /home/%d/%u
    default_shell = /bin/bash

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

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

    ad_gpo_access_control = permissive
```

domain-dns-name is the DNS domain name, such as example.com.

> **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. **ad_gpo_access_control** is set to **permissive** to prevent an invalid logon error for Linux sessions. See the man pages for `sssd.conf` and `sssd-ad`.

2. Set the file ownership and permissions on `sssd.conf`:

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

Enable SSSD Run the following commands to enable and start the SSSD daemon at system startup:

```
sudo systemctl enable sssd
sudo systemctl start sssd
```

Verify domain membership

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

```
sudo net ads testjoin
```

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

```
sudo net ads info
```

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

```
sudo klist -ke
```

This command displays the list of keys available for the various combinations of principal names and cipher suites.

Run the Kerberos **kinit** command to authenticate the machine with the domain controller using these keys:

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

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

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

```
sudo klist
```

Verify user authentication 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 to the Linux VDA using a domain user account that has not been used before.

```
ssh localhost -l domain\\username

id -u

klist

exit
```

Verify that the Kerberos tickets returned by the `klist` command are correct for that user and have not expired.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

As a root user, verify 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 the Gnome or KDE console directly. Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

PBIS

Download the required PBIS package For example:

```
wget https://github.com/BeyondTrust/pbis-open/releases/download/9.1.0/
    pbis-open-9.1.0.551.linux.x86_64.rpm.sh
```

Make the PBIS installation script executable For example:

```
chmod +x pbis-open-9.1.0.551.linux.x86_64.rpm.sh
```

Run the PBIS installation script For example:

```
sh pbis-open-9.1.0.551.linux.x86_64.rpm.sh
```

Join a 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:

```
/opt/pbis/bin/domainjoin-cli join domain-name user
```

The **user** is a domain user who has permissions to add machines to the Active Directory domain. The **domain-name** is the DNS name of the domain, for example, example.com.

Note: To set Bash as the default shell, run the **/opt/pbis/bin/config LoginShellTemplate/bin/bash** command.

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

```
/opt/pbis/bin/domainjoin-cli query
```

If the machine is joined to a domain, this command returns the information about the currently joined AD domain and OU. Otherwise, only the host name appears.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Verify user authentication Verify that PBIS can authenticate domain users through PAM. To do so, log on to the Linux VDA using a domain user account that has not been used before.

```
ssh localhost -l domain\\user

id -u
```

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

```
ls /tmp/krb5cc_uid
```

Exit the session.

```
exit
```

Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

Step 4: Install .NET

In addition to the .NET Runtime, you must install .ASP.NET Core Runtime on all supported Linux distributions before you install or upgrade the Linux VDA. Version 6 is required for Amazon Linux 2. Version 8 is required for other distributions.

If your Linux distribution contains the .NET version that you require, install it from the built-in feed. Otherwise, install .NET from the Microsoft package feed. For more information, see https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers.

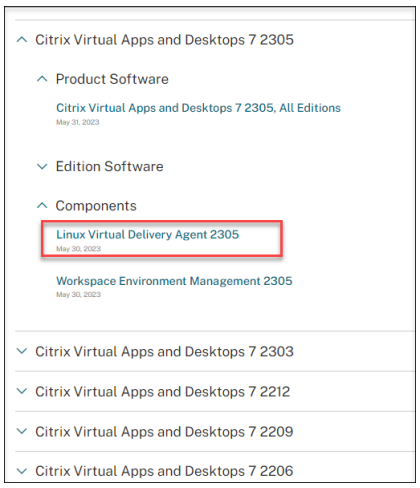
After installing .NET, run the **which dotnet** command to find your runtime path.

Based on the command output, set the .NET runtime binary path. For example, if the command output is /aa/bb/dotnet, use /aa/bb as the .NET binary path.

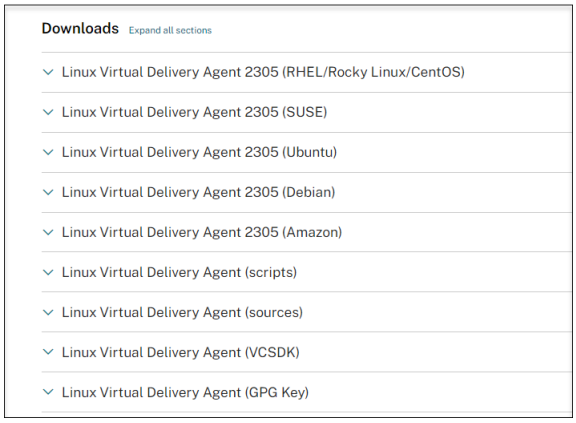
Step 5: Download the Linux VDA package

1. Go to the Citrix Virtual Apps and Desktops download page.
2. Expand the appropriate version of Citrix Virtual Apps and Desktops.
3. Expand **Components** to find the Linux VDA. For example:

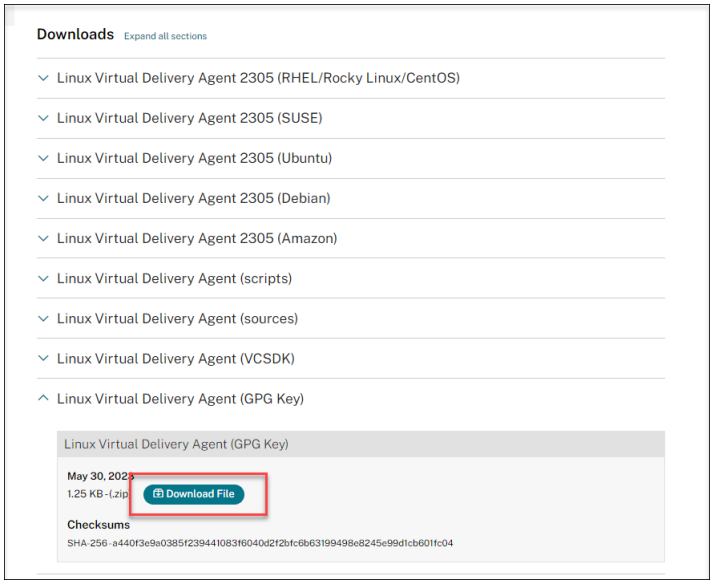
© 1999–2024 Cloud Software Group, Inc. All rights reserved.



4. Click the Linux VDA link to access the Linux VDA downloads.



5. Download the Linux VDA package that matches your Linux distribution.
6. Download the GPG public key that you can use to verify the integrity of the Linux VDA package. For example:



© 1999–2024 Cloud Software Group, Inc. All rights reserved.

To verify the integrity of the Linux VDA package by using the public key, run the following commands to import the public key into the RPM database and to check the package integrity:

```
rpmkeys --import <path to the public key>
rpm --checksig --verbose <path to the Linux VDA package>
```

Step 6: Install the Linux VDA

Step 6a: Uninstall the old version

If you installed an earlier version other than the previous two and an LTSR release, uninstall it before installing the new version.

1. Stop the Linux VDA services:

```
sudo systemctl stop ctxvda

sudo systemctl stop ctxhdx
```

> **Note:**
>
> Before you stop the **ctxvda** and **ctxhdx** services, run the **systemctl stop ctxmonitord** command to stop the monitor service daemon. Otherwise, the monitor service daemon restarts the services you stopped.

2. Uninstall the package:

```
sudo rpm -e XenDesktopVDA
```

> **Important:**
>
> Upgrading from the latest two versions is supported.
>
> **Note:**
>
> You can find installed components under **/opt/Citrix/VDA/**.
>
> To run a command, the full path is needed; alternatively, you can add **/opt/Citrix/VDA/sbin** and **/opt/Citrix/VDA/bin** to the system path.

Step 6b: Install the Linux VDA

Install the Linux VDA software using Zypper:

```
sudo zypper install XenDesktopVDA-<version>.sle15_x.x86_64.rpm
```

Install the Linux VDA software using the RPM package manager:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
sudo rpm -i XenDesktopVDA-<version>.sle15_x.x86_64.rpm
```

Step 6c: Upgrade the Linux VDA (optional)

The Linux VDA supports upgrades from the most recent version. For example, you can upgrade the Linux VDA from 2308 to 2311 and from 1912 LTSR to 2203 LTSR.

> **Note:**
>
> Upgrading an existing installation overwrites the configuration files under /etc/xdl. Before you conduct an upgrade, make sure to back up the files.

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

RPM Dependency list for SUSE 15:

```
java-11-openjdk >= 11

ImageMagick >= 7.0

dbus-1 >= 1.12.2

dbus-1-x11 >= 1.12.2

xorg-x11 >= 7.6_1

libXpm4 >= 3.5.12

libXrandr2 >= 1.5.1

libXtst6 >= 1.2.3

pam >= 1.3.0

bash >= 4.4

findutils >= 4.6

gawk >= 4.2

sed >= 4.4

cups >= 2.2

cups-filters >= 1.25

libxml2-2 >= 2.9

libmspack0 >= 0.6

```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
ibus >= 1.5

libtcmalloc4 >= 2.5

libcap-progs >= 2.26

mozilla-nss-tools >= 3.53.1

libpython3_6m1_0 >= 3.6~

libQt5Widgets5 >= 5.12

libqrencode4 >= 4.0.0

libImlib2-1 >= 1.4.10
```

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

Step 7: Install NVIDIA GRID drivers

Enabling HDX 3D Pro requires you to install the NVIDIA GRID drivers on your hypervisor and on the VDA machines.

To install and configure the NVIDIA GRID Virtual GPU Manager (the host driver) on the specific hypervisors, see the following guides:

- XenServer
- VMware ESX
- Nutanix AHV

To install and configure the NVIDIA GRID guest VM drivers, perform the following general steps:

1. Make sure that the guest VM is shut down.
2. In the hypervisor control panel, allocate a GPU to the VM.
3. Start the VM.
4. Install the guest VM driver on the VM.

Step 8: Configure the Linux VDA

> **Note:**
>
> Before setting up the runtime environment, ensure that the **en_US.UTF-8** locale is installed on your OS. If the locale is not available on your OS, run the **sudo locale-gen en_US.UTF-8** command. For Debian, edit the **/etc/locale.gen** file by uncommenting the **# en_US.UTF-8 UTF-8**

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> line and then run the **sudo locale-gen** command.

After installing the package, you must configure the Linux VDA by running the `ctxsetup.sh` script. Before the script makes any changes, it 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_NON_DOMAIN_JOINED='y|n'**–Whether to join the machine to a domain. The default value is 'n'. For domain-joined scenarios, set it to 'n'.
- **CTX_XDL_AD_INTEGRATION='winbind|sssd|centrify|pbis|quest'** –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.
- **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_VDI_MODE='y|n'**–Whether to configure the machine as a dedicated desktop delivery model (VDI) or hosted shared desktop delivery model. For HDX 3D Pro environments, set the value to **'y'**.
- **CTX_XDL_HDX_3D_PRO='y|n'**–The Linux VDA supports HDX 3D Pro, a set of GPU acceleration technologies designed to optimize the virtualization of rich graphics applications. If HDX 3D Pro is selected, the VDA is configured for VDI desktops (single-session) mode - (that is, CTX_XDL_VDI_MODE=**'y'**).

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- **CTX_XDL_START_SERVICE='y|n'**–Determines whether the Linux VDA services are started when the configuration is complete.
- **CTX_XDL_REGISTER_SERVICE='y|n'**–The Linux Virtual Desktop services are started after machine startup.
- **CTX_XDL_ADD_FIREWALL_RULES='y|n'**–The Linux VDA services require incoming network connections to be allowed through the system firewall. You can open the required ports (by default ports 80 and 1494) automatically in the system firewall for the Linux Virtual Desktop.
- **CTX_XDL_DESKTOP_ENVIRONMENT=gnome/gnome-classic/mate/'<none>'** –Specifies the GNOME, GNOME Classic, or MATE desktop environment to use in sessions. If you set it to **'<none>'**, the default desktop configured on the VDA is used.

  You can also switch between desktop environments by running commands or using the system tray. For more information, see Desktop switching commands and System tray.
- **CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime** –The path to install .NET for supporting the new broker agent service (ctxvda). The default path is **'/usr/bin'**.
- **CTX_XDL_VDA_PORT=port-number** –The Linux VDA communicates with Delivery Controllers through a TCP/IP port.
- **CTX_XDL_SITE_NAME=<dns-name>** –The Linux VDA discovers LDAP servers through DNS. To limit the DNS search results to a local Site, specify a DNS Site name. If unnecessary, set to **'<none>'**.
- **CTX_XDL_LDAP_LIST='<list-ldap-servers>'** –The Linux VDA queries DNS to discover LDAP servers. If DNS cannot provide LDAP service records, you can provide a space-separated list of LDAP FQDNs with LDAP ports. For example, ad1.mycompany.com:389 ad2.mycompany.com:3268 ad3.mycompany.com:3268. To enable faster LDAP queries within an Active Directory forest, enable Global Catalog on a domain controller and specify the relevant LDAP port number as 3268. This variable is set to **'<none>'** by default.
- **CTX_XDL_SEARCH_BASE=search-base-set** –The Linux VDA queries LDAP through a search base set to the root of the Active Directory Domain (for example, DC=mycompany,DC=com). To improve search performance, you can specify a search base (for example, OU=VDI,DC=mycompany,DC=com). If unnecessary, set to **'<none>'**.
- **CTX_XDL_SUPPORT_DDC_AS_CNAME='y|n'**–The Linux VDA supports specifying a Delivery Controller name using a DNS CNAME record.

Set the environment variable and run the configure script:

```
export CTX_XDL_NON_DOMAIN_JOINED='n'
export CTX_XDL_AD_INTEGRATION=sssd|winbind|centrify|pbis|quest
export CTX_XDL_DDC_LIST='<list-ddc-fqdns>'
export CTX_XDL_VDI_MODE='y|n'
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
export CTX_XDL_HDX_3D_PRO='y|n'
export CTX_XDL_START_SERVICE='y|n'
export CTX_XDL_REGISTER_SERVICE='y|n'
export CTX_XDL_ADD_FIREWALL_RULES='y|n'
export CTX_XDL_DESKTOP_ENVIRONMENT= gnome|gnome-classic|mate|'<none>'
export CTX_XDL_DOTNET_RUNTIME_PATH='<path-to-install-dotnet-runtime>'
export CTX_XDL_VDA_PORT='<port-number>'
export CTX_XDL_SITE_NAME='<dns-site-name>'|'<none>'
export CTX_XDL_LDAP_LIST='<list-ldap-servers>'|'<none>'
export CTX_XDL_SEARCH_BASE='<search-base-set>'|'<none>'
export CTX_XDL_SUPPORT_DDC_AS_CNAME='y|n'
sudo -E /opt/Citrix/VDA/sbin/ctxsetup.sh --silent
```

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

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

```
sudo CTX_XDL_NON_DOMAIN_JOINED='n' \
CTX_XDL_AD_INTEGRATION=winbind|centrify|sssd|pbis|quest \
CTX_XDL_DDC_LIST='<list-ddc-fqdns>' \
CTX_XDL_VDI_MODE='y|n' \
CTX_XDL_HDX_3D_PRO='y|n' \
CTX_XDL_START_SERVICE='y|n' \
CTX_XDL_REGISTER_SERVICE='y|n' \
CTX_XDL_ADD_FIREWALL_RULES='y|n' \
CTX_XDL_DESKTOP_ENVIRONMENT= gnome|gnome-classic|mate|'<none>' \
CTX_XDL_DOTNET_RUNTIME_PATH='<path-to-install-dotnet-runtime>' \
CTX_XDL_VDA_PORT='<port-number>' \
CTX_XDL_SITE_NAME='<dns-site-name>'|'<none>' \
CTX_XDL_LDAP_LIST='<list-ldap-servers>'|'<none>' \
CTX_XDL_SEARCH_BASE='<search-base-set>'|'<none>' \
CTX_XDL_SUPPORT_DDC_AS_CNAME='y|n' \
/opt/Citrix/VDA/sbin/ctxsetup.sh --silent
```

Remove configuration changes

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

Review Help about this script before proceeding:

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

To remove configuration changes:

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Important:**
>
> This script deletes all configuration data from the database and renders the Linux VDA inoperable.

Configuration logs

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

`/tmp/xdl.configure.log`

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

Step 9: Run XDPing

Run `sudo /opt/Citrix/VDA/bin/xdping` to check for common configuration issues with a Linux VDA environment. For more information, see **XDPing**.

Step 10: Run the Linux VDA

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

Start the Linux VDA:

To start the Linux VDA services:

```
sudo 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
```

> **Note:**
>
> Before you stop the **ctxvda** and **ctxhdx** services, run the **systemctl stop ctxmonitord** command to stop the monitor service daemon. Otherwise, the monitor service daemon restarts the services you stopped.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Restart the Linux VDA:

To restart the Linux VDA services:

```
sudo systemctl stop ctxvda

sudo systemctl restart ctxhdx

sudo systemctl restart ctxvda
```

Check the Linux VDA status:

To check the running status of the Linux VDA services:

```
sudo systemctl status ctxvda

sudo systemctl status ctxhdx
```

Step 11: Create machine catalogs

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

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

- For the operating system, select:
  - The **Multi-session OS** option for a hosted shared desktops delivery model.
  - The **Single-session OS** option for a VDI dedicated desktop delivery model.
- Do not mix Linux and Windows VDA machines in the same machine catalog.

> **Note:**
>
> Early versions of Citrix Studio did not support the notion of a "Linux OS."However, selecting the **Windows Server OS** or **Server OS** option implies an equivalent hosted shared desktops delivery model. Selecting the **Windows Desktop OS** or **Desktop OS** option implies a single user per machine delivery model.
>
> **Tip:**
>
> If you remove and rejoin a machine to the Active Directory domain, you must remove and add the machine to the machine catalog again.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 12: Create delivery groups

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:

- Make sure that the AD users and groups that you select have been properly configured to log on to the Linux VDA machines.
- Do not allow logon of unauthenticated (anonymous) users.
- Do not mix the delivery group with machine catalogs that contain Windows machines.

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

For information about how to create machine catalogs and delivery groups, see Citrix Virtual Apps and Desktops 7 2407.

Install the Linux VDA on Ubuntu manually

August 2, 2024

> **Important:**
>
> For fresh installations, we recommend you use easy install for a quick installation. Easy install saves time and labor and is less error-prone than the manual installation detailed in this article.

Step 1: Prepare configuration information and the Linux machine

Step 1a: Verify the network configuration

Make sure that the network is connected and configured correctly. For example, you must configure the DNS server on the Linux VDA.

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

```
preserve_hostname: true
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 1b: Set the host name

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

`hostname`

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

Make sure that the DNS domain name and Fully Qualified Domain Name (FQDN) of the machine are reported back correctly. The way is to change the following line of the **/etc/hosts** file to include the FQDN and host name as the first two entries:

`127.0.0.1 hostname-fqdn hostname localhost`

For example:

`127.0.0.1 vda01.example.com vda01 localhost`

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

> **Note:**
>
> The Linux VDA currently does not support NetBIOS name truncation. Therefore, the host name must not exceed 15 characters.
>
> **Tip:**
>
> Use a–z, A–Z, 0–9, and hyphen (-) characters only. Avoid underscores (_), spaces, and other symbols. Do not start a host name with a number and do not end with a hyphen. This rule also applies to Delivery Controller host names.

Step 1d: Check the host name

Verify that the host name is set correctly:

```
1 hostname
```

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

Verify that the FQDN is set correctly:

```
1 hostname -f
```

This command returns the FQDN of the machine.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 1e: Disable multicast DNS

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

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

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

To:

```
hosts: files dns
```

Step 1f: Check name resolution and service reachability

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

```
nslookup domain-controller-fqdn

ping domain-controller-fqdn

nslookup delivery-controller-fqdn

ping delivery-controller-fqdn
```

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 (VM) can cause clock skew problems. For this reason, synchronizing time with a remote time service is preferred.

Install chrony:

```
apt-get install chrony
```

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

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

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Save changes and restart the Chrony daemon:

```
sudo systemctl restart chrony
```

Step 1h: Install OpenJDK 11

The Linux VDA requires the presence of OpenJDK 11.

Ubuntu 22.04 includes OpenJDK 11.

To install OpenJDK 11 on Ubuntu 20.04, run the following command:

```
sudo apt-get install -y openjdk-11-jdk
```

Step 1i: Install and specify a database to use

> **Note:**
>
> - We recommend you use SQLite for VDI mode only and use PostgreSQL for a hosted shared desktops delivery model.
> - For easy install and MCS, you can specify SQLite or PostgreSQL to use without having to install them manually. Unless otherwise specified through **/etc/xdl/db.conf**, the Linux VDA uses PostgreSQL by default.
> - For manual installations, you must install SQLite, PostgreSQL, or both manually. If you install both SQLite and PostgreSQL, you can specify one of them to use by editing **/etc/xdl/db.conf** after installing the Linux VDA package.

This section describes how to install PostgreSQL and SQLite and how to specify either of them to use.

Install PostgreSQL Run the following commands to install PostgreSQL:

```
sudo apt-get install -y postgresql

sudo apt-get install -y libpostgresql-jdbc-java
```

Run the following commands to start PostgreSQL upon machine startup or immediately, respectively:

```
sudo systemctl enable postgresql

sudo systemctl restart postgresql
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Install SQLite For Ubuntu, run the following command to install SQLite:

```
sudo apt-get install -y sqlite3
```

Specify a database to use If you install both SQLite and PostgreSQL, you can specify one of them to use by editing **/etc/xdl/db.conf** after installing the Linux VDA package.

1. Run **/opt/Citrix/VDA/sbin/ctxcleanup.sh**. Omit this step if it is a fresh installation.
2. Edit **/etc/xdl/db.conf** to specify a database to use.
3. Run **ctxsetup.sh**.

> **Note:**
>
> You can also use **/etc/xdl/db.conf** to configure the port number for PostgreSQL.

Step 1j: Install Motif

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

Step 1k: Install other packages

For Ubuntu 22.04:

```
sudo apt-get install -y libsasl2-2
sudo apt-get install -y libsasl2-modules-gssapi-mit
sudo apt-get install -y libldap-2.5-0
sudo apt-get install -y krb5-user
sudo apt-get install -y libgtk2.0-0
```

For Ubuntu 20.04:

```
sudo apt-get install -y libsasl2-2
sudo apt-get install -y libsasl2-modules-gssapi-mit
sudo apt-get install -y libldap-2.4-2
sudo apt-get install -y krb5-user
sudo apt-get install -y libgtk2.0-0
```

Step 2: Prepare the hypervisor

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Fix time synchronization on XenServer (formerly Citrix Hypervisor)

When the XenServer Time Sync feature is enabled, within each paravirtualized Linux VM you experience issues with NTP and XenServer. Both try to manage the system clock. To avoid the clock becoming out of sync with other servers, make sure 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.

If you are running a paravirtualized Linux kernel with XenServer VM 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 restart, edit the **/etc/sysctl.conf file** and add the line:

```
xen.independent_wallclock = 1
```

To verify these changes, restart the system:

```
su -

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

This command returns the value 1.

Fix time synchronization on Microsoft Hyper-V

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

From the management operating system:

1. Open the Hyper-V Manager console.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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 (formerly Citrix Hypervisor), 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 try to synchronize the system clock. To avoid the clock becoming out of sync with other servers, make sure 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 VM to the Windows domain

The following methods are available for adding Linux machines to the Active Directory (AD) domain:

- Samba Winbind
- Quest authentication service
- Centrify DirectControl
- SSSD
- PBIS

Follow instructions based on your chosen method.

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

Samba Winbind

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Install or update the required packages

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

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

```
sudo systemctl enable winbind
```

> **Note:**
>
> Ensure that the `winbind` script is located under `/etc/init.d`.

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

> **Note:**
>
> Configure Kerberos based on your AD infrastructure. The following settings are meant for the single-domain, single-forest model.

```
[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** parameter 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 doesn't have a tool like **authconfig** in RHEL and yast2 in SUSE.

Open **/etc/samba/smb.conf** by running the **vim /etc/samba/smb.conf** command, and then make the following settings:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
[global]
workgroup = WORKGROUP
security = ADS
realm = REALM
encrypt passwords = yes
idmap config *:range = 16777216-33554431
kerberos method = secrets and keytab
winbind refresh tickets = yes
template shell = /bin/bash
```

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

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

```
passwd: compat winbind
group: compat winbind
```

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

```
1 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

```
1 sudo systemctl restart winbind
```

Configure PAM for Winbind Run the following command and make sure that the **Winbind NT/Active Directory authentication** and **Create home directory on login** options are selected:

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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, verify that the system **keytab** file has been created and contains valid keys:

```
sudo klist -ke
```

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

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

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

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

```
sudo klist
```

Examine the account details of the machine using:

```
sudo net ads status
```

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

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

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
ssh localhost -l domain\\username

id -u
```

> **Note:**
>
> To run an SSH command successfully, make sure that SSH is enabled and working properly.

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

```
ls /tmp/krb5cc_uid
```

Verify 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. Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

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

Quest authentication service

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

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

Configure Quest on Linux VDA

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

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

```
SELINUX=disabled
```

This change requires a machine restart:

```
1 reboot
```

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

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

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

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

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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.

Restart the Linux machine after domain joining.

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

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

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

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

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

```
ssh localhost -l domain\\username

id -u
```

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

```
ls /tmp/krb5cc_uid
```

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

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

Exit the session.

```
exit
```

Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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
```

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

More comprehensive system and diagnostic information is available using:

```
adinfo --sysinfo all

adinfo --diag
```

Test connectivity to the various Active Directory and Kerberos services.

```
adinfo --test
```

Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

SSSD

Configure Kerberos Run the following command to install Kerberos:

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

To configure Kerberos, open **/etc/krb5.conf** as root and set the parameters:

> **Note:**
>
> Configure Kerberos based on your AD infrastructure. The following settings are meant for the

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> single-domain, single-forest model.

```
[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` parameter in this context is the DNS domain name, such as example.com. The *REALM* is the Kerberos realm name in uppercase, such as EXAMPLE.COM.

Join the domain SSSD must be configured to use Active Directory as its identity provider and Kerberos for authentication. However, SSSD does not provide AD client functions for joining the domain and managing the system keytab file. You can use **adcli**, **realmd**, or **Samba** instead.

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

- **If you use adcli to join the domain, complete the following steps:**

1. Install **adcli**.

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

2. 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
   ```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

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

3. Verify domain membership.

   For Ubuntu 22.04 and Ubuntu 20.04 machines, run the `adcli testjoin` command to test whether the machines are joined to the domain.

- **If you use Samba to join the domain, complete the following steps:**

1. Install the package.

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

2. 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.

3. Join the domain with **Samba**.

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

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

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

Set up SSSD Install or update required packages:

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

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

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

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 created manually. 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
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
ad_gpo_map_remote_interactive = +ctxhdx
```

> **Note:**
>
> ldap_id_mapping is set to **true** so that SSSD itself takes care of mapping Windows SIDs to Unix UIDs. Otherwise, the Active Directory must be able to provide POSIX extensions. PAM service `ctxhdx` is added to ad_gpo_map_remote_interactive.
>
> The **domain-dns-name** parameter 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.
>
> For information about the configuration settings, see the man pages for sssd.conf and `sssd-ad`.

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

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

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

```
sudo systemctl start sssd

sudo systemctl enable sssd
```

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

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

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

- If you use **adcli** to verify domain membership, run the `sudo adcli info domain-dns-name` command to show the domain information.
- If you use **Samba** to verify domain membership, run the `sudo net ads testjoin` command to verify that the machine is joined to a domain and the `sudo net ads info` 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, verify that the system keytab file has been created and contains valid keys:

```
sudo klist -ke
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

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

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

Verify that the TGT 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 to the Linux VDA using a domain user account that has not been used before.

```
ssh localhost -l domain\\username

id -u

klist

exit
```

Verify that the Kerberos tickets returned by the **klist** command are correct for that user and have not expired.

As a root user, verify 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. Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

PBIS

Download the required PBIS package

```
sudo wget https://github.com/BeyondTrust/pbis-open/releases/download
    /9.1.0/pbis-open-9.1.0.551.linux.x86_64.deb.sh
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Make the PBIS installation script executable

```
sudo chmod +x pbis-open-9.1.0.551.linux.x86_64.deb.sh
```

Run the PBIS installation script

```
sudo sh pbis-open-9.1.0.551.linux.x86_64.deb.sh
```

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 /opt/pbis/bin/domainjoin-cli join domain-name user
```

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

Note: To set Bash as the default shell, run the **sudo /opt/pbis/bin/config LoginShellTemplate/bin/bash** command.

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

```
/opt/pbis/bin/domainjoin-cli query
```

If the machine is joined to a domain, this command returns the information about the currently joined AD domain and OU. Otherwise, only the host name appears.

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

```
sudo ssh localhost -l domain\\user

id -u
```

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

```
ls /tmp/krb5cc_uid
```

Exit the session.

```
exit
```

Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 4: Install .NET

In addition to the .NET Runtime, you must install .ASP.NET Core Runtime on all supported Linux distributions before you install or upgrade the Linux VDA. Version 6 is required for Amazon Linux 2. Version 8 is required for other distributions.

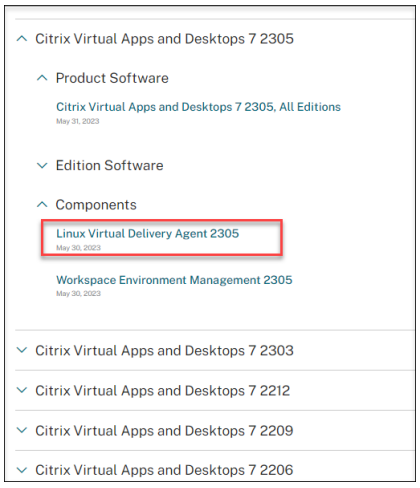
If your Linux distribution contains the .NET version that you require, install it from the built-in feed. Otherwise, install .NET from the Microsoft package feed. For more information, see https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers.

After installing .NET, run the **which dotnet** command to find your runtime path.

Based on the command output, set the .NET runtime binary path. For example, if the command output is /aa/bb/dotnet, use /aa/bb as the .NET binary path.

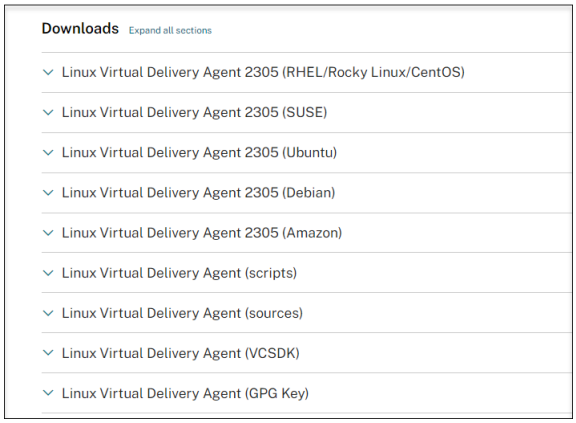
Step 5: Download the Linux VDA package

1. Go to the Citrix Virtual Apps and Desktops download page.
2. Expand the appropriate version of Citrix Virtual Apps and Desktops.
3. Expand **Components** to find the Linux VDA. For example:

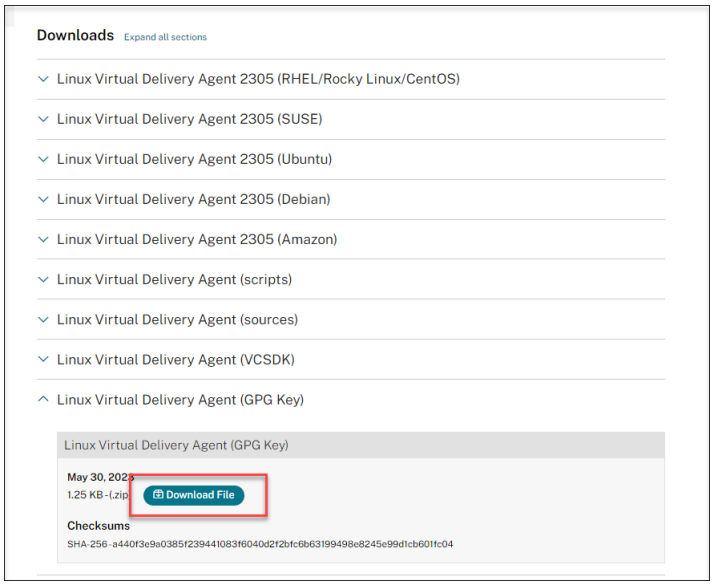


4. Click the Linux VDA link to access the Linux VDA downloads.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



5. Download the Linux VDA package that matches your Linux distribution.
6. Download the GPG public key that you can use to verify the integrity of the Linux VDA package. For example:



   To verify the integrity of the Linux VDA package, run the following commands to import the public key into the DEB database and to check the package integrity:

```
sudo apt-get install dpkg-sig
gpg --import <path to the public key>
dpkg-sig --verify <path to the Linux VDA package>
```

Step 6: Install the Linux VDA

Step 6a: Install the Linux VDA

Install the Linux VDA software using the Debian package manager:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

> **Note:**
>
> For Ubuntu 20.04 on GCP, disable RDNS. To do so, add the **rdns = false** line under **[libdefaults]** in /etc/krb5.conf.

Debian dependency list for Ubuntu 22.04:

```
openjdk-11-jdk >= 11

imagemagick >= 8:6.9.11

libgtkmm-3.0-1v5 >= 3.24.5

ufw >= 0.36

ubuntu-desktop >= 1.481

libxrandr2 >= 2:1.5.2

libxtst6 >= 2:1.2.3

libxm4 >= 2.3.8

util-linux >= 2.37

gtk3-nocsd >= 3

bash >= 5.1

findutils >= 4.8.0

sed >= 4.8

cups >= 2.4

libmspack0 >= 0.10

ibus >= 1.5

libgoogle-perftools4 >= 2.9~

libpython3.10 >= 3.10~

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

libnss3-tools >= 2:3.68

libqt5widgets5 >= 5.15~

```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
libqrencode4 >= 4.1.1

libimlib2 >= 1.7.4
```

Debian dependency list for Ubuntu 20.04:

```
openjdk-11-jdk >= 11

imagemagick >= 8:6.9.10

libgtkmm-3.0-1v5 >= 3.24.2

ufw >= 0.36

ubuntu-desktop >= 1.450

libxrandr2 >= 2:1.5.2

libxtst6 >= 2:1.2.3

libxm4 >= 2.3.8

util-linux >= 2.34

gtk3-nocsd >= 3

bash >= 5.0

findutils >= 4.7.0

sed >= 4.7

cups >= 2.3

libmspack0 >= 0.10

ibus >= 1.5

libgoogle-perftools4 >= 2.7~

libpython3.8 >= 3.8~

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

libnss3-tools >= 2:3.49

libqt5widgets5 >= 5.7~

libqrencode4 >= 4.0.0

libimlib2 >= 1.6.1
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> For a matrix of the Linux distributions and the Xorg versions that this version of the Linux VDA supports, see System requirements.

Step 6b: Upgrade the Linux VDA (optional)

The Linux VDA supports upgrades from the most recent version. For example, you can upgrade the Linux VDA from 2308 to 2311 and from 1912 LTSR to 2203 LTSR.

```
sudo dpkg -i <PATH>/<Linux VDA deb>

sudo apt-get install -f
```

> **Note:**
>
> - Upgrading an existing installation overwrites the configuration files under /etc/xdl. Before you conduct an upgrade, make sure to back up the files.
> - Starting with the 2407 release, the Linux VDA delegates package managers **rpm** or **dpkg** to handle configuration files during upgrades. The following describes how rpm and dpkg interact with changes to configuration files:
>   - **rpm**: by default keeps the local version and saves the new version from the package with a .rpmnew extension.
>   - **dpkg**: interactively prompts you with a choice on how to proceed. To silently upgrade the Linux VDA while retaining your local configuration file and saving the new package version as **.dpkg-new** or **.dpkg-dist**, use the following command:
>
>     ```
>     dpkg --force-confold -i package.deb  # Always keep your
>        version, then save new package's version as *.dpkg-new
>        or *.dpkg-dist
>     ```

Step 7: Install NVIDIA GRID drivers

Enabling HDX 3D Pro requires you to install the NVIDIA GRID drivers on your hypervisor and on the VDA machines.

To install and configure the NVIDIA GRID Virtual GPU Manager (the host driver) on the specific hypervisors, see the following guides:

- XenServer
- VMware ESX
- Nutanix AHV

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

To install and configure the NVIDIA GRID guest VM drivers, perform the following general steps:

1. Ensure that the guest VM is shut down.
2. In the hypervisor control panel, allocate a GPU to the VM.
3. Start the VM.
4. Install the guest VM driver on the VM.

Step 8: Configure the Linux VDA

> **Note:**
>
> Before setting up the runtime environment, ensure that the **en_US.UTF-8** locale is installed on your OS. If the locale is not available on your OS, run the **sudo locale-gen en_US.UTF-8** command. For Debian, edit the **/etc/locale.gen** file by uncommenting the **# en_US.UTF-8 UTF-8** line and then run the **sudo locale-gen** command.

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_NON_DOMAIN_JOINED='y|n'**–Whether to join the machine to a domain. The default value is 'n'. For domain-joined scenarios, set it to 'n'.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- **CTX_XDL_AD_INTEGRATION='winbind|sssd|centrify|pbis|quest'** –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.
- **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_VDI_MODE='y|n'**–Whether to configure the machine as a dedicated desktop delivery model (VDI) or hosted shared desktop delivery model. For HDX 3D Pro environments, set the value to **'y'**.
- **CTX_XDL_HDX_3D_PRO='y|n'**–The Linux VDA supports HDX 3D Pro, a set of GPU acceleration technologies designed to optimize the virtualization of rich graphics applications. If HDX 3D Pro is selected, the VDA is configured for VDI desktops (single-session) mode - (that is, CTX_XDL_VDI_MODE='**y**').
- **CTX_XDL_START_SERVICE='y|n'**–Determines whether the Linux VDA services are started when the configuration is complete.
- **CTX_XDL_REGISTER_SERVICE='y|n'**–The Linux Virtual Desktop services are started after machine startup.
- **CTX_XDL_ADD_FIREWALL_RULES='y|n'**–The Linux VDA services require incoming network connections to be allowed through the system firewall. You can open the required ports (by default ports 80 and 1494) automatically in the system firewall for the Linux Virtual Desktop.
- **CTX_XDL_DESKTOP_ENVIRONMENT=gnome/gnome-classic/kde/mate/xfce/'<none>'** – Specifies the GNOME, GNOME Classic, KDE, MATE, or **Xfce** desktop environment to use in sessions. If you set it to '**<none>**', the default desktop configured on the VDA is used.

  You can also switch between desktop environments by running commands or using the system tray. For more information, see Desktop switching commands and System tray.
- **CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime** –The path to install .NET for supporting the new broker agent service (ctxvda). The default path is '**/usr/bin**'.
- **CTX_XDL_VDA_PORT=port-number** –The Linux VDA communicates with Delivery Controllers through a TCP/IP port.
- **CTX_XDL_SITE_NAME=<dns-name>** –The Linux VDA discovers LDAP servers through DNS. To limit the DNS search results to a local Site, specify a DNS Site name. If unnecessary, set to '**<none>**'.
- **CTX_XDL_LDAP_LIST='<list-ldap-servers>'** –The Linux VDA queries DNS to discover LDAP servers. If DNS cannot provide LDAP service records, you can provide a space-separated list of LDAP FQDNs with LDAP ports. For example, ad1.mycompany.com:389

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

ad2.mycompany.com:3268 ad3.mycompany.com:3268. To enable faster LDAP queries within an Active Directory forest, enable Global Catalog on a domain controller and specify the relevant LDAP port number as 3268. This variable is set to '**<none>**' by default.

- **CTX_XDL_SEARCH_BASE=search-base-set** –The Linux VDA queries LDAP through a search base set to the root of the Active Directory Domain (for example, DC=mycompany,DC=com). To improve search performance, you can specify a search base (for example, OU=VDI,DC=mycompany,DC=com). If unnecessary, set to '**<none>**'.
- **CTX_XDL_SUPPORT_DDC_AS_CNAME='y|n'**–The Linux VDA supports specifying a Delivery Controller name using a DNS CNAME record.

Set the environment variable and run the configure script:

```
export CTX_XDL_NON_DOMAIN_JOINED='n'
export CTX_XDL_AD_INTEGRATION=sssd|winbind|centrify|pbis|quest
export CTX_XDL_DDC_LIST='<list-ddc-fqdns>'
export CTX_XDL_VDI_MODE='y|n'
export CTX_XDL_HDX_3D_PRO='y|n'
export CTX_XDL_START_SERVICE='y|n'
export CTX_XDL_REGISTER_SERVICE='y|n'
export CTX_XDL_ADD_FIREWALL_RULES='y|n'
export CTX_XDL_DESKTOP_ENVIRONMENT=gnome|gnome-classic|kde|mate|xfce|'<
    none>'
export CTX_XDL_DOTNET_RUNTIME_PATH='<path-to-install-dotnet-runtime>'
export CTX_XDL_VDA_PORT='<port-number>'
export CTX_XDL_SITE_NAME='<dns-site-name>'|'<none>'
export CTX_XDL_LDAP_LIST='<list-ldap-servers>'|'<none>'
export CTX_XDL_SEARCH_BASE='<search-base-set>'|'<none>'
export CTX_XDL_SUPPORT_DDC_AS_CNAME='y|n'
sudo -E /opt/Citrix/VDA/sbin/ctxsetup.sh --silent
```

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

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

```
sudo CTX_XDL_NON_DOMAIN_JOINED='n' \
CTX_XDL_AD_INTEGRATION=sssd|winbind|centrify|pbis|quest \
CTX_XDL_DDC_LIST='<list-ddc-fqdns>' \
CTX_XDL_VDI_MODE='y|n' \
CTX_XDL_HDX_3D_PRO='y|n' \
CTX_XDL_START_SERVICE='y|n' \
CTX_XDL_REGISTER_SERVICE='y|n' \
CTX_XDL_ADD_FIREWALL_RULES='y|n' \
CTX_XDL_DESKTOP_ENVIRONMENT=gnome|gnome-classic|kde|mate|xfce|'<none>'
    \
CTX_XDL_DOTNET_RUNTIME_PATH='<path-to-install-dotnet-runtime>' \
CTX_XDL_VDA_PORT='<port-number>' \
CTX_XDL_SITE_NAME='<dns-site-name>'|'<none>' \
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
CTX_XDL_LDAP_LIST='<list-ldap-servers>'|'<none>' \
CTX_XDL_SEARCH_BASE='<search-base-set>'|'<none>' \
CTX_XDL_SUPPORT_DDC_AS_CNAME='y|n' \
/opt/Citrix/VDA/sbin/ctxsetup.sh --silent
```

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

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
```

To uninstall the Linux VDA software:

```
dpkg -r xendesktopvda
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> Uninstalling the Linux VDA software deletes the associated PostgreSQL and other configuration data. However, the PostgreSQL package and other dependent packages that were set up before the installation of the Linux VDA are not deleted.
>
> **Tip:**
>
> The information in this section does not cover the removal of dependent packages including PostgreSQL.

Step 9: Run XDPing

Run `sudo /opt/Citrix/VDA/bin/xdping` to check for common configuration issues with a Linux VDA environment. For more information, see **XDPing**.

Step 10: 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
```

> **Note:**
>
> Before you stop the **ctxvda** and **ctxhdx** services, run the **systemctl stop ctxmonitord** command to stop the monitor service daemon. Otherwise, the monitor service daemon restarts the services you stopped.

Restart the Linux VDA:

To restart the Linux VDA services:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
sudo systemctl stop ctxvda

sudo systemctl restart ctxhdx

sudo systemctl restart ctxvda
```

Check the Linux VDA status:

To check the running status of the Linux VDA services:

```
sudo systemctl status ctxvda

sudo systemctl status ctxhdx
```

Step 11: Create machine catalogs

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

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

- For the operating system, select:
  - The **Multi-session OS** option for a hosted shared desktops delivery model.
  - The **Single-session OS** option for a VDI dedicated desktop delivery model.
- Do not mix Linux and Windows VDA machines in the same machine catalog.

> **Note:**
>
> Early versions of Citrix Studio did not support the notion of a "Linux OS."However, selecting the **Windows Server OS** or **Server OS** option implies an equivalent hosted shared desktops delivery model. Selecting the **Windows Desktop OS** or **Desktop OS** option implies a single user per machine delivery model.
>
> **Tip:**
>
> If you remove and rejoin a machine to the Active Directory domain, you must remove and add the machine to the machine catalog again.

Step 12: Create delivery groups

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.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

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

For information about how to create machine catalogs and delivery groups, see Citrix Virtual Apps and Desktops 7 2407.

Install the Linux VDA on Debian manually

August 2, 2024

> **Important:**
>
> For fresh installations, we recommend you use easy install for a quick installation. Easy install saves time and labor and is less error-prone than the manual installation detailed in this article.

Step 1: Prepare configuration information and the Linux machine

Step 1a: Set the host name

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

```
hostname
```

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

Make sure that the DNS domain name and Fully Qualified Domain Name (FQDN) of the machine are reported back correctly. The way is to change the following line of the **/etc/hosts** file to include the FQDN and host name as the first two entries:

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

For example:

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> The Linux VDA currently does not support NetBIOS name truncation. The host name must not exceed 15 characters.
>
> **Tip:**
>
> Use a–z, A–Z, 0–9, and hyphen (-) characters only. Avoid underscores (_), spaces, and other symbols. Do not start a host name with a number and do not end with a hyphen. This rule also applies to Delivery Controller host names.

Step 1c: Check the host name

Restart the machine and verify that the host name is set correctly:

```
hostname
```

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

Verify that the FQDN is set correctly:

```
hostname -f
```

This command returns the FQDN of the machine.

Step 1d: 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:

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

To:

```
hosts: files dns
```

Step 1e: Check name resolution and service reachability

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

```
nslookup domain-controller-fqdn

ping domain-controller-fqdn

nslookup delivery-controller-fqdn
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
6
7 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 (chrony)

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

Install chrony:

```
1 apt-get install chrony
```

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

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

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

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

Save changes and restart the Chrony daemon:

```
1 sudo systemctl restart chrony
```

Step 1g: Install packages

```
1 sudo apt-get install -y libsasl2-2
2
3 sudo apt-get install -y libgtk2.0-0
```

Step 1h: Add repositories to install necessary dependencies

For Debian 11, add the `deb http://deb.debian.org/debian/ bullseye main` line to the /etc/apt/sources.list file.

Step 1i: Install and specify a database to use

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> - We recommend you use SQLite for VDI mode only and use PostgreSQL for a hosted shared desktops delivery model.
> - For easy install and MCS, you can specify SQLite or PostgreSQL to use without having to install them manually. Unless otherwise specified through **/etc/xdl/db.conf**, the Linux VDA uses PostgreSQL by default.
> - For manual installations, you must install SQLite, PostgreSQL, or both manually. If you install both SQLite and PostgreSQL, you can specify one of them to use by editing **/etc/xdl/db.conf** after installing the Linux VDA package.

This section describes how to install PostgreSQL and SQLite and how to specify either of them to use.

Install PostgreSQL Run the following commands to install PostgreSQL:

```
sudo apt-get update

sudo apt-get install -y postgresql

sudo apt-get install -y libpostgresql-jdbc-java
```

Run the following commands to start PostgreSQL upon machine startup or immediately, respectively:

```
sudo systemctl enable postgresql

sudo systemctl start postgresql
```

Install SQLite For Debian, run the following command to install SQLite:

```
sudo apt-get install -y sqlite3
```

Specify a database to use If you install both SQLite and PostgreSQL, you can specify one of them to use by editing **/etc/xdl/db.conf** after installing the Linux VDA package.

1. Run **/opt/Citrix/VDA/sbin/ctxcleanup.sh**. Omit this step if it is a fresh installation.
2. Edit **/etc/xdl/db.conf** to specify a database to use.
3. Run **ctxsetup.sh**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> You can also use **/etc/xdl/db.conf** to configure the port number for PostgreSQL.

Step 2: Prepare the hypervisor

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

Fix time synchronization on XenServer (formerly Citrix Hypervisor)

When the XenServer Time Sync feature is enabled, within each paravirtualized Linux VM you experience issues with NTP and XenServer. Both try to manage the system clock. To avoid the clock becoming out of sync with other servers, make sure 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.

If you are running a paravirtualized Linux kernel with XenServer VM 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 restart, edit the **/etc/sysctl.conf file** and add the line:

```
xen.independent_wallclock = 1
```

To verify these changes, restart the system:

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

This command returns the value 1.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Fix time synchronization on Microsoft Hyper-V

Linux VMs with Hyper-V Linux Integration Services installed can use the Hyper-V time synchronization feature to use the host operating system's time. To ensure that the system clock remains accurate, enable this feature 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 (formerly Citrix Hypervisor), 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 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 VM to the Windows domain

The following methods are available for adding Linux machines to the Active Directory (AD) domain:

- Samba Winbind
- Quest authentication service
- Centrify DirectControl
- SSSD
- PBIS

Follow instructions based on your chosen method.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

Samba Winbind

Install or update the required packages

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

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

```
sudo systemctl enable winbind
```

> **Note:**
>
> Ensure that the `winbind` script is located under `/etc/init.d`.

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

> **Note:**
>
> Configure Kerberos based on your AD infrastructure. The following settings are meant for the single-domain, single-forest model.

```
[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** parameter 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**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Configure Winbind Authentication Open **/etc/samba/smb.conf** by running the **vim /etc/samba/smb.conf** command, and then make the following settings:

```
[global]
workgroup = WORKGROUP
security = ADS
realm = REALM
encrypt passwords = yes
idmap config *:range = 16777216-33554431
kerberos method = secrets and keytab
winbind refresh tickets = yes
template shell = /bin/bash
```

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

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

```
passwd: files systemd winbind
group: files systemd winbind
```

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

```
sudo net ads join <Kerberos realm name in uppercase> -U <domain user
    with permission to add computers to the domain>
```

Restart Winbind

```
sudo systemctl restart winbind
```

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

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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, verify that the system **keytab** file has been created and contains valid keys:

```
sudo klist -ke
```

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

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

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

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

```
sudo klist
```

Examine the account details of the machine using:

```
sudo net ads status
```

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

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

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
ssh localhost -l domain\\username

id -u
```

> **Note:**
>
> To run an SSH command successfully, ensure that SSH is enabled and working properly.

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

```
ls /tmp/krb5cc_uid
```

Verify 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. Proceed to Step 6: Install the Linux VDA after the domain joining verification.

> **Tip:**
>
> If you succeed in user authentication but cannot show your desktop when logging on with a domain account, restart the machine and 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.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

Configure Quest on Linux VDA

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

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

```
SELINUX=disabled
```

This change requires a machine restart:

```
1 reboot
```

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

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

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

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

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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.

Restart the Linux machine after domain joining.

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

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

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

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

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

```
ssh localhost -l domain\\username

id -u
```

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

```
ls /tmp/krb5cc_uid
```

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

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

Exit the session.

```
exit
```

Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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
```

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

More comprehensive system and diagnostic information is available using:

```
adinfo --sysinfo all

adinfo --diag
```

Test connectivity to the various Active Directory and Kerberos services.

```
adinfo --test
```

Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

SSSD

Configure Kerberos Run the following command to install Kerberos:

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

To configure Kerberos, open **/etc/krb5.conf** as root and set the parameters:

> **Note:**
>
> Configure Kerberos based on your AD infrastructure. The following settings are meant for the

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> single-domain, single-forest model.

```
[libdefaults]
default_realm = REALM
dns_lookup_kdc = false
rdns = 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` parameter in this context is the DNS domain name, such as example.com. The *REALM* is the Kerberos realm name in uppercase, such as EXAMPLE.COM.

Join the domain SSSD must be configured to use Active Directory as its identity provider and Kerberos for authentication. However, SSSD does not provide AD client functions for joining the domain and managing the system keytab file. You can use **adcli**, **realmd**, or **Samba** instead.

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

- **If you use adcli to join the domain, complete the following steps:**

1. Install **adcli**.

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

2. 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
   ```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

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

3. Verify system keytab.

   Run the `sudo klist -ket` command to ensure that the system keytab file has been created.

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

- **If you use Samba to join the domain, complete the following steps:**

1. Install the package.

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

2. 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.

3. Join the domain with **Samba**.

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

```
sudo net ads join <the Kerberos realm name in uppercase> -U <
    domain user with permission 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:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

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 created manually. 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, `Active Directory` must be able to provide POSIX extensions. PAM service

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> `ctxhdx` is added to ad_gpo_map_remote_interactive.
>
> The **domain-dns-name** parameter 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.
>
> For information about the configuration settings, see the man pages for sssd.conf and `sssd-ad`.

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

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

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

```
sudo systemctl start sssd

sudo systemctl enable sssd
```

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

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

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

- If you use **adcli** to verify domain membership, run the `sudo adcli info domain-dns-name` command to show the domain information.
- If you use **Samba** to verify domain membership, run the `sudo net ads testjoin` command to verify that the machine is joined to a domain and the `sudo net ads info` 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, verify 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
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

Verify that the TGT 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 to the Linux VDA using a domain user account that has not been used before.

```
ssh localhost -l domain\\username

id -u

klist

exit
```

Verify that the Kerberos tickets returned by the **klist** command are correct for that user and have not expired.

As a root user, verify 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. Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

PBIS

Download the required PBIS package

```
sudo wget https://github.com/BeyondTrust/pbis-open/releases/download
   /9.1.0/pbis-open-9.1.0.551.linux.x86_64.deb.sh
```

Make the PBIS installation script executable

```
sudo chmod +x pbis-open-9.1.0.551.linux.x86_64.deb.sh
```

Run the PBIS installation script

```
sudo sh pbis-open-9.1.0.551.linux.x86_64.deb.sh
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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 /opt/pbis/bin/domainjoin-cli join domain-name user
```

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

Note: To set Bash as the default shell, run the **sudo /opt/pbis/bin/config LoginShellTemplate/bin/bash** command.

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

```
/opt/pbis/bin/domainjoin-cli query
```

If the machine is joined to a domain, this command returns the information about the currently joined AD domain and OU. Otherwise, only the host name appears.

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

```
sudo ssh localhost -l domain\\user

id -u
```

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

```
ls /tmp/krb5cc_uid
```

Exit the session.

```
exit
```

Proceed to Step 6: Install the Linux VDA after the domain-joining verification.

Step 4: Install .NET

In addition to the .NET Runtime, you must install .ASP.NET Core Runtime on all supported Linux distributions before you install or upgrade the Linux VDA. Version 6 is required for Amazon Linux 2. Version 8 is required for other distributions.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

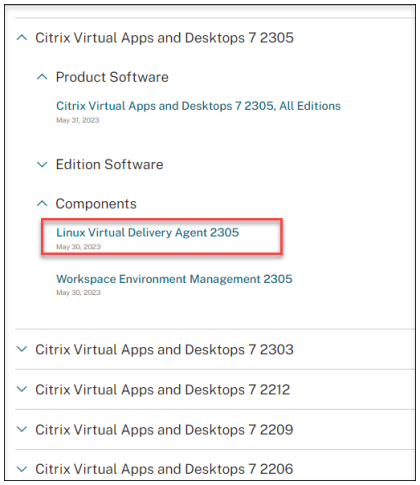
If your Linux distribution contains the .NET version that you require, install it from the built-in feed. Otherwise, install .NET from the Microsoft package feed. For more information, see https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers.

After installing .NET, run the **which dotnet** command to find your runtime path.

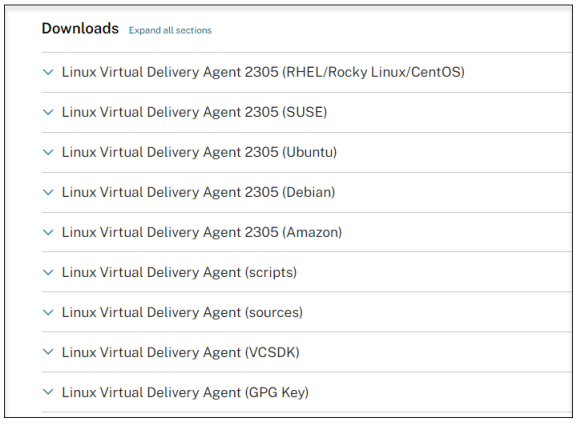
Based on the command output, set the .NET runtime binary path. For example, if the command output is /aa/bb/dotnet, use /aa/bb as the .NET binary path.

Step 5: Download the Linux VDA package

1. Go to the Citrix Virtual Apps and Desktops download page.
2. Expand the appropriate version of Citrix Virtual Apps and Desktops.
3. Expand **Components** to find the Linux VDA. For example:

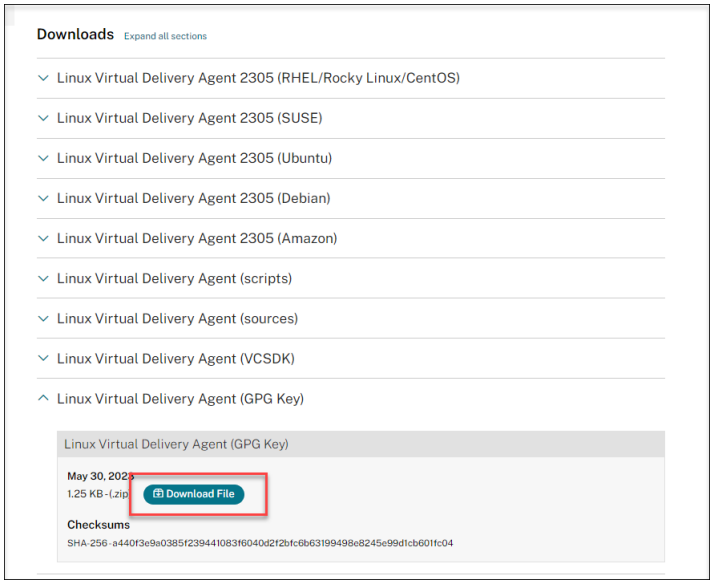


4. Click the Linux VDA link to access the Linux VDA downloads.



5. Download the Linux VDA package that matches your Linux distribution.
6. Download the GPG public key that you can use to verify the integrity of the Linux VDA package. For example:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



To verify the integrity of the Linux VDA package, run the following commands to import the public key into the DEB database and to check the package integrity:

```
sudo apt-get install dpkg-sig
gpg --import <path to the public key>
dpkg-sig --verify <path to the Linux VDA package>
```

Step 6: Install the Linux VDA

Step 6a: Install the Linux VDA

Install the Linux VDA software using the Debian package manager:

For Debian 12.5:

```
sudo dpkg -i xendesktopvda_<version>.debian12_amd64.deb
```

For Debian 11:

```
sudo dpkg -i xendesktopvda_<version>.debian11_amd64.deb
```

Dependency list for Debian 12.5:

```
openjdk-17-jdk >= 17

imagemagick >= 8:6.9.11

ufw >= 0.36

desktop-base >= 12.0.6

libxrandr2 >= 2:1.5.2
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
libxtst6 >= 2:1.2.3

libxm4 >= 2.3.8

util-linux >= 2.38

gtk3-nocsd >= 3

bash >= 5.2

findutils >= 4.9.0

sed >= 4.9

cups >= 2.4

ghostscript >= 10.0.0~

libmspack0 >= 0.11

ibus >= 1.5

libgoogle-perftools4 >= 2.10~

libpython3.11 >= 3.11~

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

libnss3-tools >= 2:3.87

libqt5widgets5 >= 5.15~

mutter >= 43.8

libqrencode4 >= 4.1.1

libimlib2 >= 1.10.0

libfuse2 >= 2.9.9

pulseaudio-utils >= 16.1
```

Dependency list for Debian 11:

```
libnss3-tools >= 2:3.61

libfuse2 >= 2.9

openjdk-11-jdk >= 11

imagemagick >= 8:6.9.10

```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
ufw >= 0.36

desktop-base >= 10.0.2

libxrandr2 >= 2:1.5.1

libxtst6 >= 2:1.2.3

libxm4 >= 2.3.8

util-linux >= 2.33

gtk3-nocsd >= 3

bash >= 5.0

findutils >= 4.6.0

sed >= 4.7

cups >= 2.2

ghostscript >= 9.53~

libmspack0 >= 0.10

ibus >= 1.5

libgoogle-perftools4 >= 2.7~

libpython3.9 >= 3.9~

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

libqt5widgets5 >= 5.5~

mutter >= 3.38.6~

libqrencode4 >= 4.0.0

libimlib2 >= 1.5.1
```

> **Note:**
>
> For a matrix of the Linux distributions and the Xorg versions that this version of the Linux VDA supports, see System requirements.

Step 6b: Upgrade the Linux VDA (optional)

The Linux VDA supports upgrades from the most recent version. For example, you can upgrade the Linux VDA from 2308 to 2311 and from 1912 LTSR to 2203 LTSR.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
sudo dpkg -i <PATH>/<Linux VDA deb>

sudo apt-get install -f
```

Note:

- Upgrading an existing installation overwrites the configuration files under /etc/xdl. Before you conduct an upgrade, make sure to back up the files.
- Starting with the 2407 release, the Linux VDA delegates package managers **rpm** or **dpkg** to handle configuration files during upgrades. The following describes how rpm and dpkg interact with changes to configuration files:
  - **rpm**: by default keeps the local version and saves the new version from the package with a .rpmnew extension.
  - **dpkg**: interactively prompts you with a choice on how to proceed. To silently upgrade the Linux VDA while retaining your local configuration file and saving the new package version as **.dpkg-new** or **.dpkg-dist**, use the following command:

    ```
    dpkg --force-confold -i package.deb  # Always keep your version, then save new package's version as *.dpkg-new or *.dpkg-dist
    ```

Step 7: Install NVIDIA GRID drivers

Enabling HDX 3D Pro requires you to install the NVIDIA GRID drivers on your hypervisor and on the VDA machines.

To install and configure the NVIDIA GRID Virtual GPU Manager (the host driver) on the specific hypervisors, see the following guides:

- XenServer
- VMware ESX
- Nutanix AHV

To install and configure the NVIDIA GRID guest VM drivers, perform the following general steps:

1. Ensure that the guest VM is shut down.
2. In the hypervisor control panel, allocate a GPU to the VM.
3. Start the VM.
4. Install the guest VM driver on the VM.

Step 8: Configure the Linux VDA

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> Before setting up the runtime environment, ensure that the **en_US.UTF-8** locale is installed on your OS. If the locale is not available on your OS, run the **sudo locale-gen en_US.UTF-8** command. For Debian, edit the **/etc/locale.gen** file by uncommenting the **# en_US.UTF-8 UTF-8** line and then run the **sudo locale-gen** command.

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_NON_DOMAIN_JOINED='y|n'**–Whether to join the machine to a domain. The default value is 'n'. For domain-joined scenarios, set it to 'n'.
- **CTX_XDL_AD_INTEGRATION='winbind|sssd|centrify|pbis|quest'** –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.
- **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_VDI_MODE='y|n'**–Whether to configure the machine as a dedicated desktop delivery model (VDI) or hosted shared desktop delivery model. For HDX 3D Pro environments, set the value to **'y'**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- **CTX_XDL_HDX_3D_PRO='y|n'**–The Linux VDA supports HDX 3D Pro, a set of GPU acceleration technologies designed to optimize the virtualization of rich graphics applications. If HDX 3D Pro is selected, the VDA is configured for VDI desktops (single-session) mode - (that is, CTX_XDL_VDI_MODE=**'y'**).
- **CTX_XDL_START_SERVICE='y|n'**–Determines whether the Linux VDA services are started when the configuration is complete.
- **CTX_XDL_REGISTER_SERVICE='y|n'**–The Linux Virtual Desktop services are started after machine startup.
- **CTX_XDL_ADD_FIREWALL_RULES='y|n'**–The Linux VDA services require incoming network connections to be allowed through the system firewall. You can open the required ports (by default ports 80 and 1494) automatically in the system firewall for the Linux Virtual Desktop.
- **CTX_XDL_DESKTOP_ENVIRONMENT=gnome/gnome-classic/kde/mate/'<none>'** –Specifies the GNOME, GNOME Classic, or MATE desktop environment to use in sessions. If you set it to '**<none>**', the default desktop configured on the VDA is used.

  You can also switch between desktop environments by running commands or using the system tray. For more information, see Desktop switching commands and System tray.
- **CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime** –The path to install .NET for supporting the new broker agent service (ctxvda). The default path is **'/usr/bin'**.
- **CTX_XDL_VDA_PORT=port-number** –The Linux VDA communicates with Delivery Controllers through a TCP/IP port.
- **CTX_XDL_SITE_NAME=<dns-name>** –The Linux VDA discovers LDAP servers through DNS. To limit the DNS search results to a local Site, specify a DNS Site name. If unnecessary, set to **'<none>'**.
- **CTX_XDL_LDAP_LIST='<list-ldap-servers>'** –The Linux VDA queries DNS to discover LDAP servers. If DNS cannot provide LDAP service records, you can provide a space-separated list of LDAP FQDNs with LDAP ports. For example, ad1.mycompany.com:389 ad2.mycompany.com:3268 ad3.mycompany.com:3268. To enable faster LDAP queries within an Active Directory forest, enable Global Catalog on a domain controller and specify the relevant LDAP port number as 3268. This variable is set to **'<none>'** by default.
- **CTX_XDL_SEARCH_BASE=search-base-set** –The Linux VDA queries LDAP through a search base set to the root of the Active Directory Domain (for example, DC=mycompany,DC=com). To improve search performance, you can specify a search base (for example, OU=VDI,DC=mycompany,DC=com). If unnecessary, set to **'<none>'**.
- **CTX_XDL_SUPPORT_DDC_AS_CNAME='y|n'**–The Linux VDA supports specifying a Delivery Controller name using a DNS CNAME record.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Set the environment variable and run the configure script:

```
export CTX_XDL_NON_DOMAIN_JOINED='n'
export CTX_XDL_AD_INTEGRATION=sssd|winbind|centrify|pbis|quest
export CTX_XDL_DDC_LIST='<list-ddc-fqdns>'
export CTX_XDL_VDI_MODE='y|n'
export CTX_XDL_HDX_3D_PRO='y|n'
export CTX_XDL_START_SERVICE='y|n'
export CTX_XDL_REGISTER_SERVICE='y|n'
export CTX_XDL_ADD_FIREWALL_RULES='y|n'
export CTX_XDL_DESKTOP_ENVIRONMENT=gnome|gnome-classic|kde|mate|'<none>
    '
export CTX_XDL_DOTNET_RUNTIME_PATH='<path-to-install-dotnet-runtime>'
export CTX_XDL_VDA_PORT='<port-number>'
export CTX_XDL_SITE_NAME='<dns-site-name>'|'<none>'
export CTX_XDL_LDAP_LIST='<list-ldap-servers>'|'<none>'
export CTX_XDL_SEARCH_BASE='<search-base-set>'|'<none>'
export CTX_XDL_SUPPORT_DDC_AS_CNAME='y|n'
sudo -E /opt/Citrix/VDA/sbin/ctxsetup.sh --silent
```

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

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

```
sudo CTX_XDL_NON_DOMAIN_JOINED='n' \
CTX_XDL_AD_INTEGRATION=sssd|winbind|centrify|pbis|quest \
CTX_XDL_DDC_LIST='<list-ddc-fqdns>' \
CTX_XDL_VDI_MODE='y|n' \
CTX_XDL_HDX_3D_PRO='y|n' \
CTX_XDL_START_SERVICE='y|n' \
CTX_XDL_REGISTER_SERVICE='y|n' \
CTX_XDL_ADD_FIREWALL_RULES='y|n' \
CTX_XDL_DESKTOP_ENVIRONMENT=gnome|gnome-classic|kde|mate|'<none>' \
CTX_XDL_DOTNET_RUNTIME_PATH='<path-to-install-dotnet-runtime>' \
CTX_XDL_VDA_PORT='<port-number>' \
CTX_XDL_SITE_NAME='<dns-site-name>'|'<none>' \
CTX_XDL_LDAP_LIST='<list-ldap-servers>'|'<none>' \
CTX_XDL_SEARCH_BASE='<search-base-set>'|'<none>' \
CTX_XDL_SUPPORT_DDC_AS_CNAME='y|n' \
/opt/Citrix/VDA/sbin/ctxsetup.sh --silent
```

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
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

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
```

To uninstall the Linux VDA software:

```
dpkg -r xendesktopvda
```

> **Note:**
>
> Uninstalling the Linux VDA software deletes the associated PostgreSQL and other configuration data. However, the PostgreSQL package and other dependent packages that were set up before the installation of the Linux VDA are not deleted.
>
> **Tip:**
>
> The information in this section does not cover the removal of dependent packages including PostgreSQL.

Step 9: Run XDPing

Run `sudo /opt/Citrix/VDA/bin/xdping` to check for common configuration issues with a Linux VDA environment. For more information, see **XDPing**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 10: 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
```

> **Note:**
>
> Before you stop the **ctxvda** and **ctxhdx** services, run the **systemctl stop ctxmonitord** command to stop the monitor service daemon. Otherwise, the monitor service daemon restarts the services you stopped.

Restart the Linux VDA:

To restart the Linux VDA services:

```
sudo systemctl stop ctxvda

sudo systemctl restart ctxhdx

sudo systemctl restart ctxvda
```

Check the Linux VDA status:

To check the running status of the Linux VDA services:

```
sudo systemctl status ctxvda

sudo systemctl status ctxhdx
```

Step 11: Create machine catalogs

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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 **Multi-session OS** option for a hosted shared desktops delivery model.
  - The **Single-session OS** option for a VDI dedicated desktop delivery model.
- Do not mix Linux and Windows VDA machines in the same machine catalog.

> **Note:**
>
> Early versions of Citrix Studio did not support the notion of a "Linux OS."However, selecting the **Windows Server OS** or **Server OS** option implies an equivalent hosted shared desktops delivery model. Selecting the **Windows Desktop OS** or **Desktop OS** option implies a single user per machine delivery model.
>
> **Tip:**
>
> If you remove and rejoin a machine to the Active Directory domain, you must remove and add the machine to the machine catalog again.

Step 12: Create delivery groups

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:

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

For information about how to create machine catalogs and delivery groups, see Citrix Virtual Apps and Desktops 7 2407.

Configure

June 20, 2024

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

Administration

June 20, 2024

This section contains the following topics:

- Linux VDA data collection program
- HDX Insight
- Integration with the Citrix Telemetry Service
- Linux VDA self-update for Citrix DaaS Standard for Azure
- Linux VM and Linux session metrics
- Log collection
- Session shadowing
- The monitor service daemon
- Troubleshooting
- Others
  - Citrix Workspace app for HTML5 support
  - Create a **Python3** virtual environment
  - Integrate NIS with Active Directory
  - **IPv6**
  - LDAPS
  - **Xauthority**

Linux VDA data collection program

July 3, 2024

You automatically participate in the data collection program after installing the Linux VDA. The data collection program collects statistics and usage data and sends the data to Citrix Analytics to help

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

improve the quality and performance of Citrix products. The upload of data occurs when you launch a session.

Registry settings

The Linux VDA data collection program is enabled by default. You can enable or disable the program through the following registry setting:

Location: HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\CEIP

Name: **CEIPSwitch**

Value: 1 = disabled, 0 = enabled (default)

When unspecified, the Linux VDA data collection program is enabled.

You can run the following command to disable the Linux VDA data collection program:

```
1 /opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SOFTWARE\
     Citrix\CEIP" -v "CEIPSwitch" -d "1"
```

Data collected from the Linux VDA

The following table gives an example of the types of data collected.

Data Point	Key Name	Description
Machine GUID	**machine_guid**	A GUID string identifying the machine where the data originates
Linux OS name and version	**os_name_version**	A string denoting the Linux OS name and version of the machine
AD solution	**ad_solution**	A string denoting the machine's domain-joining method
Linux kernel version	**kernel_version**	A string denoting the machine's kernel version
GPU type	**gpu_type**	Denoting the GPU type of the VDA machine
CPU cores	**cpu_cores**	An integer denoting the number of CPU cores of the machine

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Data Point	Key Name	Description
CPU frequency	**cpu_frequency**	Float denoting the CPU frequency in MHz
Physical memory size	**memory_size**	An integer denoting the physical memory size in KB
LVDA version	**vda_version**	A string denoting the installed version of the Linux VDA.
LVDA update or fresh install	**update_or_fresh_install**	A string denoting the current Linux VDA package is being freshly installed or updated
VDI mode enabled or not	**vdi_mode**	A string denoting whether VDI mode is enabled
LVDA installed method	**install_method**	A string denoting that the current Linux VDA package is installed by using MCS, PVS, easy install, or manual installation.
HDX 3D Pro enabled or not	**hdx_3d_pro**	A string denoting whether HDX 3D Pro is enabled on the machine
BCR enabled or not	**bcr**	A string denoting whether Browser content redirection (BCR) is enabled on this machine
System Locale	**system_locale**	A string denoting the locale of this machine
Farm Id	**farm_id**	A string denoting the farm ID.
VDA virtualization type	**vda_virtualization**	A string denoting the hypervisor that created the VDA
Session key	**session_key**	Identifying the session where the data originates
Resource type	**resource_type**	A string denoting the resource type of the launched session: desktop or <appname>
Receiver client type	**receiver_type**	An integer denoting the type of Citrix Workspace app used to launch the session

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Data Point	Key Name	Description
Receiver client version	**receiver_version**	A string denoting the version of Citrix Workspace app used to launch the session
Printing count	**printing_count**	An integer denoting the number of times the session uses the printing function
USB redirection count	**usb_redirecting_count**	An integer denoting the number of times the session uses a USB device
Gfx provider type	**gfx_provider_type**	A string denoting the graphics provider type of the session
Shadowing count	**shadow_count**	An integer denoting the number of times the session has been shadowed
User selected Language	**ctxism_select**	A composed long string that contains all languages that users have selected
Smartcard redirecting count	**scard_redirecting_count**	An integer denoting the number of times smart card redirection is used for session logons and user authentication for in-session apps
Video codec type	**grahpic_video_codec_type**	A string denoting which video codec is being used for Thinwire.
Logon credential type	**credentials_type**	An integer denoting the credential type used to log on to the Linux VDA
Watermark	**watermark**	A string denoting whether session watermark is enabled or not
Watermark transparency	**watermark_transparency**	An integer denoting the watermark transparency
Watermark custom text len	**watermark_custom_text_len**	An integer denoting the length of watermark custom text
MTU	**mtu**	A string denoting whether the Maximum Transmission Unit (MTU) is used in this session

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Data Point	Key Name	Description
MTU MSS	**mtu_mss**	An integer denoting the Maximum Segment Size (MSS)
File Transfer	**filetrans**	An integer denoting the file transfer policy settings
File Transfer Upload Count	**filetrans_upload_count**	An integer denoting the number of times the “Upload” icon is used for file transfers in the session
File Transfer Download Count	**filetrans_download_count**	An integer denoting the number of times the “Download”icon is used for file transfers in the session

HDX Insight

June 20, 2024

Overview

The Linux VDA partially supports the HDX Insight feature.

Installation

No dependent packages need installation.

Usage

HDX Insight analyzes the ICA messages passed through the Citrix ADC between Citrix Workspace app and the Linux VDA. All HDX Insight data is sourced from the NSAP virtual channel and sent uncompressed. The NSAP virtual channel is enabled by default.

The following commands disable and enable the NSAP virtual channel, respectively:

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\
    VirtualDesktopAgent" -t "REG_DWORD" -v "EnableNSAP" -d "0x00000000"
    --force
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

Troubleshooting

No data points are displayed

There might be two causes:

- HDX Insight is not configured correctly.

  For example, AppFlow is not enabled on the Citrix ADC or an incorrect Citrix ADC instance is configured on the Citrix ADM.

- The ICA Control Virtual Channel is not started on the Linux VDA.

  ```
  ps aux | grep -i ctxctl
  ```

  If `ctxctl` is not running, contact your administrator to report a bug to Citrix.

No application data points are displayed

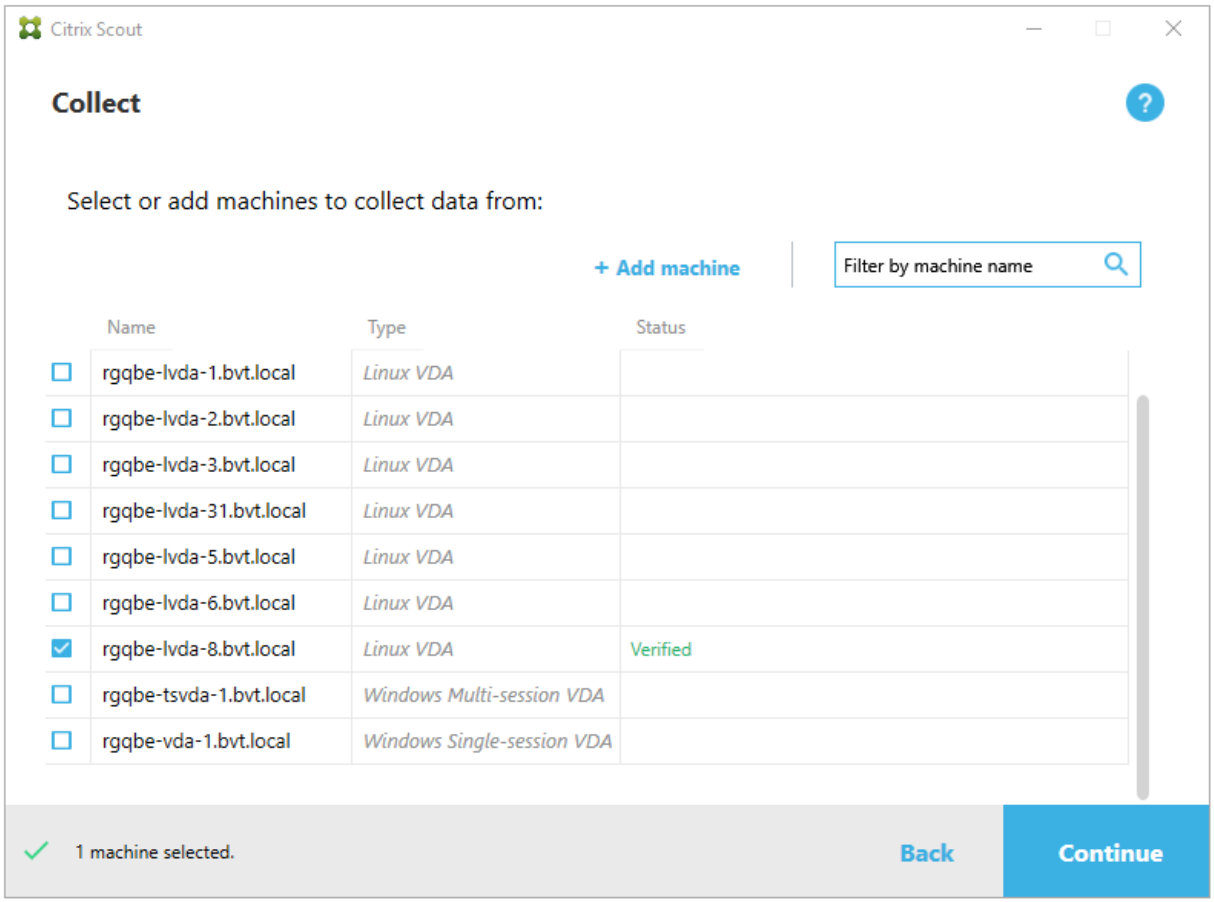
Verify that the seamless virtual channel is enabled and a seamless application is running.

Integration with the Citrix Telemetry Service

June 20, 2024

With the Citrix Telemetry Service (`ctxtelemetry`) integrated with the Linux VDA software, you can run Citrix Scout, which then uses the /opt/Citrix/VDA/bin/xdlcollect.sh script, to collect logs about the Linux VDA.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



Enable and disable the Citrix Telemetry Service

- To enable the service, run the **sudo systemctl enable ctxtelemetry.socket** command.
- To disable the service, run **sudo systemctl disable ctxtelemetry.socket**.

Ports

The Citrix Telemetry Service (`ctxtelemetry`), by default, uses TCP/IP port 7503 to listen for Citrix Scout. It uses TCP/IP port 7502 on the Delivery Controller to communicate with Citrix Scout.

To change ports after you have your VDA installed, do the following:

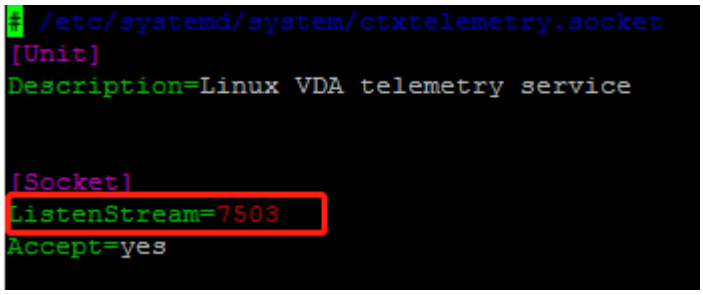
1. To change a port for communicating with Scout, run the following command.

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\
    VirtualDesktopAgent" -t REG_DWORD -v "TelemetryServicePort" -d
    <port number> --force
```

2. To change the socket port for listening for Scout, run the following command to open and edit the ctxtelemetry.socket file.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
sudo vi /etc/systemd/system/ctxtelemetry.socket
```



3. Run the following commands to restart the socket port.

```
sudo systemctl daemon-reload
sudo systemctl stop ctxtelemetry.socket
sudo systemctl start ctxtelemetry.socket
```

4. Enable the new ports in your firewall configuration.

   If you are using Ubuntu, for example, run the **sudo ufw allow 7503** command to enable port 7503.

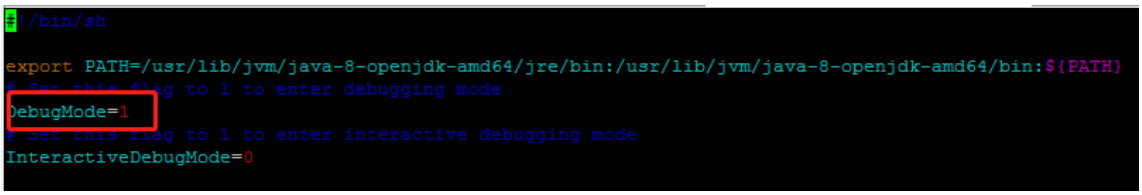
> **Note:**
>
> You can also run ctxsetup.sh instead to automate the preceding steps 3 and 4.

Debug mode

If the Citrix Telemetry Service does not work as expected, you can enable debug mode to determine the causes.

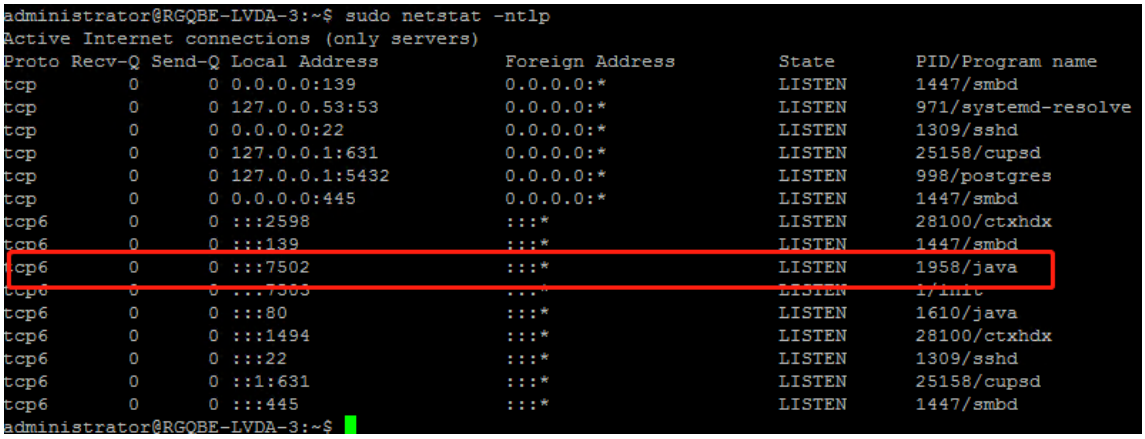
1. To enable debug mode, run the following command to open the `ctxtelemetry` file and then change the DebugMode value to 1.

```
sudo vi /opt/Citrix/VDA/sbin/ctxtelemetry
```



2. Manually stop the Citrix Telemetry Service, or wait 15 minutes for the service to stop automatically.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



In this example, you can run the following commands to stop the Citrix Telemetry Service.

```
sudo netstat -ntlp
Kill -9 1958
```

3. To restart the Citrix Telemetry Service, select your Linux VDA on Scout and find telemetry-debug.log in /var/log/xdl/.

Service wait time

The `systemd` daemon that opens the socket port starts by default and uses few resources. The Citrix Telemetry Service stops by default and starts only when there is a log collection request from the Delivery Controller. After log collection completes, the service awaits new collection requests for a duration of 15 minutes and stops again if there are not any. You can configure the wait time through the following command. The minimum value is 10 minutes. If you set a value less than 10 minutes, the minimum value, 10 minutes, takes effect. After setting the wait time, stop and restart the service.

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\
   VirtualDesktopAgent" -v "TelemetryServiceIdleTimeoutInMinutes" -d <
   number> -t REG_DWORD
```

Verification tests

Before a collection starts, verification tests run automatically for each selected machine. These tests ensure that the requirements are met. If a test fails for a machine, Scout displays a message with suggested corrective actions. For more information about verification tests, see the Verification tests section in the Citrix Scout documentation.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Linux VDA self-update through Azure

June 26, 2024

This feature helps to automatically update your Linux VDA software - immediately, or at a scheduled time. It is beneficial when you create Linux VDAs in Citrix DaaS Standard for Azure (formerly Citrix Virtual Apps and Desktops Standard for Azure). You need no administrator privileges of the VMs in Azure. For more information, see Create Linux VDAs in Citrix DaaS Standard for Azure.

Configuration

To use this feature, complete the following steps:

Step 1: Upload update information and new VDA packages to your Azure container

Step 1a: Create a container under your Azure storage account and set your container access level to **Blob (Anonymous read access for blobs only)**.

> **Note:**
>
> Azure containers and blobs are exclusively held and managed by customers. Citrix is not liable for any security issues with them. To ensure data security and cost efficiency, set your container access level to **Private (no anonymous access)** after each **self-update**.

Step 1b: Incorporate your VDA update information to a JSON file named UpdateInfo.json. For an example of the file format, see the following block:

```
{

 "Version": "21.04.200.4",
"Distributions":[
{

"TargetOS": "RHEL7_9",
"PackageName": "",
"PackageHash": ""
 }
 ,
{

"TargetOS": "UBUNTU20_04",
"PackageName": "",
"PackageHash": ""
 }

```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
19 ]
20  }
```

Where, **"Version"** indicates the new VDA version and **"Distributions"** is an array of update objects. Each object contains three items:

- **"TargetOS"**: must be "RHEL7_9"(for RHEL 7, CentOS 7, and Amazon Linux 2) or "UBUNTU20_04." The **ctxmonitord**' does not recognize any other distributions.
- **"PackageName"**: Full name of the VDA package of the specified version.
- **"PackageHash"**: SHA-256 value that you compute by using the `shasum -a 256 <pkgname>` command.

Step 1c: Upload the JSON file and the new version of Linux VDA packages to your Azure container.

Step 2: Enable the self-update feature on the master image or on each VDA

By default, **self-update** is disabled. If you create Linux VDAs in Citrix DaaS Standard for Azure, the feature enablement must be conducted on the master image. Otherwise, enable the feature on each target VDA directly.

To enable **self-update**, run commands similar to the following to edit the registry key at HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\SelfUpdate.

```
1 /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet\
     Control\Citrix\SelfUpdate" -t "REG_DWORD" -v "fEnabled" -d "0
     x00000001" --force
2
3 /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet\
     Control\Citrix\SelfUpdate" -t "REG_SZ" -v "ScheduledTime" -d "
     Immediately" --force
4
5 /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet\
     Control\Citrix\SelfUpdate" -t "REG_SZ" -v "Url" -d "<Your-Azure-
     Container-Url>" --force
6
7 /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet\
     Control\Citrix\SelfUpdate" -t "REG_SZ" -v "CaCertificate" -d "<Local
     -Certificate-Path-of-PortalAzureCom>" --force
```

The following table describes the registry settings.

Registry setting	Description
fEnabled	This setting is required. By default, the value is 0, which means **self-update** is disabled. You can set it to 1 to enable **self-update**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Registry setting	Description
Url	This setting is required. It sets the URL of your Azure container to get the update information and new VDA packages.
ScheduledTime	This setting is required. You can set it to **Immediately** or **NextStart**. **Immediately** means to run an update immediately after downloading VDA packages. This option is appropriate when the download speed is high and your update is urgent. But it can disrupt the user experience if there are any live sessions when you download the package. **NextStart** means to run an update upon the next start of the **ctxmonitord**. This option is appropriate when the download speed is not high and your update is not urgent.
CaCertificate	This setting is optional. It sets the full path of a PEM certificate to verify the URL of your Azure container. For Azure blobs, it can be the certificate of portal.azure.com that is retrieved from the browser and then converted to PEM. For security, we recommend you add this registry setting, but it is NOT supported on Ubuntu or Debian, because the `curl` command can't work as expected when verifying the peer using the specified certificate file on Ubuntu or Debian. Make sure to set the least privileges of the certificate.

When the **ctxmonitord** restarts, it first queries **Url** to get the UpdateInfo.json file and retrieves the update version from the JSON file. Then the **ctxmonitord** compares the update version with the current version. If the current version is earlier, the service downloads the new version of the VDA package from Azure and saves it locally. After that, it runs an update according to the setting of **ScheduledTime**. For an on-premises deployment, you can restart the **ctxmonitord** directly to trigger the update. However, in Citrix DaaS Standard for Azure where you have no administrator privileges to the VMs, the **ctxmonitord** can be restarted only after the VDA machine is restarted. If an update fails, your VDA is rolled back to the existing version.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> - The registry settings you configured on the master image cannot be changed.
> - If all VMs in an environment download a package at the same time, the local network can be congested.
> - To download from the provided URL, you must install `curl` on the target VM.
> - User data is lost if both an update and rollback fail.
> - If an update fails but rollback succeeds, users on the same network might have different versions of the Linux VDA. This case is suboptimal.
> - An update typically takes several minutes to complete. There is no status indicator in Citrix Studio.

Linux VM and Linux session metrics

June 20, 2024

The following table lists the metrics available for Linux VMs and Linux sessions.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Metric	Min. VDA Version Required	Description	Remarks
Logon duration	2402	The total logon time is available with the Linux VDA 2109 and later. It is a measure of the logon process from the time that a user connects from Citrix Workspace app to the time a session is ready to use. The time taken for each phase of the logon process, such as **HDX connection**, **Authentication**, and **GPOs**, is available with the Linux VDA 2402 and later. To view the logon duration metrics, select the **Logon Performance** tab on the **Trends** view.	Available in Citrix Director and Monitor.
ICA latency	2311	ICA latency is the network latency which indicates if the network is sluggish. Open the **Session Details** view to access this metric.	Available in Citrix Director and Monitor.
Policies	2311	All policies that are in effect for the current session are displayed on the **Policies** tab in the **Session Details** view.	Available in Citrix Director and Monitor.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Metric	Min. VDA Version Required	Description	Remarks
Session auto reconnect count	2109	To view the number of auto reconnects in a session, access the **Trends** view. Set conditions and click **Apply** to narrow the search results. The **Session Auto Reconnect Count** column displays the number of auto reconnects in a session. **Auto reconnect** is enabled when the **Session Reliability** or the **Auto Client Reconnect** policies are in effect. For more information about session reconnections and relevant policies, see the following articles. Sessions Auto client reconnect policy settings Session reliability policy settings	Available in Citrix Director and Monitor.
Idle time	2103	To access this metric, open the **All Sessions** page by selecting **Filters > Sessions > All Sessions**.	Available in Citrix Director and Monitor.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Metric	Min. VDA Version Required	Description	Remarks
Metrics of a Linux VM	2103	The metrics for Linux VMs are: the number of CPU cores, memory size, hard disk capacity, and current and historical CPU and memory utilization	Available in Citrix Director and Monitor.
Protocol	1909	The transport protocol of a Linux session appears as UDP or TCP in the **Session Details** view.	Available in Citrix Director and Monitor.
ICA RTT	1903	ICA Round Trip Time (RTT) is the elapsed time from when you press a key until the response appears on the endpoint. To obtain ICA RTT metrics, create the **ICA round trip calculation** and **ICA round trip calculation interval** policies in Citrix Studio.	Available in Citrix Director and Monitor.

Log collection

June 26, 2024

Logging mechanism overview

The following table presents an overview of the logging mechanism for the Linux VDA.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Logging module	Log file name format	Log file name example	Logging scope	Maximum single file size	Rotation threshold (defaults)	Configuration
HDX	hdx.log[.n]	hdx.log, hdx.log.1	Graphics, login, audio, keyboard, mouse	200 MiB	1 current, 2 old	Configurable through the setlog utility
Jproxy	jproxy.log[.n.log]	jproxy.log, jproxy.log.1.log	VDA registration, user authentication	20 MiB	1 current, 10 old	Configurable through the setlog utility or /etc/xdl/log4j2.xml
VDA	vda.YYYY-MM-DD.hh.mm.ss.log	vda.2024-05-06.20.18.40.log	VDA registration	50 MiB	1 current, 1 old	Configurable through the setlog utility or /etc/xdl/brokeragent.conf

Note:

- The first log file doesn't have a number in its name, and subsequent files are numbered with ".n"where "n"represents the file number. For example, "hdx.log"is the first HDX log file and "hdx.log.1"is the second.
- The maximum size for a single log file is measured in Mebibytes (MiB).
- A log file that is being generated and has not yet reached the maximum size for a single file is referred to as a "current"log file. When a"current"log file reaches the maximum size for a single file, it is rolled over and becomes an "old"log file.
- The rotation threshold is configurable to limit the number of "old"log files that can be retained. The oldest log files will be deleted when the limit is reached.

Logging configuration

This section provides additional information about logging configuration, complementing the details outlined in the table above.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Logging enabled by default for the Linux VDA

The ctxlogd daemon and the setlog utility are 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.

Configure VDA logging through /etc/xdl/brokeragent.conf

> **Note:**
>
> If you're looking to configure only log levels for VDA logging without delving into other logging parameters such as the maximum size for a single log file, you can use the setlog utility described later in this article. Otherwise, use /etc/xdl/brokeragent.conf.

The /etc/xdl/brokeragent.conf file on the VDA is available for configuring VDA logging. For example:

```
<!-- VDA trace settings -->
<add key="TraceProvider.Type" value="QueuedLogFileProvider"/>
<add key="TraceProvider.BaseFilename" value="/var/log/xdl/vda.log"/>
<add key="TraceProvider.AppendMode" value="true"/>
<add key="TraceProvider.SizeLimit" value="50"/>
<add key="TraceProvider.TrailCount" value="2"/>
<add key="TraceProvider.ProcessSafe" value="true"/>
<add key="TraceProvider.CategoryFilter" value="All" />
```

Configure Jproxy logging through /etc/xdl/log4j2.xml

> **Note:**
>
> If you're looking to configure only log levels for Jproxy logging without delving into other logging parameters such as the maximum size for a single log file, you can use the setlog utility described later in this article. Otherwise, use /etc/xdl/log4j2.xml.

The following is an example of configuring Jproxy logging through /etc/xdl/log4j2.xml, of which the **SizeBasedTriggeringPolicy** parameter specifies the maximum size for a single Jproxy log file and the **DefaultRollerStrategy** parameter sets the total number of Jproxy log files that can be retained.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
<!-- Logging Properties -->
<Properties>
    <Property name="LOG_PATTERN">%d{yyyy-MM-dd HH:mm:ss.SSS}{GMT} [%-5p] [%tid] - %m%n</Property>
    <Property name="logfile">/var/log/xdl/jproxy.log</Property>
    <Property name="perflogfile">/var/log/xdl/jproxyperf.log</Property>
</Properties>

<Appenders>

  <Console name="CONSOLE" target="SYSTEM_OUT">
    <PatternLayout pattern="${LOG_PATTERN}"/>
  </Console>

  <RollingFile name="logfile" fileName="${sys:logfile}"
    filePattern="${sys:logfile}.%i.log">
    <PatternLayout pattern="${LOG_PATTERN}"/>
    <Policies>
        <SizeBasedTriggeringPolicy size="19500KB" />
    </Policies>
    <DefaultRolloverStrategy max="10"/>
  </RollingFile>

  <RollingFile name="perflogfile" fileName="${sys:perflogfile}"
    filePattern="${sys:perflogfile}.%i.log">
    <PatternLayout pattern="${LOG_PATTERN}"/>
    <Policies>
        <SizeBasedTriggeringPolicy size="19500KB" />
    </Policies>
    <DefaultRolloverStrategy max="10"/>
  </RollingFile>
```

Configure HDX logging through the setlog utility

The setlog utility resides 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 your configuration options including values (log file path, single file size limit, and rotation threshold) and log levels. Run the following command for help with the setlog utility:

```
setlog help
```

Values By default, HDX logs are saved under /**var/log/xdl/hdx.log**, the size limit for a single HDX log file is 200 MiB, and you can save up to two "old"HDX log files under **/var/log/xdl/hdx.log**.

To view all the current setlog values, run the following command:

```
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
```

To view or set a single setlog value, run the following command:

```
setlog value <name> [<value>]
```

For example:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
setlog value log_size 100
```

Levels By default, log levels are set to **info** (case-insensitive).

When you encounter an issue requiring troubleshooting, we recommend that you set the log level to **verbose** in most scenarios. Afterwards, reproduce the issue and collect logs.

To set log levels (including Disabled, Inherited, Trace, Verbose, Information, Warnings, Errors, and Fatal Errors), run the following command:

```
setlog level <class> [<level>]
```

Log Level	Command Parameter (Case-Insensitive)
Disabled	none
Inherited	inherit
Trace	trace
Verbose	verbose
Information	info
Warnings	warning
Errors	error
Fatal Errors	fatal

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

```
setlog level all error
```

To view all supported classes or components, run the following command:

```
setlog levels
```

Restore Defaults Revert all levels 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. We recommend that root users not

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> give write permission to other users. Failure to comply can cause the arbitrary or malicious configuration to **ctxlogd**, which can affect server performance and therefore the user experience.

Log collection

You can run the **bash /opt/Citrix/VDA/bin/xdlcollect.sh** command to collect logs. The **xdlcollect** Bash script used to collect logs is integrated into the Linux VDA software and located under **/opt/Citrix/VDA/bin**.

After the log collection completes, a ZIP file is generated under **/tmp/xdlcollect** on the VDA.

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.

Session shadowing

August 16, 2024

Shadowing sessions allows domain administrators to view users'ICA sessions in an intranet. The feature uses noVNC to connect to the ICA sessions.

> **Note:**
>
> To use the feature, use `Citrix Director` 7.16 or later.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Installation and configuration

Dependencies

Two new dependencies, `python-websockify` and `x11vnc`, are required for session shadowing. Install `python-websockify` and `x11vnc` manually after you install the Linux VDA.

For Amazon Linux2:

Run the following commands to install `python-websockify` and `x11vnc` (`x11vnc` version 0.9.13 or later):

```
sudo pip3 install websockify
sudo yum install x11vnc
```

For RHEL 9.x/8.x and Rocky Linux 9.x/8.x:

Run the following commands to install `python-websockify` and `x11vnc` (`x11vnc` version 0.9.13 or later).

```
sudo pip3 install websockify
sudo yum install x11vnc
```

Resolve `x11vnc` by enabling the EPEL and CodeReady Linux Builder repositories:

```
dnf install -y --nogpgcheck https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm

subscription-manager repos --enable "codeready-builder  -for-rhel-8-x86_64-rpms"
```

For Ubuntu:

Run the following commands to install `python-websockify` and `x11vnc` (`x11vnc` version 0.9.13 or later):

```
sudo pip3 install websockify
sudo apt-get install x11vnc
```

For SUSE:

Run the following commands to install `python-websockify` and `x11vnc` (`x11vnc` version 0.9.13 or later):

```
sudo pip3 install websockify
sudo zypper install x11vnc
```

For Debian 12:

Run the following commands to install `python-websockify` and `x11vnc` (`x11vnc` version 0.9.13 or later):

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
apt install python3-websockify
sudo apt-get install x11vnc
```

For Debian 11:

Run the following commands to install `python-websockify` and `x11vnc` (`x11vnc` version 0.9.13 or later):

```
sudo pip3 install websockify
sudo apt-get install x11vnc
```

Port

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

Registry

The following table lists related registries:

Registry	Description	Default Value
EnableSessionShadowing	Enables or disables the session-shadowing feature	1 (Enabled)
ShadowingUseSSL	Determines whether to encrypt the connection between the Linux VDA and Citrix Director	0 (Disabled)

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

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\Software\Citrix\
   VirtualDesktopAgent" -v "EnableSessionShadowing" -d "0x00000000"
```

SSL

The noVNC connection between the Linux VDA and Citrix Director uses the WebSocket protocol. For session shadowing, whether `ws://` or `wss://` is chosen depends on the previously mentioned "ShadowingUseSSL"registry. By default, `ws://` is chosen. However, for security reasons, we

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

recommend that you use `wss://` and install certificates on each Citrix Director client and on each Linux VDA server. Citrix disclaims any security responsibility for the Linux VDA session shadowing by using `ws://`.

To enable SSL, run the following command:

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\Software\Citrix\
   VirtualDesktopAgent" -v "ShadowingUseSSL" -d "0x00000001"
```

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, consider using a wildcard certificate for the entire domain.

A root certificate is also required for each Citrix Director client that communicates with the Linux VDA. Root certificates are available from the same CAs that issue the server certificates.

You can install server and client certificates from the following CAs:

- A CA that is bundled with your operating system
- An enterprise CA (a CA that your organization makes accessible to you)
- 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 getting certificates.

> **Important:**
>
> - The Common Name for a server certificate must be the exact FQDN of the Linux VDA 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.
> - Chrome has stopped accepting self-signed SSL certificates, considering them insecure. The `NET::ERR_CERT_COMMON_NAME_INVALID` error occurs because the generated certificate lacks the SAN (subjectAltName) field. To resolve this issue, provide a certificate with extended attributes (X509 v3 extensions) that include the SAN field.

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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 CA certificates on the **Authorities** tab of Certificate Manager.

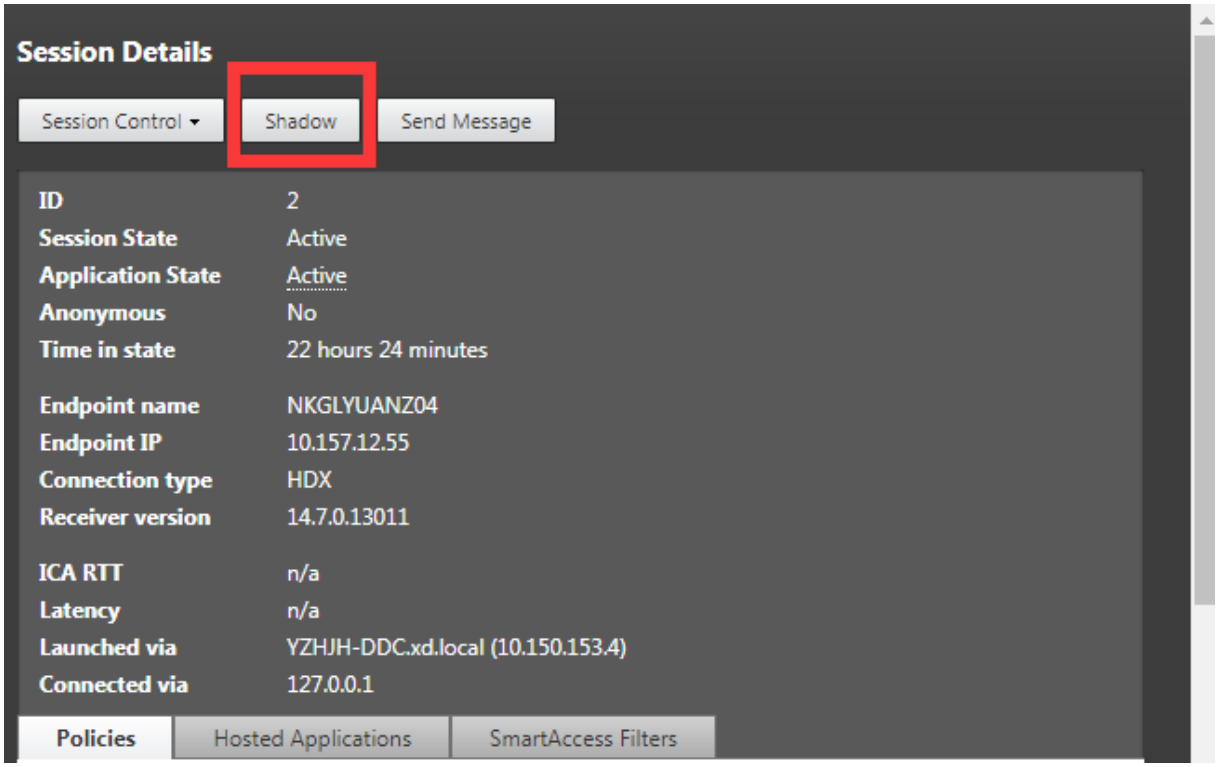
Install a server certificate and its key on each Linux VDA server Name the server certificates "shadowingcert.*"and the key file "shadowingkey.*"(* indicates the format, for example, shadowingcert.pem and shadowingkey.key). Put server certificates and key files under the path **/etc/xdl/shadowingssl** and protect them properly with restricted permissions, allowing only **ctxsrvr** to have read access. An incorrect name or path makes the Linux VDA unable to find a specific certificate or key file and therefore causes connection failure with Citrix Director. Commands are as follows:

```
cp <vda's-public-key> /etc/xdl/shadowingssl/shadowingcert.pem
cp <vda's-server-private-key> /etc/xdl/shadowingssl/shadowingkey.key
sudo chown ctxsrvr:ctxadm  /etc/xdl/shadowingssl/shadowingcert.pem
sudo chown ctxsrvr:ctxadm  /etc/xdl/shadowingssl/shadowingkey.key
```

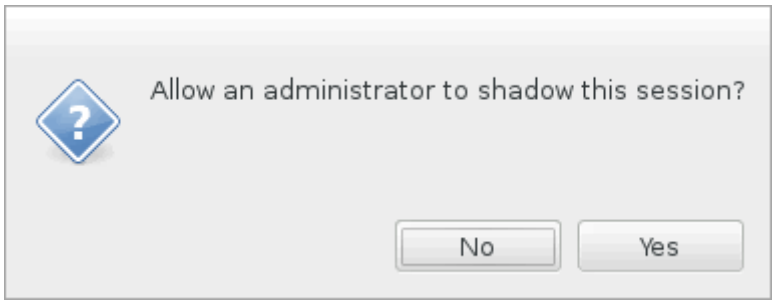
Usage

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



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



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

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

Limitations

- Session shadowing is designed for use in an Intranet only. It does not work for external networks even connecting through Citrix Gateway. Citrix disclaims any responsibility for the Linux VDA session shadowing in an external network.
- With session shadowing enabled, a domain administrator can only view the ICA sessions, but has no permission to write or control it.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- After an administrator clicks **Shadow** from Citrix Director, a confirmation appears to ask the user for permission to shadow the session. A session can be shadowed only when the session user gives the permission.
- The previously mentioned confirmation has a timeout limitation, which is 20s. A shadowing request fails when the time runs out.
- One session can be shadowed by only one administrator. For example, if administrator B sends a shadowing request for a session administrator A is shadowing, the confirmation for getting the user permission reappears on the user device. If the user agrees, the shadowing connection for administrator A stops and a new shadowing connection is built for administrator B. If an administrator sends another shadowing request for the same session, a new shadowing connection can also be built.
- To use session shadowing, install Citrix Director 7.16 or later.
- A Citrix Director client uses an FQDN rather than an IP address to connect to the target Linux VDA server. Therefore, the Citrix Director client must be able to resolve the FQDN of the Linux VDA server.

Troubleshooting

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

On the Citrix Director client

Through the developer tools of the browser, check the output logs on the **Console** tab. Or, check the response of the ShadowLinuxSession API on the **Network** tab. If the confirmation for getting user permission appears but the connection build-up fails, ping the VDA's FQDN manually to verify that Citrix Director can resolve the FQDN. If there's an issue with the wss://connection, check your certificates.

On the Linux VDA

1. Check the /var/log/xdl/vda.log file for clues.
2. Edit the /var/xdl/sessionshadowing.sh file and change the 'logFile'variable to specify a log file that can be tailed for clues during session shadowing from the director.
3. Also, you can manually verify whether your certificates work correctly with the noVNC connection:

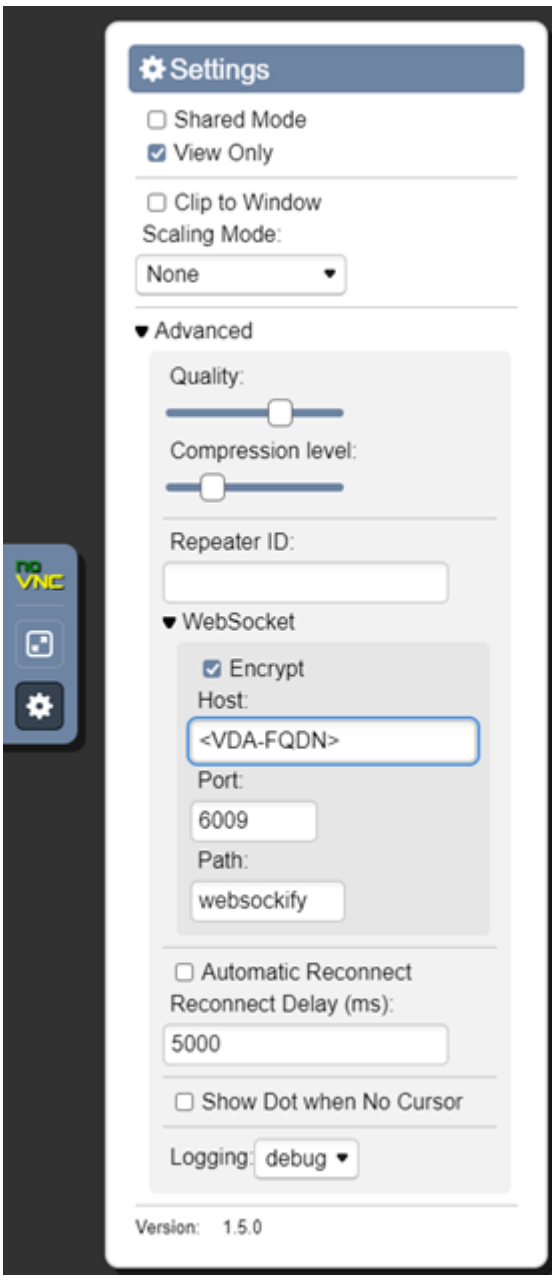
    a) Run **ps aux | grep xorg** to find the current session's Xorg display number <display-num>, for example, :3.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

b) Run the following command to start an x11vnc server and wait for an incoming connection.

```
runuser -l "ctxsrvr" -s /bin/bash -c "websockify <port> -v --cert /etc/xdl/shadowingssl/shadowingcert.pem --key /etc/xdl/shadowingssl/shadowingkey.key -- x11vnc -viewonly -shared -passwd <passwd> -rfbport <port> -display <display-num> -many -o /var/log/xdl/x11vnc.log"
```

c) Try to connect using noVNC to verify SSL mode as follows. Enter your VDA's FQDN and the port number. In this example, the port number is 6009.



d) Resolve any errors printed by Websockify on the VDA or reported by the browser on the

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

client.

Key checkpoints during connection establishment:

i. Check for any firewall limitation that stops session shadowing from opening the port.

ii. Verify that you have named certificates and key files properly and put them under the correct path if it's the SSL scenario.

iii. Verify that there are enough ports left between 6001-6099 for new shadowing requests.

iv. Verify that 'netstat' is installed because it is required by `/var/xdl/sessionshadowing.sh`.

v. Run `openssl x509 -in shadowingcert.pem -text -noout` to verify that the certificates are configured correctly, paying particular attention to the CN and SAN fields.

vi. On RHEL 8, there might be an issue where `rebind.so` cannot be found. To resolve this issue, run the following command:

```
ln -s /usr/bin/rebind.so /usr/local/bin/rebind.so
```

The monitor service daemon

July 8, 2024

The monitor service, **ctxmonitord**, is a daemon that monitors the health status of the Linux VDA. It consists of three modules:

- Key process monitor module: This module delegates Systemd to automatically restart key services upon unexpected failures, periodically scans and records the status of target processes, and ensures timely cleanup of Xorg residuals. Logs are stored in **/var/log/xdl/ms.log**.
- XDPing integration module: This module periodically executes XDPing analysis and backup tasks. All output from these tasks can be found in **/var/log/xdl/msxdping.log**.
- Linux VDA self-update module: For more information, see Linux VDA self-update through Azure.

Configuration

By default, the monitor service **ctxmonitord** starts automatically when you start the VDA. With administrator privileges, you can configure it through /etc/xdl/ctxmonitord.conf and /etc/xdl/whitelist.conf.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

ctxmonitord.conf

This configuration file specifies the behaviors of the monitor service ctxmonitord and its modules. By default, it is configured as follows:

```
1       ; This is the configuration file for ctxmonitord service
2
3
4  ; Section 'service' configures the key process monitor module
5  [service]
6  MonitorEnable=true      ; true or false
7  DetectInterval=300       ; in seconds, minimum is 60
8
9  ; Section 'xdping' configures the XDPing integration module
10 [xdping]
11 XdpingEnable=true       ; true or false
12 AnalysisInterval=600     ; in seconds, minimum is 60, 0 means disable
13 BackupInterval=1         ; in days, 0 means disable
14
15 ; Section 'rules' configures the rules about how to monitor each key
      process
16 ; Each rule should be named as rules.<ProcessName> where <ProcessName>
      is the name of the process to be monitored
17 ; 'ProcessName' should be the name of the target process that is picked
       up from whitelist.conf
18 ; 'MonitorType' should be 'HealthCheck' or 'ResidueCleanup'. All '
      MonitorType' should be 'HealthCheck' except for 'Xorg'
19 [rules]
20 [rules.ctxhdx]
21 ProcessName=ctxhdx
22 MonitorType=HealthCheck
23
24 [rules.ctxvda]
25 ProcessName=ctxvda
26 MonitorType=HealthCheck
27
28 [rules.ctxjproxy]
29 ProcessName=ctxjproxy
30 MonitorType=HealthCheck
31
32 [rules.ctxpolicyd]
33 ProcessName=ctxjproxy
34 MonitorType=HealthCheck
35
36 [rules.ctxlogd]
37 ProcessName=ctxlogd
38 MonitorType=HealthCheck
39
40 [rules.xorg]
41 ProcessName=Xorg
42 MonitorType=ResidueCleanup
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

whitelist.conf

This configuration file specifies the white list of the target processes to monitor. The target services specified in the ctxmonitord.conf file must also be listed in the whitelist.conf file. By default, it is configured as follows:

```
ctxhdx
ctxvda
ctxjproxy
ctxlogd
ctxcdm
ctxcups
ctxpolicyd
ctxusbsd
ctxceip
ctxsdcd
ctxrunatboot
ctxgdtd
ctxbcrd
ctxfidod
ctxpfwd
ctxwcamsd
ctxcertmgr
ctxscardsd  # Except for SLES
ctxfedsd    # Only for RHEL 8/9
Xorg
```

Note:

- To make your changes in the ctxmonitord.conf and whitelist.conf files take effect, run the **systemctl restart ctxmonitord** command to restart the monitor service daemon.
- To configure how services restart upon unexpected failures, edit the **/usr/lib/systemd/system/ctx*.service** file. For example, the following are the default restart options:

  ```
  Restart=on-failure
  RestartSec=5
  StartLimitInterval=60
  StartLimitBurst=3
  ```

Troubleshooting

June 20, 2024

This article describes how to use **XDPing** to troubleshoot and how to query session metrics using the **ctxsdcutil** utility.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

XDPing

The Linux **XDPing** tool is a command-line application. It automates the process of checking for common configuration issues with a Linux VDA environment.

Install the Linux XDPing tool

Running ctxsetup.sh does not install **XDPing**. To install **XDPing**, run `sudo /opt/Citrix/VDA/bin/xdping`.
This command also creates a **Python3** virtual environment that is required for **XDPing**. If this command fails to create a **Python3** virtual environment, create it manually following the instructions at Create a **Python3** virtual environment.

To address SSL connection errors that you might encounter when using the pip tool, consider adding the following trusted hosts to the /etc/pip.conf file:

```
[global]
trusted-host =
pypi.org
files.pythonhosted.org
```

Tasks that can be done with XDPing

XDPing comes with the single executable named **xdping** that is run from the command shell.

The following table describes the various tasks that can be done with the corresponding **XDPing** commands:

Task	**XDPing** Command	Remarks
To display the command-line options	**sudo /opt/Citrix/VDA/bin/xdping -h**	N/A
To run the full suite of tests	**sudo /opt/Citrix/VDA/bin/xdping** (run **XDPing** without any command-line option)	The Linux **XDPing** tool performs over 150 individual tests on the system. For more information, see Individual tests later in this article.
To run a VDA registration status check	**sudo /opt/Citrix/VDA/bin/xdping -a**	For more information, see Scope of registration status checks later in this article.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Task	**XDPing** Command	Remarks
To back up the key data of a VDA	**sudo /opt/Citrix/VDA/bin/xdping -b**	For more information, see Backup and comparison of VDA data later in this article.
To compare the latest two copies of VDA backup data	**sudo /opt/Citrix/VDA/bin/xdping -diff**	For more information, see Backup and comparison of VDA data later in this article.
To compare two specific copies of VDA backup data	**sudo /opt/Citrix/VDA/bin/xdping -diff=:**	For more information, see Backup and comparison of VDA data later in this article.
To check the environment before installing the Linux VDA package	**sudo /opt/Citrix/VDA/bin/xdping –preflight**	N/A
To run specific test categories only, for example, the time, Kerberos, and database tests	**sudo /opt/Citrix/VDA/bin/xdping -T time,kerberos,database**	N/A
To probe a particular Delivery Controller	**sudo /opt/Citrix/VDA/bin/xdping -d **	N/A
To display what dependencies are installed on the VDA	**sudo /opt/Citrix/VDA/bin/xdping –query-pkgs or sudo /opt/Citrix/VDA/bin/xdping -q**	N/A
To display what dependencies are installed on the VDA and save the query results in a specific path	**sudo /opt/Citrix/VDA/bin/xdping -f ** or **sudo /opt/Citrix/VDA/bin/xdping –query-pkgs-to **	N/A

Individual tests The Linux **XDPing** tool performs over 150 individual tests on the system, which are broadly categorized as follows:

- Check whether Linux VDA system requirements are met.
- Identify and display machine information including the Linux distributions.
- Check the Linux kernel compatibility.
- Check for any known Linux distribution issues that can impact the Linux VDA operation.
- Check the Security-Enhanced Linux (SELinux) mode and compatibility.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- Identify network interfaces and check network settings.
- Check storage partitioning and available disk space.
- Check machine host and domain name configuration.
- Check DNS configuration and perform lookup tests.
- Identify underlying hypervisors and check virtual machine configuration. Support for:
  - XenServer (formerly Citrix Hypervisor)
  - Microsoft HyperV
  - VMware vSphere
- Check time settings and check whether network time synchronization is operational.
- Check whether the PostgreSQL service is properly configured and operational.
- Check whether SQLite is properly configured and operational.
- Check whether the firewall is enabled and the required ports are open.
- Check Kerberos configuration and perform authentication tests.
- Check the LDAP search environment for the group policy service engine.
- Check whether Active Directory integration is set up properly and the current machine is joined to the domain. Support for:
  - Samba Winbind
  - Dell Quest Authentication Services
  - Centrify DirectControl
  - SSSD
- Check the integrity of the Linux computer object in the Active Directory.
- Check Pluggable Authentication Module (PAM) configuration.
- Check the core dump pattern.
- Check whether packages required by the Linux VDA are installed.
- Identify the Linux VDA package and check the integrity of the installation.
- Check the integrity of the PostgreSQL registry database.
- Check whether the Linux VDA services are properly configured and operational.
- Check the integrity of the VDA and HDX configuration.
- Probe each configured Delivery Controller to test that the Broker Service is reachable, operational, and responsive.
- Check whether the machine is registered with the Delivery Controller farm.
- Check the state of each active or disconnected HDX session.
- Scan log files for the Linux VDA related errors and warnings.
- Check whether the version of Xorg is suitable.
- Check whether required dependencies are installed.

Sample output The following is a sample output from running the Kerberos test:

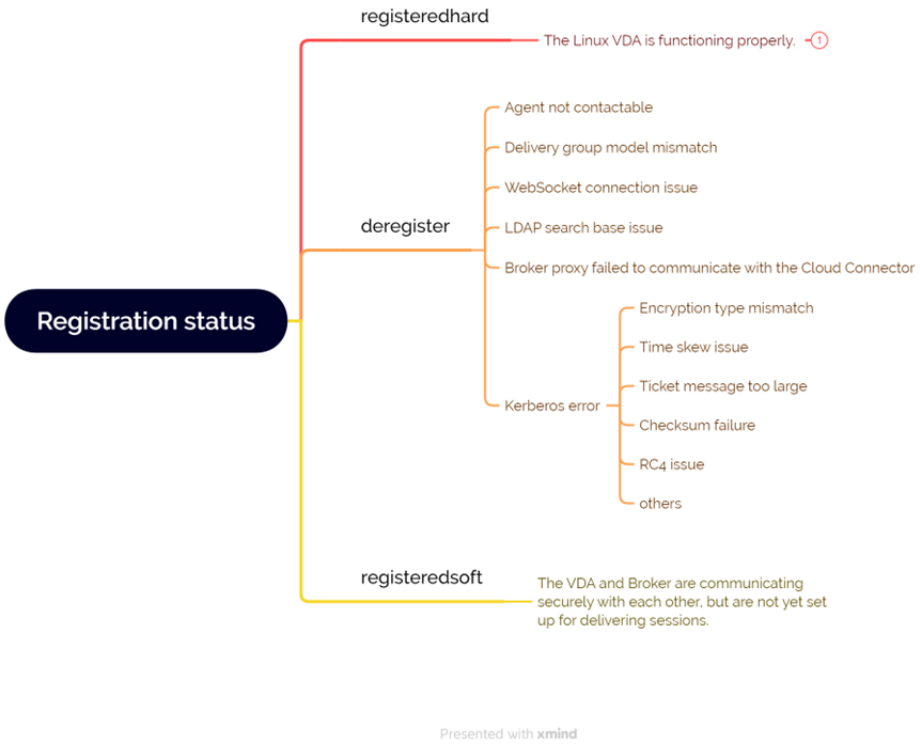
© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
sudo xdping -T kerberos

Root User ------------------------------------------------------------------
    User:         root
    EUID:         0
      Verify user is root                                              [Pass]

Kerberos -------------------------------------------------------------------
    Kerberos version: 5
      Verify Kerberos available                                        [Pass]
      Verify Kerberos version 5                                        [Pass]
    KRB5CCNAME:   [Not set]
                  Distro default FILE:/tmp/krb5cc_%{uid}
    KRB5CCNAME type: [Supported]
    KRB5CCNAME format: [Default]
      Verify KRB5CCNAME cache type                                     [Pass]
      Verify KRB5CCNAME format                                         [Pass]
    Configuration file: /etc/krb5.conf [Exists]


  Verify Kerberos configuration file found                            [Pass]
Keytab file:  /etc/krb5.keytab [Exists]
Default realm: XD2.LOCAL
Default realm KDCs: [NONE SPECIFIED]
Default realm domains: [NONE SPECIFIED]
DNS lookup realm: [Enabled]
DNS lookup KDC: [Enabled]
Weak crypto:  [Disabled]
Clock skew limit:  300 s
  Verify system keytab file exists                                    [Pass]
  Verify default realm set                                            [Pass]
  Verify default realm in upper-case                                  [Pass]
  Verify default realm not EXAMPLE.COM                                [Pass]
  Verify default realm domain mappings                                [Pass]
  Verify default realm master KDC configured                          [Pass]
  Verify Kerberos weak crypto disabled                                [Pass]
  Verify Kerberos clock skew setting                                  [Pass]
Default ccache: [Not set]
                Distro default FILE:/tmp/krb5cc_%{uid}
Default ccache type: [Supported]
Default ccache format: [Default]
  Verify default credential cache cache type                          [Pass]
  Verify default credential cache format                              [Pass]
UPN system key [MYVDA1$@[redacted]]: [MISSING]
SPN system key [host/[redacted]l]: [Exists]
  Verify Kerberos system keys for UPN exist                          [ERROR]
  No system keys were found for the user principal name (UPN) of
  the machine account. For the Linux VDA to mutually authenticate
  with the Delivery Controller, the system keytab file must
  contain keys for both the UPN and host-based SPN of the machine
  account.
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
  Verify Kerberos system keys for SPN exist                    [Pass]
Kerberos login: [FAILED AUTHENTICATION]
                Keytab contains no suitable keys for MYVDA1$@[redacted]
                while getting initial credentials
  Verify KDC authentication                                    [ERROR]
  Failed to authenticate and obtain a Ticket Granting Ticket (TGT)
  from the KDC authentication service for the machine account UPN
  MYVDA1$@[redacted]. Check that the Kerberos configuration is
  valid and the keys in the system keytab are current.


Summary ------------------------------------------------------------------------
  The following tests did not pass:
     Verify Kerberos system keys for UPN exist                 [ERROR]
     Verify KDC authentication                                 [ERROR]
```

Scope of VDA registration status checks The Linux **XDPing** tool also provides an analysis module to help you check and analyze your VDA registration status. For a scope of registration status checks, see the following screen capture:



Backup and comparison of VDA data Starting with the Linux VDA 2305, the **XDPing** tool introduces a VDA backup module. This module lets you back up the key data of a VDA at any time, such as the configuration, database, and binary permission data. You can back up the key data of a VDA when

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

it is running properly. In case the VDA fails later, back up another copy of the data and compare the two copies of data to facilitate troubleshooting. The following table describes VDA data backup and comparison with the corresponding **XDPing** commands:

Task	**XDPing** Command	Remarks
To back up the key data of a VDA	**sudo /opt/Citrix/VDA/bin/xdping -b**	Each time you run the backup command, a copy of the backup data is generated and saved in a directory under **/var/ctxbackup**. The backup data directories are named the current date and time in **yyyy-mm-dd-hh_mm_ss** format, for example, **2023-02-27-16_31_27**. By default, the maximum number of backup data directories is 30 and the **XDPing** tool rotates or deletes old backup data directories when the number is exceeded. To customize the number for directory rotation, run the following command: `sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\VirtualDesktopAgent\Backup"-t "REG_DWORD" -v " MaxDirRotationCount"-d "0x0000005"--force`
To compare the latest two copies of VDA backup data	**sudo /opt/Citrix/VDA/bin/xdping -diff**	N/A
To compare two specific copies of VDA backup data	**sudo /opt/Citrix/VDA/bin/xdping -diff=:**	N/A

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Session metric query utilities

ctxsession

This utility provides a Windows user experience. With this utility, you can access session metrics as shown in the following screen capture:

```
root@u22:~# /opt/Citrix/VDA/bin/ctxsession -v

            Session Id:    4
             Protocols:    TCP-TLS-CGP-ICA
             User Name:    st\wcx
           Client Name:    BLR3-EWS-WPD030
        Client Address:    10.109.143.143:58535
         Local Address:    10.158.180.61:443
        Remote Address:    10.109.143.143:58535
     Security Protocol:    TLSv1.2
       Security Cipher:    ECDHE-RSA-AES256-GCM-SHA384
       Cipher Strength:    256 bits
        ICA Encryption:    Basic
    Rendezvous Version:    None
       Reducer Version:    4

     ICA Statistics:
        ICA RTT: 0 ms
        Sent Bandwidth: 112 bps
        Received Bandwidth: 0 bps
```

ctxqsession, ctxquser, ctxqfull, ctxquery

For instructions on how to use these utilities, run the help command, for example:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
root@ubuntu2204-33:~# /opt/Citrix/VDA/bin/ctxqsession --help
Usage: ctxqsession [-f short_format_options | -o long_format_options] [-m] [user user_name]
Display information about ICA connections to the local server.

Options:
    -f         Change the displayed information specified by short format options
    -o         Change the displayed information specified by long format options
    -m         Alter the column headers to remove spaces
    user       Display information about the specified user

Short format options:
    d    Client device name (client name)
    i    Session ID
    n    Session name
    p    Published application name
    P    Protocols
    S    Session state
    u    User name
    x    X display number

Long format options(comma separated):
    app     Published application name
    dev     Client device name (client name)
    id      Session ID
    proto   Protocols
    sess    Session name
    state   Session state
    user    User name
    xdpy    X display number
```

Users can query only their own session metrics. Only the root and **ctxadm** group users have permission to query other users'session metrics.

ctxsdcutil

This utility helps to query the following metrics of all sessions or a specific session hosted on a VDA. To do so, run the `/opt/Citrix/VDA/bin/ctxsdcutil -q <all | SessionID> [-c]` command. The `[-c]` argument means to query metrics every second.

- **Input Session Bandwidth**
- **Output Session Bandwidth**
- **Output Session Line Speed**
- **Latency - Last Recorded**
- **Round Trip Time**
- **Output ThinWire Bandwidth**
- **Output Audio Bandwidth**
- **Output Printer Bandwidth**
- **Input Drive Bandwidth**
- **Output Drive Bandwidth**

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Others

June 20, 2024

This section contains the following topics:

- Citrix Workspace app for HTML5 support
- Create a Python3 virtual environment
- Integrate NIS with Active Directory
- IPv6
- LDAPS
- **Xauthority**

Citrix Workspace app for HTML5 support

June 20, 2024

You can use Citrix Workspace app for HTML5 to access Linux virtual apps and desktops directly without connecting your client to Citrix Gateway. For information about Citrix Workspace app for HTML5, see the Citrix documentation.

Enable this feature

This feature is disabled by default. To enable it, do the following:

1. In Citrix StoreFront, enable Citrix Workspace app for HTML5.

   For the detailed procedure, see Step 1 of the Knowledge Center article CTX208163.

2. Enable WebSocket connections.

   a) In Citrix Studio, set the **WebSockets connections** policy to **Allowed**.

   You can also set the other WebSocket policies. For a full list of the WebSocket policies, see WebSockets policy settings.

   b) On the VDA, restart the `ctxvda` service and the `ctxhdx` service, in this order, for your setting to take effect.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

c) On the VDA, run the following command to check whether the WebSocket listener is running.

```
netstat -an | grep 8008
```

When the WebSocket listener is running, the command output is similar to the following:

```
tcp 0 0 :::8008 :::* LISTEN
```

Note: You can also enable TLS encryption to secure WebSocket connections. For information about enabling TLS encryption, see Secure user sessions using TLS.

Create a Python3 virtual environment

June 20, 2024

If you are connecting to the network, running the `sudo /opt/Citrix/VDA/bin/xdping` or `/opt/Citrix/VDA/sbin/enable_ldaps.sh` command can create a Python3 virtual environment. However, if the commands fail to create a Python3 virtual environment, you can create it manually even without a network connection. This article details the prerequisites and steps to create a Python3 virtual environment without a network connection.

Prerequisites

- You must have administrative privileges to access the `/opt/Citrix/VDA/sbin/ctxpython3` directory.
- The wheel files of `Python3` packages are in place. You can download the wheel files from https://pypi.org/.

Create a Python3 virtual environment

Complete the following steps to create a Python3 virtual environment:

1. Install Python3 dependencies.

   For Amazon Linux 2:

   ```
   1 yum -y install python3 python3-devel krb5-devel gcc
   ```

   For RHEL and Rocky Linux:

   ```
   1 yum -y install python3-devel krb5-devel gcc
   ```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> You might have to enable a particular repository to install some dependencies. For RHEL 7, run the `subscription-manager repos --enable rhel-7-server-optional-rpms` command. For RHEL 8, run the `subscription-manager repos --enable=rhel-8-for-x86_64-appstream-rpms` command.

For Debian, Ubuntu:

```
apt-get -y install python3-dev python3-pip python3-venv libkrb5-dev
```

For SUSE:

```
zypper -n install lsb-release python3-devel python3-setuptools krb5-devel gcc libffi-devel libopenssl-devel
```

2. Create a Python3 virtual environment.

> **Note:**
>
> To address SSL connection errors that you might encounter when using the pip tool, consider adding the following trusted hosts to the /etc/pip.conf file:
>
> ```
> [global]
> trusted-host =
> pypi.org
> files.pythonhosted.org
> ```

For Amazon Linux 2, Debian, RHEL, Rocky Linux, Ubuntu:

```
sudo python3 -m venv /opt/Citrix/VDA/sbin/ctxpython3
```

For SUSE:

```
sudo ln -s /usr/lib/mit/bin/krb5-config /usr/bin/krb5-config

export PATH=$PATH:/usr/lib/mit/bin:/usr/lib/mit/sbin

sudo mkdir -p /usr/lib/mit/include/gssapi/

sudo ln -s /usr/include/gssapi/gssapi_ext.h/usr/lib/mit/include/gssapi/gssapi_ext.h

sudo python3 -m venv /opt/Citrix/VDA/sbin/ctxpython3
```

3. Install LDAPS dependencies.

```
sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m pip install --upgrade pip==21.3.1
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
2
3 sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m pip install
    cffi==1.15.0 cryptography==36.0.2 decorator==5.1.1 gssapi
    ==1.7.3 ldap3==2.9.1 pyasn1==0.4.8 pycparser==2.21 six==1.16.0
```

4. Install **XDPing** dependencies.

```
1 sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m pip install --
    upgrade pip==21.3.1
2
3 sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m pip install
    asn1crypto==1.5.1 cffi==1.15.0 cryptography==36.0.2 decorator
    ==5.1.1 gssapi==1.7.3 ldap3==2.9.1 netifaces==0.11.0 packaging
    ==21.3 pg8000==1.26.0 psutil==5.9.0 pyasn1==0.4.8 pycparser
    ==2.21 pyparsing==3.0.8 scramp==1.4.1 six==1.16.0 termcolor
    ==1.1.0
4
5 sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m pip install /
    opt/Citrix/VDA/sbin/ctxpython3/packages/xdping-*.whl
```

Integrate NIS with Active Directory

June 20, 2024

This article describes how to integrate NIS with Windows Active Directory (AD) on the Linux VDA by using SSSD. The Linux VDA is considered a component of Citrix Virtual Apps and Desktops. As a result, it fits tightly into the Windows AD environment.

Using NIS instead of AD as a UID and GID provider requires the account information (user name and password combinations) to be the same in AD and NIS.

> **Note:**
>
> Authentication is still performed by the AD server. NIS+ is not supported. If you use NIS as the UID and GID provider, the POSIX attributes from the Windows server are no longer used.
>
> **Tip:**
>
> This method represents a deprecated way to deploy the Linux VDA, which is used only for special use cases. For an RHEL/CentOS distribution, follow the instructions in Install the Linux VDA on Amazon Linux 2, CentOS, RHEL, and Rocky Linux manually. For an Ubuntu distribution, follow the instructions in Install the Linux VDA on Ubuntu manually.

What is SSSD?

SSSD is a system daemon. Its primary function is to provide access to identify and authenticate remote resources through a common framework that can provide caching and offline support for the

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

system. It provides both PAM and NSS modules, and in the future can support D-BUS based interfaces for extended user information. It also provides a better database to store local user accounts and extended user data.

Integrate NIS with AD

To integrate NIS with AD, complete the following steps:

Step 1: 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
```

The **nis.domain** represents the domain name of the NIS server. The **server.nis.domain** is the host name of the NIS server, which can also be the IP address of the NIS server.

Configure the NIS services:

```
sudo systemctl start rpcbind ypbind

sudo systemctl enable rpcbind ypbind
```

Ensure that the NIS configuration is correct:

```
ypwhich
```

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

```
getent passwd nisaccount
```

> **Note:**
>
> The **nisaccount** represents the real NIS account on the NIS server. Ensure that the UID, GID, home

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> directory, and login shell are configured correctly.

Step 2: Join the domain and create a host keytab using Samba

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

- `adcli`
- `realmd`
- `Winbind`
- `Samba`

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

Join the domain and create host keytab using Samba:

On the Linux client with properly configured files:

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

Configure the machine for Samba and Kerberos authentication:

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

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

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

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

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

```
kerberos method = secrets and keytab
winbind offline logon = no
```

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

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 3: Set up SSSD

> **Note**
>
> Using SSSD with the Name Service Cache Daemon (NSCD) can result in unexpected behavior. For more information, see Using NSCD with SSSD.

Setting up SSSD consists of the following steps:

- Install the **sssd-ad** and **sssd-proxy** packages on the Linux client machine.
- Make configuration changes to various files (for example, **sssd.conf**).
- Start the **sssd service**.

/etc/sssd/sssd.conf An example **sssd.conf** configuration (more options can be added as needed):

```
[sssd]
config_file_version = 2
domains = EXAMPLE
services = nss, pam

[domain/EXAMPLE]
# Uncomment if you need offline logins
# cache_credentials = true
re_expression = (((?P<domain>[^\\]+)\\(?P<name>.+$))|((?P<name>[^@]+)@(?P<domain>.+$))|(^(?P<name>[^@\\]+)$))
id_provider = proxy
proxy_lib_name = nis
auth_provider = ad
access_provider = ad

# Should be specified as the long version of the Active Directory domain.
ad_domain = 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
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

Set the file ownership and permissions on **sssd.conf**:

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

Step 4: Configure NSS/PAM

RHEL/CentOS:

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

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

sudo systemctl start sssd

sudo systemctl enable sssd
```

> **Tip:**
>
> When configuring Linux VDA settings, consider that for SSSD, there has no special settings for the Linux VDA client. For extra solutions in the **ctxsetup.sh** script, use the default value.

Step 5: Verify the Kerberos configuration

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

```
sudo klist -ke
```

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

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

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
sudo klist -ke
```

Step 6: Verify user authentication

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

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

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

The supported logon formats are:

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

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

```
sudo 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
```

IPv6

June 20, 2024

The Linux VDA supports IPv6 to align with Citrix Virtual Apps and Desktops. When using this feature, consider the following:

- For dual stack environments, IPv4 is used unless IPv6 is explicitly enabled.
- If IPv6 is enabled in an IPv4 environment, the Linux VDA fails to function.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Important:**
>
> - The whole network environment must be IPv6, not only for the Linux VDA.
> - Centrify does not support pure IPv6.

No special setup tasks are required for IPv6 when you install the Linux VDA.

Configure IPv6 for the Linux VDA

Before changing the configuration for the Linux VDA, ensure that your Linux virtual machine has previously worked in an IPv6 network. There are two registry keys related to IPv6 configuration:

```
"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" -d "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 Citrix Virtual Apps and Desktops, see IPv4/IPv6 support.

Troubleshooting

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

LDAPS

June 20, 2024

LDAPS is the secure version of the Lightweight Directory Access Protocol (LDAP) where LDAP communications are encrypted using TLS/SSL.

By default, LDAP communications between client and server applications are not encrypted. LDAPS enables you to protect the LDAP query content between the Linux VDA and the LDAP servers.

The following Linux VDA components have dependencies on LDAPS:

- Broker agent: Linux VDA registration with a 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 the Linux VDA
- Configure LDAPS for third-party platforms
- Configure SSSD
- Configure Winbind
- Configure Centrify
- Configure Quest

> **Note:**
>
> You can run the following command to set a monitoring cycle for your LDAP servers. The default value is 15 minutes. Set it to 10 minutes at least.

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\
   VirtualDesktopAgent" -v "ListOfLDAPServersMonitorPeroid" -t "
   REG_DWORD" -d "0x0000000f" --force
```

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:**
>
> LDAPS is enabled automatically when you install an Enterprise Root CA on a domain controller.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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.

When you have a multi-tier certificate authority hierarchy, you do not automatically have the appropriate certificate for LDAPS authentication on the domain controller.

For information about how to enable LDAPS for domain controllers using a multi-tier certificate authority hierarchy, see the LDAP over SSL (LDAPS) Certificate article.

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 a trusted 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 on the Linux VDA, run the **enable_ldaps.sh** script (while logged on as an administrator). To run the **enable_ldaps.sh** script in a non-interactive mode, export the following variables to your environment:

```
#CTX_LDAPS_KEYSTORE_PASSWORD=

#CTX_LDAPS_LDAP_SERVERS=
```

- To enable LDAP over SSL/TLS with the root CA certificate provided:

  ```
  /opt/Citrix/VDA/sbin/enable_ldaps.sh -Enable pathToRootCA
  ```

- To enable LDAP over SSL/TLS with channel binding:

  ```
  /opt/Citrix/VDA/sbin/enable_ldaps.sh -Enablecb pathToRootCA
  ```

  > **Note:**
  >
  > The root CA certificate for channel binding must be in PEM format. If enabling LDAPS does not create a Python3 virtual environment successfully, create it manually following the instructions at Create a Python3 virtual environment.
  >
  > To address SSL connection errors that you might encounter when using the pip tool, consider adding the following trusted hosts to the /etc/pip.conf file:
  > `[global]`
  > `trusted-host =`

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
pypi.org
files.pythonhosted.org
```

- To fall back to LDAP without SSL/TLS

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

The Java keystore dedicated for LDAPS resides 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

HKLM\Software\Citrix\VirtualDesktopAgent\EnableChannelBinding
```

Configure LDAPS for third-party platform

Besides the Linux VDA components, several third-party software components that adhere to the VDA might also require secure LDAP, such as SSSD, Winbind, Centrify, and Quest. The following sections describe how to configure secure LDAP with LDAPS, STARTTLS, or SASL sign and seal.

> **Tip:**
>
> Not all of these software components prefer to use SSL port 636 to ensure secure LDAP. And most of the time, LDAPS (LDAP over SSL on port 636) cannot coexist with STARTTLS on port 389.

SSSD

Configure the SSSD secure LDAP traffic on port 636 or port 389 as per the options. For more information, see the SSSD LDAP Linux man page.

Winbind

The Winbind LDAP query uses the ADS method. Winbind supports only the StartTLS method on port 389. Affected configuration files are **/etc/samba/smb.conf** and **/etc/openldap/ldap.conf** (for Amazon Linux 2, RHEL, Rocky Linux, CentOS, and SUSE) or **/etc/ldap/ldap.conf** (for Debian and Ubuntu). Change the files as follows:

- smb.conf

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

- ldap.conf

```
TLS_REQCERT never
```

Alternately, you can configure secure LDAP by SASL GSSAPI sign and seal, but it cannot coexist with TLS/SSL. To use SASL encryption, change the **smb.conf** configuration:

```
ldap ssl = off
ldap ssl ads = no
client ldap sasl wrapping = seal
```

Centrify

Centrify does not support LDAPS on port 636. However, it does provide secure encryption on port 389. For more information, see the Centrify site.

Quest

Quest Authentication Service does not support LDAPS on port 636, but it provides secure encryption on port 389 using a different method.

Troubleshooting

The following issues might arise when you use this feature:

- **LDAPS service availability**

  Verify that the LDAPS connection is available on the AD/LDAP server. The port is on 636 by default.

- **Linux VDA registration failed when LDAPS is enabled**

  Verify that the LDAP server and ports are configured correctly. Check the Root CA Certificate first and ensure that it matches the AD/LDAP server.

- **Incorrect registry change by accident**

  If you updated the LDAPS related keys by accident without using **enable_ldaps.sh**, it might break the dependency of LDAPS components.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- **LDAP traffic is not encrypted through SSL/TLS from Wireshark or any other network monitoring tools**

  By default, LDAPS is disabled. Run **/opt/Citrix/VDA/sbin/enable_ldaps.sh** to force it.

- **There is no LDAPS traffic from Wireshark or any other networking monitoring tool**

  LDAP/LDAPS traffic occurs when Linux VDA registration and Group Policy evaluation occur.

- **Failed to verify LDAPS availability by running ldp connect on the AD server**

  Use the AD FQDN instead of the IP Address.

- **Failed to import Root CA certificate by running the /opt/Citrix/VDA/sbin/enable_ldaps.sh script**

  Provide the full path of the CA certificate, and verify that the Root CA Certificate is the correct type. It is supposed to be compatible with most of the Java Keytool types supported. If it is not listed in the support list, you can convert the type first. We recommend the base64 encoded PEM format if you encounter a certificate format problem.

- **Failed to show the Root CA certificate with Keytool -list**

  When you enable LDAPS by running `/opt/Citrix/VDA/sbin/enable_ldaps.sh`, the certificate is imported to /etc/xdl/.keystore, and the password is set to protect the keystore. If you forget the password, you can rerun the script to create a keystore.

Xauthority

June 20, 2024

The Linux VDA supports environments that use X11 display functionality (including `xterm` and `gvim`) for interactive remoting. This feature provides a security mechanism necessary to ensure secure communication between XClient and XServer.

There are two methods to secure permission for this secure communication:

- **Xhost**. By default, Xhost allows only the localhost XClient to communicate with XServer. If you choose to allow a remote XClient to access XServer, the Xhost command must be run to grant permission on the specific machine. Or, you can alternately use **xhost +** to allow any XClient to connect to XServer.
- **Xauthority**. The `.Xauthority` file can be found in each user's home directory. It is used to store credentials in cookies used by xauth for authentication of XServer. When an XServer instance (Xorg) is started, the cookie is used to authenticate connections to that specific display.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

How it works

When Xorg starts up, a `.Xauthority` file is passed to the Xorg. This `.Xauthority` file contains the following elements:

- Display number
- Remote request protocol
- Cookie number

You can browse this file using the `xauth` command. For example:

```
# xauth -f ~/.Xauthority

# > list

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

If **XClient** connects to the Xorg remotely, two prerequisites must be met:

- Set the **DISPLAY** environment variable to the remote XServer.
- Get the `.Xauthority` file which contains one of the cookie numbers in Xorg.

Configure Xauthority

To enable **Xauthority** on the Linux VDA for remote X11 display, you must create the following two registry keys:

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

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

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

- Pass the `.Xauthority` file to the XClient manually

  After launching an ICA session, the Linux VDA generates the `.Xauthority` file for the XClient and stores the file in the logon user's home directory. You can copy this `.Xauthority` file to the remote XClient machine, and set the **DISPLAY** and **XAUTHORITY** environment variables. `DISPLAY` is the display number stored in the `.Xauthority` file and `XAUTHORITY` is the file path of **Xauthority**. For an example, see the following command:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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


export XAUTHORITY={
the file path of .Xauthority }
```

> **Note:**
>
> If the **XAUTHORITY** environment variable is not set, the ~/.Xauthority file is used by default.

- Pass the .Xauthority file to the 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 the XClient, the user does not need to transmit this .Xauthority file to the XClient manually. After the **DISPLAY** and **XAUTHORITY** environment variables are set correctly, the GUI is displayed in the XServer desktop automatically.

Troubleshooting

If **Xauthority** does not work, follow the troubleshooting steps:

1. As an administrator with root privilege, retrieve all Xorg cookies:

   ```
   ps aux | grep -i xorg
   ```

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

   ```
   /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 by using the IP address of the Linux VDA and the published desktop display number.

   For example, run the following command on the XClient machine:

   ```
   telnet 10.158.11.11 6160
   ```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

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

Authentication

June 20, 2024

This section contains the following topics:

- Authentication with Azure Active Directory
- Double-hop single sign-on authentication
- Federated Authentication Service
- Non-SSO authentication
- Smart cards
- Access by unauthenticated (anonymous) users

Authentication with Azure Active Directory

June 20, 2024

> **Note:**
>
> This feature is available only for Azure-hosted VDAs.

Based on your needs, you can deploy two types of Linux VDAs in Azure:

- Azure AD DS-joined VMs. The VMs are joined to an Azure Active Directory (AAD) Domain Services (DS) managed domain. Users use their domain credentials to log on to the VMs.
- Non-domain-joined VMs. The VMs integrate with the AAD identity service to provide user authentication. Users use their AAD credentials to log on to the VMs.

For more information about AAD DS and AAD, see this Microsoft article.

This article shows you how to enable and configure the AAD identity service on non-domain-joined VDAs.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Supported distributions

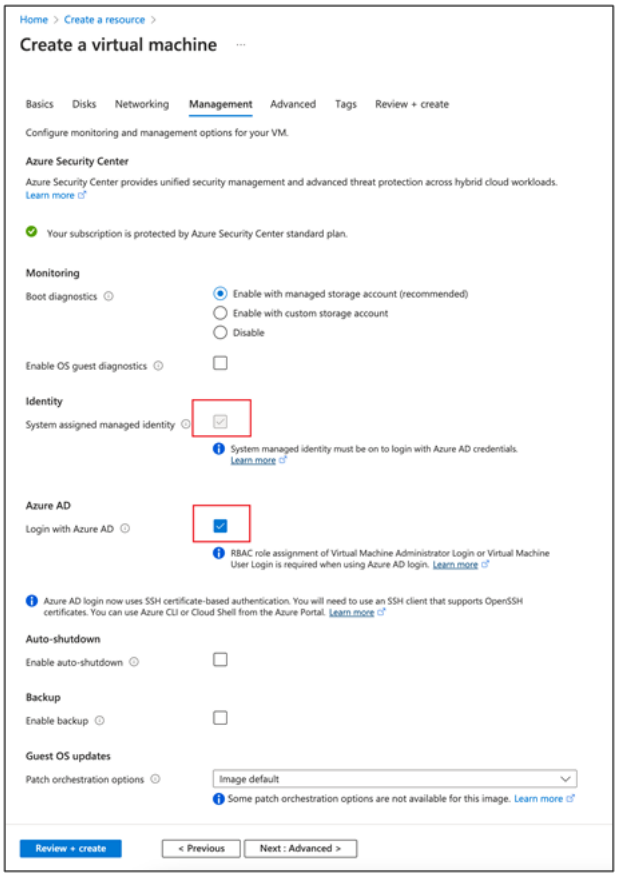
- Ubuntu 22.04, 20.04
- RHEL 8.10, 8.9, 8.8
- SUSE 15.5

For more information, see this Microsoft article.

Step 1: Create a template VM on the Azure portal

Create a template VM and install the Azure CLI on the VM.

1. On the Azure portal, create a template VM. Be sure to select **Login with Azure AD** on the **Management** tab before clicking **Review + create**.



2. Install the Azure CLI on the template VM.
   For more information, see this Microsoft article.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 2: Prepare a master image on the template VM

To prepare a master image, follow **Step 3: Prepare a master image** in Use MCS to create Linux VMs on Azure.

Step 3: Set the template VM to non-domain-joined mode

After you create a master image, follow these steps to set the VM to non-domain-joined mode:

1. Run the following script from the command prompt.

```
Modify /var/xdl/mcs/mcs_util.sh
```

2. Locate `function read_non_domain_joined_info()`, and then change the value of `NonDomainJoined` to 2. See the following code block for an example.

```
function read_non_domain_joined_info()
{

log "Debug: Enter read_non_domain_joined_info"
# check if websocket enabled
TrustIdentity=`cat ${
 id_disk_mnt_point }
 ${
 ad_info_file_path }
  | grep '\[TrustIdentity\]' | sed 's/\s//g'`
if [ "$TrustIdentity" == "[TrustIdentity]" ]; then
NonDomainJoined=2
fi
...
 }
```

3. Save the change.
4. Shut down the template VM.

Step 4: Create the Linux VMs from the template VM

After you have the non-domain-joined template VM ready, follow these steps to create VMs:

1. Sign in to Citrix Cloud.
2. Double-click Citrix DaaS, and then access the Full Configuration management console.
3. In **Machine Catalogs**, choose to use Machine Creation Services to create the Linux VMs from the template VM. For more information, see Non-domain-joined VDAs in the Citrix DaaS document.

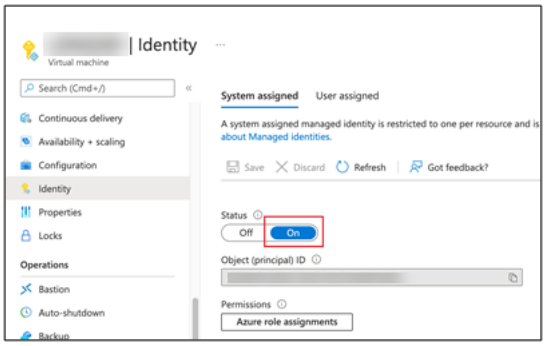
© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 5: Assign AAD user accounts to the Linux VMs

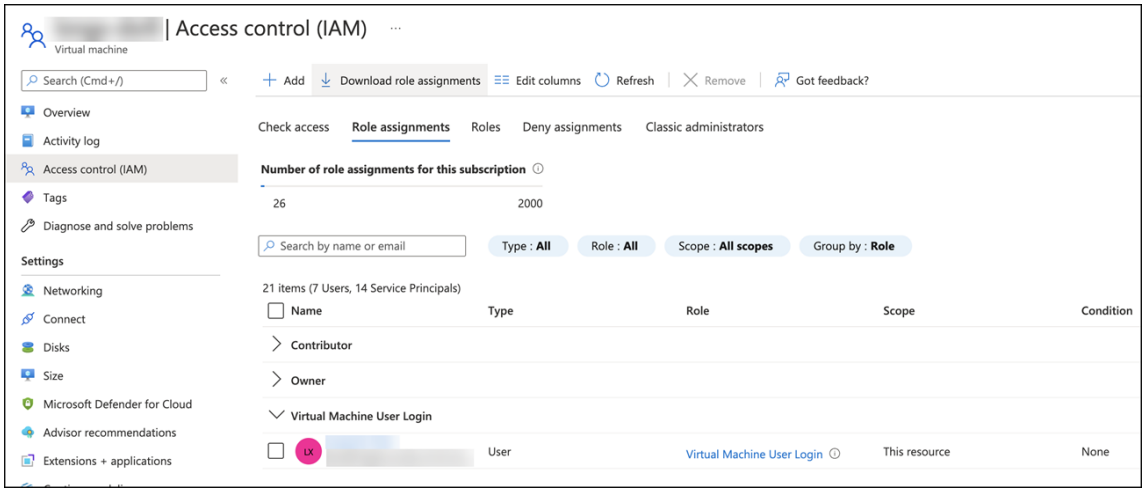
After you create the non-domain-joined VMs, assign AAD user accounts to them.

To assign AAD user accounts to a VM, follow these steps:

1. Access the VM using an administrator account.
2. On the **Identify > System assigned** tab, enable **System Identity**.



3. On the **Access control (IAM) > Role assignments** tab, locate the **Virtual Machine User Login** area, and then add the AAD user accounts as needed.



Log on to non-domain-joined VDAs

End users in your organization can log on to a non-domain-joined VDA in two ways. Detailed steps are as follows:

1. Start the Workspace app, and then log on to the workspace by entering the AAD user name and password. The Workspace page appears.
2. Double-click a non-domain-joined desktop. The AAD LOGIN page appears.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

The page varies depending on the login mode set on the VDA: Device Code or AAD account/password. By default, Linux VDAs authenticate AAD users using Device Code login mode as follows. As the administrator, you can change the login mode to AAD account/password if needed. See the following section for detailed steps.

3. Based on the onscreen instructions, log on to the desktop session in one of the following ways:

   - Scan the QR code and enter the code.
   - Enter the AAD user name and password.

Change to AAD account/password login mode

By default, Linux VDAs authenticate AAD users with device codes. See this Microsoft article for details. To change the login mode to *AAD account/password*, follow these steps:

Run the following command on the VDA, locate the key `AADAcctPwdAuthEnable`, and change its value to `0x00000001`.

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet\
   Services\CitrixBrokerAgent\WebSocket" -t "REG_DWORD" -v "
   AADAcctPwdAuthEnable" -d "0x00000001" --force
```

> **Note:**
>
> This approach doesn't work with Microsoft accounts or accounts that have two-factor authentication enabled.

Double-hop single sign-on authentication

June 20, 2024

User credentials for accessing a StoreFront store can be injected to the AuthManager module of Citrix Workspace app for Linux and Citrix Receiver for Linux 13.10. After injection, you can use the client to access virtual desktops and applications from within a Linux virtual desktop session, without entering user credentials for a second time.

> **Note:**
>
> This feature is supported on Citrix Workspace app for Linux and Citrix Receiver for Linux 13.10.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

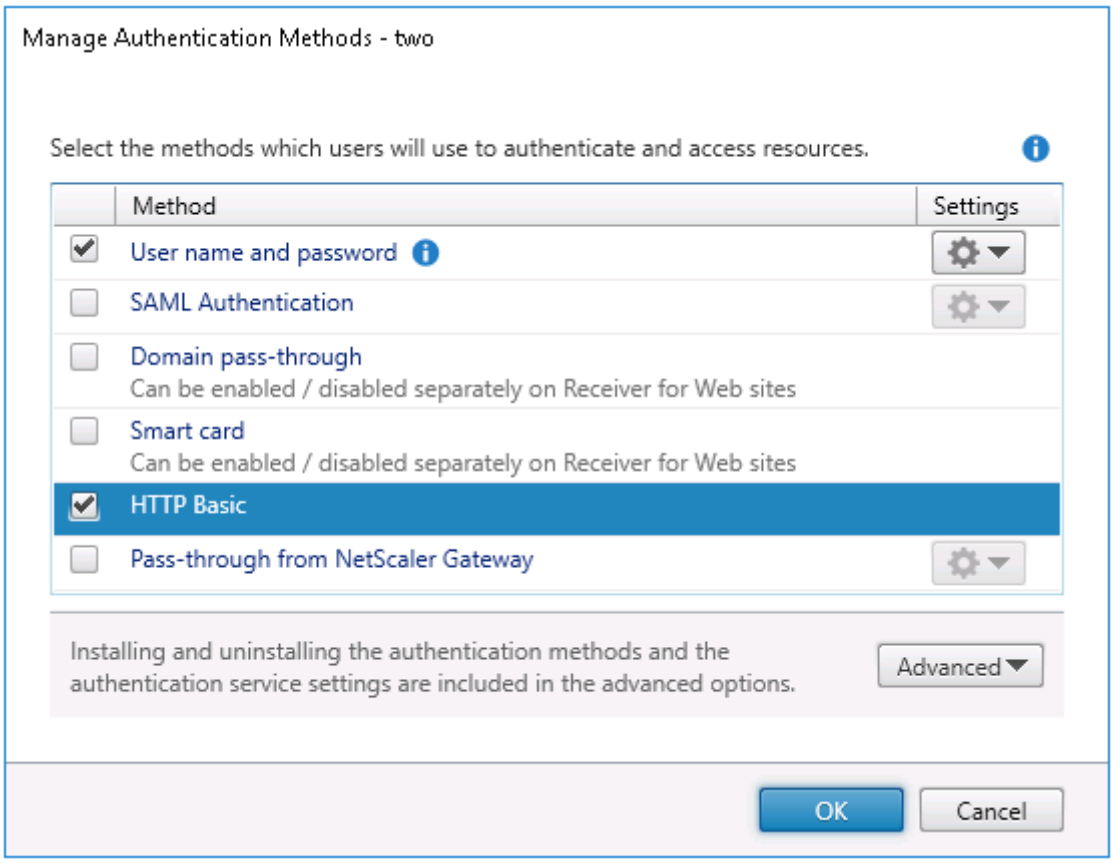
To enable the feature:

1. On the Linux VDA, install Citrix Workspace app for Linux or Citrix Receiver for Linux 13.10.

   Download the app from the Citrix download page for Citrix Workspace app or for Citrix Receiver.

   The default installation path is /opt/Citrix/ICAClient/. If you install the app to a different path, set the ICAROOT environment variable to point to the actual installation path.

2. In the Citrix StoreFront management console, add the **HTTP Basic** authentication method for the target store.



3. Add the following key to the AuthManager configuration file ($ICAROOT/config/AuthManConfig.xml) for allowing the HTTP Basic authentication:

```
<Protocols>
     <HTTPBasic>
         <Enabled>True</Enabled>
    </HTTPBasic>
</Protocols>
```

4. Run the following commands to install the root certificate in the specified directory.

```
cp rootcert.pem $ICAROOT/keystore/cacerts/
$ICAROOT/util/ctx_rehash  $ICAROOT/keystore/cacerts/
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

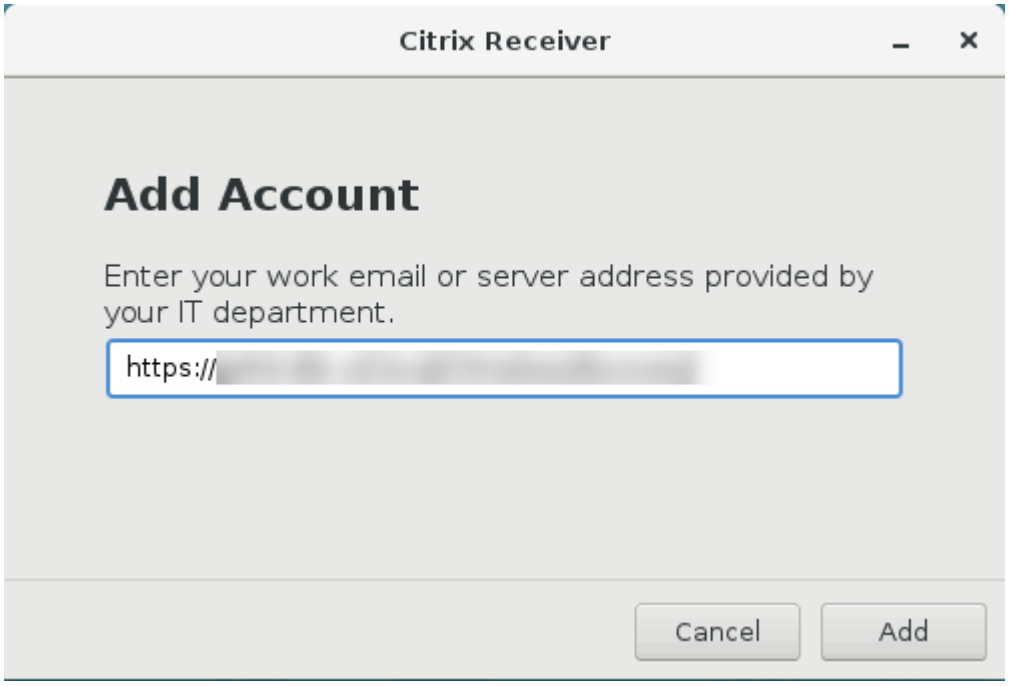
5. Run the following command to enable the feature:

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\
   CurrentControlSet\Control\Citrix" -v "LurSsonEnabled" -d "0
   x00000001"
```

6. Launch a Linux virtual desktop session and start Citrix Workspace app for Linux or Citrix Receiver for Linux 13.10 within that session.

   You are prompted for a store account when you start the Citrix Workspace app for the first time. Later on, you are logged on to the store you specified earlier automatically.

   > **Note:**
   >
   > Enter an HTTPS URL as your store account.



Federated Authentication Service

July 15, 2024

You can use the Federated Authentication Service (FAS) to authenticate users logging on to a Linux VDA. The Linux VDA uses the same Windows environment as the Windows VDA for the FAS logon feature. For information about configuring the Windows environment for FAS, see Federated Authentication Service. This article provides extra information specific to the Linux VDA.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> - The Linux VDA does not support the **In-session Behavior** policy.
> - The Linux VDA uses short connections to transmit data with FAS servers.

Supported distributions

FAS supports limited Linux distributions and domain joining methods. See the following matrix:

	Winbind	SSSD	Centrify	PBIS	Quest
Amazon Linux 2	Yes	Yes	Yes	Yes	No
Debian 12.5/11.9	Yes	Yes	Yes	Yes	No
RHEL 9.4/9.3/9.2	Yes	Yes	No	No	Yes
RHEL 8.10/8.9/8.8	Yes	Yes	Yes	Yes	Yes
Rocky Linux 9.4/9.3/9.2	Yes	Yes	No	No	No
Rocky Linux 8.10/8.9/8.8	Yes	Yes	No	No	No
SUSE 15.5	Yes	Yes	Yes	No	No
Ubuntu 22.04/20.04	Yes	Yes	Yes	Yes	Yes

Configure FAS on the Linux VDA

Install certificates

For the verification of users' certificates, install the root CA certificate and all intermediate certificates on the VDA. For example, to install the root CA certificate, get the AD root certificate from the preceding **Retrieve the CA Certificate from the Microsoft CA (on AD)** step, or download it from the root CA server `http://CA-SERVER/certsrv`.

> **Note:**
>
> The following commands also apply to configuring an intermediate certificate.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

For example, to convert a DER file (.crt, .cer, .der) to PEM, run a command similar to the following:

```
sudo openssl x509 -inform der -in root.cer -out root.pem
```

Then, install the root CA certificate to the `openssl` directory by running the command similar to the following:

```
sudo cp root.pem /etc/pki/CA/certs/
```

> **Note:**
>
> Do not put the root CA certificate under the **/root** path. Otherwise, FAS does not have the **read** permission to the root CA certificate.

Run ctxfascfg.sh

Run the ctxfascfg.sh script to configure FAS:

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

You can run `ctxfascfg.sh` in silent mode. Before you run the script in silent mode, set the following environment variables:

- **CTX_FAS_ADINTEGRATIONWAY=winbind | sssd | centrify | pbis | quest**: Denotes the Active Directory integration method, which equals to `CTX_EASYINSTALL_ADINTEGRATIONWAY` when `CTX_EASYINSTALL_ADINTEGRATIONWAY` is specified. If `CTX_EASYINSTALL_ADINTEGRATIO` is not specified, `CTX_FAS_ADINTEGRATIONWAY` uses its own value setting.
- **CTX_FAS_CERT_PATH =<certificate path>**: Specifies the full path that stores the root certificate and all intermediate certificates.
- **CTX_FAS_KDC_HOSTNAME**: Specifies the host name of the Key Distribution Center (KDC) when you select PBIS and Quest.
- **CTX_FAS_PKINIT_KDC_HOSTNAME**: Specifies the PKINIT KDC host name, which equals to CTX_FAS_KDC_HOSTNAME unless otherwise specified. If you have multiple Delivery Controllers, add the host names of all KDCs of the domain to pkinit_kdc_hostname in the **/etc/krb5.conf** file. For more information, see Knowledge Center article CTX322129.
- **CTX_FAS_SERVER_LIST='list-fas-servers'** –The Federated Authentication Service (FAS) servers are configured through the AD Group Policy. For information about FAS policy setting on the domain GPO, see Configure Group Policy. The Linux VDA does not support the AD Group Policy, but you can provide a semicolon-separated list of FAS servers instead. The sequence must be the same as configured in the AD Group Policy. If any server address is removed, fill its blank with the **'<none>'** text string and do not modify the order of server addresses. To communicate with FAS servers properly, make sure you append

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

a port number consistent with the port number specified on the FAS servers, for example, CTX_XDL_FAS_LIST='fas_server_1_url:port_number; fas_server_2_url: port_number; fas_server_3_url: port_number'.

To upgrade an existing Linux VDA installation, you can run the following commands to set the FAS servers and to restart the `ctxvda` service to make your setting take effect.

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\
    VirtualDesktopAgent\Authentication\UserCredentialService" -t "
    REG_SZ" -v "Addresses" -d "<Your-FAS-Server-List>" --force

systemctl restart ctxjproxy

systemctl restart ctxvda
```

To update the FAS servers through `ctxreg`, run the following commands:

```
sudo /opt/Citrix/VDA/bin/ctxreg update -k "HKLM\Software\Citrix\
    VirtualDesktopAgent\Authentication\UserCredentialService" -v "
    Addresses" -d "<Your-FAS-Server-List>"

systemctl restart ctxjproxy

systemctl restart ctxvda
```

Choose the correct Active Directory integration method and then type the correct path of certificates (for example, `/etc/pki/CA/certs/`).

The script then installs the krb5-pkinit and pam_krb5 packages and sets the relevant configuration files. For RHEL 8 and later, as PAM_KRB5 is moved to the EPEL repository, the script tries to enable EPEL in those distributions.

Disable FAS

To disable FAS on the Linux VDA, remove all FAS servers from ConfDB using the following commands:

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\
    VirtualDesktopAgent\Authentication\UserCredentialService" -t "REG_SZ
    " -v "Addresses" -d "" --force

systemctl restart ctxjproxy

systemctl restart ctxvda
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Limitation

- FAS doesn't support the lock screen yet. If you click the lock button in a session, you can't log back on to the session again by using FAS.
- This release supports only the common FAS deployments summarized in the Federated Authentication Service architectural overview article and doesn't include **Windows 10 Azure AD Join**.

Troubleshooting

Before troubleshooting FAS, make sure that:

- The Linux VDA is installed and configured correctly.
- A non-FAS session can be launched successfully on the common store by using password authentication.

If non-FAS sessions work properly, set the HDX log level of the **Login** class to VERBOSE and the VDA log level to TRACE. For information on enabling trace logging for the Linux VDA, see Knowledge Center article CTX220130.

You can also use the Linux **XDPing** tool to check for common configuration issues that might exist in your Linux VDA environment.

FAS server configuration error

Launching a session from the FAS store fails.

Check **/var/log/xdl/hdx.log** and find the error log similar to the following:

```
2021-01-28 01:42:16.164 <P26422:S4> citrix-ctxlogin: validate_user: [Logon Type] Federated Authentication Logon.

2021-01-28 01:42:16.164 <P26422:S4> citrix-ctxlogin: validate_fas: entry

2021-01-28 01:42:16.164 <P26422:S4> citrix-ctxlogin: connect_fas: start connect to server 0

2021-01-28 01:42:16.164 <P26422:S4> citrix-ctxlogin: connect_fas0: failed to connect: Connection refused.

2021-01-28 01:42:16.164 <P26422:S4> citrix-ctxlogin: validate_fas: failed to connect to server [0], please confirm if fas service list is well configurated in condb

2021-01-28 01:42:16.164 <P26422:S4> citrix-ctxlogin: validate_fas: exit , 43

```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
2021-01-28 01:42:16.164 <P26422:S4> citrix-ctxlogin: validate_user:
    failed to validate fas credential

2021-01-28 01:42:16.164 <P26422:S4> citrix-ctxlogin: LoginBoxValidate:
    failed validation of user 'user1@CTXDEV.LOCAL', INVALID_PARAMETER
```

Solution Run the following command to verify that the Citrix registry value "HKEY_LOCAL_MACHINE\SOFTWARE\
is set to <Your-FAS-Server-List>.

```
sudo /opt/Citrix/VDA/bin/ctxreg dump | grep "UserCredentialService"
```

If the existing setting is incorrect, follow the preceding Set FAS servers step to set it again.

Incorrect CA certificate configuration

Launching a session from the FAS store fails. A gray window appears and disappears several seconds later.

Check **/var/log/xdl/hdx.log** and find the error log similar to the following:

```
2021-01-28 01:47:46.210 <P30656:S5> citrix-ctxlogin:
    get_logon_certificate: entry

2021-01-28 01:47:46.210 <P30656:S5> citrix-ctxlogin: check_caller:
    current process: pid [30656], name [/opt/Citrix/VDA/bin/ctxlogin]

2021-01-28 01:47:46.210 <P30656:S5> citrix-ctxlogin:
    get_public_certificate: entry

2021-01-28 01:47:46.211 <P30656:S5> citrix-ctxlogin: query_fas: waiting
     for response...

2021-01-28 01:47:46.270 <P30656:S5> citrix-ctxlogin: query_fas: query
    to server success

2021-01-28 01:47:46.270 <P30656:S5> citrix-ctxlogin:
    get_public_certificate: exit

```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
2021-01-28 01:47:46.270 <P30656:S5> citrix-ctxlogin: fas_base64_decode:
    input size 1888

2021-01-28 01:47:46.271 <P30656:S5> citrix-ctxlogin: fas_base64_decode:
    output size 1415

2021-01-28 01:47:46.271 <P30656:S5> citrix-ctxlogin:
   get_logon_certificate: get logon certificate success

2021-01-28 01:47:46.271 <P30656:S5> citrix-ctxlogin: cache_certificate:
    cache certificate success

2021-01-28 01:47:46.271 <P30656:S5> citrix-ctxlogin:
   get_logon_certificate: exit, 0

2021-01-28 01:47:48.060 <P30656:S5> citrix-ctxlogin: validate_user:
   pam_authenticate err,can retry for user user1@CTXDEV.LOCAL
```

Solution Verify that you have correctly set in `/etc/krb5.conf` the full path that stores the root CA certificate and all intermediate certificates. The full path is similar to the following:

```
 [realms]

EXAMPLE.COM = {


    ......

    pkinit_anchors = DIR:/etc/pki/CA/certs/

    ......

 }
```

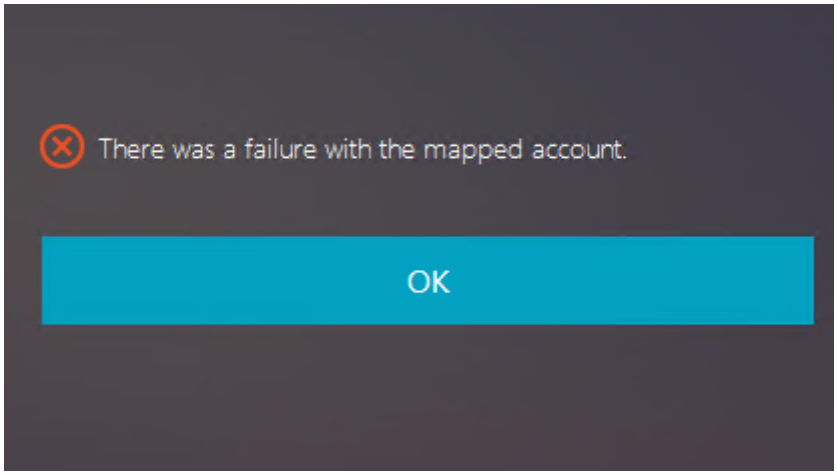
If the existing setting is incorrect, follow the preceding Install certificates step to set it again.

Alternatively, check whether the root CA certificate is valid.

Shadow account mapping error

FAS is configured by SAML authentication. The following error might occur after an ADFS user enters the user name and password on the ADFS logon page.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



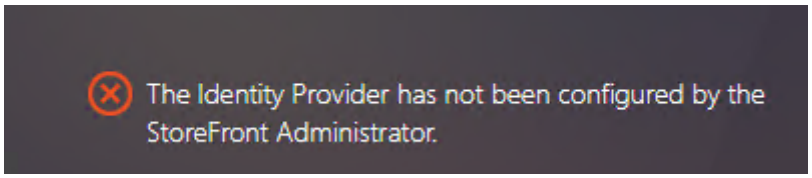
The error indicates that:

- The ADFS user has been verified successfully, but there is no shadow user configured on AD.

Solution Set the Shadow Account on AD.

ADFS not configured

The following error occurs during a logon attempt to the FAS store:



The issue occurs when you configure the FAS store to use SAML authentication but the ADFS deployment is missing.

Solution Deploy the ADFS IdP for Federated Authentication Service. For more information, see Federated Authentication Service ADFS deployment.

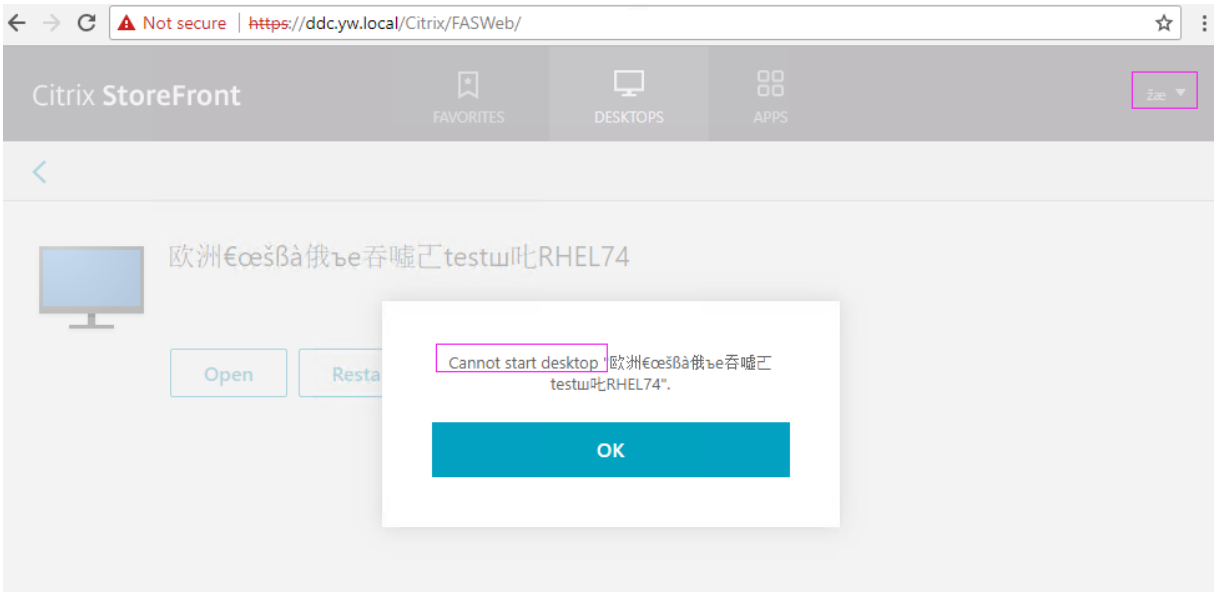
Related information

- The common FAS deployments are summarized in the Federated Authentication Service architectural overview article.
- "How-to"articles are introduced in the Federated Authentication Service advanced configuration chapter.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Known issues

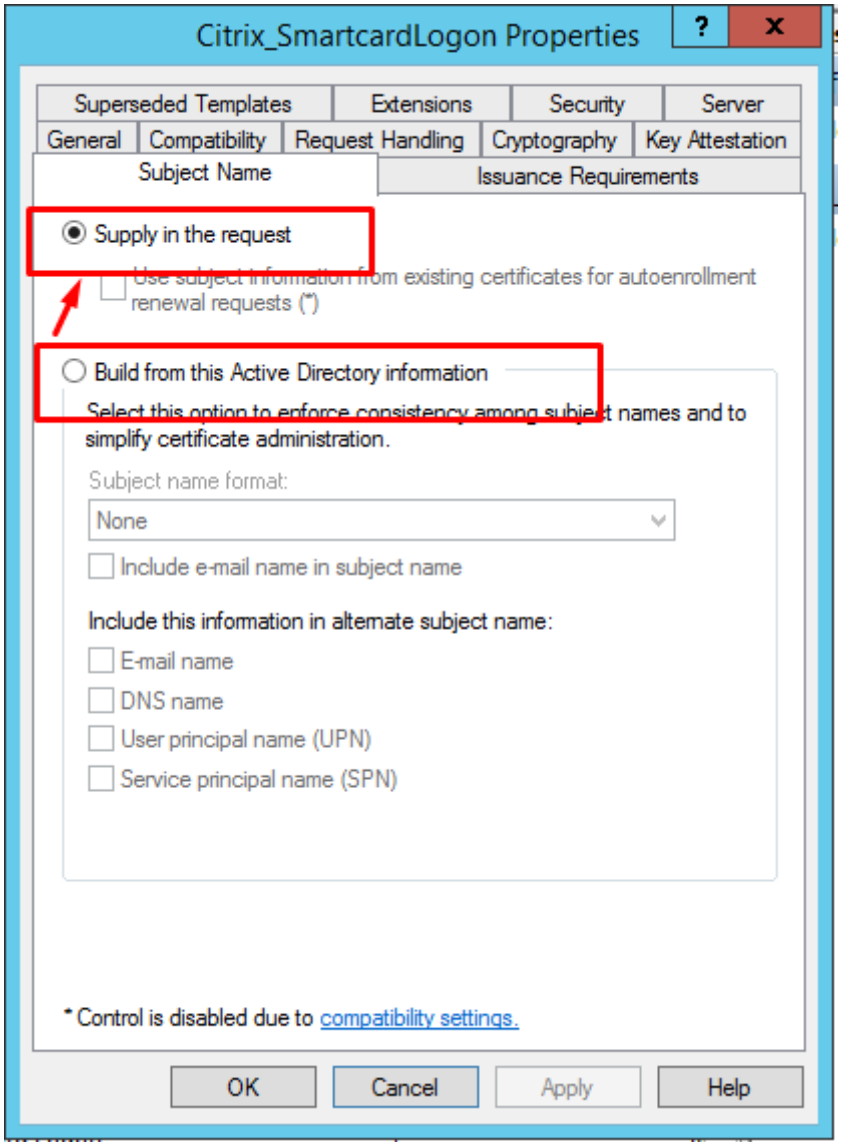
When FAS is in use, you can fail when trying to launch a published desktop or app session with non-English characters.



Workaround

Right-click **Manage Templates** in the CA tool to change the **Citrix_SmartcardLogon** template from **Build from this Active Directory information** to **Supply in the request**:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



FIDO2 (preview)

June 20, 2024

You can set up FIDO2 authentication to access websites using Google Chrome hosted on the Linux VDA.

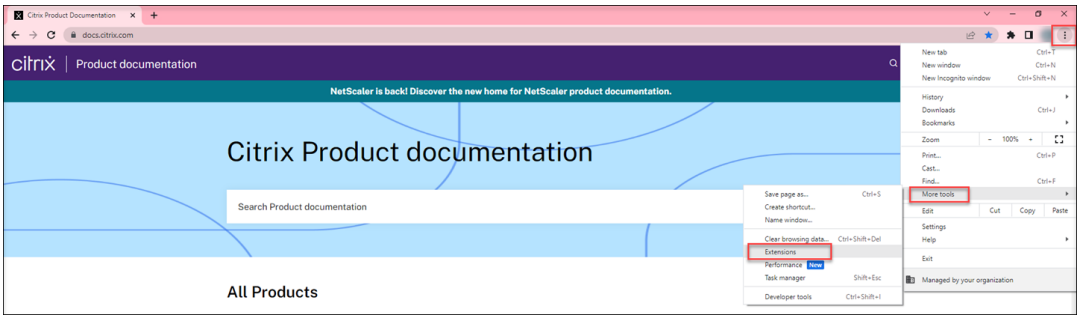
> **Note:**
>
> This feature is in preview. Preview features might not be fully localized and are recommended for use in non-production environments. Citrix Technical Support doesn't support issues found

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

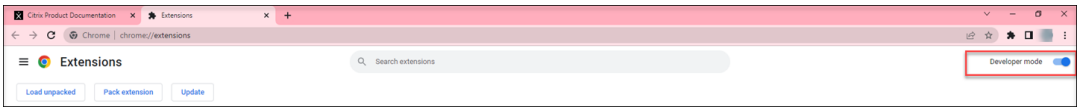
> with preview features.
>
> The Linux VDA supports only the combination of FIDO2 and Google Chrome.

To set up FIDO2 authentication, complete the following steps:

1. Download the Citrix FIDO2 extension package.
   a) Go to the Citrix Virtual Apps and Desktops download page.
   b) Expand the appropriate version of Citrix Virtual Apps and Desktops.
   c) Click **Components** to find the Linux VDA.
   d) Click the Linux VDA to open its download page.
   e) Download the sources package.
   f) Unzip the sources package to find **FIDO2-JavaScript-Extensions.zip**.
   g) Unzip the FIDO2 extension package. You can find the FIDO2 extension directory at **extensions > chrome > fido2**.
2. Add the Citrix FIDO2 extension in Google Chrome:
   a) Open Google Chrome hosted on the Linux VDA.
   b) Click the three-dot menu to the right of the address bar and then select **More Tools > Extensions**.



   c) Toggle on the **Developer mode**.



   d) Click **Load unpacked** and select the extension directory at **extensions > chrome > fido2**.
3. In the website that you want to use FIDO2 authentication for, register a FIDO2 security key to use FIDO2 authentication.
   a) Insert a FIDO2 security key to the client where Citrix Workspace app is installed.
   b) Enable multifactor authentication and add FIDO2 as the authentication method.

After FIDO2 authentication is set up, you will be prompted to touch the security key to access the website successfully.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Non-SSO authentication

June 20, 2024

This article provides guidance on how to enable non-SSO authentication on the Linux VDA.

Overview

By default, the Linux VDA has single sign-on (SSO) enabled. Users log on to Citrix Workspace app and to VDA sessions using one set of credentials.

To have users log on to VDA sessions using a different set of credentials, disable SSO on the Linux VDA. The following table lists combinations of user authentication methods supported in non-SSO scenarios.

Citrix Workspace app	VDA session
user name	user name
smart card	user name
user name	smart card
FAS	user name
FAS	smart card

Disable SSO

Run the following command on your Linux VDA:

```
1 /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet\
     Control\Citrix\WinStations\tcp" -t "REG_DWORD" -v "
     fPromptForDifferentUser" -d "0x00000001" --force
```

Smart cards

July 17, 2024

You can use a smart card connected to the client device for authentication when logging on to a Linux virtual desktop session. This feature is implemented through smart card redirection over the ICA smart card virtual channel. You can also use the smart card within the session. Use cases include:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- Adding a digital signature to a document
- Encrypting or decrypting an email
- Authenticating to a website

The Linux VDA uses the same configuration as the Windows VDA for this feature. For more information, see the Configure the smart card environment section in this article.

> **Note:**
>
> Using a mapped smart card within a Linux VDA session to sign on to Citrix Gateway isn't supported.

Prerequisites

The availability of smart card pass-through authentication is contingent on the following conditions:

- Your Linux VDA is installed on one of the following distributions:
  - RHEL 9.4/ 9.3/9.2
  - RHEL 8.10/8.9/8.8
  - Rocky Linux 9.4/ 9.3/9.2
  - Rocky Linux 8.10/8.9/8.8
  - Ubuntu 22.04
  - Ubuntu 20.04
  - Debian 12.5/11.9

  After you complete installing the VDA, verify that your VDA can register with the Delivery Controller and you can open the published Linux desktop sessions using Windows credentials.
- Smart cards supported by OpenSC are used. For more information, see Ensure that OpenSC supports your smart card.

Ensure that OpenSC supports your smart card

OpenSC is a widely used smart card driver on RHEL 7.4+. As a fully compatible replacement of CoolKey, OpenSC supports many types of smart cards (see Smart Card Support in Red Hat Enterprise Linux).

In this article, the YubiKey smart card is used as an example to illustrate the configuration. YubiKey is an all-in-one USB CCID PIV device that can easily be purchased from Amazon or other retail vendors. The OpenSC driver supports YubiKey.

If your organization requires some other more advanced smart card, prepare a physical machine with a supported Linux distribution and the OpenSC package installed. For information about the OpenSC

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

installation, see Install the smart card driver. Insert your smart card, and run the following command to verify that OpenSC supports your smart card:

```
pkcs11-tool --module opensc-pkcs11.so --list-slots
```

Configuration

Prepare a root certificate

A root certificate is used to verify the certificate on the smart card. Complete the following steps to download and install a root certificate.

1. Get a root certificate in PEM format, typically from your CA server.

   You can run a command similar to the following to convert a DER file (*.crt, *.cer, *.der) to PEM. In the following command example, **certnew.cer** is a DER file.

   ```
   openssl x509 -inform der -in certnew.cer -out certnew.pem
   ```

2. Install the root certificate to the `openssl` directory. The **certnew.pem** file is used as an example.

   ```
   cp certnew.pem <path where you install the root certificate>
   ```

   To create a path for installing the root certificate, run `sudo mdkir -p <path where you install the root certificate>`.

Configure the smart card environment

You can use the ctxsmartlogon.sh script to configure the smart card environment or complete the configuration manually.

(Option 1) Use the ctxsmartlogon.sh script to configure the smart card environment

> **Note:**
>
> The ctxsmartlogon.sh script adds PKINIT information to the default realm. You can change this setting through the **/etc/krb5.conf** configuration file.

Before using smart cards for the first time, run the ctxsmartlogon.sh script to configure the smart card environment.

> **Tip:**
>
> If you have used SSSD for domain joining, restart the SSSD service after you run ctxsmartlo-

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> gon.sh.
> For RHEL 8 and later, as PAM_KRB5 is moved to the EPEL repository, the script will try to enable EPEL in those distributions.

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

The results resemble the following:

```
#*****************************************************************************
# ctxsmartlogon.sh sets up smart card logon for the Linux VDA, which
# includes automatic installation of the necessary packages and changes to
# the configuration files.
#
# Note:
# The ctxsmartlogon.sh adds pkinit information to the default realm. You can
# change this setting through the /etc/krb5.conf configuration file.
#*****************************************************************************
Step 1:Enable smart card logon.
  Do you want enable smart card logon? (y/n)[y] y
Step 2:Select the Active Directory integration method.
  Please select which Active Directory integration method to use:
     1: Winbind
     2: SSSD
     3: Centrify
  Select one of the above options (1-3)[1] 1
Step 3:Install dependent packages.
  Installing, please wait...
    [krb5-pkinit][pam_krb5] already installed.
    [pcsc-lite][pcsc-lite-ccid][pcsc-lite-libs][coolkey] already installed.
  Packages installed. [Success]
Step 4:Configure krb5.conf.
  Specify the path to the root CA (e.g., /etc/pki/CA/certs/root.pem):/etc/pki/CA/certs/root.pem
  /etc/krb5.conf configure successfully.
Step 5:Configure PAM files.
  Specify the path to the smart card PKCS11 driver (e.g., /usr/lib64/pkcs11/libcoolkeypk11.so):/usr/lib64/pkcs11/libcoolkeypk11.so
  /etc/pam.d/ctxfsc configure successfully.
  /etc/pam.d/smartcard-auth configure successfully.
ctxsmartlogon.sh executed successfully. SmartCard is ready.
```

> **Note:**
>
> If you use Quest as the domain joining method, you are required to specify the host name of the Key Distribution Center (KDC). For example:

```
[root@jx-rh79qu110 ~]# /opt/Citrix/VDA/sbin/ctxsmartlogon.sh
#*****************************************************************************
# ctxsmartlogon.sh sets up smart card logon for the Linux VDA, which
# includes automatic installation of the necessary packages and changes to
# the configuration files.
#
# Note:
# The ctxsmartlogon.sh adds pkinit information to the default realm. You can
# change this setting through the /etc/krb5.conf configuration file.
#*****************************************************************************
Step 1:Enable smart card logon.
  Do you want enable smart card logon? (y/n)[y]
Step 2:Select the Active Directory integration method.
  Please select which Active Directory integration method to use:
     1: Winbind
     2: SSSD
     3: Centrify
     4: Quest
  Select one of the above options (1-4)[1] 4
Step 3:Install dependent packages.
  Installing, please wait...
    [krb5-pkinit][pam_krb5] already installed.
    [pcsc-lite][pcsc-lite-ccid][pcsc-lite-libs][opensc] already installed.
  Packages installed. [Success]
Step 4:Configure krb5.conf.
  Specify the path to the root CA (e.g., /etc/pki/CA/certs/root.pem):/etc/pki/CA/certs/root.pem
  Specify the KDC hostname:kdc.test.local
  /etc/krb5.conf configure successfully.
Step 5:Configure PAM files.
  Specify the path to the smart card PKCS11 driver (e.g., /usr/lib64/pkcs11/opensc-pkcs11.so):/usr/lib64/pkcs11/opensc-pkcs11.so
  /etc/pam.d/ctxfsc configure successfully.
  /etc/pam.d/smartcard-auth configure successfully.
ctxsmartlogon.sh executed successfully. SmartCard is ready.
```

You can also disable smart cards by running the ctxsmartlogon.sh script:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

The results resemble the following:

```
#*******************************************************************************
# ctxsmartlogon.sh sets up smart card logon for the Linux VDA, which
# includes automatic installation of the necessary packages and changes to
# the configuration files.
#
# Note:
# The ctxsmartlogon.sh adds pkinit information to the default realm. You can
# change this setting through the /etc/krb5.conf configuration file.
#*******************************************************************************
Step 1:Enable smart card logon.
  Do you want enable smart card logon? (y/n)[y] n
ctxsmartlogon.sh exit.
```

(Option 2) Configure the smart card environment manually The Linux VDA uses the same smart card environment with the Windows VDA. In the environment, multiple components must be configured, including the Domain Controller, Microsoft Certificate Authority (CA), Internet Information Services, Citrix StoreFront, and Citrix Workspace app. For information about the configuration based on the YubiKey smart card, see Knowledge Center article CTX206156.

Before moving to the next step, make sure that:

- You have configured all components correctly.
- You have downloaded the private key and user certificate to the smart card.
- You can successfully log on to the VDA using the smart card.

Install the PC/SC Lite packages PCSC Lite is an implementation of the Personal Computer/Smart Card (PC/SC) specification in Linux. It provides a Windows smart card interface for communicating to smart cards and readers. Smart card redirection in the Linux VDA is implemented on the PC/SC level.

Run the following command to install the PC/SC Lite packages:

RHEL 9.4/9.3/9.2/8.x, Rocky Linux 9.4/9.3/9.2/8.x:

```
yum install pcsc-lite pcsc-lite-ccid pcsc-lite-libs
```

Ubuntu 22.04, Ubuntu 20.04, Debian 12.5, Debian 11.9:

```
apt-get install -y libpcsclite1 libccid
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Install the smart card driver OpenSC is a widely used smart card driver. If OpenSC is not installed, run the following command to install it:

RHEL 9.4/9.3/9.2/8.x, Rocky Linux 9.4/9.3/9.2/8.x:

```
yum install opensc
```

Ubuntu 22.04, Ubuntu 20.04, Debian 12.5, Debian 11.9:

```
apt-get install -y opensc
```

Install the PAM modules for smart card authentication Run the following command to install the pam_krb5 and krb5-pkinit modules.

RHEL 9.4/9.3/9.2/8.x, Rocky Linux 9.4/9.3/9.2/8.x:

```
yum install krb5-pkinit
```

Ubuntu 22.04, Ubuntu 20.04:

```
apt-get install libpam-krb5 krb5-pkinit
```

Debian 12.5, Debian 11.9:

```
apt-get install -y libpam-krb5 krb5-pkinit
```

The pam_krb5 module is a pluggable authentication module. PAM-aware applications can use pam_krb5 to check passwords and obtain ticket-granting tickets from the Key Distribution Center (KDC). The krb5-pkinit module contains the PKINIT plug-in that allows clients to obtain initial credentials from the KDC using a private key and a certificate.

Configure the pam_krb5 module The pam_krb5 module interacts with the KDC to get Kerberos tickets using certificates in the smart card. To enable pam_krb5 authentication in PAM, run the following command:

```
authconfig --enablekrb5 --update
```

In the **/etc/krb5.conf** configuration file, add PKINIT information according to the actual realm.

> **Note:**
>
> The **pkinit_cert_match** option specifies matching rules that the client certificate must match before it is used to attempt PKINIT authentication. The syntax of the matching rules is:
>
> **[relation-operator] component-rule ...**
>
> where **relation-operator** can be either **&&**, meaning all component rules must match, or **||**,

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> meaning only one component rule must match.

Here is an example of a generic krb5.conf file:

```
EXAMPLE.COM = {


    kdc = KDC.EXAMPLE.COM

    auth_to_local = RULE:[1:$1@$0]

    pkinit_anchors = FILE:<path where you install the root certificate
       >/certnew.pem

    pkinit_kdc_hostname = KDC.EXAMPLE.COM

    pkinit_cert_match = ||<EKU>msScLogin,<KU>digitalSignature

    pkinit_eku_checking = kpServerAuth

 }
```

The configuration file resembles the following after you add the PKINIT information.

```
CTXDEV.LOCAL = {
    kdc = [redacted]
    auth_to_local = RULE:[1:$1@$0]
    pkinit_kdc_hostname = ctx-ad.ctxdev.local
    pkinit_anchors = FILE:/etc/pki/CA/certs/certnew.pem
    pkinit_eku_checking = kpServerAuth
    pkinit_cert_match = ||<EKU>msScLogin,<KU>digitalSignature
}
```

Configure PAM authentication PAM configuration files tell what modules are used for PAM authentication. To add pam_krb5 as an authentication module, add the following line to the **/etc/pam.d/smartcard-auth** file:

```
auth [success=done ignore=ignore default=die] pam_krb5.so preauth_options
=X509_user_identity=PKCS11:<path to the pkcs11 driver>/opensc-pkcs11.
so
```

The configuration file resembles the following after modification if SSSD is used.

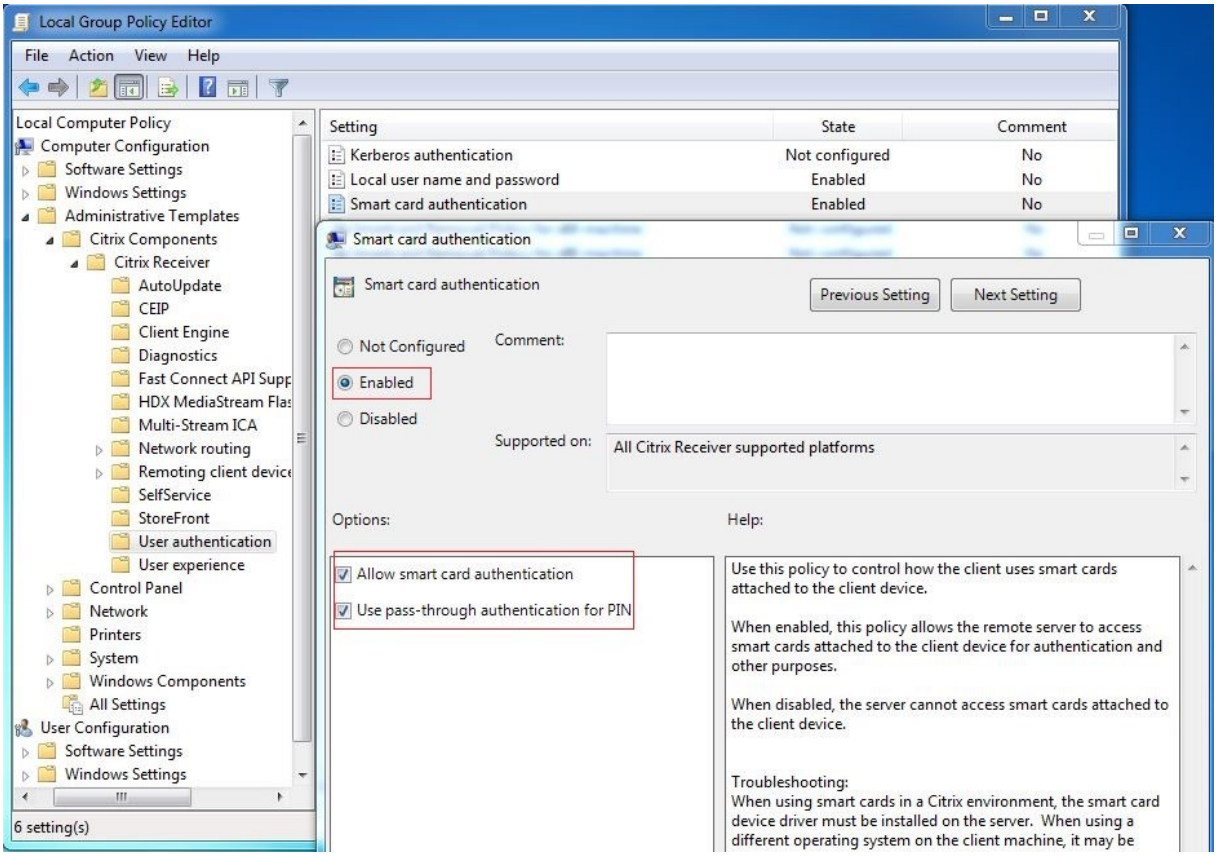
© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
#%PAM-1.0
# This file is auto-generated.
# User changes will be destroyed the next time authconfig is run.
auth        required      pam_env.so
auth        [success=done ignore=ignore default=die]  pam_krb5.so preauth_opt=X509_user_identity=PKCS11:/usr/lib/x86_64-linux-gnu/pkcs11/opensc-pkcs11.so
auth        sufficient    pam_permit.so
auth        required      pam_deny.so

account     required      pam_unix.so
account     sufficient    pam_localuser.so
account     sufficient    pam_succeed_if.so uid < 1000 quiet
account     [default=bad success=ok user_unknown=ignore] pam_sss.so
account     [default=bad success=ok auth_err=ignore user_unknown=ignore ignore=ignore]  pam_krb5.so
account     required      pam_permit.so

session     optional      pam_keyinit.so revoke
session     required      pam_limits.so
-session     optional      pam_systemd.so
#session     optional      pam_oddjob_mkhomedir.so umask=0077
session     optional      pam_mkhomedir.so umask=0077
session     [success=1 default=ignore] pam_succeed_if.so service in crond quiet use_uid
session     required      pam_unix.so
session     optional      pam_sss.so
session     optional      pam_krb5.so
```

(Optional) Single sign-on by using smart cards

Single sign-on (SSO) is a Citrix feature that implements pass-through authentication with virtual desktop and application launches. This feature reduces the number of times that users type their PIN. To use SSO with the Linux VDA, configure Citrix Workspace app. The configuration is the same with the Windows VDA. For more information, see Knowledge Center article CTX133982.

Enable the smart card authentication as follows when configuring the group policy in Citrix Workspace app.



© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Fast smart card logon

Fast smart card is an improvement over the existing HDX PC/SC-based smart card redirection. It improves performance when smart cards are used in high-latency WAN environments. For more information, see Smart cards.

The Linux VDA supports fast smart card on the following versions of Citrix Workspace app:

- Citrix Receiver for Windows 4.12
- Citrix Workspace app 1808 for Windows and later

Enable fast smart card logon on the client Fast smart card logon is enabled by default on the VDA and disabled by default on the client. On the client, to enable fast smart card logon, include the following parameter in the default.ica file of the associated StoreFront site:

```
[WFClient]
SmartCardCryptographicRedirection=On
```

Disable fast smart card logon on the client To disable fast smart card logon on the client, remove the **SmartCardCryptographicRedirection** parameter from the default.ica file of the associated StoreFront site.

Run XDPing

After completing the preceding steps, you can use the Linux **XDPing** tool to check for common configuration issues that might exist in your Linux VDA environment.

Usage

Log on to the Linux VDA by using a smart card

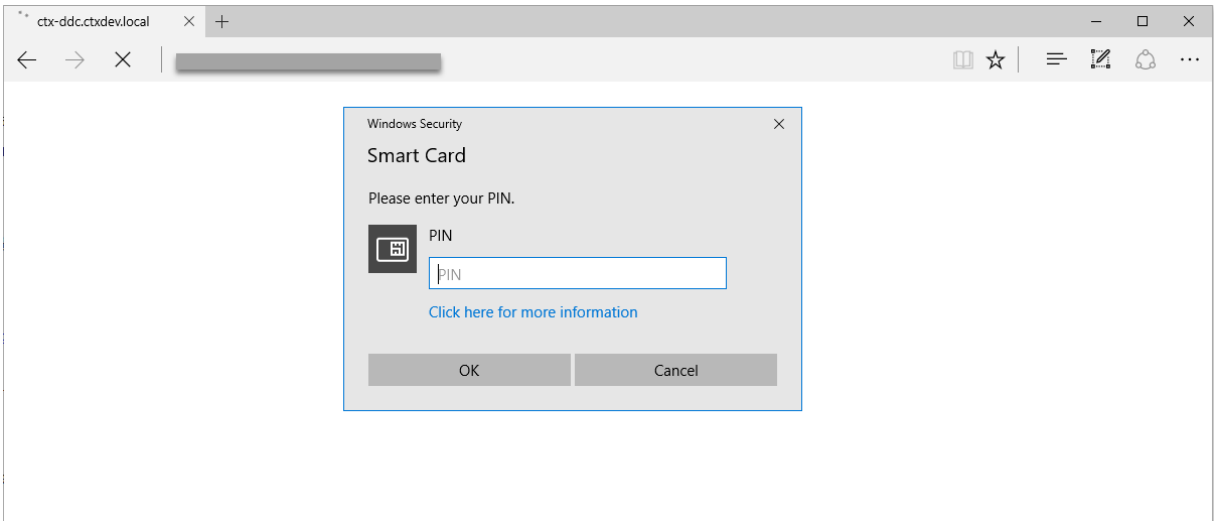
You can use a smart card to log on to the Linux VDA in both SSO and non-SSO scenarios.

- In the SSO scenario, you are logged on to StoreFront automatically by using the cached smart card certificate and PIN. When you launch a Linux virtual desktop session in StoreFront, the PIN is passed to the Linux VDA for smart card authentication.
- In the non-SSO scenario, you are prompted to select a certificate and type a PIN to log on to StoreFront.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

When you launch a Linux virtual desktop session in StoreFront, a dialog box for logon to the Linux VDA appears as follows. The user name is extracted from the certificate in the smart card and you must type the PIN again for logon authentication.

This behavior is the same with the Windows VDA.



Reconnect to a session by using a smart card

To reconnect to a session, ensure that the smart card is connected to the client device. Otherwise, a gray caching window appears on the Linux VDA side and exits quickly because reauthentication fails without the smart card connected. No other prompt is provided in this case to remind you to connect the smart card.

On the StoreFront side, however, if a smart card is not connected when you reconnect to a session, the StoreFront web might give an alert as follows:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Limitation

Support for limited Linux distributions and AD integration methods

- Smart card pass-through authentication supports limited Linux distributions and AD integration methods. See the following matrix:

	Winbind	SSSD	Centrify	Quest
Debian 12.5/11.9	Yes	Yes	Yes	No
RHEL 9.4/9.3/9.2	Yes	Yes	No	Yes
RHEL 8.10/8.9/8.8	Yes	Yes	Yes	Yes
Rocky Linux 9.4/9.3/9.2	Yes	Yes	No	No
Rocky Linux 8.10/8.9/8.8	Yes	Yes	No	No
Ubuntu 22.04/20.04	Yes	Yes	Yes	Yes

Smart card removal settings

You can determine what happens when the smart card for a signed-in user is removed from the smart card reader during a session by editing the following registry key on the Linux VDA:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\LocalPolicies\SecurityOptions

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Value name	Type	Default	Description
SCardRemoveOption	REG_DWORD	0x00000000	This setting determines what happens when the smart card for a signed-in user is removed from the smart card reader during a session. The options are as follows. 0 **No Action** 1 **Force Logoff**: The user is signed out automatically upon removal of the smart card. 2 **Disconnect Session**: The session is disconnected without signing out the user upon removal of the smart card. The user can reinsert the smart card to reconnect the session later.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Value name	Type	Default	Description
SCardRemoveActionDelaySeconds	REG_DWORD	15	If the removal option is set as **Force Logoff** or **Disconnect Session**, you can further specify a delay of several seconds before the action is taken. With this setting, a message box appears in the user session after the smart card is removed, indicating that the session will be force logged off/disconnected after the specified seconds. If you reinsert the smart card before that time, the session continues uninterrupted

Limitations The Linux VDA supports only one smart card reader at a time.

Support for other smartcards and the PKCS#11 library

Citrix provides a generic smart card redirection solution. Although only the OpenSC smart card is listed on our support list, you can try using other smart cards and the PKCS#11 library. To switch to your specific smart card or the PKCS#11 library:

1. Replace all the `opensc-pkcs11.so` instances with your PKCS#11 library.
2. To set the path of your PKCS#11 library to the registry, run the following command:

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\
   CurrentControlSet\Control\Citrix\VirtualChannels\Scard" -v "
   PKCS11LibPath" -d "PATH"
```

where **PATH** points to your PKCS#11 library such as /usr/lib64/pkcs11/opensc-pkcs11.so

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

3. Disable fast smart card logon on the client.

Access by unauthenticated (anonymous) users

June 20, 2024

You can allow users to access applications and desktops without presenting credentials to StoreFront or Citrix Workspace app. To grant access to unauthenticated users, you must have an unauthenticated StoreFront store and enable access for unauthenticated users in a delivery group.

> **Note:**
>
> Access by unauthenticated users is supported for domain-joined VDAs only.
>
> Session prelaunch is not supported for unauthenticated users. Session prelaunch is also not supported on Citrix Workspace app for Android.

Create an unauthenticated StoreFront store

1. Select the **Stores** node in the left pane of the Citrix StoreFront management console and, in the **Actions** pane, click **Create Store**.
2. On the **Store Name** page, specify a name for your store, select **Allow only unauthenticated (anonymous) users to access this store**, and click **Next**.

For more information, see Create an unauthenticated store.

Enable access for unauthenticated users in a delivery group

A delivery group is a collection of machines selected from one or more machine catalogs. When you specify who can use the applications and desktops in a delivery group, you can grant access to unauthenticated users. For more information, see
Create delivery groups.

Set the idle timeout for unauthenticated user sessions

Unauthenticated user sessions have a default idle timeout of 10 minutes, and are logged off automatically when the client disconnects. You can configure a custom idle timeout through the registry setting **AnonymousUserIdleTime**. For example, to set a custom idle timeout of five minutes, run the following command:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

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 user 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, run the following command:

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

> **Important:**
>
> Limit the number of unauthenticated user 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 user sessions:

- **Failed to log on to an unauthenticated user 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 user session with KDE**.

The lock screen button and menu are disabled by default in an unauthenticated user session. However, they can still be displayed in KDE. In KDE, to disable the lock screen button and menu for a particular user, add the following lines to the configuration file **$Home/.kde/share/config/kdeglobals**. For example:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

File

June 20, 2024

This section contains the following topics:

- File copy and paste
- File transfer

File copy and paste

June 20, 2024

Users can copy and paste files between a session and a local client by using the right-click menu or keyboard shortcuts. This feature requires Citrix Virtual Apps and Desktops 2006 or later and Citrix Workspace app 1903 or later for Windows.

To copy and paste files successfully, ensure that:

- The maximum number of files does not exceed 20.
- The maximum file size does not exceed 200 MB.
- The Nautilus file manager is available on the machine where you installed the Linux VDA.
- The file names contain only ASCII characters and no special characters.

Supported Linux distributions

The **file copy and paste feature** is available for all Linux distributions that the Linux VDA supports.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Relevant policies

The following clipboard policies are relevant to configuring the feature. For more information about the clipboard policies, see the Policy support list.

- Client clipboard redirection
- Clipboard selection update mode
- Primary selection update mode

> **Note:**
>
> To disable the file copy and paste feature, set the **Client clipboard redirection** policy to **Prohibited** in Citrix Studio.

Limitations

- Cut is not supported. Requests to cut a file are treated as copy operations.
- Drag and drop is not supported.
- Copying directories is not supported.
- File copy and paste must be performed sequentially. Only after the previous file is copied and pasted successfully can the next file be copied.

File transfer

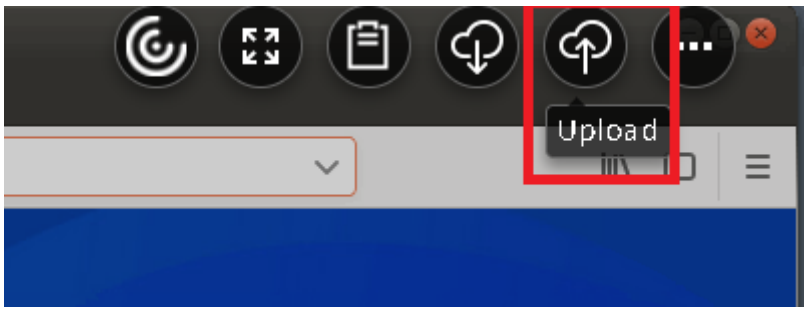
June 20, 2024

File transfer is supported between the Linux VDA and the client device. This feature is available when the client device runs a web browser that supports the HTML5 sandbox attribute. The HTML5 sandbox attribute allows users to access virtual desktops and apps using Citrix Workspace app for HTML5 and for Chrome.

> **Note:**
>
> File transfer is available for Citrix Workspace app for HTML5 and for Chrome.

Within published app and desktop sessions, file transfer allows file uploads and downloads between the Linux VDA and the client device. To upload files from the client device to the Linux VDA, click the **Upload** icon on the toolbar of Citrix Workspace app and select the file you want from the file dialogs. To download files from the Linux VDA to the client device, click the **Download** icon. You can add files during uploading or downloading. You can transfer up to 10 files at any one time.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



> **Note:**
>
> To upload and download files between the Linux VDA and the client device, enable the toolbar of Citrix Workspace app.
> You can use a version of Citrix Workspace app that lets you drag and drop files.

Autodownload is an enhancement for file transfer. Files you download or move to the **Save to My Device** directory on the VDA are transferred to the client device automatically.

> **Note:**
>
> Autodownload requires the **File transfer for Citrix Workspace app for ChromeOS/HTML5** and **Download file for Citrix Workspace app for ChromeOS/HTML5** policies to be set to **Allowed**.

Here are some use cases for auto-download:

- Download files to **Save to My Device**

  Within published desktop and web browser app sessions, files you download from websites can be saved to the **Save to My Device** directory on the VDA for automatic transfer to the client device. To achieve auto-download, set the default download directory of the in-session web browser to **Save to My Device** and set a local download directory in the web browser that runs your Citrix Workspace app for HTML5 or for Chrome.

- Move or copy files to **Save to My Device**

  Within published desktop sessions, choose the target files and move or copy them to the **Save to My Device** directory for availability on the client device.

File transfer policies

By default, file transfer is enabled. Use Citrix Studio to change these policies, located under **User Setting > ICA\File Redirection**. Consider the following when using file transfer policies:

File transfer for Citrix Workspace app for ChromeOS/HTML5: Allows or prevents end users from transferring files between Citrix Virtual Apps and Desktops session or Citrix DaaS session and the end user's device.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

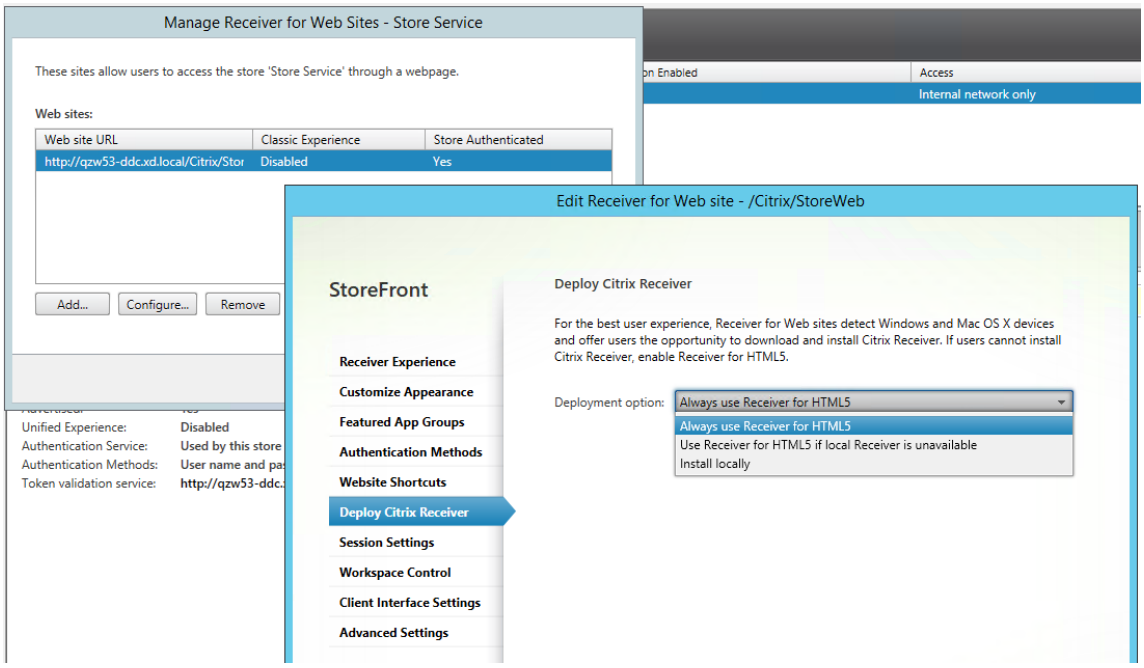
Upload file for Citrix Workspace app for ChromeOS/HTML5: Allows or prevents users from uploading files. It can be from the user's device to a Citrix Virtual Apps and Desktops session or Citrix DaaS session.

Download file for Citrix Workspace app for ChromeOS/HTML5: Allows or prevents users from downloading files. It can be from a Citrix Virtual Apps and Desktops session or Citrix DaaS session to the user's device.

Usage

To use the file transfer feature through Citrix Workspace app for HTML5:

1. In Citrix Studio, set the **WebSockets connections** policy to **Allowed**.
2. In Citrix Studio, enable file transfer through the file transfer policies described earlier.
3. In the Citrix StoreFront management console, click **Stores**, select the **Manage Receiver for Web Sites** node, and enable Citrix Receiver for HTML5 by selecting the **Always use Receiver for HTML5** option.



4. Launch a virtual desktop or web browser app session. Perform one or more file transfers between the Linux VDA and your client device.

To use the file transfer feature through Citrix Workspace app for Chrome:

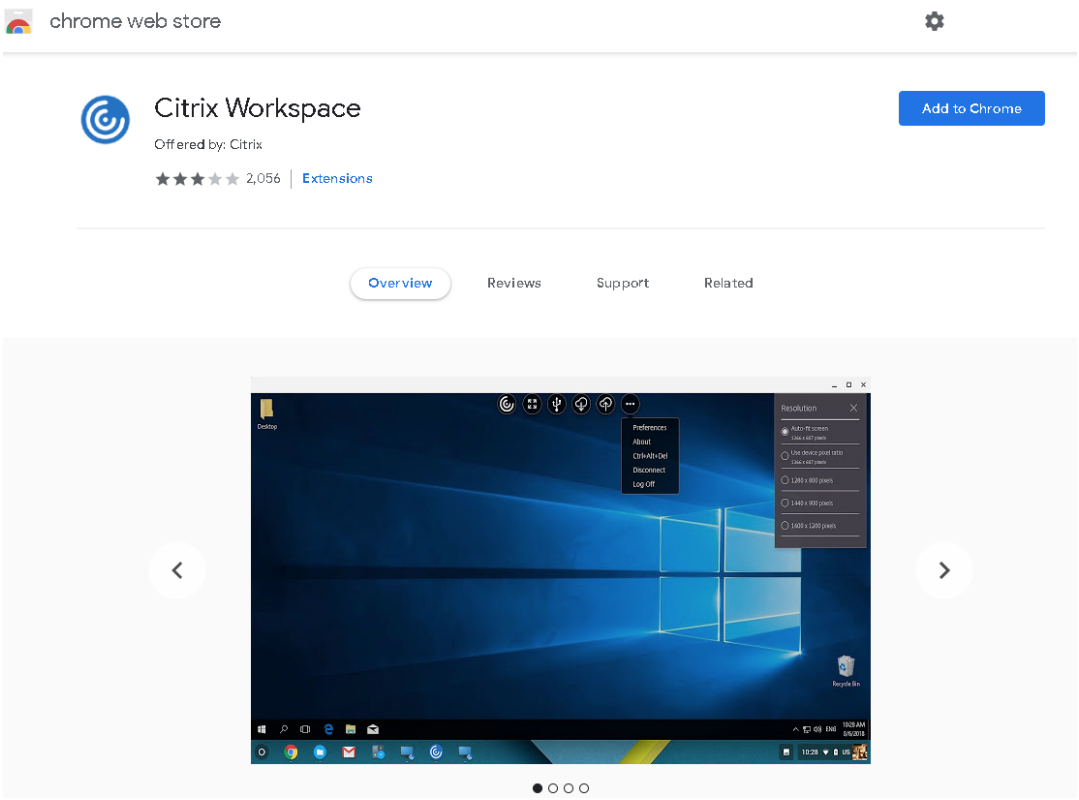
1. Enable file transfer through the file transfer policies described earlier.
2. Obtain Citrix Workspace app from the Chrome Web Store.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Skip this step if you already added Citrix Workspace app for Chrome to the Chrome Apps page.

a) Type **Citrix Workspace for Chrome** in the search box of Google Chrome. Click the search icon.

b) Among the search results, click the URL to the Chrome Web Store where Citrix Workspace app is available.



c) Click **Add to Chrome** to add Citrix Workspace app to Google Chrome.

3. Click Citrix Workspace app for Chrome on the Chrome Apps page.
4. Type the URL of your StoreFront store to connect.

   Skip this step if you typed the URL before.
5. Launch a virtual desktop or app session. Perform one or more file transfers between the Linux VDA and your client device.

Graphics

June 20, 2024

This section contains the following topics:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- Automatic DPI scaling
- Client battery status display
- Graphics configuration and fine-tuning
- HDX screen sharing
- Non-virtualized GPUs
- Session watermark
- Thinwire progressive display

Automatic DPI scaling

June 20, 2024

The Linux VDA supports automatic DPI scaling. When a user opens a virtual desktop or application session, the DPI value in the session automatically changes to match the DPI setting on the client side.

The following are considerations related to this feature:

- The feature requires that you enable DPI matching for Citrix Workspace. In the case of Citrix Workspace app for Windows, make sure that the **No, use the native resolution** option is selected. For more information about configuring DPI scaling for Citrix Workspace app for Windows, see DPI scaling.
- For the feature to work in multi-monitor scenarios, each monitor must be configured with the same DPI setting. Mixed DPI scenarios are not supported. If monitors are configured with different DPI settings, the Linux VDA applies the smallest DPI value for all screens.
- The feature is enabled for MATE, GNOME, GNOME Classic, and KDE. When using KDE or MATE, consider the following:
  - For Linux virtual desktops running in a KDE desktop environment:
    * We recommend using KDE Plasma 5 or later.
    * Changing DPI settings on the client side while sessions are running requires users to log off and log back on.
  - For Linux virtual desktops running in a MATE desktop environment:
    * Only scale factors of 1 and 2 are supported.
    * Changing DPI settings on the client side while sessions are running requires users to log off and log back on.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- The DPI value in the virtual session automatically changes according to the DPI setting on the client side. Currently, the feature supports only scale factors of type integer, for example, 100% and 200%. If the scale factor configured on the client side is of type fractional, the virtual session DPI changes to an integer scale factor according to the following table. Example: If the scale factor is 125%, the DPI value changes to 100%.

Client-side scale factor	Remote session DPI	Display scaling
Less than or equal to 174%	96 (1 x 96)	100%
175%–274%	192 (2 x 96)	200%
275%–399%	288 (3 x 96)	300%
Greater than or equal to 400%	384 (4 x 96)	400%

> **Note:**
>
> The Linux VDA supports up to 200% scaling for MATE desktops.

Client battery status display

June 20, 2024

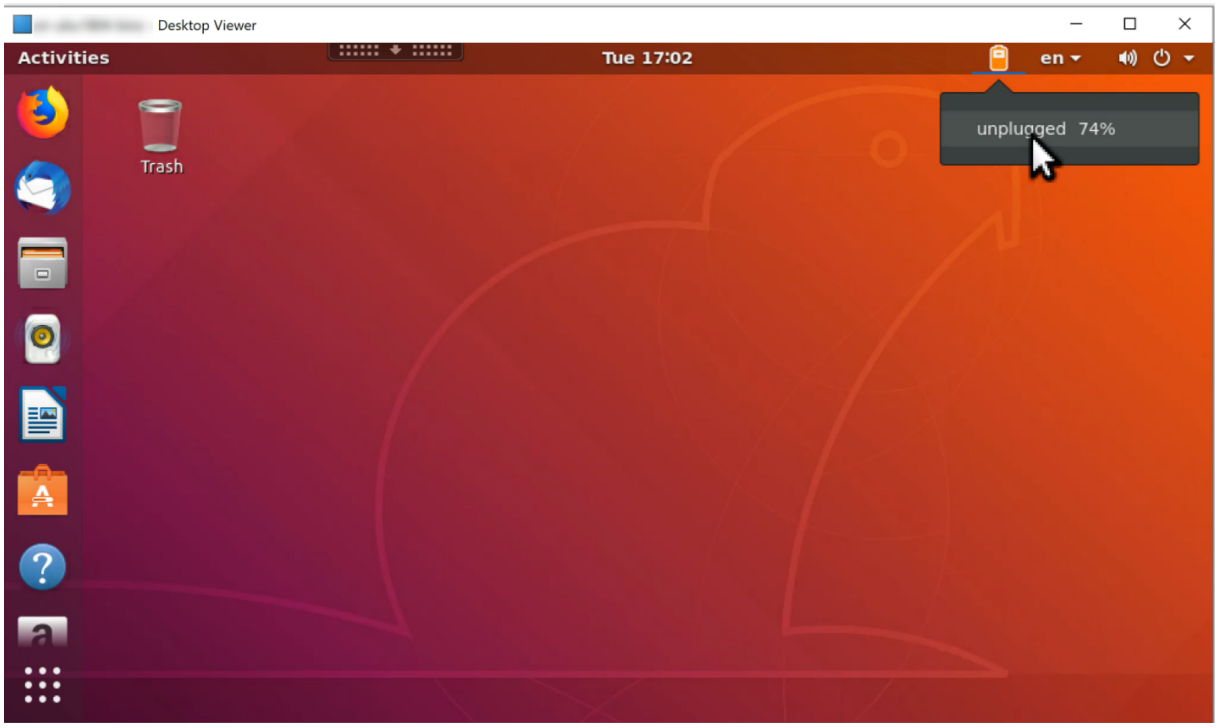
The Linux VDA can redirect and display the battery status of client devices in virtual desktops. This feature is enabled by default and available for the following versions of Citrix Workspace app:

- Citrix Workspace app for iOS
- Citrix Workspace app for Linux
- Citrix Workspace app for Mac (version 2204.1 is not supported)
- Citrix Workspace app for Windows (version 2204.1 is not supported)

Overview

When users open a virtual desktop, they can see a battery icon in the Linux system tray. The battery icon indicates the battery status of their client devices. To check for the percentage of remaining battery life, click the battery icon. For example, see the following screen capture:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



Different battery icons indicate different battery statuses. For an overview, see the following table:

Battery icon	Charging status	Level of remaining battery life	Percentage of remaining battery life
	Charging, indicated with a “+”symbol	High, indicated with a green color	=80%
	Charging, indicated with a “+”symbol	Medium, indicated with an amber color	=20% and <80%
	Charging, indicated with a “+”symbol	Low, indicated with a red color	< 20%
	Not charging, indicated with a “-“symbol	High, indicated with a green color	=80%
	Not charging, indicated with a “-“symbol	Medium, indicated with an amber color	=20% and <80%

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Battery icon	Charging status	Level of remaining battery life	Percentage of remaining battery life
	Not charging, indicated with a “-“symbol	Low, indicated with a red color	< 20%
	Unknown	Unknown	Unknown

Configuration

The client battery status display is enabled by default.

To disable the feature, run the following command:

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\CurrentControlSet\
   Control\Citrix\VirtualChannels\MrVc"  -v "Enabled" -d "0x00000000"
```

To enable the feature, run the following command:

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\CurrentControlSet\
   Control\Citrix\VirtualChannels\MrVc"  -v "Enabled" -d "0x00000001"
```

> **Note:**
>
> The preceding commands impact the soft keyboard feature, which shares the Mobile Receiver Virtual Channel (MRVC) with the client battery status display.

Based on your distribution, complete the extra steps to enable the feature. For more information, see the Enable the system tray section.

Graphics configuration and fine-tuning

June 20, 2024

This article describes the Linux VDA graphics configuration and fine-tuning.

For more information, see System requirements and the Installation overview section.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Configuration

Optimize for 3D graphics workload

This setting configures the appropriate default values that best suit graphics-intensive workloads. Enable this setting for users whose workload focuses on graphics-intensive applications. Apply this policy only in cases where a GPU is available to the session. Any other settings that explicitly override the default settings set by this policy take precedence.

By default, **Optimize for 3D graphics workload** is disabled.

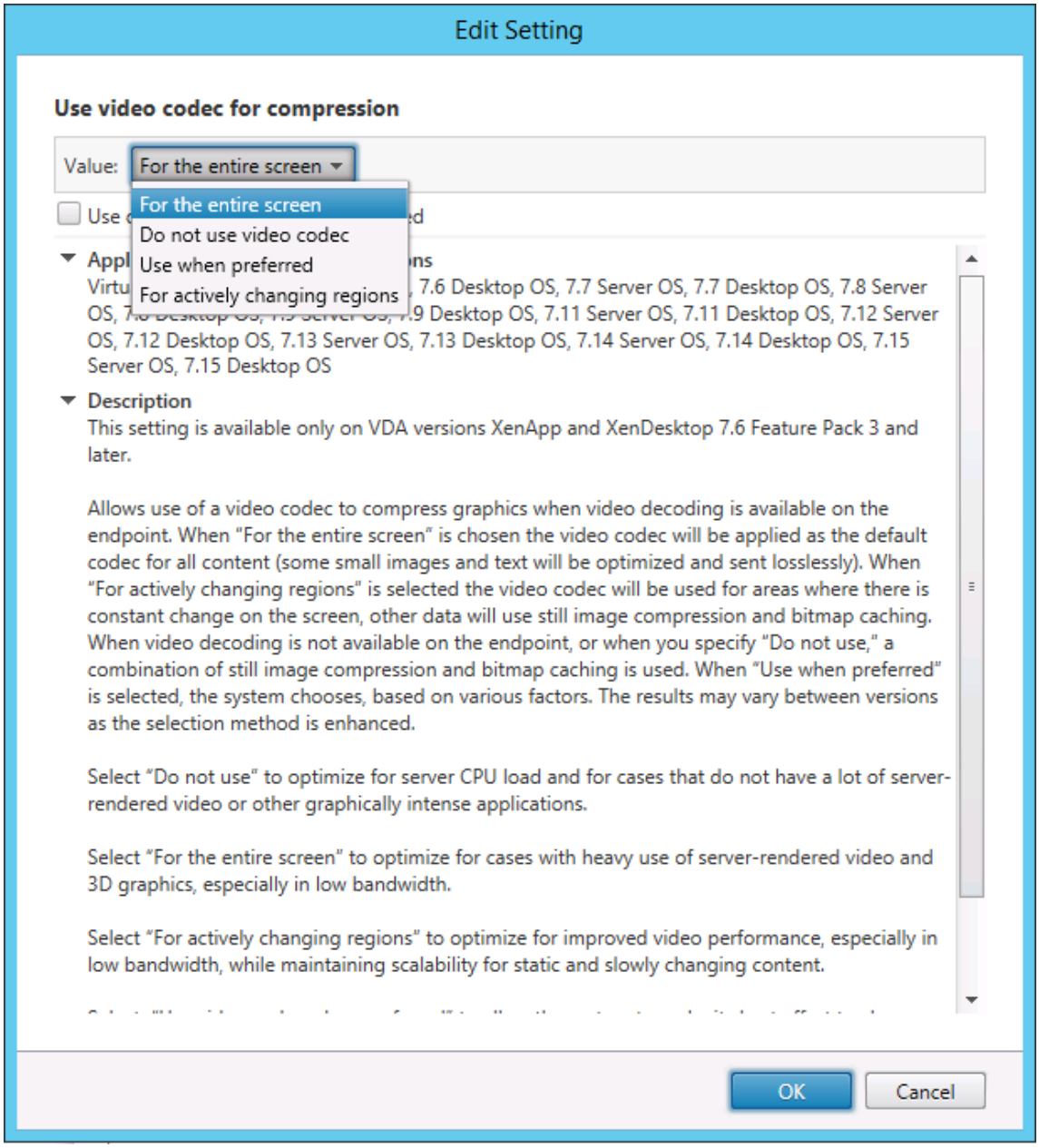
Video codec for compression

Thinwire is the display-remoting technology used in the Linux VDA. The technology allows graphics generated on one machine to be transmitted, typically across a network, to another machine for display.

The **Use video codec for compression** graphics policy sets the default graphics mode and provides the following options for different use cases:

- **Use when preferred**. This setting is the default. No additional configuration is required. It ensures that Thinwire is selected for all Citrix connections, and optimized for scalability, bandwidth, and superior image quality for typical desktop workloads.
- **For the entire screen**. Delivers Thinwire with full-screen H.264 or H.265 to optimize for improved user experience and bandwidth, especially in cases with heavy use of 3D graphics. Session watermark is supported when **For the entire screen** is selected, or when **Use when preferred** is selected and Optimize for 3D graphics workload is enabled.
- **For actively changing regions**. The adaptive display technology in Thinwire identifies moving images (video, 3D in motion). It uses AV1, H.265, or H.264 only in the part of the screen where the image is moving. The selective use of the AV1, H.265, or H.264 video codec enables HDX Thinwire to detect and encode parts of the screen that are frequently updated. Still image compression (JPEG, RLE) and bitmap caching continue to be used for the rest of the screen, including text and photographic imagery. Users get the benefit of lower bandwidth consumption and better quality for video content combined with lossless text or high-quality imagery elsewhere. The selective use of AV1 and H.265 is not supported when the **Visual quality** policy is set to **Always Lossless** or **Build to Lossless**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



Some other policy settings, including the following visual display policy settings can be used to fine-tune the performance of display remoting:

- **Preferred color depth for simple graphics**
- **Target frame rate**
- **Visual quality**

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

AV1/H.265/H.264 hardware encoding

The **Use hardware encoding for video codec** policy allows the use of GPU hardware acceleration, if available, to compress screen elements with the video codec. GPU hardware acceleration optimizes hardware resource utilization and highly improves the performance of frames per second (FPS).

GPU hardware acceleration covers all the graphics modes set by the **Use video codec for compression** policy:

- **Use when preferred**
- **For the entire screen**
- **For actively changing regions**

To enable hardware video compression, complete the following steps:

1. Set the **Use hardware encoding for video codec** policy to **Enabled**.
2. Set **Use video codec for compression** to **Use when preferred**, **For the entire screen**, or **For actively changing regions**. Ensure that it is not set to **Do not use video codec**.

To be usable, AV1 or H.265 video codec must be supported and enabled on both the VDA and Citrix Workspace app. AV1 gets preference over H.265 and H.264 during codec negotiation. When AV1 is not supported, H.265 gets negotiated. If both AV1 and H.265 are not supported, sessions fall back to using the H.264 video codec. If GPU hardware is not available, the VDA falls back to CPU-based encoding using the software video codec.

Requirements for AV1 hardware encoding

VDA

- VDA: 2311 or later
- GPU: **NVIDIA Ada Lovelace or later** (For a matrix of the video codecs that NVIDIA GPUs support, see the NVIDIA document at https://developer.nvidia.com/video-encode-and-decode-gpu-support-matrix-new.)
- NVIDIA graphics driver 522.25 or later (Video Codec SDK v12.0)

Client

- Citrix Workspace app 2305 for Windows or later
- Client GPU that supports AV1 decoding:
  - NVIDIA Ampere or later
  - Intel 11th Gen / Arc or later
  - AMD Radeon RX 6000 / Radeon Pro W6000 series (RDNA2) or later

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Requirements for H.265 hardware encoding

Client

- Citrix Receiver for Windows 4.10 through 4.12
- Citrix Workspace app 1808 for Windows and later

To enable H.265 hardware encoding on your client, see H.265 video encoding.

H.265/H.264 lossless compression

H.265/H.264 lossless compression is available for HDX 3D PRO hardware acceleration by NVIDIA GPUs. H.265 lossless compression requires Citrix Workspace app 2305 for Windows and later. H.264 lossless compression requires the following clients:

- Citrix Workspace app 2303 for Windows and later
- Citrix Workspace app 2301 for Mac and later with the Apple M1 chip

To enable H.265/H.264 lossless compression, complete the following steps:

1. Set the **Use hardware encoding for video codec** policy to **Enabled**.
2. Set the **Use video codec for compression** policy to **For the entire screen**.
3. Set the **Visual quality** policy to **Always losses** or **Build to Lossless**.

Allow visually lossless compression

The **Allow visually lossless compression** policy allows **visually** lossless compression to be used instead of true lossless compression for graphics. Visually lossless improves performance over true lossless, but has minor loss that is unnoticeable by sight. This setting changes the way the values of the **Visual quality** setting are used.

The **Allow visually lossless compression** policy is disabled by default. To enable **visually** lossless compression, set **Allow visually lossless compression** to **Enabled** and the **Visual quality** policy to **Build to Lossless**.

If the **Use video codec for compression** policy is set to **Do not use video codec**, **visually** lossless compression applies to static image encoding. If the **Use video codec for compression** policy is set to a graphics mode other than **Do not use video codec**, **visually** lossless compression applies to H.264 encoding.

For more information about the **Visual quality** and **Use video codec for compression** policy settings, see Visual display policy settings and Graphics policy settings.

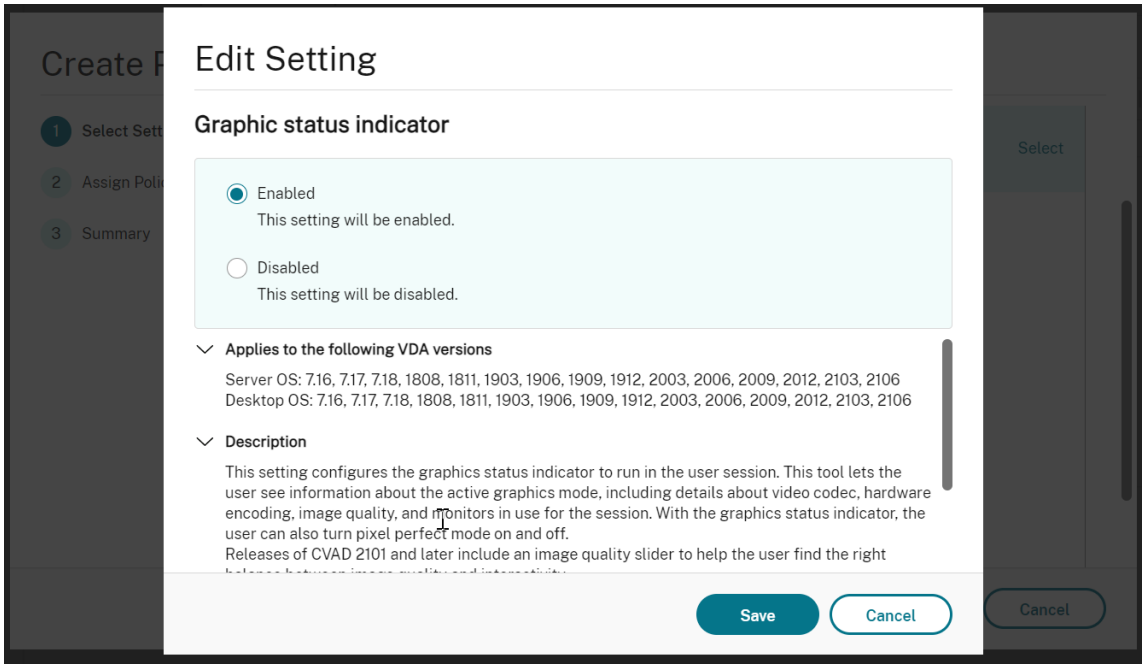
© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Graphics quality slider

We have included a graphics quality slider in the graphics status indicator tool that runs in your virtual Linux sessions. The slider helps to find the right balance between image quality and interactivity.

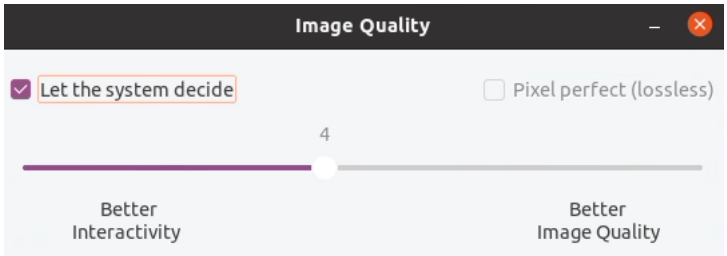
To use the slider, complete the following steps:

1. Enable the **Graphic status indicator** policy in Citrix Studio.



2. Open the Terminal and run the `ctxslider` command. The slider UI appears.

> **Note:**
>
> - If you have set the **Visual Quality** policy to **Always Lossless** or **Build to Lossless**, the slider UI is not showing.
> - You can launch the slider UI from both the Terminal and the system tray.



The following choices are now available:

- To change the image quality, move the slider. The slider supports a range of 0–9.
- To use system-defined settings, select **Let the system decide**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- To switch to lossless mode, select **Pixel perfect**.

Adjust average bit rates based on bandwidth estimates

Citrix enhances HDX 3D Pro hardware encoding by adjusting average bit rates based on bandwidth estimates.

When HDX 3D Pro hardware encoding is in use, the VDA can intermittently estimate the bandwidth of the network and adjust the bit rates of encoded frames accordingly. This new feature provides a mechanism to balance between sharpness and fluency.

This feature is enabled by default. To disable it, run the following command:

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\SYSTEM\
    CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "
    DisableReconfigureEncoder" -d "0x00000001" --force
```

In addition to using this feature, you can also run the following commands to adjust between sharpness and fluency. The **AverageBitRatePercent** and **MaxBitRatePercent** parameters set the percentage of bandwidth usage. The higher values you set, the sharper graphics and lower fluency you get. The recommended setting range is 50–100.

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\SYSTEM\
    CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "
    AverageBitRatePercent" -d "90" --force

sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\SYSTEM\
    CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "
    MaxBitRatePercent" -d "100" --force
```

In the average bit rate adjustment, when your screen holds still, the most recent frame stays in a low-quality state because no new frames are sent. Sharpening support can address this issue by reconfiguring and immediately sending the most recent frame at the highest quality.

For a full list of the policies supported by the Linux VDA Thinwire, see Policy support list.

For information on the configuration of multi-monitor support on the Linux VDA, see CTX220128.

Parallel processing

Thinwire can improve the number of Frames Per Second (FPS) by parallelizing certain tasks, with the overhead of slightly higher overall CPU consumption. This feature is disabled by default. To enable the feature, run the following command on your VDA:

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Troubleshooting

Check which graphics mode is in use

Run the following command to check which graphics mode is in use (**0** means TW+. **1** means full-screen video codec):

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

The result resembles:

```
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"
-v "GraphicsMode"-d "0x00000000"--force
```

Verify that AV1 is in use

> **Note:**
>
> To verify which video codec is in use for the current session, either run a command provided below or check the graphics status through the System tray.

Run the following command to verify that AV1 is in use (**0** means not in use. **1** means in use):

```
sudo /opt/Citrix/VDA/bin/ctxreg dump | grep AV1
```

For example, the result can resemble:

```
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"
-v "AV1"-d "0x00000000"--force
```

Verify that H.265 is in use

Run the following command to verify that full-screen H.265 is in use (**0** means not in use. **1** means in use):

```
sudo /opt/Citrix/VDA/bin/ctxreg dump | grep H265
```

For example, the result can resemble:

```
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"
-v "H265"-d "0x00000000"--force
```

Verify that H.264 is in use

Run the following command to verify that H.264 is in use (**0** means not in use. **1** means in use):

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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

For example, the result can resemble:

```
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"
-v "H264"-d "0x00000000"--force
```

Check which YUV encoding scheme is in use

Run the following command to check which YUV encoding scheme is in use (**0** means YUV420. **1** means YUV422. **2** means YUV444):

> **Note:**
>
> The value of **YUVFormat** is meaningful only when a video codec is in use.

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

For example, the result can resemble:

```
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"
-v "YUVFormat"-d "0x00000000"--force
```

Verify that YUV444 software encoding is in use

Run the following command to verify that the YUV444 software encoding is in use:

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

When YUV444 is in use, the result resembles:

```
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"
-v "GraphicsMode"-d "0x00000001"--force

create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"
-v "H264"-d "0x00000001"--force

create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"
-v "HardwareEncoding"-d "0x00000000"--force

create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"
-v "YUVFormat"-d "0x00000002"--force
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Verify that HDX 3D Pro is enabled

Run the following commands to verify that HDX 3D Pro is enabled:

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

sudo /opt/Citrix/VDA/bin/ctxreg dump | grep StackSessionMode

sudo /opt/Citrix/VDA/bin/ctxreg dump | grep 3DPro
```

When HDX 3D Pro is enabled, the result resembles:

```
create -k "HKLM\Software\Citrix\VirtualDesktopAgent\State"-t "REG_SZ"
-v "ProductEdition"-d "<PLT or ENT>"--force

create -k "HKLM\System\CurrentControlSet\Control\Citrix\WinStations\
tcp"-t "REG_DWORD"-v "StackSessionMode"-d "0x00000000"--force

create -k "HKLM\System\CurrentControlSet\Control\Citrix"-t "REG_DWORD
"-v "3DPro"-d "0x00000000"--force
```

To verify that the required NVIDIA libraries are loaded for HDX 3D Pro, run the **nvidia-smi** command on the Linux VDA. The result resembles:

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

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

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


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

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

|    0      2164  C+G   /usr/local/bin/ctxgfx
   106MiB |
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
|    0      2187    G   Xorg
     85MiB |
+-----------------------------------------------------------------------------+
```

Verify that the 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
```

When 3D Pro is in use, the result resembles:

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

Verify that the NVIDIA GRID graphics driver is installed correctly

To verify that the NVIDIA GRID graphics driver is installed correctly, run **nvidia-smi**. The result resembles:

```
+-----------------------------------------------------+
| NVIDIA-SMI 352.70     Driver Version: 352.70        |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile
   Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util
   Compute M. |
|===============================+======================+======================|
|   0  Tesla M60           Off  | 0000:00:05.0     Off |
                    Off |
| N/A   20C    P0    37W / 150W |     19MiB /  8191MiB |      0%
   Default |
+-------------------------------+----------------------+----------------------+

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
+-------------------------------------------------------------------------------+
```

Set the correct configuration for the card:

`etc/X11/ctx-nvidia.sh`

HDX 3D Pro multi-monitor redraw issues

If you are seeing redraw issues on screens other than the primary monitor, check that the NVIDIA GRID license is available.

Check Xorg error logs

The log file of Xorg is named similar to **Xorg.{DISPLAY}.log** in the **/var/log/** folder.

Known issues and limitations

For vGPU, the XenServer (formerly Citrix Hypervisor) local console shows the ICA desktop session screen

Workaround: Disable the VM's local VGA console by running the following command:

```
[root@xenserver ~]# xe vgpu-param-set uuid=vgpu-uuid extra_args=
   disable_vnc=1
```

Gnome 3 desktop popups slow when logging on

It is a limitation of Gnome 3 desktop session startup.

Some OpenGL/WebGL applications do not render well upon resizing the Citrix Workspace app window

Resizing the window of Citrix Workspace app changes the screen resolution. The NVIDIA proprietary driver changes some internal states and might require applications to respond accordingly. For example, the WebGL library element **lightgl.js** might spawn an error saying that **Rendering to this texture is not supported (incomplete frame buffer)**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

HDX screen sharing

June 20, 2024

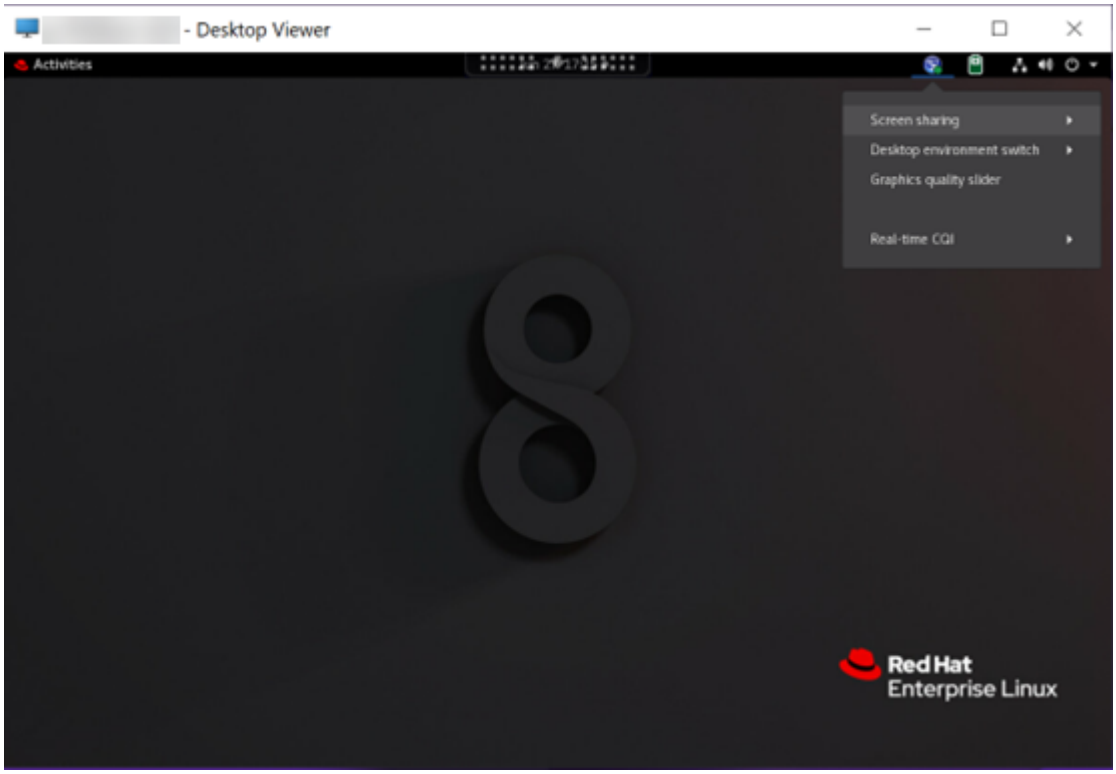
Overview

The Linux VDA lets you share the screen of your virtual desktop with session users on other virtual desktops.

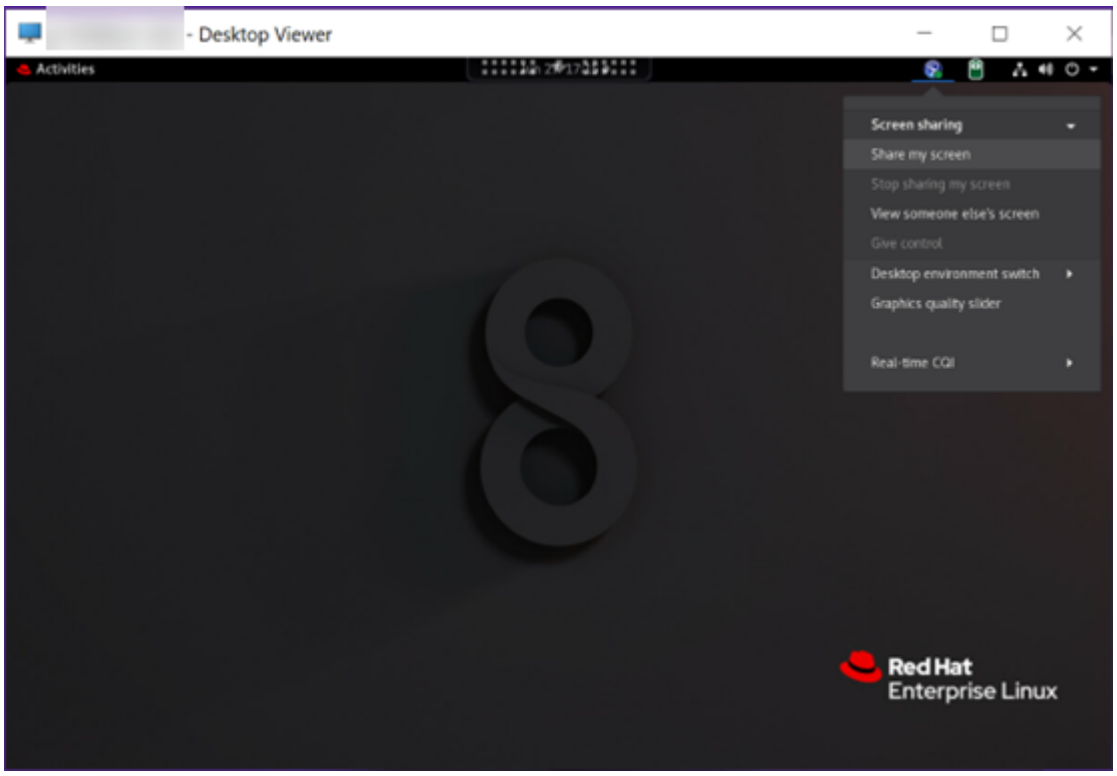
The following example walks you through the procedure of sharing a screen and viewing someone else's screen.

To share a screen:

1. In the notification area of your virtual desktop, click the following system tray icon and select **Screen sharing > Share my screen**.

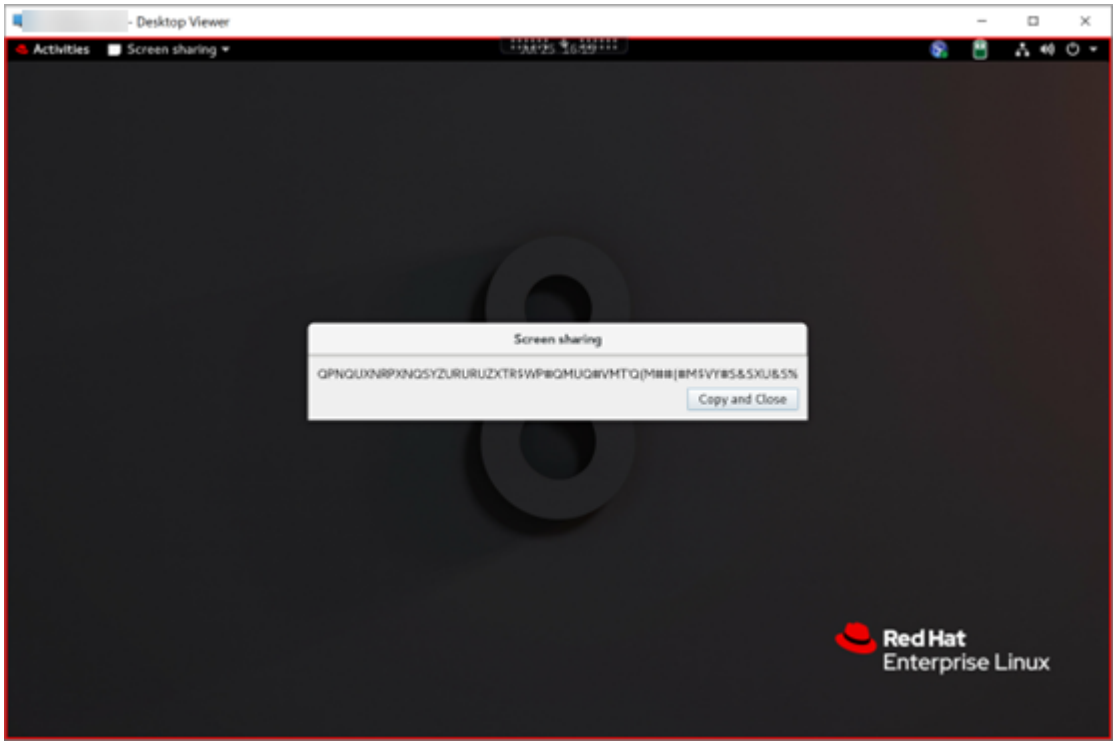


© 1999–2024 Cloud Software Group, Inc. All rights reserved.



2. Click **Copy and Close**.

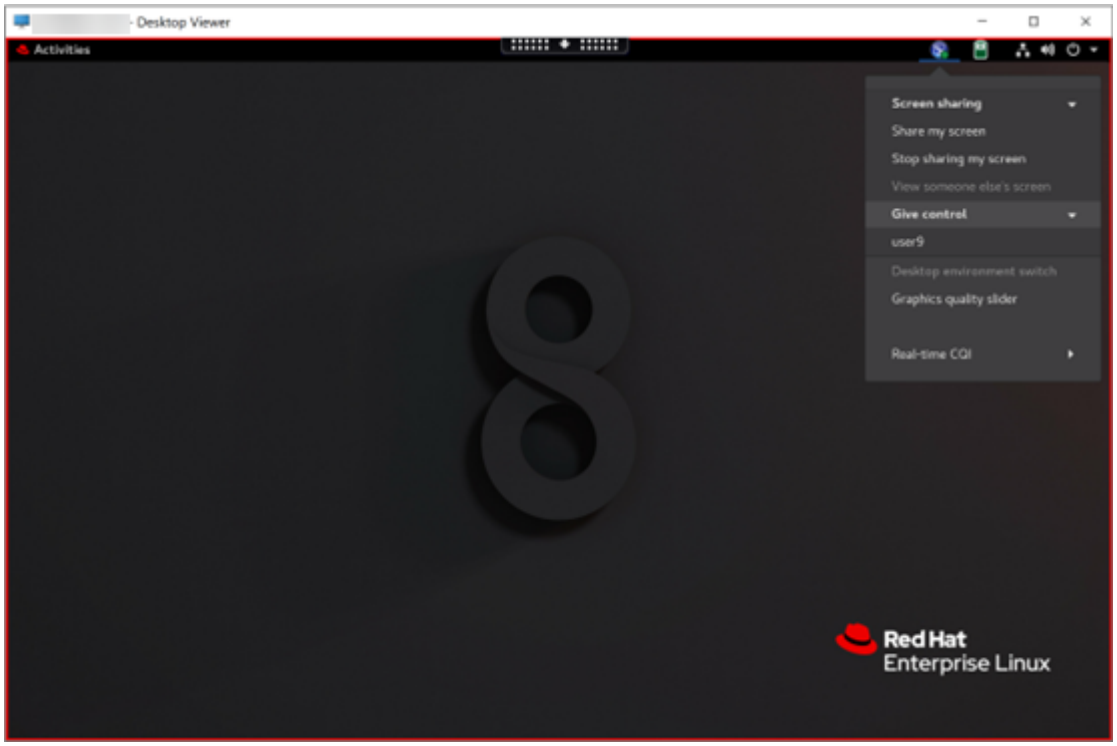
   The current screen sharing code persists until you stop and restart sharing your screen.



© 1999–2024 Cloud Software Group, Inc. All rights reserved.

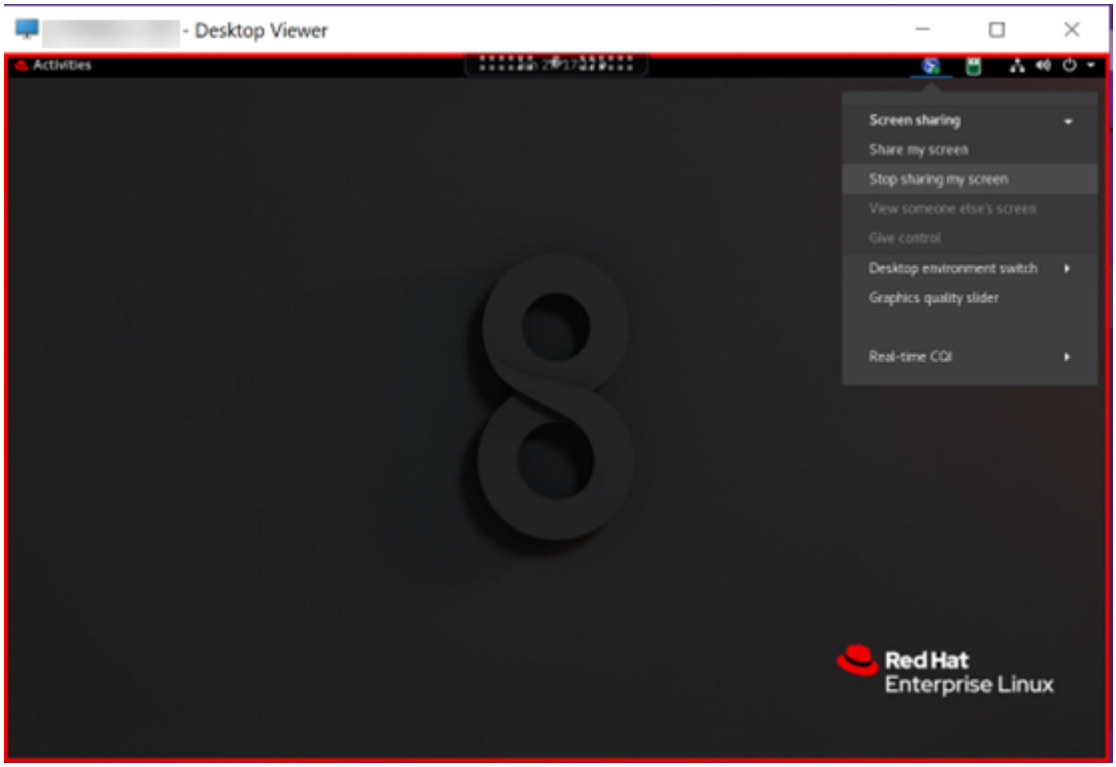
> **Tip:**
>
> While you are sharing your screen, there is a red border around it, indicating that sharing is in progress.

3. Share the copied code with session users on other virtual desktops that you want to share your screen with.
4. To let a viewer control your screen, select **Give control** and then the viewer's name. To stop giving control, clear the viewer's name.



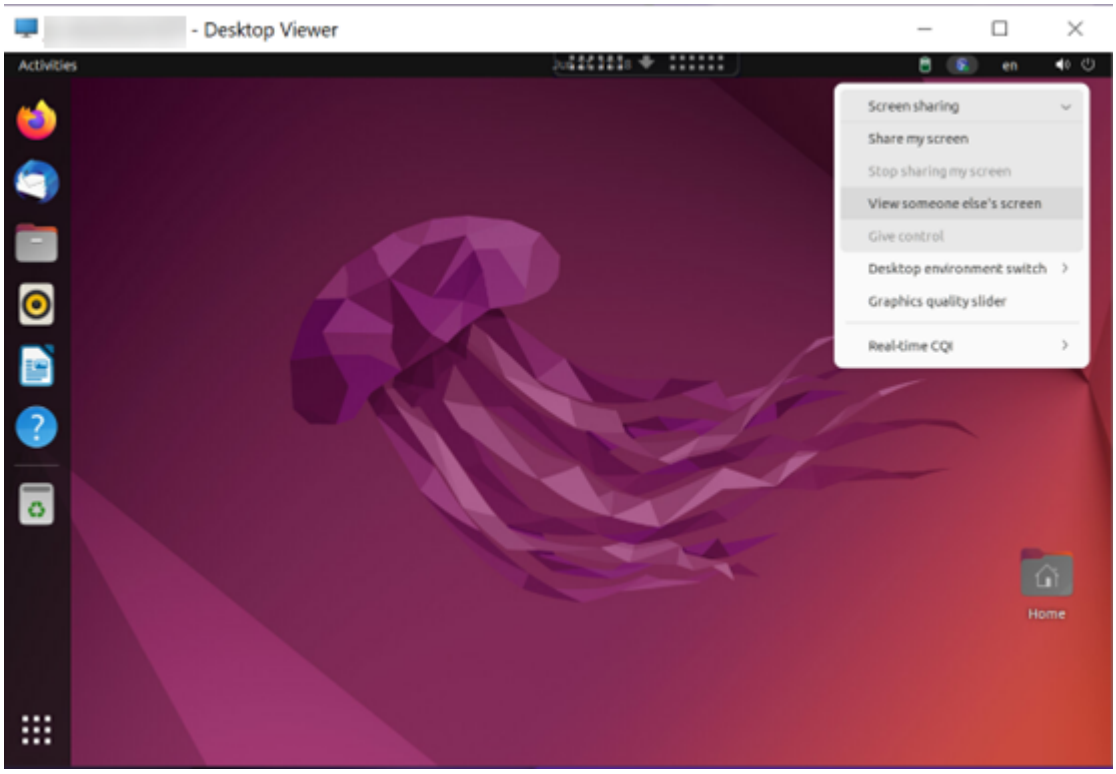
5. To stop sharing your screen, select **Stop sharing my screen**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



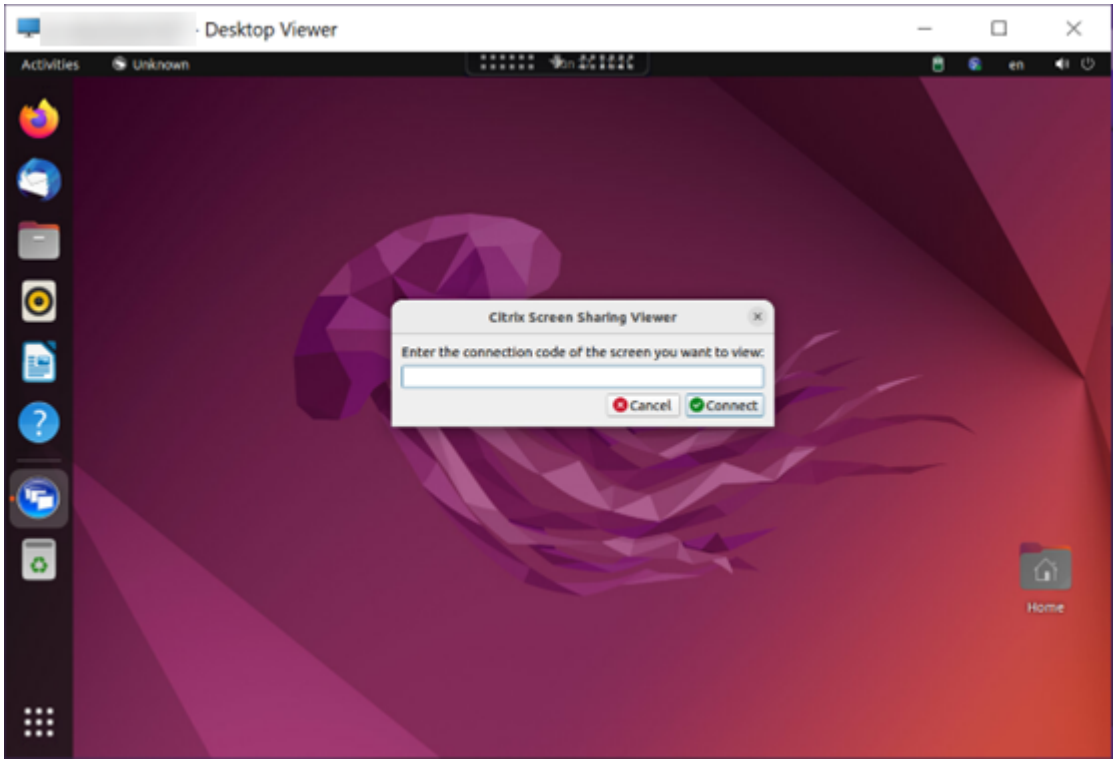
To view someone else's screen:

1. In the notification area of your virtual desktop, click the **screen sharing** icon and select **View someone else's screen**.

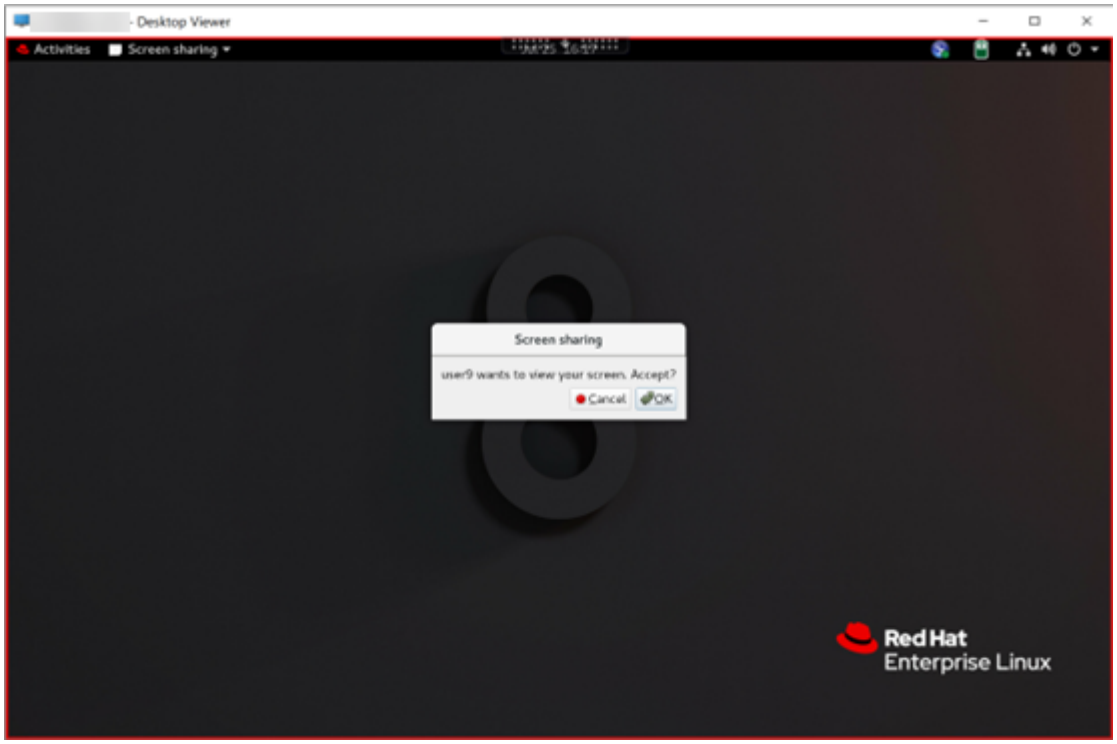


© 1999–2024 Cloud Software Group, Inc. All rights reserved.

2. Enter the connection code of the screen that you want to view and then click **Connect**.



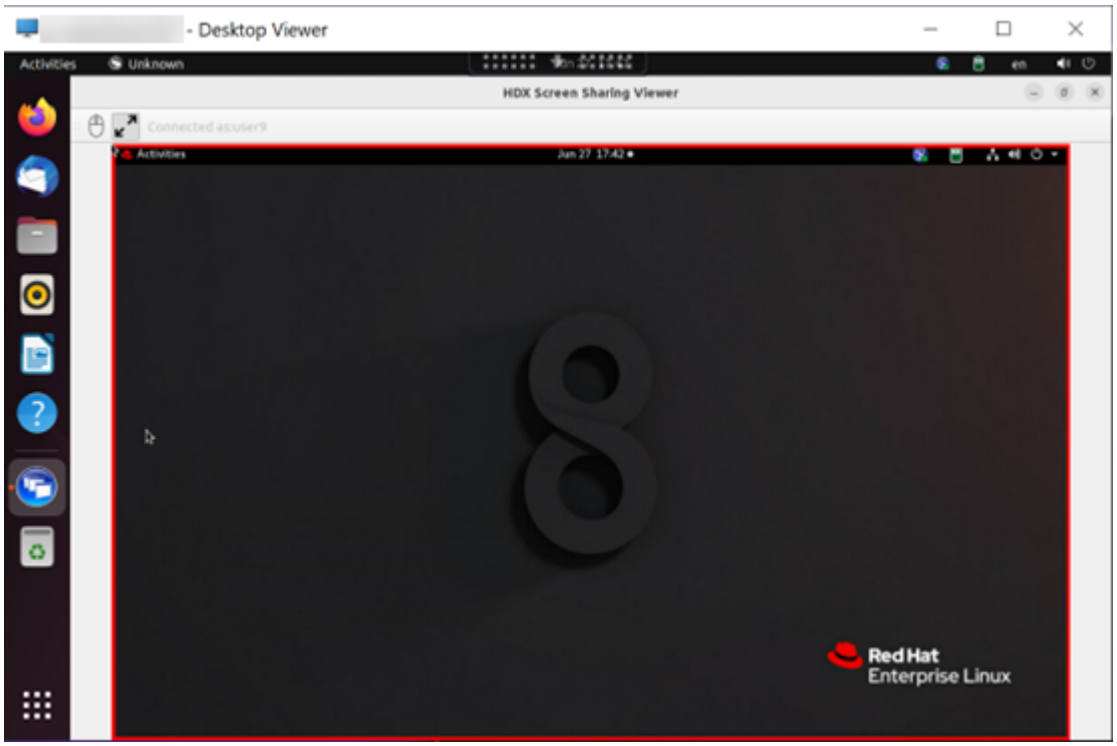
3. Wait for the screen sharer to accept your request. For example:



© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Tip:**
>
> - On the sharer side, the Linux system issues a notification of your request.
> - If the sharer does not accept your request within 30 seconds, your request expires and a prompt appears.

4. After the screen sharer accepts your request by clicking **OK**, the shared screen appears in your Desktop Viewer. You are connected as a viewer with an automatically assigned user name.

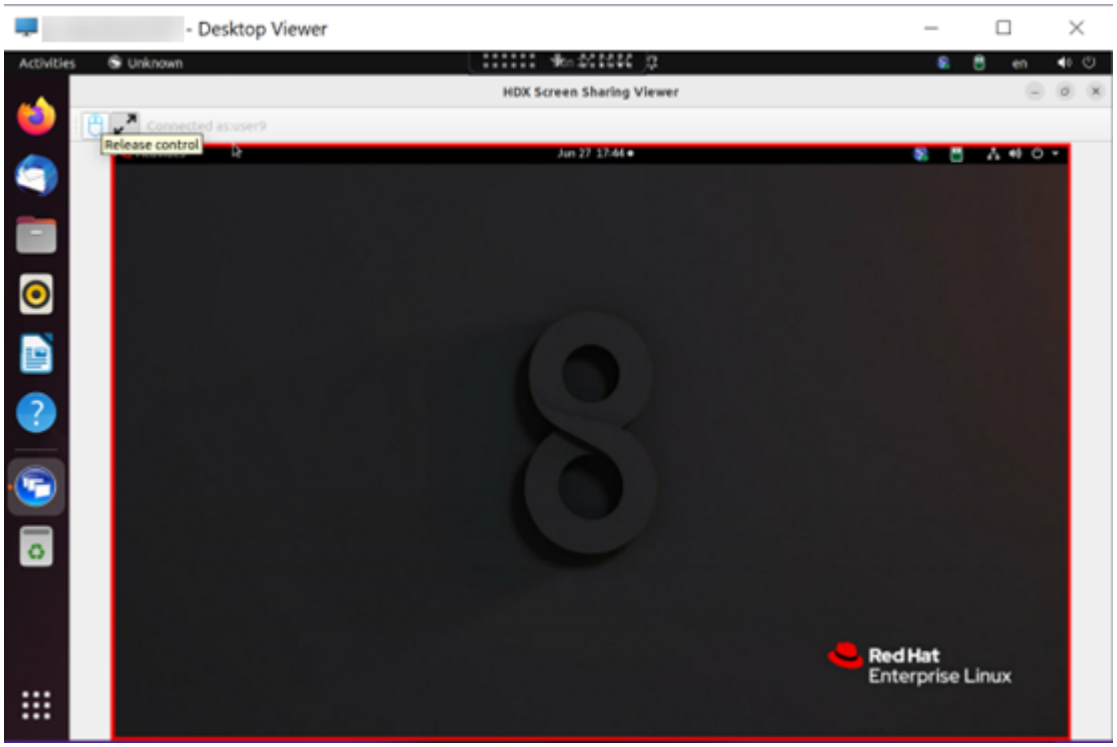


5. To request control over the shared screen, click the mouse icon in the upper left corner.

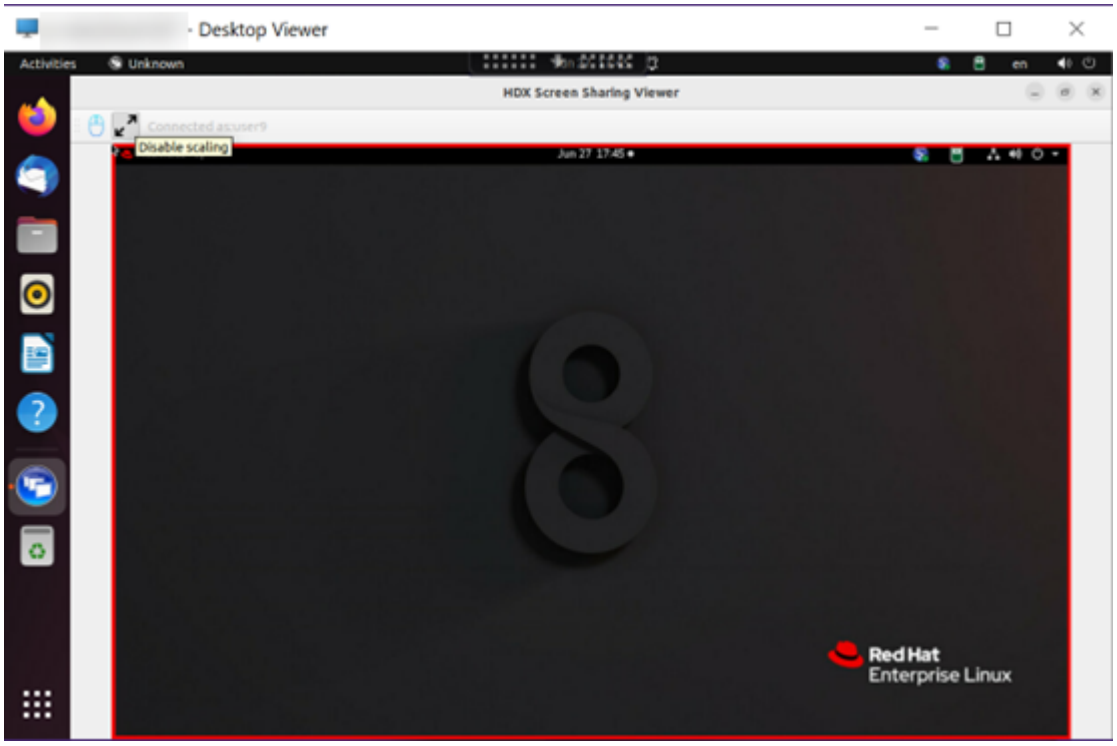
> **Tip:**
>
> - If the sharer does not accept your request within 30 seconds, your request expires.
> - Only one viewer is allowed to control a shared screen at a time.

Click the mouse icon again to release control over the shared screen.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



6. To disable display scaling or scale to the window size, click the icon next to the mouse icon.



© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Configuration

The screen sharing feature is disabled by default. To enable it, complete the following settings:

1. Enable the system tray.
2. For Citrix Virtual Apps and Desktops 2112 and later, enable the **ScreenSharing** policy in Citrix Studio.
3. (Optional) For Citrix Virtual Apps and Desktops 2109 and earlier, enable screen sharing on the Linux VDA by running the following command:

```
sudo /opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\
    CurrentControlSet\Control\Citrix\Thinwire" -v "
    EnableScreenSharing" -d "0x00000001"
```

4. Allow ports 52525–52625 in your firewall.

Considerations

- The screen sharing feature does not support the H.265 video codec.
- The screen sharing feature is not available for app sessions.
- Users of desktop sessions can share their session screens with up to 10 viewers by default. The maximum number of viewers is configurable through `ctxreg update -k "HKLM\System\CurrentControlSet\Control\Citrix\Thinwire"-v "ScreenSharingViewerMaxNum"-d <hex_value>`. When the maximum number is reached, a prompt appears when users try to accept extra connection requests.

Multi-monitor support

June 20, 2024

Overview

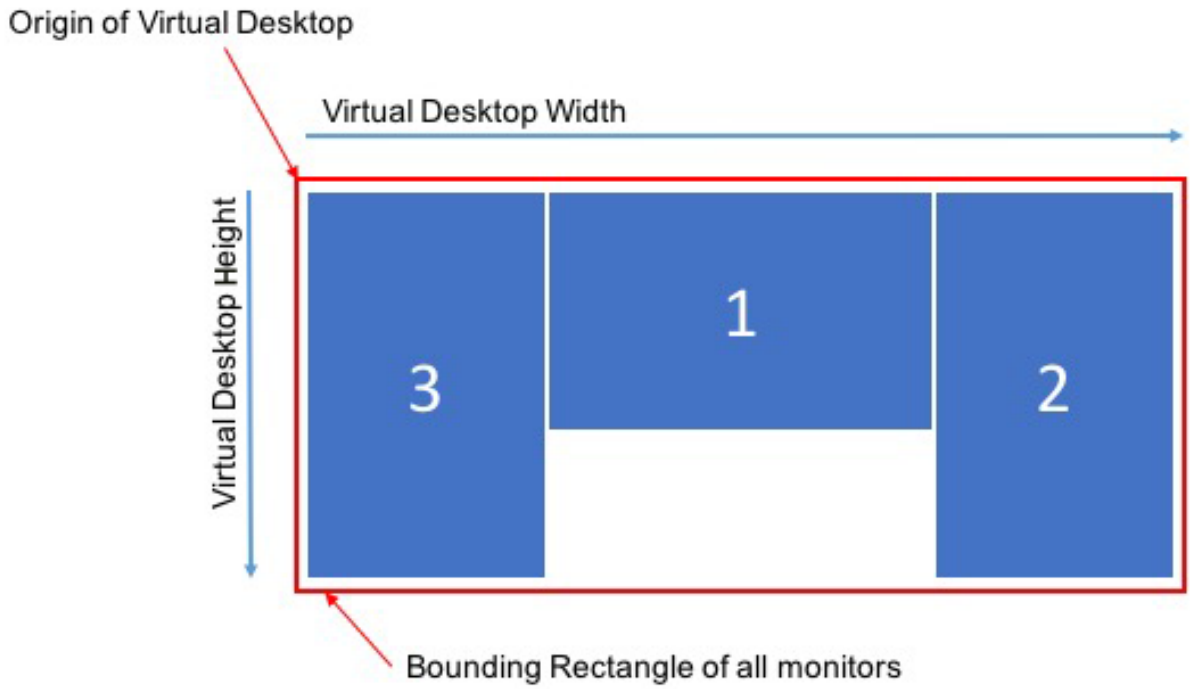
The Linux VDA provides the out-of-the-box multi-monitor support with a default resolution of 2560×1600 per monitor. Standard VDAs support up to nine monitors, and HDX 3D Pro VDAs support up to four monitors.

This article describes how to configure the Linux VDA for different monitor resolutions and layouts.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Virtual session desktop

Like the Windows VDA, the Linux VDA has the concept of a multi-monitor virtual desktop. It is based on the bounding rectangle of all monitors, not the actual layout of the monitors. Thus, the area of the virtual desktop can theoretically be larger than the area covered by the monitors of the client.



Virtual session desktop size

The origin of the virtual session desktop is calculated from the top-left corner of the bounding rectangle of all monitors. That point locates at X = 0, Y = 0, where X and Y are the horizontal and vertical axes, respectively.

The width of the virtual session desktop is the horizontal distance, in pixels, from the origin to the top-right corner of the bounding rectangle of all monitors.

Similarly, the height of the virtual session desktop is the vertical distance, in pixels, from the origin to the bottom-left corner of the bounding rectangle of all monitors.

This calculation is important for the following reasons:

- Allowing for different client monitor layouts
- Understanding memory usage on the Linux VDA

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

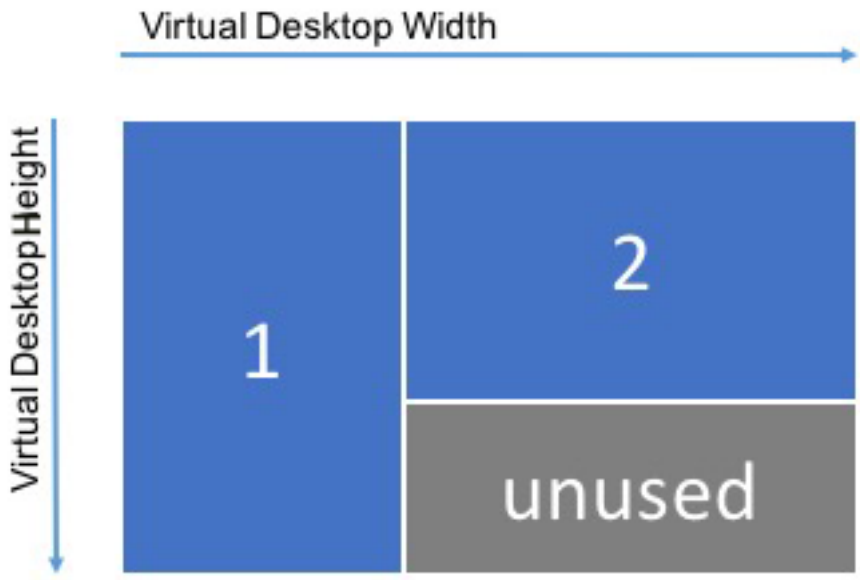
Allowing for different client monitor configurations

Knowing the maximum virtual desktop size for your various client monitor configurations allows you to configure the Linux VDA to be flexible in terms of client monitor configurations.

Consider the following client monitor configuration:

The diagram above shows an out-of-the-box multi-monitor configuration with two monitors, each with a resolution of 2560×1600.

Now, consider connecting to the same Linux VDA with the following client monitor configuration:



If each monitor in the above diagram has a resolution of 2560×1600, the out-of-the-box multi-monitor configuration parameters are insufficient. The maximum height is too small to accommodate the virtual session desktop for this monitor layout. To accommodate the client monitor configuration in this example, you must set the Linux VDA virtual desktop to a size of 4160×2560.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

For the greatest flexibility in a multi-monitor configuration, find the smallest bounding rectangle of all monitor layouts you want to support. For configurations with two 2560×1600 monitors, the possible layouts include:

- **Monitor1** 2560×1600 and **Monitor2** 2560×1600
- **Monitor1** 1600×2560 and **Monitor2** 2560×1600
- **Monitor1** 2560×1600 and **Monitor2** 1600×2560
- **Monitor1** 1600×2560 and **Monitor2** 1600×2560

To accommodate all the layouts above, you need a virtual session desktop of 5120×2560. It is the smallest bounding rectangle that can contain all the desired layouts.

If all your users have only one monitor in the typical landscape layout, set the maximum virtual desktop size to the highest resolution of the monitor.

In this example, the virtual desktop needs to be set to a size of 2560×1600. Because the default configuration is 5120×1600 and 2 monitors, a configuration change is required to optimize memory usage for single-monitor deployments.

> **Note:**
>
> If a desktop displays at an improper resolution in a multi-monitor setup, adjust Dots Per Inch (DPI) settings on the Citrix Workspace app. For more information, see Knowledge Center article CTX230017.

Understanding memory usage on the Linux VDA

Knowing the virtual desktop size allows you to calculate the amount of memory used by each HDX session. This memory is the memory allocated to each session for its graphics data when the session begins. It does not change for the life of the session. While this memory is not the total amount of memory used for the session, it is the easiest way of calculating per-session memory usage.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

To calculate how much memory is allocated to each HDX session, use the following formula:

M = X × Y × Z,

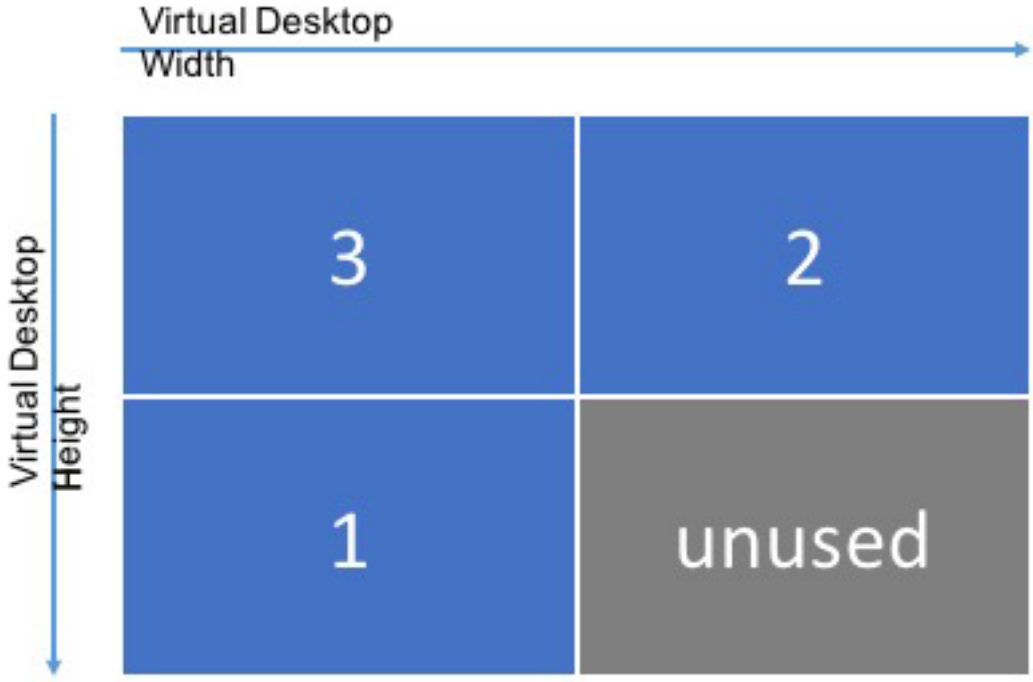
Where:

- **M** is the amount of memory used for session graphics.
- **X** is the width of the virtual session desktop.
- **Y** is the height of the virtual session desktop.
- **Z** is the color depth of the HDX session window. The value is in bytes, not bits, so use 4 for 32-bit color.

> **NOTE:**
>
> The color depth of the X server starts and cannot change with the life of the session (**from login through disconnects/reconnects until logoff**). Hence, the Linux VDA always allocates the virtual session desktop as 32-bit and down samples to the color depth requested for the session.
>
> For example, for a 1024×768 session, the memory used is:
>
> 1024 × 768 × 4 / 2^20 MB = 3 MB

Understanding memory usage is important for increasing session density on each Linux VDA.

Consider the following client monitor configuration:



If each monitor has a resolution of 2560×1600, to accommodate this client monitor configuration, the virtual session desktop size needs to be 5120×3200. Notice that the gray area is unused and equates to 16,384,000 (that is, 2560 x 1600 x 4) bytes of wasted memory.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Citrix multi-monitor configuration parameters

You can control the multi-monitor functionality of the Linux VDA by using the following configuration parameters:

- **MaxScreenNum**

  Parameter: HKEY_LOCAL_MACHINE/System/CurrentControlSet/Control/Citrix/Thinwire/MaxScreenNum

  Description: Number of monitors to support

  Type: DWORD

  Default: 4

  Maximum: 9 for standard VDA, 4 for HDX 3D Pro VDA

- **MaxFbWidth**

  Parameter: HKEY_LOCAL_MACHINE /System/CurrentControlSet/Control/Citrix/Thinwire/-MaxFbWidth

  Description: Maximum width of a virtual session desktop

  Type: DWORD

  Default: 5,120

  Maximum: 16,384 (8,192 x 2)

- **MaxFbHeight**

  Parameter: HKEY_LOCAL_MACHINE /System/CurrentControlSet/Control/Citrix/Thinwire/-MaxFbHeight

  Description: Maximum height of a virtual session desktop

  Type: DWORD

  Default: 1,600

  Maximum: 16,384 (8,192 x 2)

Changing the Linux VDA multi-monitor configuration

The following section outlines how to enable, configure, and disable the multi-monitor functionality on the Linux VDA.

Set the maximum number of monitors by using:

```
1 sudo ctxreg create -k " HKEY_LOCAL_MACHINE \System\CurrentControlSet\
     Control\Citrix\Thinwire" -t "REG_DWORD" -v "MaxScreenNum" -d "
     NumMons" --force
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Where **NumMons** is a value between 1 and 9 for standard VDA or 1 and 4 for HDX 3D Pro VDA.

Set the maximum width of a virtual session desktop by using:

```
sudo ctxreg create -k " HKEY_LOCAL_MACHINE \System\CurrentControlSet\
   Control\Citrix\Thinwire" -t "REG_DWORD" -v "MaxFbWidth" -d "
   MaxWidth" --force
```

Where **MaxWidth** is a value between **1,024** and **16,384**.

Set the maximum height of a virtual session desktop by using:

```
sudo ctxreg create -k " HKEY_LOCAL_MACHINE \System\CurrentControlSet\
   Control\Citrix\Thinwire" -t "REG_DWORD" -v "MaxFbHeight" -d "
   MaxHeight" --force
```

Where **MaxHeight** is a value between **1,024** and **16,384**.

Non-virtualized GPUs

June 20, 2024

In the Linux VDA documentation, non-virtualized GPUs refer to:

- GPUs used in Remote PC Access scenarios
- GPUs passed through from a hypervisor

This article provides information on supporting non-virtualized GPUs.

Enable HDX 3D Pro for NVIDIA GPUs that support the NVIDIA Capture SDK for Linux

For NVIDIA GPUs that support the NVIDIA Capture SDK for Linux, enable HDX 3D Pro simply by setting **CTX_XDL_HDX_3D_PRO** to **Y** when installing the Linux VDA. No additional configuration is required. Hardware acceleration is enabled by default after you enable HDX 3D Pro.

Compatible with NVIDIA GPUs that don't support the NVIDIA Capture SDK for Linux and GPUs from other manufacturers such as AMD and Intel

> **Note:**
>
> In this scenario, only software encoding is supported.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 1: Set CTX_XDL_HDX_3D_PRO to Y when installing the Linux VDA

For information about environment variables, see Step 8: Set up the runtime environment to complete the installation.

Step 2: Install Xdamage

For example, you can run **sudo apt-get install -y libxdamage1** to install XDamage on Ubuntu 20.04. Typically, XDamage exists as an extension of XServer.

Step 3: Enable XDamage by running the following command

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\System\
   CurrentControlSet\Control\Citrix\XDamage" -t "REG_DWORD" -v "
   XDamageEnabled" -d "0x00000001" --force
```

Step 4: Modify Xorg configuration files

You can find the following four template configuration files under **/etc/X11**. Based on the number of connected monitors, modify one of the template configuration files with the corresponding number in its name. For example, if only one monitor is connected, modify the template configuration file with the number 1 in its name, that is, ctx-driver_name-1.conf. If two monitors are connected, modify the template configuration file with the number 2 in its name, that is, ctx-driver_name-2.conf.

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

Using **ctx-driver_name-1.conf** as an example, do the following to modify the template configuration files:

1. Replace ***driver_name*** with your actual driver name.

   For example, if your driver name is `intel`, you can change the configuration file name to `ctx-intel-1.conf`.

2. Add the video driver information.

   Each template configuration file contains a section named "Device,"which is commented out. This section describes the video driver information. Enable this section before adding your video driver information. To enable this section:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

a) See the guide provided by the GPU manufacturer for configuration information. A native configuration file can be generated. Verify that your GPU can work in a local environment with the native configuration file.

b) Copy the "Device"section of the native configuration file to **ctx-driver_name-1.conf**.

3. Run the following command to set the registry key so that the Linux VDA can recognize the configuration file name set in Step 1.

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

Monitor blanking for Remote PC Access VDAs

The Linux VDA supports physical monitor blanking for Remote PC Access VDAs that use non-virtualized GPUs.

Fully tested Linux distributions that support the feature include Ubuntu 20.04 and Debian 11.

The feature is disabled by default. To enable it, complete the following two steps:

1. Install the `evdi-dkms` package based on your Linux distribution:

```
sudo apt install evdi-dkms
```

2. Enable graphics display offloading to EVDI:

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

3. If you are using an Intel GPU, disable the display manager. Otherwise, the display manager occupies and makes the Intel GPU unavailable for Citrix remote sessions.

```
sudo systemctl disable --now gdm
```

Troubleshooting

No or garbled graphic output

If you can run 3D applications locally and all configurations are correct, then missing or garbled graphic output is the result of a bug. Use /opt/Citrix/VDA/bin/setlog and set GFX_X11 to verbose to collect the trace information for debugging.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Session watermark

June 20, 2024

Session watermark helps to deter and enable tracking of data theft. Traceable information appears on session desktops as a deterrent to users who employ photographs and screen captures to steal data. You can specify a watermark as a layer of text or a PNG image with alpha channel. The watermark displays over the entire session screen without changing the content of the original document.

> **Important:**
>
> Session watermark is not a security feature. It does not prevent data theft completely, but it provides some level of deterrent and traceability. We do not guarantee complete information traceability when using this feature. Instead, we recommend that you combine this feature with other security solutions as applicable.

The session watermark carries information for tracking data theft. The most important data is the identity of the user, as tracked by their logon credentials, of the session where the screen image was taken. To trace data leakage more effectively, include other information such as the server or client Internet protocol address and a connect time.

To adjust the user experience, use the following session watermark policy settings to configure the placement and watermark appearance on the screen:

Session watermark policy settings

Enable session watermark

When you enable this setting, the session display has an opaque watermark displaying session-specific information. The other watermark settings depend on this one being enabled.

By default, the session watermark is disabled.

Include client IP address

When you enable this setting, the session displays the current client IP address as a watermark.

By default, **Include client IP address** is disabled.

Include connection time

When you enable this setting, the session watermark displays a connect time. The format is yyyy/mm/dd hh:mm. The time displayed is based on the system clock and time zone.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

By default, **Include connection time** is disabled.

Include logon user name

When you enable this setting, the session displays the current logon user name as a watermark. The display format is USERNAME@DOMAINNAME. We recommend that the user name is a maximum of 20 characters. When a user name is longer than 20 characters, smaller font sizes or truncation might occur, which lessens the effectiveness of the watermark.

By default, **Include logon user name** is enabled.

Include VDA host name

When you enable this setting, the session displays the VDA host name of the current ICA session as a watermark.

By default, **Include VDA host name** is enabled.

Include VDA IP address

When you enable this setting, the session displays the VDA IP address of the current ICA session as a watermark.

By default, **Include VDA IP address** is disabled.

Session watermark style

This setting controls whether you display a single watermark text label or multiple labels. Choose **Multiple** or **Single** from the **Value** drop-down menu.
For additional style options, see the **Watermark custom text** section in this article.

Multiple displays five watermark labels in the session. One in the center and four in the corners.

Single displays a single watermark label in the center of the session.

By default, the **Session watermark style** is **Multiple**.

Watermark transparency

You can specify watermark opacity from 0 through 100. The larger the value specified, the more opaque the watermark.

By default, the value is 17.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Watermark custom text

The value is empty by default. You can type a non-empty string, set a syntax to form a string, or use the combination to display in the session watermark. Non-empty strings support up to 25 Unicode characters per line. Longer strings are truncated to 25 characters.

For example, you can set the policy to the following value:

```
<date> <time><newline><username><style=single><fontsize=40><font=
Ubuntu><position=center><rotation=0><newline><serverip><newline><
clientip><newline>Citrix Linux VDA<newline>Version 2207
```

For a description of all syntax options, see the following table:

Syntax option	Description	Valid setting (case-sensitive)	Default value	Remarks
<style>	Watermark layout style	xstyle, single, tile, horizontal	xstyle	-
<position>	Watermark position	center, topleft, topright, bottomleft, bottomright	center	Valid only when the layout style is set to **single**.
<rotation>	Watermark rotation to a certain angle	-180–180	0	-
<transparency>	Watermark opacity	0–100	17	-
<font>	-	A system supported font	Sans	-
<fontsize>	-	20–50	0 (auto calculated)	-
<fontzoom>	Percentage of the font and image sizes you set through <fontsize> and <image>	0 –	100	-

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Syntax option	Description	Valid setting (case-sensitive)	Default value	Remarks
<image>	PNG watermark	Path to a PNG image on the VDA	N/A	This syntax configures a PNG watermark. Only PNG with an alpha channel is supported. With a PNG watermark in use, only the <style>, <position>, <rotation>, <transparency>, and <fontzoom> syntax options can be effective.
<date>	Placeholder for the session connection date (YYYY/MM/DD)	N/A	N/A	-
<time>	Placeholder for the session connection time (HH:MM)	N/A	N/A	-
<domain>	Placeholder for the user account domain	N/A	N/A	-
<username>	Placeholder for the current logon user name (excluding the user account domain)	N/A	N/A	-
<hostname>	Placeholder for the host name of the VDA	N/A	N/A	-

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Syntax option	Description	Valid setting (case-sensitive)	Default value	Remarks
<clientip>	Placeholder for the IP address of the client	N/A	N/A	-
<serverip>	Placeholder for the IP address of the VDA	N/A	N/A	-

> **Note:**
>
> If **Watermark custom text** is specified with at least one valid syntax option, all other session watermark policies - except **Enable session watermark** - are ignored.
>
> If you leave a syntax option unspecified or set it to an unsupported value, their default value is used.

Limitations

- Session watermark is supported in either of the following cases:
  - When **Use video codec for compression** is set to **For the entire screen**.
  - When **Use video codec for compression** is set to **Use when preferred** and Optimize for 3D graphics workload is enabled.
- Session watermark is not supported in sessions where browser content redirection is used. To use the session watermark feature, ensure that browser content redirection is disabled.
- Session watermark is not supported and does not appear if the session is running in full-screen hardware accelerated H.264 or H.265 encoding mode with legacy NVIDIA drivers. (In this case, NvCaptureType is set to 2 in the registry.)
- Watermark is not visible for session shadowing.
- If you press the Print Screen key to capture a screen, the screen captured at the VDA side does not include the watermark. We recommend that you take measures to avoid screen captures being copied.

Shared GPU acceleration on a multi-session Linux VDA

June 20, 2024

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

HDX 3D PRO supports only the Linux VDAs that are configured for VDI desktops (single-session mode). For a multi-session Linux VDA, you can enable shared GPU acceleration on it to accelerate OpenGL 3D applications.

> **Note:**
>
> Wayland display server is not supported for shared GPU acceleration.

Configuration

To enable shared GPU acceleration on a multi-session Linux VDA to accelerate OpenGL 3D applications, complete the configuration steps:

Step 1: Install VirtualGL

Download and install **VirtualGL** from https://sourceforge.net/projects/virtualgl/files. Download **.deb** packages for Debian-based Linux distributions and **.rpm** packages for RHEL-based Linux distributions.

Step 2: Configure VirtualGL

1. Stop the Linux display manager, for example, LightDM or GNOME Display Manager (GDM).
2. Execute the VirtualGL configuration script by running:

   ```
   #/opt/VirtualGL/bin/vglserver_config
   ```

   We recommend you make the following selections during the script execution:

   - Select "1"to "Configure server for use with VirtualGL (GLX + EGL back ends)"
   - Select "n"to "Restrict 3D X server access to **vglusers** group"
   - Select "n"to "Restrict framebuffer devices access to **vglusers** group"
   - Select "n"to "Disable XTEST extension"
3. Exit the configuration script and restart the Linux display manager.

Step 3: Run OpenGL 3D applications with GPU acceleration

There are two methods to run OpenGL 3D applications with GPU acceleration in a Linux VDA session:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- **Method 1:** Enable shared GPU acceleration for all OpenGL 3D applications

  To do so, open a bash terminal on the Linux VDA, run the following command, and then restart the bash terminal. This approach enables shared GPU acceleration for all OpenGL 3D applications launched from the bash terminal.

  ```
  #/opt/Citrix/VDA/sbin/ctxgpushare.sh enable
  ```

- **Method 2:** Enable shared GPU acceleration for a specific OpenGL 3D application:

  To do so, open a terminal on the Linux VDA and run the following command with the name of the application specified:

  ```
  #vglrun <AppName>
  ```

Limitations

- Shared GPU acceleration works closely with the display manager on the Linux VDA. It is recommended to use LightDM as the display manager for shared GPU acceleration to achieve the expected functionality and performance.
- WebGL hardware acceleration is supported in Firefox on Ubuntu and Debian only.

Scalability

The maximum-supported number of concurrent sessions that can share a GPU depends on the CPU and system memory. It also highly depends on the maximum video memory of the GPU.

For example:

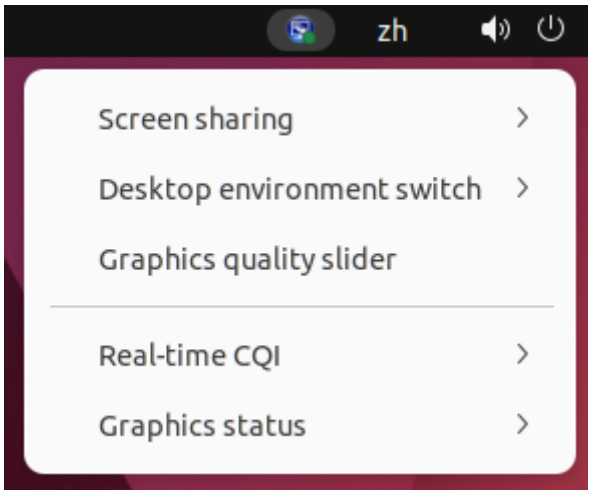
If	Then
the NVIDIA **M10-2B** vGPU has 2,048 MB of video memory and an OpenGL application such as VariCAD viewer uses 100 MB of video memory for its workload in each session,	in theory the maximum-supported number of concurrent sessions cannot exceed 20.

System tray

June 20, 2024

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Session users can click the following system tray icon to perform the actions or view the indicators:



Introduction to items in the system tray

Each item corresponds to a feature with a toggle. When the feature corresponding to an item is disabled, the item is hidden and not displayed.

- **Screen sharing**

  For more information about this feature, see HDX screen sharing.

- **Desktop environment switch**

  This item is a GUI for **ctxdesktopswitch.sh**. For more information, see Custom desktop environments by session users.

  Desktop environment customization by session users is enabled by default. To disable it, run the following command:

  ```
  1  /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\
        CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "
        EnableDesktopSwitch" -d "0x00000000" --force
  ```

- **Graphics quality slider**

  For more information, see the Graphics quality slider section in the graphics configuration article.

- **Real-time CQI**

  Currently, the ICA Round Trip Time (RTT) and latency data are displayed. For more information, see the session data query utility.

  The system tray icon displays differently according to the latency in real-time CQI:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

There are thresholds that control when the icon changes display. By default, they are set as follows:

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\
   CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "
   HighLatencyThreshold" -d "0x000000dc" --force
```

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\
   CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "
   LowLatencyThreshold" -d "0x00000078" --force
```

When the actual latency is less than or equal to **LowLatencyThreshold**, the icon is marked green. When the actual latency is greater than **HighLatencyThreshold**, the icon is marked red. In other circumstances, the icon is marked yellow. If real-time CQI is disabled, the icon does not have a color mark.

The real-time CQI is enabled and displayed by default. To disable and hide it and make the tray icon not have a color mark, run the following command:

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\
   CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "
   EnableCqiShow" -d "0x00000000" --force
```

- **Graphics status**

  This indicator shows the graphics settings for the current session. It is enabled by default. To disable it, run the following command:

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\
   CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "
   EnableGfxInfo" -d "0x00000000" --force
```

Enable the system tray

The system tray as well as the client battery status display is enabled by default. Under certain conditions, however, you must do extra configuration to enable the system tray as well as the client battery status display. Details are as follows:

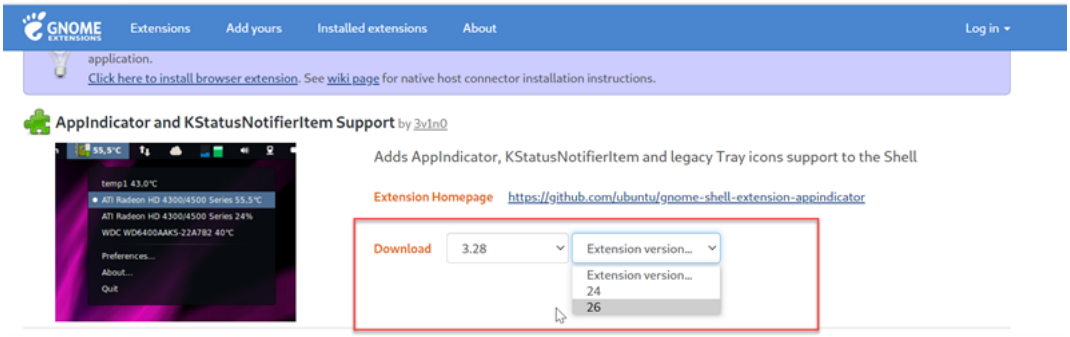
1. Enable the graphics status indicator policy in Citrix Studio.
2. (**This step is required only when you are using RHEL 8.x/9.x, Rocky Linux 8.x/9.x, Debian 11, or SUSE 15.x installed with GNOME.**) Install a compatible extension for your GNOME shell

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

to enable AppIndicator support.

a) Run the `gnome-shell --version` command to check your GNOME shell version.

b) Download a compatible extension for your GNOME shell from https://extensions.gnome.org/extension/615/appindicator-support. For example, if your shell version is 3.28, you can select 24 or 26 for the extension version.



c) Unzip the downloaded package and rename the unzipped directory to **appindicatorsupport@rgcjonas.gmail.com**. Verify that the **"uuid"** value in the **metadata.json** file in the package is set to **appindicatorsupport@rgcjonas.gmail.com**.

d) Run the `mv` command to move the **appindicatorsupport@rgcjonas.gmail.com** directory to the location under `/usr/share/gnome-shell/extensions/`.

e) Run the `chmod a+r metadata.json` command to make the **metadata.json** file readable to other users.

> **Tip:**
>
> By default, the **metadata.json** file in the **appindicatorsupport@rgcjonas.gmail.com** directory is readable only to the root user. To support screen sharing, make the **metadata.json** file readable to other users as well.

f) Install GNOME Tweaks.

g) In the desktop environment, reload your GNOME shell by pressing the `Alt+F2`, `r`, and `Enter` keys in sequence or by running the `killall -SIGQUIT gnome-shell` command.

h) In the desktop environment, run GNOME Tweaks and then enable **KStatusNotifierItem/AppIndicator Support** in the Tweaks or Extensions tool.

> **Note:**
>
> For RHEL 9.x, Rocky Linux 9.x, and SUSE 15.x installed with GNOME, Extensions management has been moved out of Tweaks. We recommend you download GNOME Extensions from the GNOME Software app store and then enbale AppIndicator support in GNOME Ex-

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> tensions.

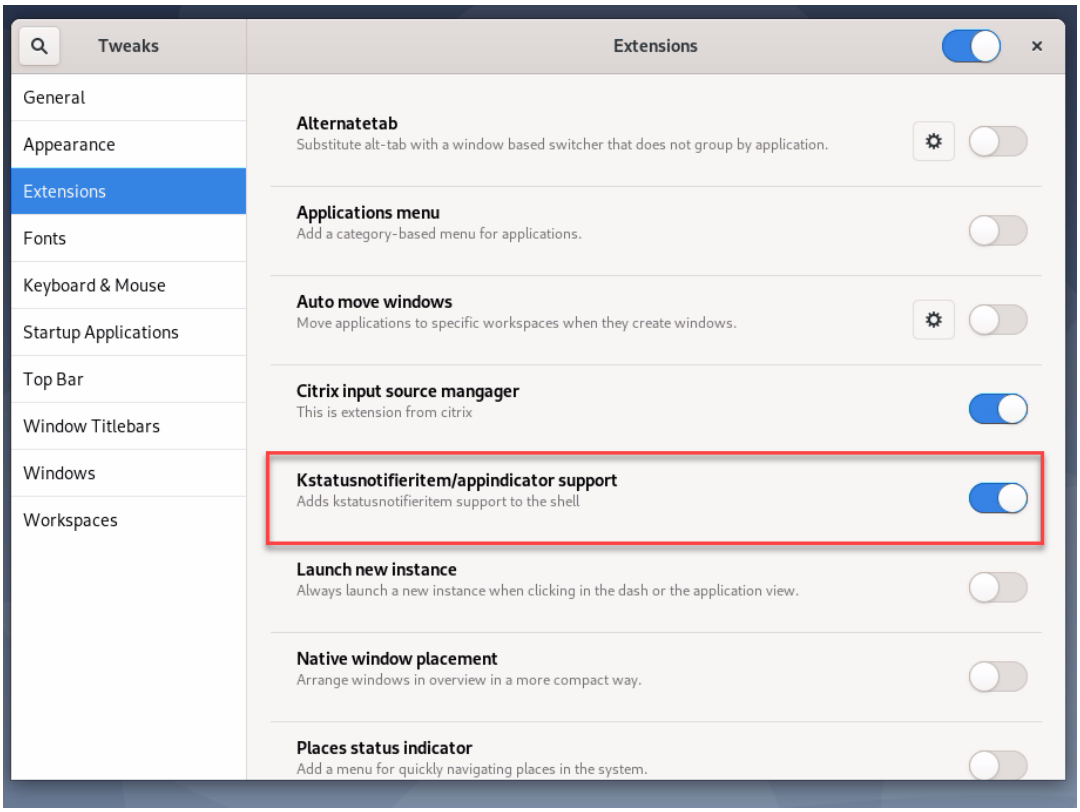
3. (**This step is required only when you are using Debian 11 installed with GNOME.**) Complete the following steps to install and enable GNOME system tray icons.

   a) Run the `sudo apt install gnome-shell-extension-appindicator` command. You might have to log out and then back in again for GNOME to see the extension.

   b) Search for Tweaks in your **Activities** screen.

   c) Select **Extensions** in the Tweaks tool.

   d) Enable **Kstatusnotifieritem/appindicator support**.



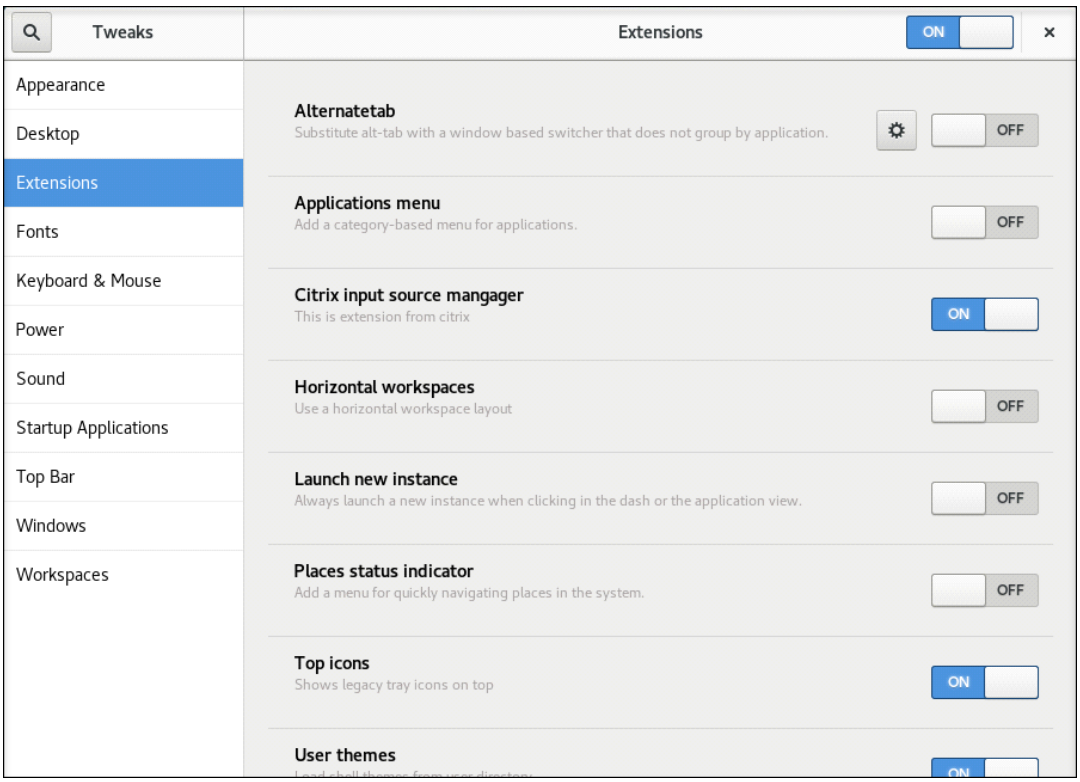
4. (**This step is required if you are using RHEL 7.9 installed with GNOME.**) Complete the following steps to install and enable GNOME system tray icons.

   a) Search for Tweaks in your **Activities** screen.

   b) Select **Extensions** in the Tweaks tool.

   c) Enable **Top icons**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



d) Log out of and back on to the session.

Thinwire progressive display

June 20, 2024

Session interactivity can degrade on low-bandwidth or high-latency connections. For example, scrolling on a webpage can become slow, unresponsive, or choppy. Keyboard and mouse operations can lag behind graphics updates.

Through version 7.17, you were able to use policy settings to reduce bandwidth consumption by configuring the session to **Low** visual quality, or setting a lower color depth (16-bit or 8-bit graphics). However, you had to know that a user was on a weak connection. HDX Thinwire did not dynamically adjust static image quality based on network conditions.

Starting with version 7.18, HDX Thinwire switches to a progressive update mode by default in either of the following cases:

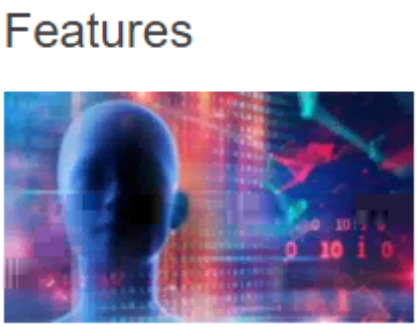
- Available bandwidth falls below 2 Mbps.
- Network latency exceeds 200 ms.

In this mode:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

For example, in the following graphic where progressive update mode is active, the letters **F** and **e** have blue artifacts, and the image is heavily compressed. This approach significantly reduces bandwidth consumption, which allows images and text to be received more quickly, and session interactivity improves.

When you stop interacting with the session, the degraded images and text are progressively sharpened to lossless. For example, in the following graphic, the letters no longer contain blue artifacts, and the image appears at source quality.



For images, sharpening uses a random block-like method. For text, individual letters or parts of words are sharpened. The sharpening process occurs over several frames. This approach avoids introducing a delay with a single large sharpening frame.

Transient imagery (video) is still managed with adaptive display or Selective H.264.

How progressive mode is used

By default, progressive mode is on standby for the **Visual quality** policy settings: **High**, **Medium** (default), and **Low**.

Progressive mode is forced off (not used) when:

- **Visual quality** = **Always Lossless** or **Build to Lossless**
- **Preferred color depth for simple graphics** = 8-bit
- **Use video codec for compression** = **For the entire screen** (when full-screen H.264 is desired)

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

When progressive mode is on standby, by default it is enabled when either of the following conditions occurs:

- Available bandwidth drops below 2 Mbps
- Network latency increases above 200 ms

After a mode switch occurs, a minimum of 10 s is spent in that mode, even if the adverse network conditions are momentary.

Change progressive mode behavior

You can change the progressive mode behavior by running the following command:

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\SOFTWARE\
    CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "
    ProgressiveDisplay" -d "<value>" --force
```

Where <value>:

0 = Always off (do not use under any circumstances)

1 = Automatic (toggle based on network conditions, default value)

2 = Always on

When in automatic mode (1), you can run either of the following commands to change the thresholds at which progressive mode is toggled:

```
sudo /opt/Citrix/VDA/bin/ctxreg  create -k "HKEY_LOCAL_MACHINE\SYSTEM\
    CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "
    ProgressiveDisplayBandwidthThreshold" -d "<value>" --force
```

Where <value> is <threshold in Kbps> (default = 2,048)

Example: 4096 = toggle progressive mode on if bandwidth falls below 4 Mbps

```
sudo /opt/Citrix/VDA/bin/ctxreg  create -k "HKEY_LOCAL_MACHINE\SOFTWARE
    \CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "
    ProgressiveDisplayLatencyThreshold" -d "<value>" --force
```

Where <value> is <threshold in ms> (default = 200)

Example: 100 = toggle progressive mode on if network latency drops below 100 ms.

General content redirection

June 20, 2024

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Client drive mapping and client folder redirection

If	Then
you enable only client drive mapping on the host (VDA),	client-side full volumes are automatically mapped to the sessions under the **ctxmnt** subdirectory in the home directory.
you enable client folder redirection on the host (VDA) and the user configures it on the user device (client),	the portion of the local volume specified by the user is redirected.

USB device redirection

USB devices are shared between Citrix Workspace app and the Linux VDA desktop. When a USB device is redirected to the desktop, you can use the USB device as if it were locally connected.

Client drive mapping

June 20, 2024

You can use client drive mapping and client folder redirection to make client-side files accessible on the host-side session. The comparison between client drive mapping and client folder redirection is as follows:

If	Then
you enable only client drive mapping on the host (VDA),	client-side full volumes are automatically mapped to the sessions under the **ctxmnt** subdirectory in the home directory.
you enable client folder redirection on the host (VDA) and the user configures it on the user device (client),	the portion of the local volume specified by the user is redirected.

Enable client drive mapping

To enable client drive mapping, set the **Client drive redirection** policy to **Allowed** in Citrix Studio. For more information about the policy, see File Redirection policy settings.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Enable client folder redirection and specify folders to redirect

To enable client folder redirection, run the following command on the VDA:

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\Client Folder Redirection" -t "REG_DWORD" -v "CFROnlyModeAvailable" -d "0x00000001" --force
```

To specify which folders to redirect from the client to the host-side session, complete the following steps on the user device:

1. Ensure that the latest version of Citrix Workspace app is installed.
2. From the Citrix Workspace app installation directory, start **CtxCFRUI.exe**.
3. Choose the **Custom** radio button and add, edit, or remove folders.
4. Disconnect and reconnect your sessions for the setting to take effect.

USB device redirection

August 9, 2024

USB devices are shared between Citrix Workspace app and the Linux VDA desktop. When a USB device is redirected to the desktop, you can use the USB device as if it were locally connected.

> **Tip:**
>
> We recommend using USB device redirection when the network latency is lower than 100 milliseconds. Do not use USB device redirection when the network latency is higher than 200 milliseconds.

USB device redirection includes three main areas of functionality:

- Open-source USB/IP project
- Citrix USB session module
- Citrix USB service module

Open-source USB/IP project:

The USB/IP project consists of a Linux kernel driver and some user mode libraries that let you communicate with the kernel driver to get all USB data.

The Linux VDA implements USB device redirection based on the open-source USB/IP project and reuses the kernel driver and user mode libraries of USB/IP. However, all USB data transfers between the Linux VDA and Citrix Workspace app are encapsulated by the Citrix ICA USB protocol.

Citrix USB session module:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

The Citrix USB session module acts as a communication bridge between the USB/IP kernel module and Citrix Workspace app.

Citrix USB service module:

The Citrix USB service module manages all operations on USB devices, for example, attach or detach USB devices.

How USB device redirection works

Typically, if a USB device is redirected successfully to the Linux VDA, one or more device nodes are created in the system /dev path. Sometimes, however, the redirected device isn't usable for an active Linux VDA session. USB devices rely on drivers to function properly and some devices require special drivers. If drivers aren't provided, the redirected USB devices are inaccessible to the active Linux VDA session. To make sure of USB device connectivity, install the drivers and configure the system properly.

The Linux VDA supports a list of USB devices that are successfully redirected from the client.

Supported USB devices

> **Tip:**
>
> We have added support for USB 3.0 ports. You can insert USB 3.0 devices into USB 3.0 ports on a client device.

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

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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

USB mass storage device	VID:PID	File system
Toshiba Kingston DataTraveler 3.0 USB device	0930:6545	FAT32, NTFS
Taiwan OEM – OBSOLETE VendorCo ProductCode Disk 2.0	FFFF:5678	FAT32, NTFS
TD-RDF5A Transcend USB device	8564:4000	FAT32, NTFS

> **Note:**
>
> To use NTFS on Amazon Linux 2, CentOS, RHEL, Rocky Linux, and SUSE, enable NTFS support on these distributions first.

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

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

Yubico USB	VID:PID
Yubico YubiKey OTP+FIDO+CCID - Keyboard, HID	1050:0407

Webcam USB	VID:PID
Logitech composite USB device – WebCam, Audio	0460:0825

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Configure USB device redirection

(For CentOS, RHEL, and Rocky Linux only) Install or compile the USB/IP kernel module

The Linux VDA uses USB/IP as the virtual host controller for USB device redirection. Because in most cases the USB/IP kernel module is released with the Linux kernel version 3.17 and later, you don't have to build the kernel module by default. However, the USB/IP kernel module is not available for CentOS, RHEL, and Rocky Linux. To use USB device redirection with these Linux distributions, you must install or compile the USB/IP kernel module. Download and install USB/IP from https://pkgs.org/download/kmod-usbip based on your Linux distribution.

Set USB device redirection policies

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

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

Enable USB device redirection In Citrix Studio, enable (or disable) USB device redirection from the client (for workstation hosts only).

In the **Edit Setting** dialog:

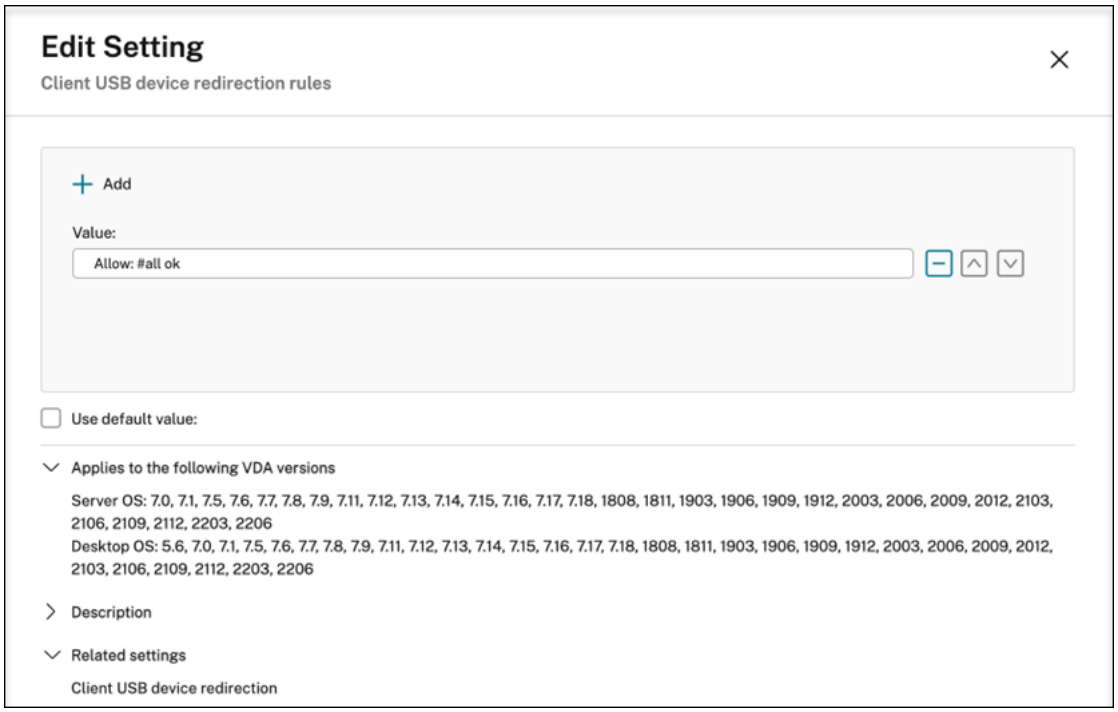
1. Select **Allowed**.
2. Click **OK**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Set USB device redirection rules After enabling the USB redirection policy, set the redirection rules using Citrix Studio by specifying which devices are allowed (or denied) on the Linux VDA.

In the **Client USB device redirection rules** dialog:

1. Click **New** to add a redirection rule, or click **Edit** to review an existing rule.
2. After creating (or editing) a rule, click **OK**.



For more information about configuring generic USB device redirection, see the Citrix Generic USB Redirection Configuration Guide.

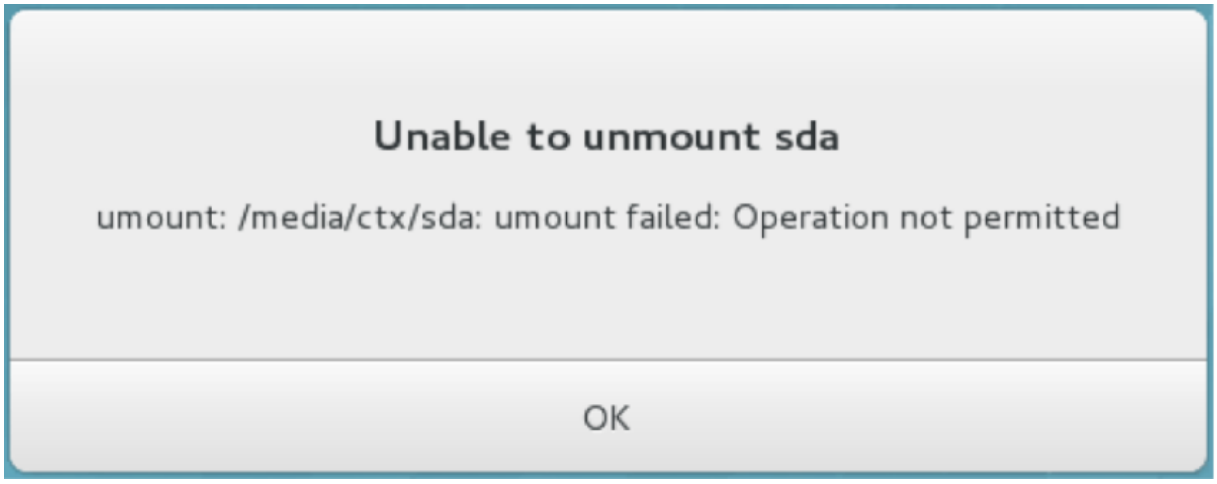
Troubleshoot USB device redirection issues

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

Unable to unmount the redirected USB disk

The Linux VDA manages all USB disks redirected from Citrix Workspace app under the administrative privilege to make sure that only the owner can access the redirected device. As a result, you can unmount the device only with the administrative privilege.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



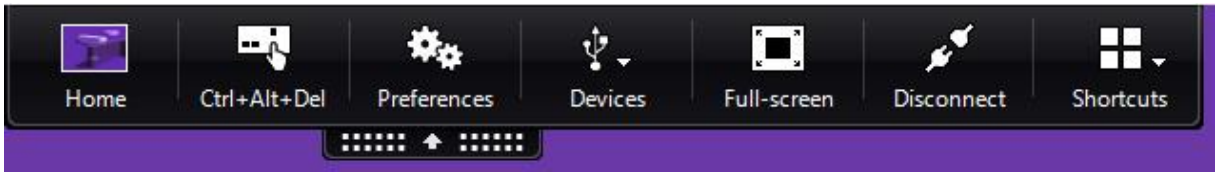
File lost when you stop redirecting a USB disk

If you stop redirecting a USB disk immediately using the toolbar of Citrix Workspace app, the files you modified or created on the disk 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 isn't written to the disk itself. If you stop redirecting using the toolbar of Citrix Workspace app, there's no time remaining for data being flushed to the disk, which results in lost data.

To resolve this issue, use the **sync** command in a terminal to flush data to the disk before stopping USB redirection.

No devices in the toolbar of Citrix Workspace app

Sometimes, you might not be able to see devices listed in the toolbar of Citrix Workspace app, which indicates that no USB redirection is taking place.



If you come across the issue, verify the following:

- The policy is configured to allow USB device redirection.
- The Citrix USB service module is running.

If the policy is not set correctly, correct it by referencing the Set USB device redirection policies section in this article.

If the Citrix USB service module is not running, complete the following steps:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

1. Check whether a USB/IP kernel module is available on your Linux distribution using the following command:

```
modinfo usbip-core
```

2. If the output is shown as follows, install or compile the USB/IP kernel module based on your Linux distribution:

```
modinfo: ERROR: Module usbip-core not found.
```

   - For Amazon Linux 2, CentOS, RHEL, and Rocky Linux, see the Install or compile the USB/IP kernel module section in this article.
   - For SUSE, download and install the USB/IP package from https://software.opensuse.org/package/usbip.
   - For Ubuntu/Debian, complete the following steps to compile and install the USB/IP kernel module:

     a) Download the USB/IP kernel module source code.

     Go to the Linux kernel repository at https://github.com/torvalds/linux/tree/master/drivers/usb/usbip, select the target Linux kernel version (v4.15 or later) tag, and get the link such as https://github.com/torvalds/linux/tree/v4.15/drivers/usb/usbip.

     Go to DownGit and enter the preceding link to create a download link for downloading the USB/IP source code.

     b) Unzip the source file using the following commands:

```
unzip ${
 USBIP_SRC }
 .zip

cd usbip
```

     c) Modify the **Makefile** file as follows:

```
# SPDX-License-Identifier: GPL-2.0

ccflags-$(CONFIG_USBIP_DEBUG) := -DDEBUG

obj-$(CONFIG_USBIP_CORE) += usbip-core.o

usbip-core-y := usbip_common.o usbip_event.o

obj-$(CONFIG_USBIP_VHCI_HCD) += vhci-hcd.o

vhci-hcd-y := vhci_sysfs.o vhci_tx.o vhci_rx.o vhci_hcd.o

#obj-$(CONFIG_USBIP_HOST) += usbip-host.o
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
14
15 #usbip-host-y := stub_dev.o stub_main.o stub_rx.o stub_tx.o
16
17 #obj-$(CONFIG_USBIP_VUDC) += usbip-vudc.o
18
19 #usbip-vudc-y := vudc_dev.o vudc_sysfs.o vudc_tx.o vudc_rx.
      o vudc_transfer.o vudc_main.o
```

d) Compile the source code:

```
1 apt-get install linux-headers-`uname -r`
2
3 make -C /lib/modules/`uname -r`/build M=$PWD
```

e) Install the USB/IP kernel module:

```
1 cp usbip-core.ko vhci-hcd.ko /opt/Citrix/VDA/lib64/
```

f) Restart the **ctxusbsd** service to load the USB/IP kernel module:

```
1 systemctl restart ctxusbsd
```

Redirection failure when USB devices are visible in the toolbar of Citrix Workspace app, but are labeled "policy restricted"

When the issue occurs, do the following:

- Configure the Linux VDA policy to enable redirection.
- Check whether any additional policy restrictions are configured in the registry of Citrix Workspace app. Check **DeviceRules** in the registry path to make sure that the device isn't denied access by this setting:

  HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\ICA Client\GenericUSB

A USB device is redirected successfully, but I can't use it in my session

Typically, only supported USB devices can be redirected. Other devices might be redirected to an active Linux VDA session too. For every redirected device, a node owned by the user is created in the system **/dev** path. However, it's 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:**
>
> For 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. It might not be the case for

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> devices that aren't in the supported device list.

Clipboard redirection

June 20, 2024

Clipboard redirection lets you copy and paste data between applications that are running in the VDA session and applications running on the client device.

This article describes the available Citrix policies that allow you to achieve clipboard redirection.

Citrix policies for clipboard redirection

Client clipboard redirection

This setting allows or prevents the clipboard on the client device being mapped to the clipboard on the VDA.

By default, clipboard redirection is set to **Allowed**.

To prevent copy-and-paste data transfer between a session and the local clipboard, select **Prohibited**. Users can still copy and paste data between applications running in sessions.

Clipboard redirection bandwidth limit

This setting specifies the maximum allowed bandwidth (kbps) for data transfer between the session and the local clipboards.

Clipboard redirection bandwidth limit percent

This setting specifies the maximum allowed bandwidth for data transfer between the session and the local clipboards, as a percentage of the total session bandwidth.

Limit clipboard client to session transfer size

This setting specifies the maximum size of clipboard data that a single copy-and-paste operation can transfer from a client device to a virtual session.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

To limit the clipboard transfer size, enable the **Limit clipboard client to session transfer size** setting. Then, in the **Size Limit** field, enter a value in kilobytes to define the size of data transfer between the local clipboard and a session.

By default, this setting is disabled and there's no limit on client-to-session transfers.

Limit clipboard session to client transfer size

This setting specifies the maximum size of clipboard data that a single copy-and-paste operation can transfer from a virtual session to a client device.

To limit the clipboard transfer size, enable the **Limit clipboard session to client transfer size** setting. Then, in the **Size Limit** field, enter a value in kilobytes to define the size of data transfer between a session and the local clipboard.

By default, this setting is disabled and there's no limit on session to client transfers.

Restrict client clipboard write AND Client clipboard write allowed formats

Enabling the two settings lets you allow specific data formats to be copied and pasted from the session to the client (writing to the client).

The following clipboard formats are system defined:

- CF_TEXT
- CF_BITMAP
- CF_METAFILEPICT
- CF_SYLK
- CF_OEMTEXT
- CF_DIB
- CF_PALETTE
- CF_UNICODETEXT
- CF_LOCALE
- CF_DIBV5
- CF_HDROP

Restrict session clipboard write AND Session clipboard write allowed formats

Enabling the two settings lets you allow specific data formats to be copied and pasted from the client to the session (writing to the session).

The following clipboard formats are system defined:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- CF_TEXT
- CF_BITMAP
- CF_METAFILEPICT
- CF_SYLK
- CF_OEMTEXT
- CF_DIB
- CF_PALETTE
- CF_UNICODETEXT
- CF_LOCALE
- CF_DIBV5
- CF_HDROP

Keyboard

June 20, 2024

This section contains the following topics:

- Client IME
- Client IME user interface synchronization
- Dynamic keyboard layout synchronization
- Soft keyboard
- Support for multiple language inputs

X Keyboard Extension (XKB) configuration

July 12, 2024

Starting with the 2407 release, the Linux VDA specifies evdev as the default XKB rule for keyboard configuration. This decision can be justified for several reasons:

- Most modern Linux distributions use the evdev XKB rule by default.
- Almost all applications are following the key code pattern of the evdev XKB rule.
- The evdev XKB rule provides extended support for a wide range of key symbols beyond basic alphanumeric characters and typical function keys, such as 'Cancel', 'Redo', 'Undo', 'XF86Copy', 'XF86Open', and 'XF86Paste'.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- The evdev XKB rule in Linux addresses various keyboard-related issues, particularly those concerning incorrect responses to key inputs.

If you want to switch back to the XFree86 rule, you can make the following registry setting:

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\System\
   CurrentControlSet\Control\Citrix\LanguageBar" -v "XkbRuleName" -d "
   xorg"
```

In this command, only the exact case-sensitive text 'xorg'is recognized as the 'xfree86'ruleset; any other input values are ineffective.

Client Input Method Editor (IME)

June 20, 2024

Overview

Double-byte characters such as Chinese, Japanese, and Korean characters must be typed through an IME. Type such characters with any IME that is compatible with Citrix Workspace app on the client side, such as the Windows native CJK IME.

Installation

This feature is installed automatically when you install the Linux VDA.

Usage

Open a Citrix Virtual Apps or Citrix Virtual Desktops session as per usual.

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

Known issues

- Double-clicking a cell in a Google spreadsheet is a must before you can use the client IME feature to type characters in the cell.
- The client IME feature is not disabled automatically in Password fields.
- The IME user interface does not follow the cursor in the input area.

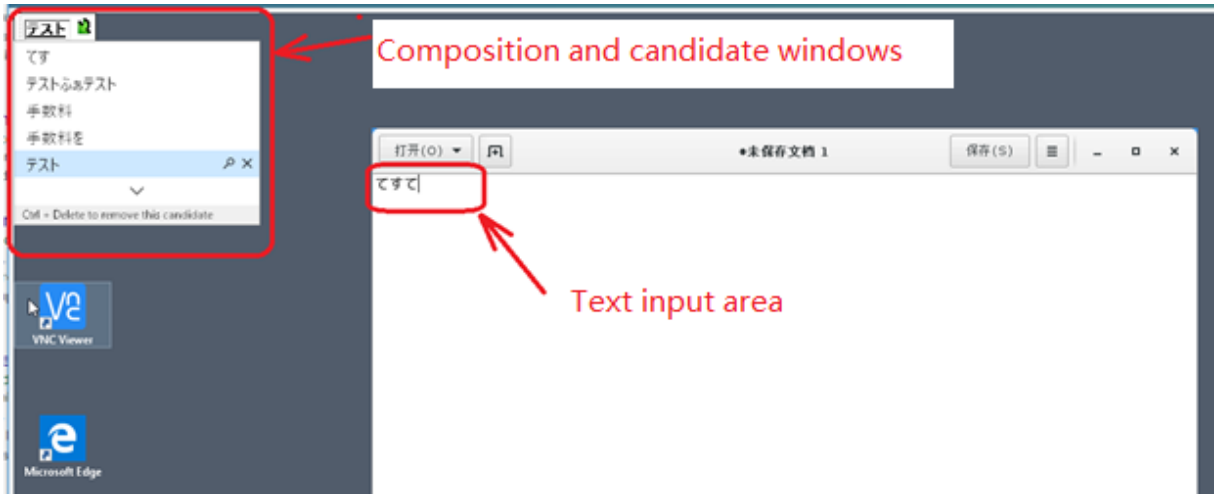
© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Client IME user interface synchronization

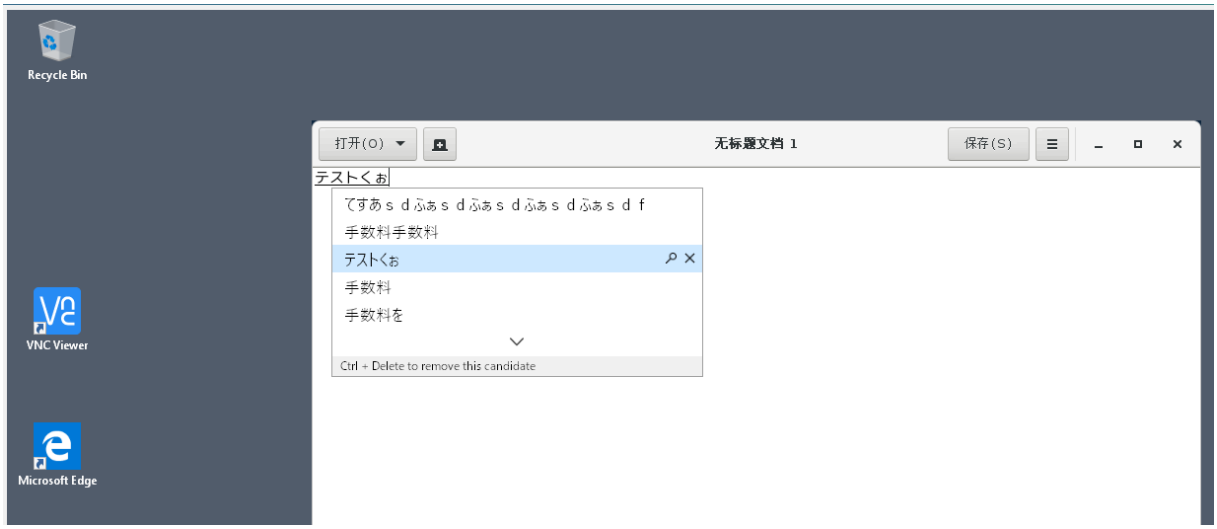
June 20, 2024

Overview

To date, the client IME user interface (including the composition window and candidate window) was positioned in the upper left corner of the screen. It did not follow the cursor and sometimes was located far from the cursor in the text input area:



Citrix enhances usability and further improves the user experience with the client IME as follows:



© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Prerequisites for using the feature

1. Enable Intelligent Input Bus (`IBus`) on your Linux VDA. For information on how to enable `IBus` on a Linux OS, see the OS vendor's documentation. For example:
   - Ubuntu: https://help.ubuntu.com/community/ibus
   - CentOS, RHEL: https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/7.0_release_notes/sect-red_hat_enterprise_linux-7.0_release_notes-internationalization-input_methods
   - Debian: https://wiki.debian.org/I18n/ibus
   - SUSE: https://documentation.suse.com/sles/15-SP2/html/SLES-all/cha-gnome-settings.html#sec-gnome-settings-lang
2. The feature installs automatically but you must enable it before you can use it.

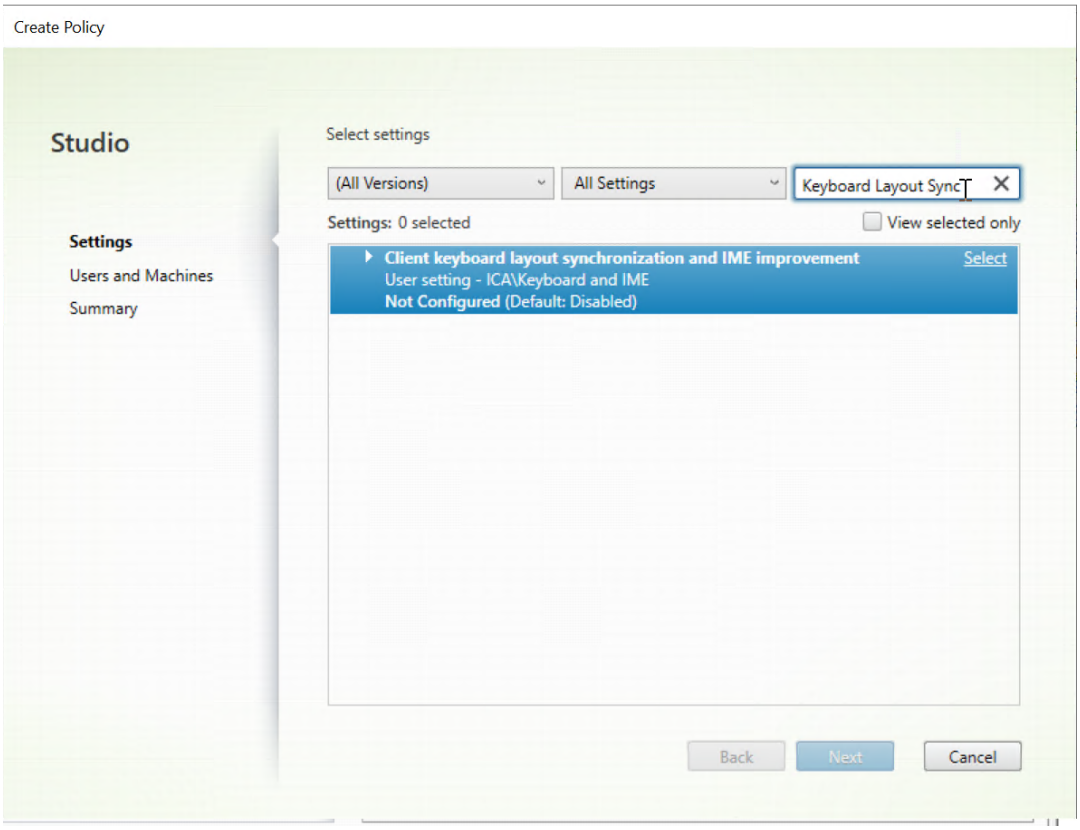
Enable and disable the feature

The client IME user interface synchronization feature is disabled by default. To enable or disable the feature, set the **Client Keyboard Layout Sync and IME Improvement** policy or edit the registry through the `ctxreg` utility.

> **Note:**
>
> The **Client Keyboard Layout Sync and IME Improvement** policy takes priority over registry settings and can be applied to user and machine objects you specify or all objects in your site. Registry settings on a given Linux VDA apply to all sessions on that VDA.

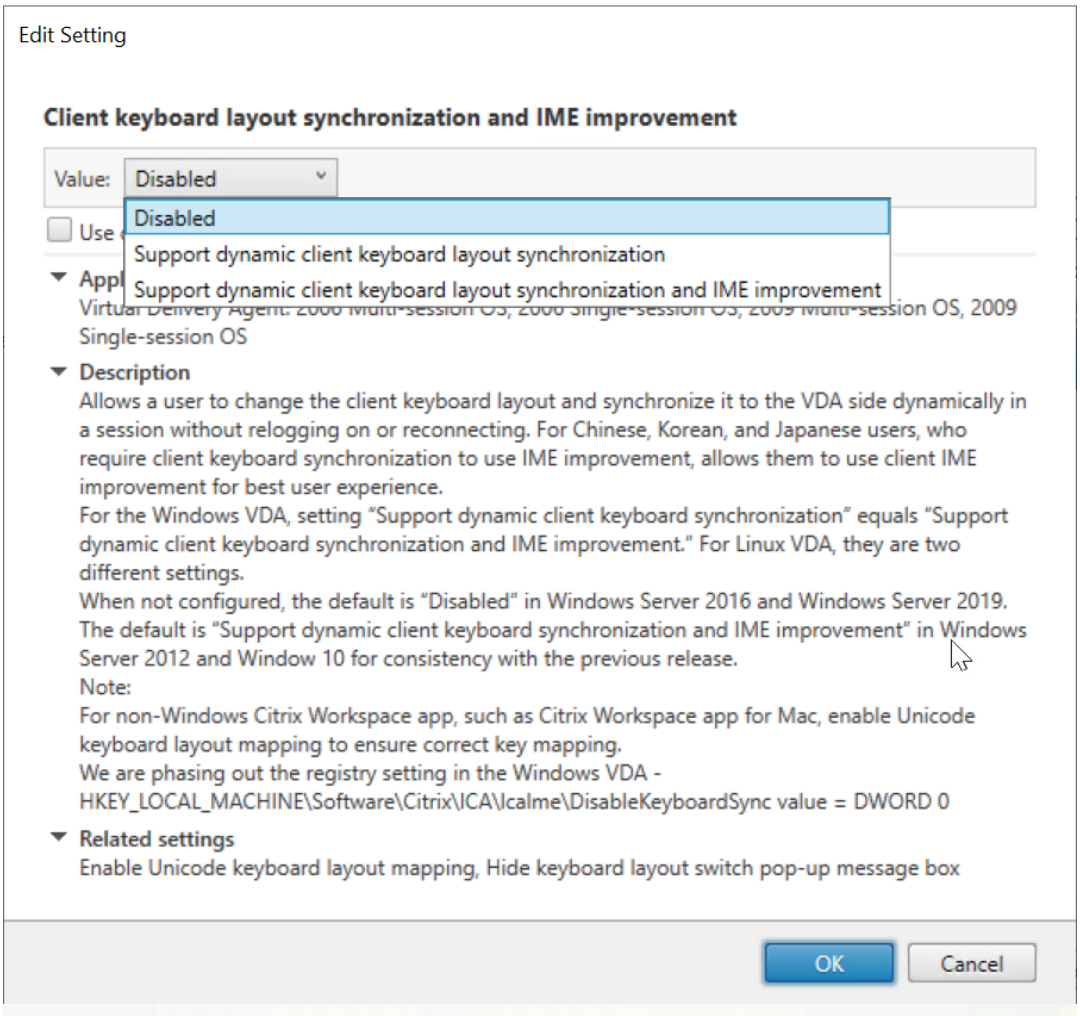
- Set the **Client Keyboard Layout Sync and IME Improvement** policy to enable or disable the client IME user interface synchronization feature:
  1. In Studio, right-click **Policies** and select **Create Policy**.
  2. Search for the **Client Keyboard Layout Sync and IME Improvement** policy.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



3. Click **Select** next to the policy name.
4. Set the policy.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



There are three options available:

- **Disabled**: disables dynamic keyboard layout synchronization and client IME user interface synchronization.
- **Support dynamic client keyboard layout synchronization**: enables dynamic keyboard layout synchronization regardless of the DWORD value of the **SyncKeyboardLayout** registry key at `HKEY_LOCAL_MACHINE\SYSTEM \CurrentControlSet\Control\Citrix\LanguageBar`.
- **Support dynamic client keyboard layout synchronization and IME improvement**: enables both dynamic keyboard layout synchronization and client IME user interface synchronization regardless of the DWORD values of the **SyncKeyboardLayout** and **SyncClientIME** registry keys at `HKEY_LOCAL_MACHINE\SYSTEM \CurrentControlSet\Control\Citrix\LanguageBar`.

- Edit the registry through the `ctxreg` utility to enable or disable the client IME user interface synchronization feature:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

To enable the feature, run the command:

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

To disable the feature, run the command:

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

Dynamic keyboard layout synchronization

June 20, 2024

Previously, the keyboard layouts on the Linux VDA and on the client device had to be the same. Key mapping issues might occur, for example, when the keyboard layout changed from English to French on the client device but not on the VDA.

Citrix addresses the issue by synchronizing the keyboard layout of the VDA with the keyboard layout of the client device automatically. Anytime the keyboard layout on the client device changes, the layout on the VDA follows suit.

> **Note:**
>
> Citrix Workspace app for HTML5 does not support the dynamic keyboard layout synchronization feature.

Configuration

The dynamic keyboard layout synchronization feature is disabled by default. To enable or disable the feature, set the **Client Keyboard Layout Sync and IME Improvement** policy or edit the registry through the `ctxreg` utility.

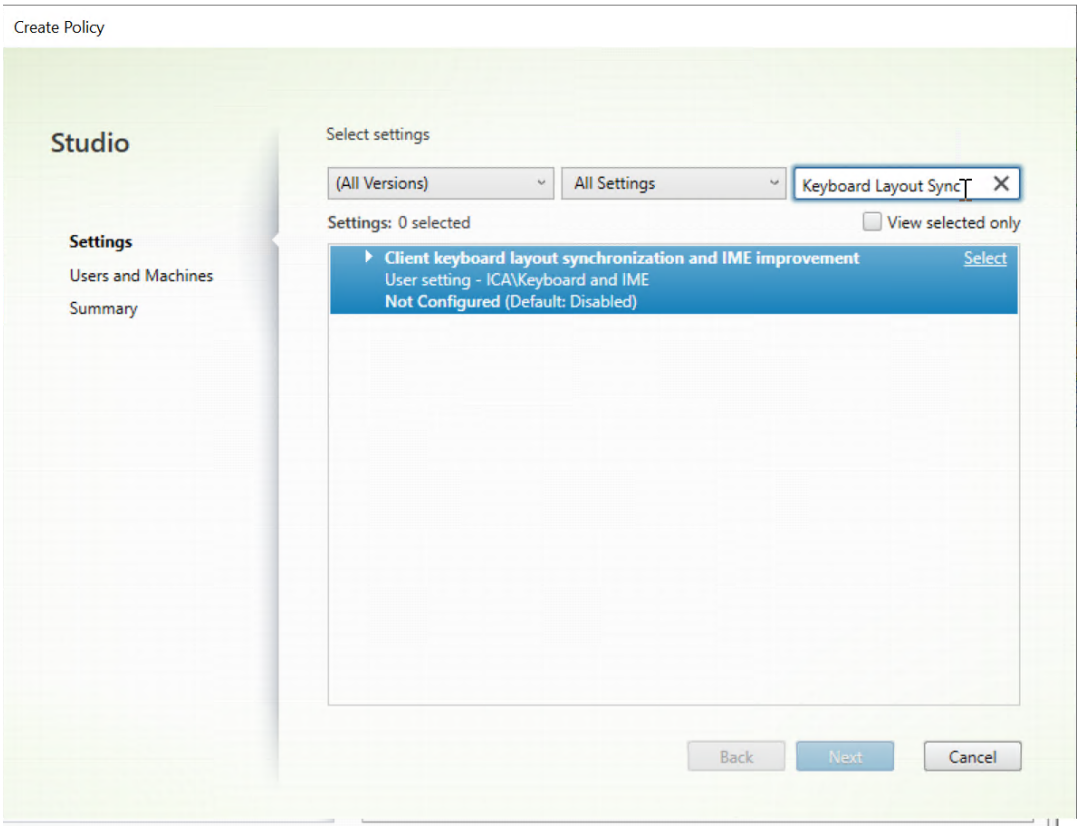
> **Note:**
>
> The **Client Keyboard Layout Sync and IME Improvement** policy takes priority over registry settings and can be applied to user and machine objects you specify or all objects in your site. Registry settings on a given Linux VDA apply to all sessions on that VDA.

- Set the **Client Keyboard Layout Sync and IME Improvement** policy to enable or disable the dynamic keyboard layout synchronization feature:

  1. In Studio, right-click **Policies** and select **Create Policy**.

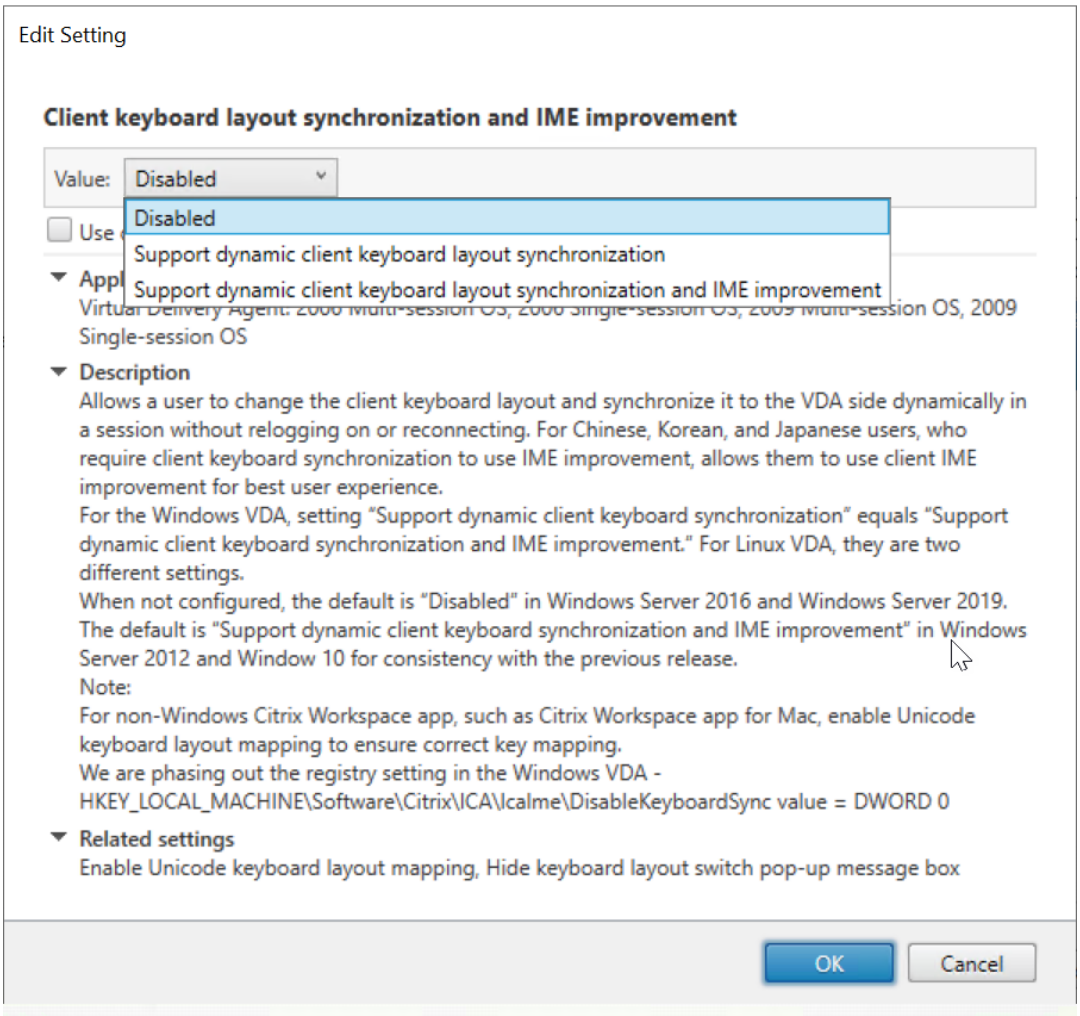
© 1999–2024 Cloud Software Group, Inc. All rights reserved.

2. Search for the **Client Keyboard Layout Sync and IME Improvement** policy.



3. Click **Select** next to the policy name.
4. Set the policy.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



There are three options available:

- **Disabled**: disables dynamic keyboard layout synchronization and client IME user interface synchronization.
- **Support dynamic client keyboard layout synchronization**: enables dynamic keyboard layout synchronization regardless of the DWORD value of the **SyncKeyboardLayout** registry key at `HKEY_LOCAL_MACHINE\SYSTEM \CurrentControlSet\Control\Citrix\LanguageBar`.
- **Support dynamic client keyboard layout synchronization and IME improvement**: enables both dynamic keyboard layout synchronization and client IME user interface synchronization regardless of the DWORD values of the **SyncKeyboardLayout** and **SyncClientIME** registry keys at `HKEY_LOCAL_MACHINE\SYSTEM \CurrentControlSet\Control\Citrix\LanguageBar`.

- Edit the registry through the `ctxreg` utility to enable or disable the dynamic keyboard layout synchronization feature:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

To enable this feature, run the command:

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

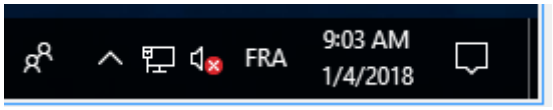
To disable this feature, run the command:

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

Usage

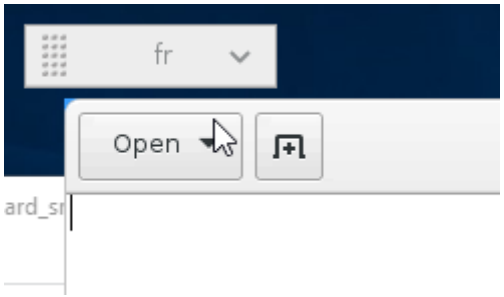
With this feature enabled, when the keyboard layout changes on the client device during a session, the keyboard layout of the session changes accordingly.

For example, if you change the keyboard layout on a client device to French (FR):

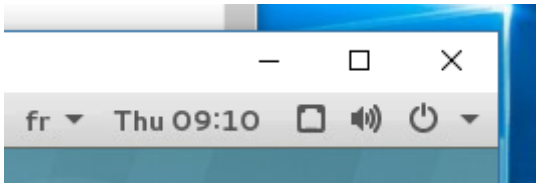


Then the keyboard layout of the Linux VDA session also changes to “fr.”

In an application session, you can see this automatic change if you have enabled the language bar:



In a desktop session, you can see this automatic change in the task bar:

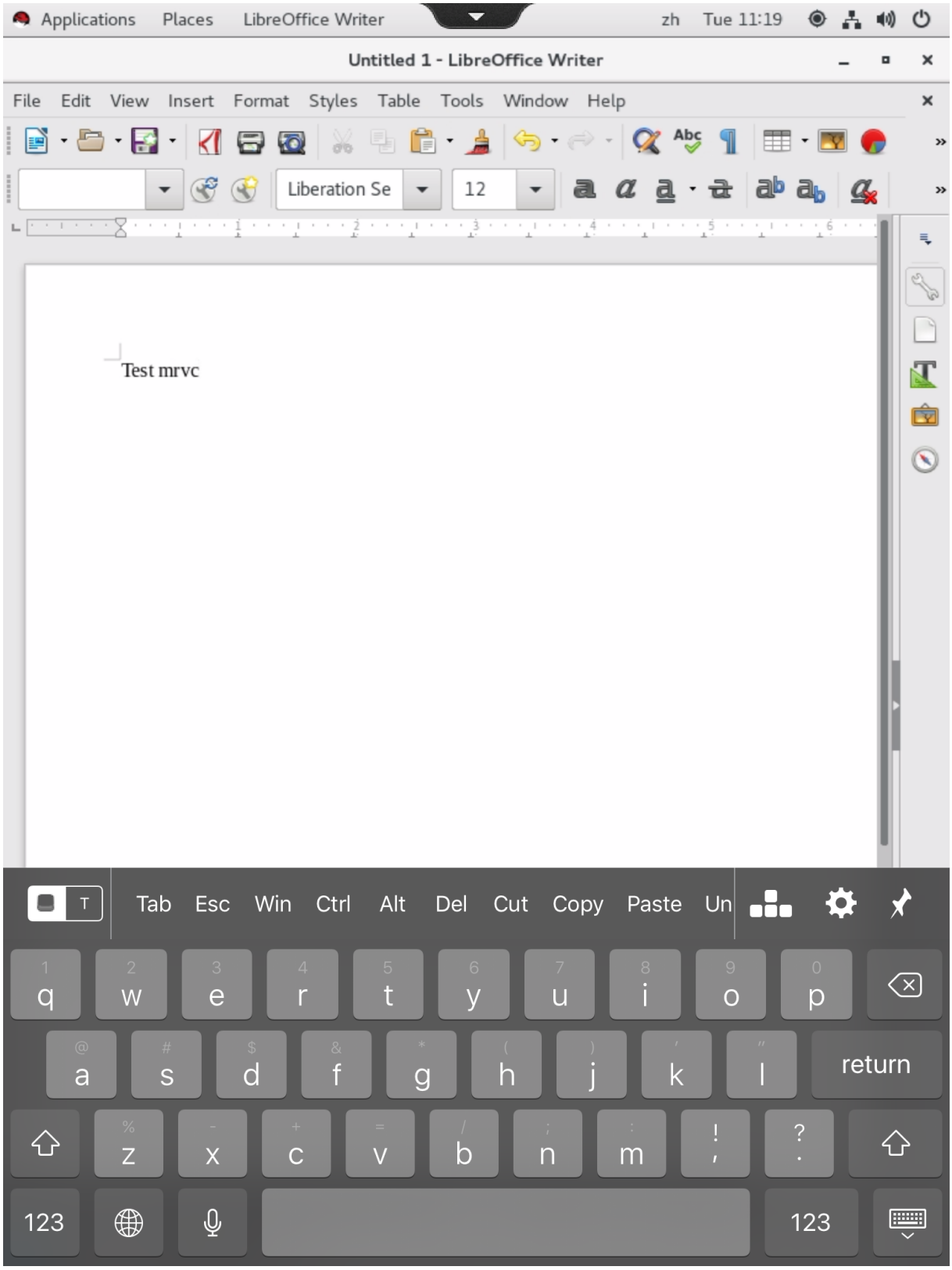


Soft keyboard

June 20, 2024

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

The soft keyboard feature is available in a Linux virtual desktop or application session. The soft keyboard shows or hides automatically when you enter or leave an input field.



© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> The feature is supported on Citrix Workspace app for iOS and for Android.

Enable and disable the feature

The feature is disabled by default. Use the **ctxreg** utility to enable or disable the feature. The feature configuration on a given Linux VDA applies to all sessions on that VDA.

To enable the feature:

1. Run the command:

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\
   CurrentControlSet\Control\Citrix\VirtualChannels\MrVc"  -v "
   Enabled" -d "0x00000001"
```

2. In Citrix Studio, set the **Automatic keyboard display** policy to **Allowed**.
3. (Optional) For RHEL 7 and CentOS 7, run the following command to configure the Intelligent Input Bus (IBus) as the default IM service:

```
echo "GTK_IM_MODULE=ibus" >>/etc/bashrc
```

To disable the feature, run the command:

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\CurrentControlSet\
   Control\Citrix\VirtualChannels\MrVc"  -v "Enabled" -d "0x00000000"
```

> **Note:**
>
> The preceding settings take effect when you log on to a new session or log off and back on to the current session.

Limitations

- The feature might not work as expected with Google Chrome, LibreOffice, and other apps.
- To display the soft keyboard again after hiding it manually, click a non-input field and then the current input field again.
- The soft keyboard might not appear when you click from one input field to another in a web browser. To work around this issue, click a non-input field and then the target input field.
- The feature does not support Unicode characters and double-byte characters (such as Chinese, Japanese, and Korean characters).
- The soft keyboard is not available for password input fields.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- The soft keyboard might overlap the current input field. In this case, move the app window or scroll up your screen to move the input field to an accessible position.
- Due to compatibility issues between Citrix Workspace app and Huawei tablets, the soft keyboard appears on Huawei tablets even with a physical keyboard connected.

Support for multiple language inputs

June 20, 2024

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

However, the native language bar on the Linux VDA was unavailable to the published application because the language bar is highly integrated with the Linux desktop environment. As a result, users were unable to input text in a language that requires IME such as Chinese, Japanese, or Korean. 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:

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

To disable this feature, run the command:

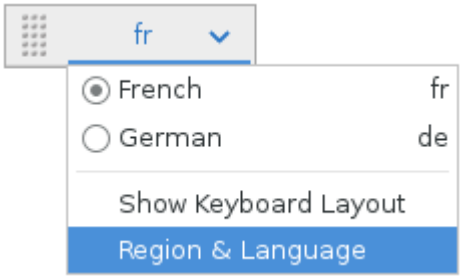
```
1 /opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SYSTEM\
     CurrentControlSet\Control\Citrix\LanguageBar" -v "Enabled" -d "0
     x00000000"
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Usage

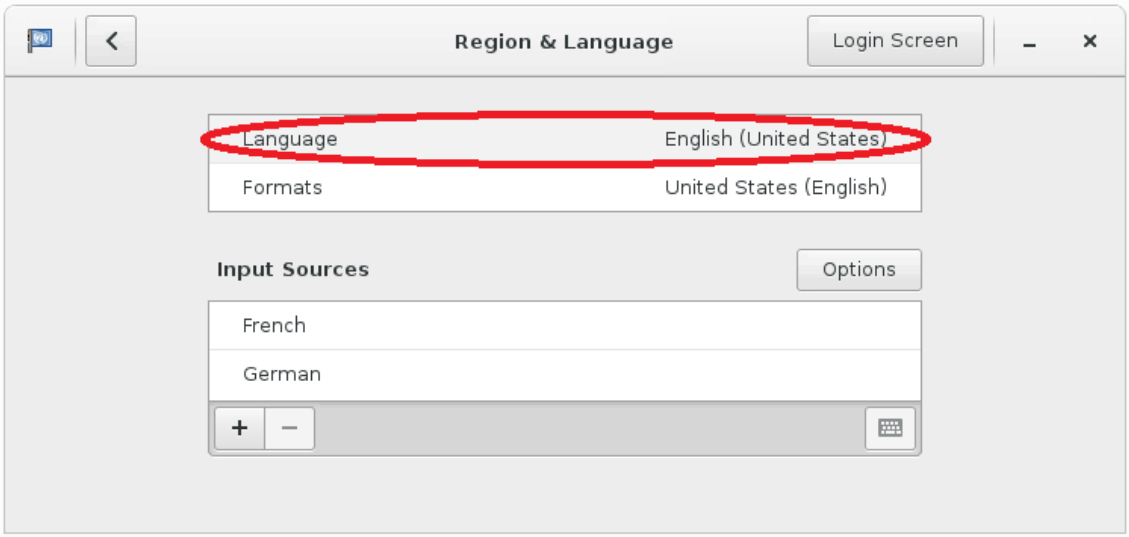
The usage is straightforward.

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



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

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



© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Multimedia

June 20, 2024

This section contains the following topics:

- Audio features
- Browser content redirection
- HDX webcam video compression

Audio features

August 1, 2024

Adaptive audio

Adaptive audio is enabled by default. It supports the following Citrix Workspace app clients:

- Citrix Workspace app for Windows –2109 and later versions
- Citrix Workspace app for Linux –2109 and later versions
- Citrix Workspace app for Mac –2109 and later versions

Adaptive audio falls back to legacy audio when you use a client not included on the list.

Adaptive audio dynamically adjusts audio sampling bitrates based on network conditions to enhance the audio experience, eliminating the need for manual configuration of audio quality policies on the VDA.

The following table shows a comparison between adaptive audio and legacy audio:

Adaptive audio	Legacy audio
Max. audio sample rate: 48 kHz	Max. audio sample rate: 8 kHz
Stereo channel	Mono channel

Audio service considerations

Starting with version 2407, the Linux VDA supports PipeWire in Debian 12.5, RHEL 9.x, and Rocky Linux 9.x, where it is the default audio service. For the other distributions, continue using PulseAudio.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

PulseAudio

- Use PulseAudio 13.99 or later on RHEL 8.x and Rocky Linux 8.x.

PipeWire

The Linux VDA supports PipeWire only in Debian 12.5, RHEL 9.x, and Rocky Linux 9.x. Here are some considerations to take:

- If you have the Linux VDA version 2407 or later installed on Debian 12.5, RHEL 9.x, or Rocky Linux 9.x, you are using PipeWire.
- If you have a version of the Linux VDA earlier than 2407 installed on RHEL 9.x or Rocky Linux 9.x, you are using PulseAudio. To continue using PulseAudio, do nothing. To switch to PipeWire, complete the following steps:

  1. Replace the PulseAudio package with pipewire-pulseaudio:

     ```
     dnf swap --allowerasing pulseaudio pipewire-pulseaudio
     ```

  2. Enable PipeWire-related services globally:

     ```
     systemctl --global --now enable pipewire pipewire-pulse
        pipewire-pulse.socket wireplumber
     ```

  3. Restart the Linux VDA.

Loss tolerant mode for audio (preview)

The loss tolerant mode supports audio. This feature increases the user experience for real-time streaming and improves audio quality over EDT when users are connecting through networks with high latency and packet loss.

For more information about the loss tolerant mode and EDT, see Additional information in the Citrix Virtual Apps and Desktops documentation.

Enable the loss tolerant mode for audio feature

Loss tolerant mode for audio is disabled by default. To enable it, complete the following steps:

1. Enable adaptive transport by setting the HDX adaptive transport policy. Adaptive transport is enabled by default.
2. Enable adaptive audio by setting the Adaptive audio policy. adaptive audio is enabled by default.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

3. Enable the loss tolerant mode (EDT unreliable transport) through the following registry setting:

```
ctxreg create -k "HKLM\System\CurrentControlSet\Control\Citrix\
    VirtualChannels\Audio" -t "REG_DWORD" -v "fEdtUnreliableAllowed
    " -d "0x00000001" --force
```

Client requirements and settings

To use loss tolerant mode for audio, ensure that your Citrix Workspace app supports and is configured to enable this feature; otherwise, audio defaults to EDT Reliable transport.

The following are the minimum Citrix Workspace app versions that support loss tolerant mode:

- Citrix Workspace app for Windows minimum version 2309
- Citrix Workspace app for Linux minimum version 2311
- Citrix Workspace app for Mac minimum version 2311

Packet Loss Concealment for improving audio performance (preview)

Starting with the 2407 release, the jitter buffer mechanism is improved, and the Packet Loss Concealment (PLC) is added for the Adaptive audio codec. PLC helps to reconstruct the lost data packets. This enhancement helps to improve the packet loss tolerance and jitter tolerance and thus improves audio performance for loss tolerant mode (EDT lossy) for audio.

This feature is disabled by default. To enable this feature, complete the following steps:

1. Enable the loss tolerant mode for audio feature.
2. Enable the audio redirection feature that allows multiple audio device support.

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\
    CurrentControlSet\Control\Citrix\VirtualChannels\Audio" -t "
    REG_DWORD" -v "fEnableAudioRedirectionV4" -d "0x00000001" --
    force
```

3. To enhance audio performance in loss tolerant mode, enable Read/Write V2 for audio:

```
ctxreg create -k "HKLM\System\CurrentControlSet\Control\Citrix\
    VirtualChannels\Audio" -t "REG_DWORD" -v "fEnableReadWriteV2" -
    d "0x00000001" --force
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Support for multiple audio devices (preview)

Overview

Starting with Version 2311, the Linux VDA introduces an audio redirection feature. The feature allows multiple audio devices on the client machine where Citrix Workspace app is installed to be redirected to the remote Linux VDA session.

With the feature enabled:

- All local audio devices on the client machine are displayed in a session. Instead of CitrixAudioSink (audio output) or CitrixAudioSource (audio input), the audio devices appear with their respective device names. You can select an audio device in an app in a session or use the default audio device during a session which is also the default audio device of the client machine. If necessary, you can change the default audio device from the system settings of the client machine. After the default audio device of the client machine is updated, the new device appears as the default audio device in the session.
- Audio devices within sessions update dynamically when you plug in or remove one.

Configuration

To use the feature, enable it on the Linux VDA and choose a supported Citrix Workspace app.

Enable the feature on the Linux VDA The audio redirection feature that allows multiple audio device support is disabled by default. To enable it, run the following command on the Linux VDA:

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\\System\\
    CurrentControlSet\\Control\\Citrix\\VirtualChannels\\Audio"  -v "
    fEnableAudioRedirectionV4" -t BIN -d "1"
```

To disable or re-enable the feature, run the following commands, respectively:

```
sudo /opt/Citrix/VDA/bin/ctxreg update -k "HKLM\\System\\
    CurrentControlSet\\Control\\Citrix\\VirtualChannels\\Audio" -v "
    fEnableAudioRedirectionV4" -d "0"
```

```
sudo /opt/Citrix/VDA/bin/ctxreg update -k "HKLM\\System\\
    CurrentControlSet\\Control\\Citrix\\VirtualChannels\\Audio" -v "
    fEnableAudioRedirectionV4" -d "1"
```

Client requirements and settings The feature is supported only for the following clients:

- Citrix Workspace app for Windows

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- Citrix Workspace app for Linux minimum version 2212
- Citrix Workspace app for HTML5 minimum version 2306
- Citrix Workspace app for Chrome minimum version 2306
- Citrix Workspace app for Mac minimum version 2311

Proper settings are required on Citrix Workspace app to make the feature function as expected. For more information, see the Citrix Workspace app documentation.

Known issues

Due to the issue with PulseAudio, attempts to switch between audio devices might fail in a Ubuntu 22.04 session. To address the issue, remove the PulseAudio configuration for the current session user from the VDA and then reopen the session. To remove the PulseAudio configuration, run the `$ rm -r ~/.config/pulse` command.

Browser content redirection

June 20, 2024

Overview

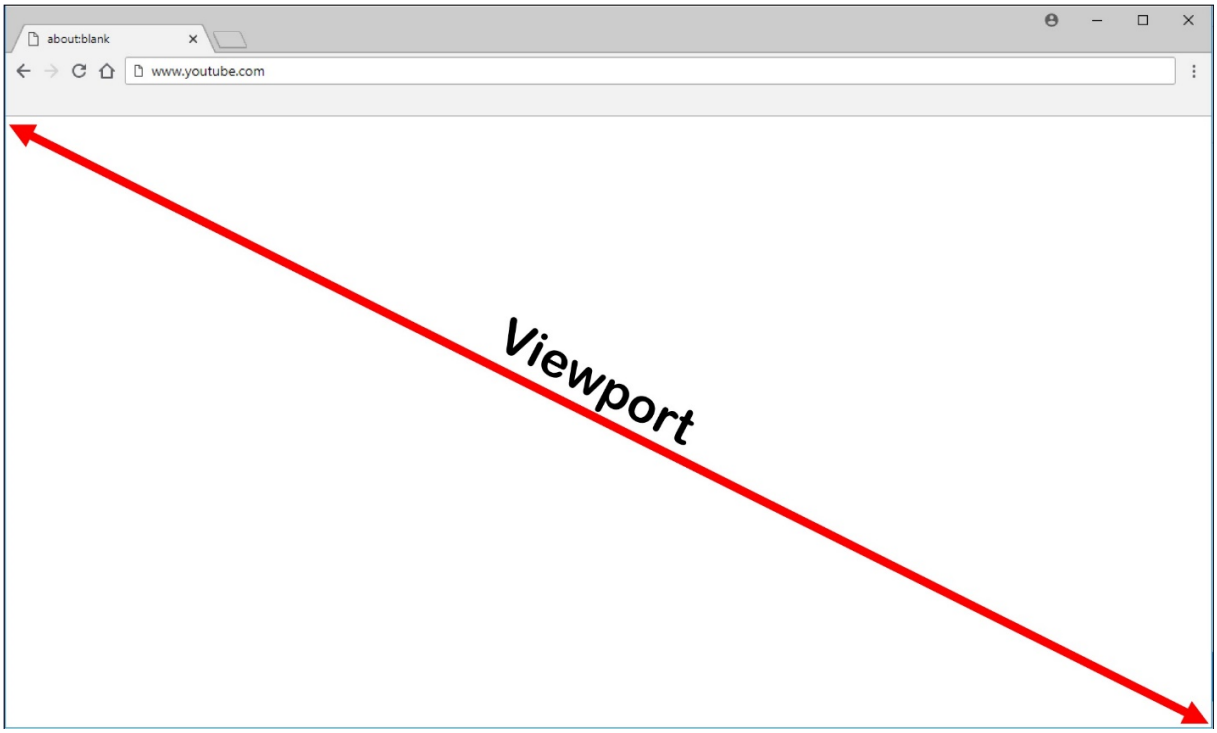
Browser content redirection provides the ability of rendering webpages in the allow list on the client side. This feature uses Citrix Workspace app to instantiate a corresponding rendering engine on the client side, which fetches the HTTP and HTTPS content from the URL.

> **Note:**
>
> The Linux VDA supports browser content redirection in Google Chrome.

This overlay web layout engine runs on the client instead of on the VDA and uses the client CPU, GPU, RAM, and network.

Only the browser viewport is redirected. The viewport is the rectangular area in your browser where content displays. The viewport does not include items such as the address bar, favorites bar, and status bar. Those items are still running in the browser on the VDA.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

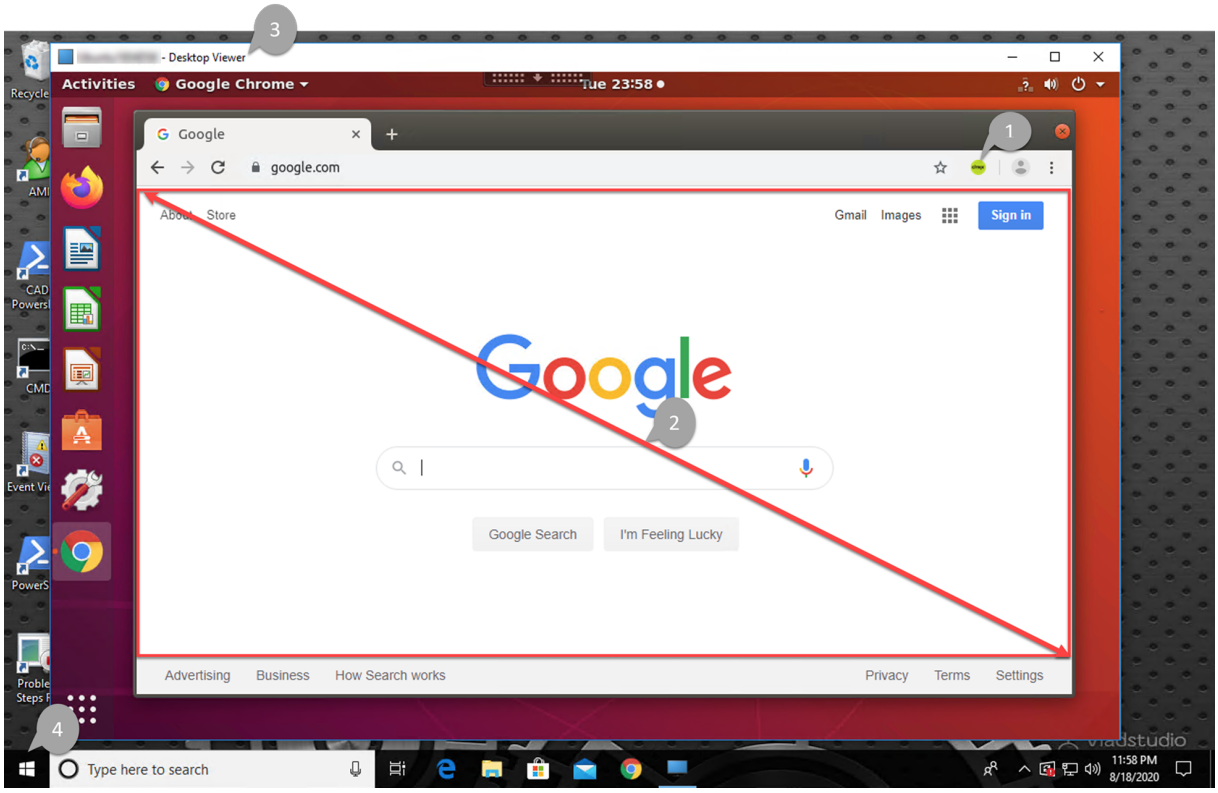


Configure a Studio policy that specifies an Access Control List containing the URLs in the allow list for redirection. Configure a block list that disables redirection for specific URLs.

If a match to a URL is found in an allow list but not in any block list, a virtual channel (CTXCSB) instructs the Citrix Workspace app that a redirection is required and relays the URL. Citrix Workspace app then instantiates a local rendering engine and displays the website.

Citrix Workspace app then blends back the website into the virtual desktop browser content area seamlessly.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



1. Icon of the Citrix browser content redirection extension

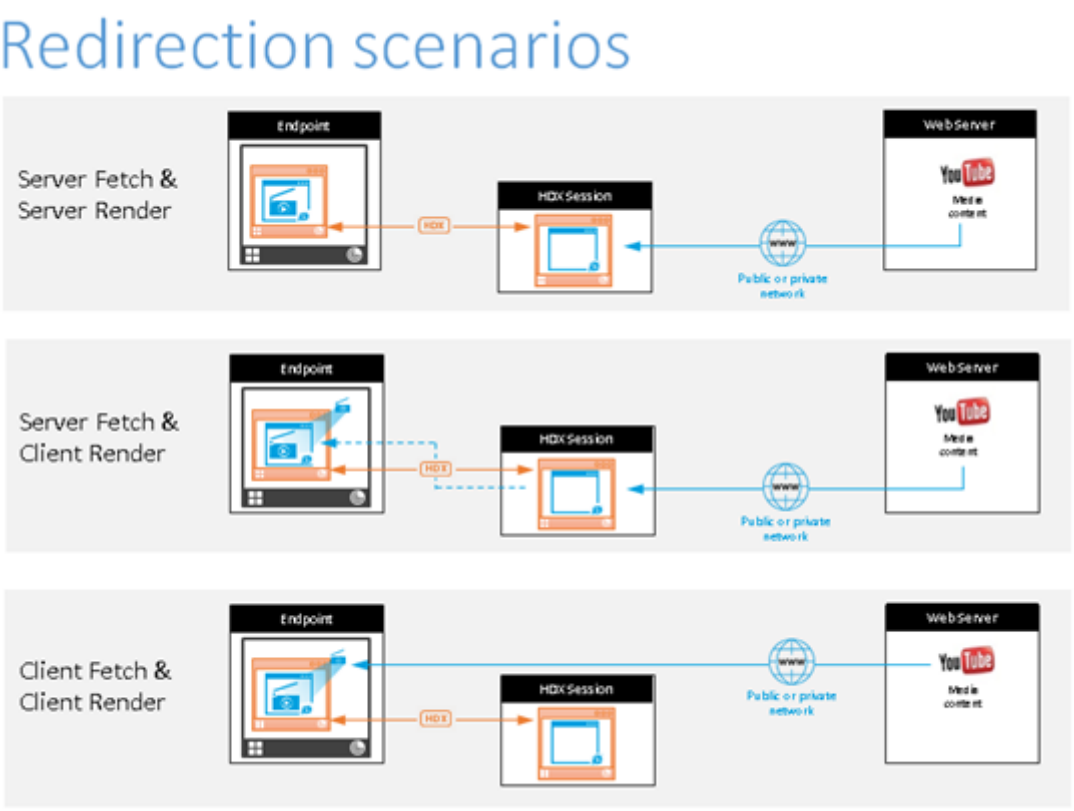
   The color of the extension icon specifies the status of the Chrome extension. It is one of the three colors:

   - Green: Active and connected
   - Gray: Not active/idle on the current tab
   - Red: Broken/Not working

2. Viewport rendered on the client or blended back to the virtual desktop
3. Linux VDA
4. Windows client

Here are scenarios of how the Citrix Workspace app fetches content:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



- **Server fetch and server render:** There is no redirection because you did not add the site to the allow list or the redirection failed. We fall back to rendering the webpage on the VDA and use Thinwire to remote the graphics. Use policies to control the fallback behavior. This scenario causes high CPU, RAM, and bandwidth consumption on the VDA.
- **Server fetch and client render**: Citrix Workspace app contacts and fetches content from the web server through the VDA using a virtual channel (CTXPFWD). This option is useful when the client doesn't have access to the web server (for example, thin clients). It lowers CPU and RAM consumption on the VDA, but bandwidth is consumed on the ICA virtual channel.

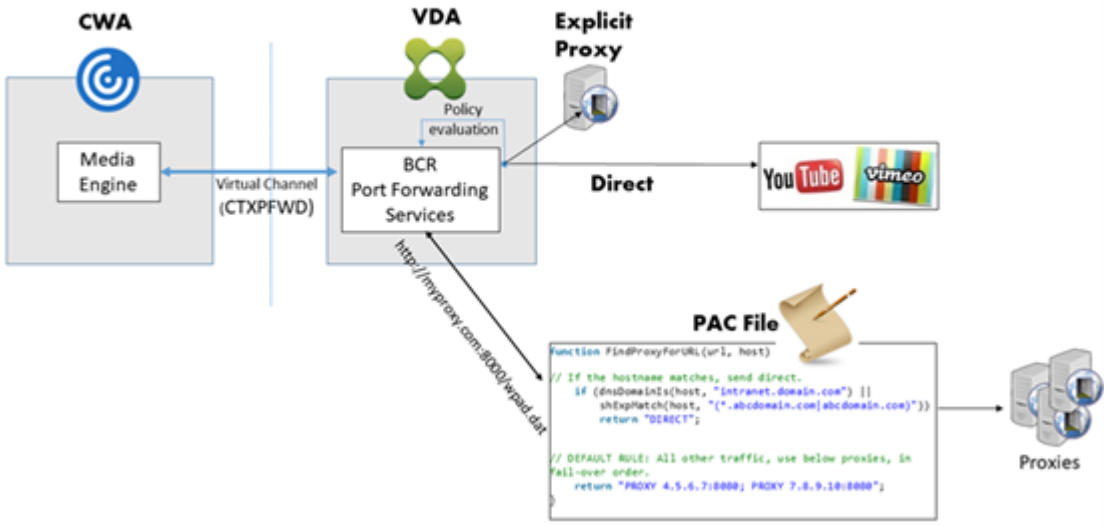
  There are three modes of operation for this scenario. CTXPFWD forwards data to a proxy device that the VDA accesses to gain access to the web server.

  Which policy option to choose:

  - Explicit Proxy - If you have a single explicit proxy in your data center.
  - Direct or Transparent - If you do not have proxies, or if you use transparent proxies.
  - PAC files - If you rely on PAC files so browsers in the VDA can automatically choose the appropriate proxy server for fetching a specified URL.

  For more information, see the **Browser Content Redirection Proxy Configuration** setting later in this article.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



- **Client fetch and client render:** Because the Citrix Workspace app contacts the web server directly, it requires Internet access. This scenario offloads all the network, CPU, and RAM usage from your Citrix Virtual Apps and Desktops site.

  Benefits:

  - Better end user experience (Adaptive Bit Rate (ABR))
  - Reduced VDA resource usage (CPU/RAM/IO)
  - Reduced bandwidth consumption

System requirements

Windows client:

- Citrix Workspace app 1809 for Windows or later

Linux VDA:

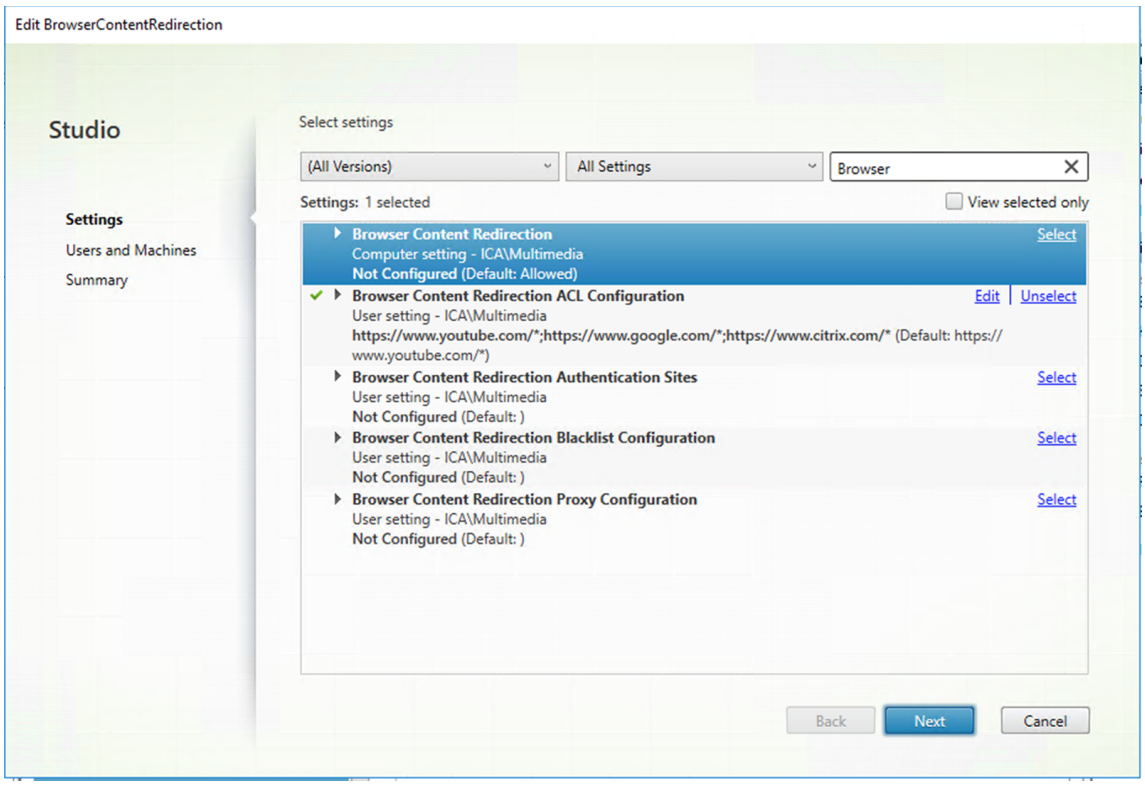
- Browser on the VDA: Google Chrome v66 or later with the Citrix browser content redirection extension added

Configure browser content redirection

To use browser content redirection, configure relevant policies and install the browser content redirection extension in Google Chrome. To do so, complete the following steps:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

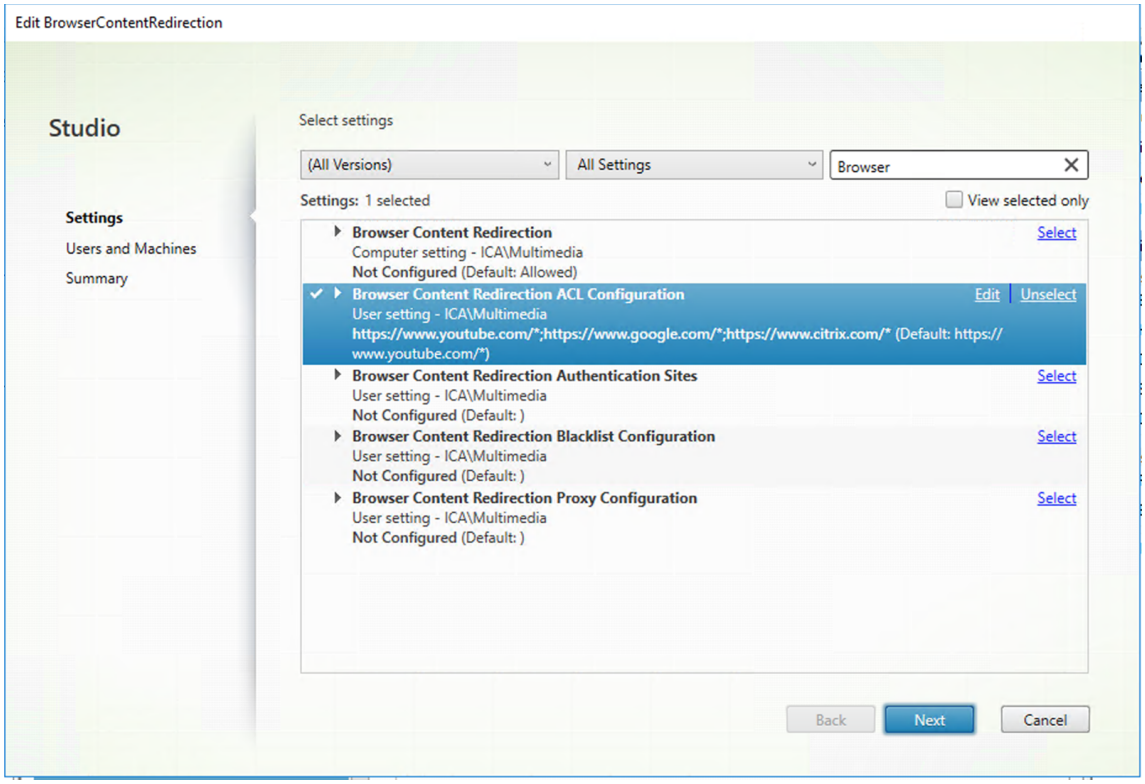
1. In Citrix Studio, set **Browser Content Redirection** to **Allowed** to enable browser content redirection.



2. Specify an allow list of URLs whose content can be redirected to the client and a block list that disables redirection for specific URLs. Configuring a block list is optional.

   The **Browser Content Redirection ACL Configuration** setting specifies an allow list of URLs whose content can be redirected to the client. When specifying URLs, you can use the * wildcard to represent all URL components except the protocol.

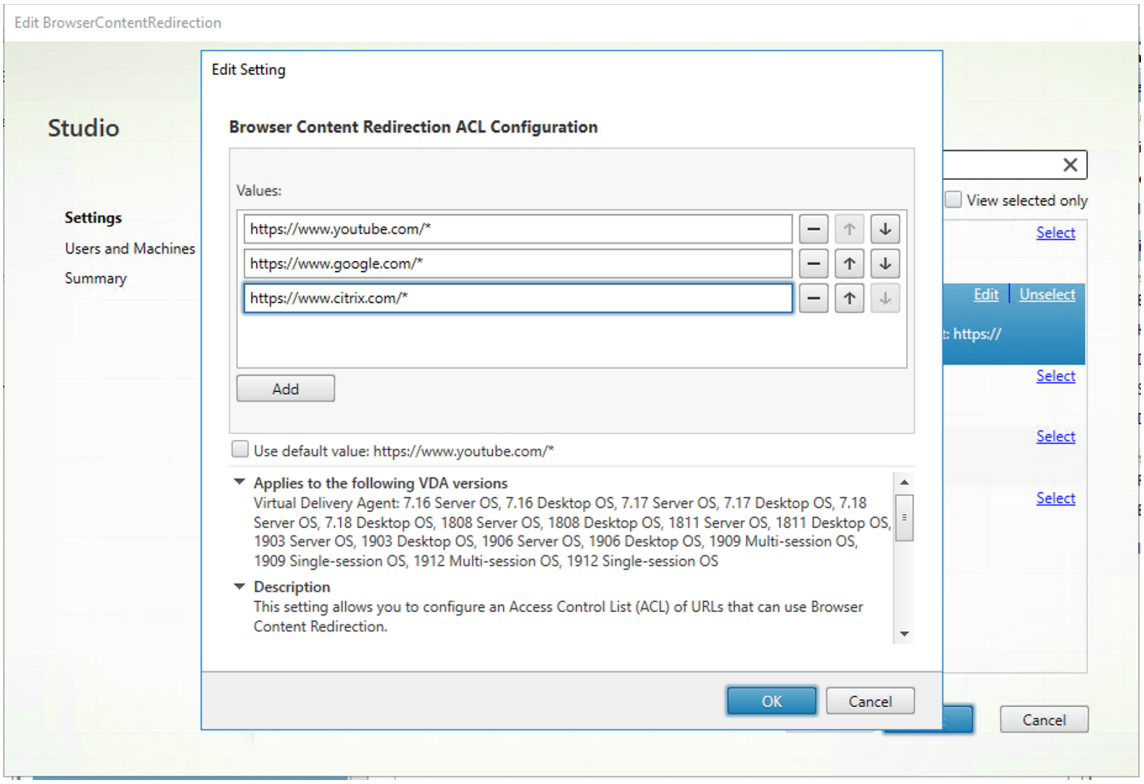
© 1999–2024 Cloud Software Group, Inc. All rights reserved.



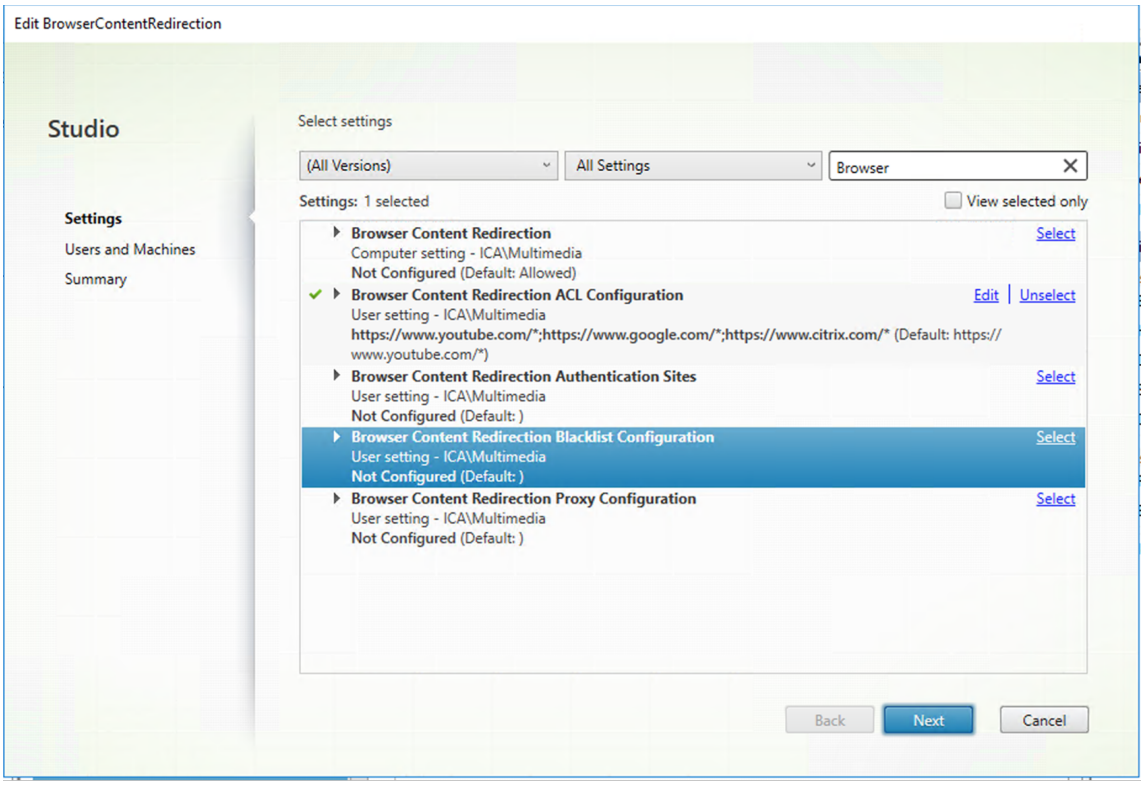
The following are allowed examples:

- `http://www.xyz.com/index.html` (You can achieve better granularity by specifying paths in the URL. For example, if you specify `https://www.xyz.com/sports/index.html`, only the `index.html` page is redirected.)
- `https://www.xyz.com/*`
- `http://www.xyz.com/*videos*`
- `http://*.xyz.com/`
- `http://*.*.com/`

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



The **Browser Content Redirection Blacklist Configuration** setting specifies a block list that disables redirection for specific URLs.



© 1999–2024 Cloud Software Group, Inc. All rights reserved.

3. To enable server fetch and client render, configure the **Browser Content Redirection Proxy Configuration** setting.

   This setting provides configuration options for proxy settings on the VDA for browser content redirection. If enabled with a valid proxy address and port number, PAC/WPAD URL, or Direct/-Transparent setting, Citrix Workspace app always attempts server fetch and client render first. For more information, see the Fallback mechanism.

   If disabled or not configured and using a default value, Citrix Workspace app attempts client fetch and client render.

   By default, this setting is **Prohibited**.

   Allowed pattern for an explicit proxy:

   `http://\<hostname/ip address\>:\<port\>`

   Example:

   `http://proxy.example.citrix.com:80 http://10.10.10.10:8080`

   Allowed patterns for PAC/WPAD files:

   `http://<hostname/ip address>:<port>/<path>/<Proxy.pac>`

   Example: `http://wpad.myproxy.com:30/configuration/pac/Proxy.pac`

   `https://<hostname/ip address>:<port>/<path>/<wpad.dat>`

   Example: `http://10.10.10.10/configuration/pac/wpad.dat`

   Allowed patterns for direct or transparent proxies:

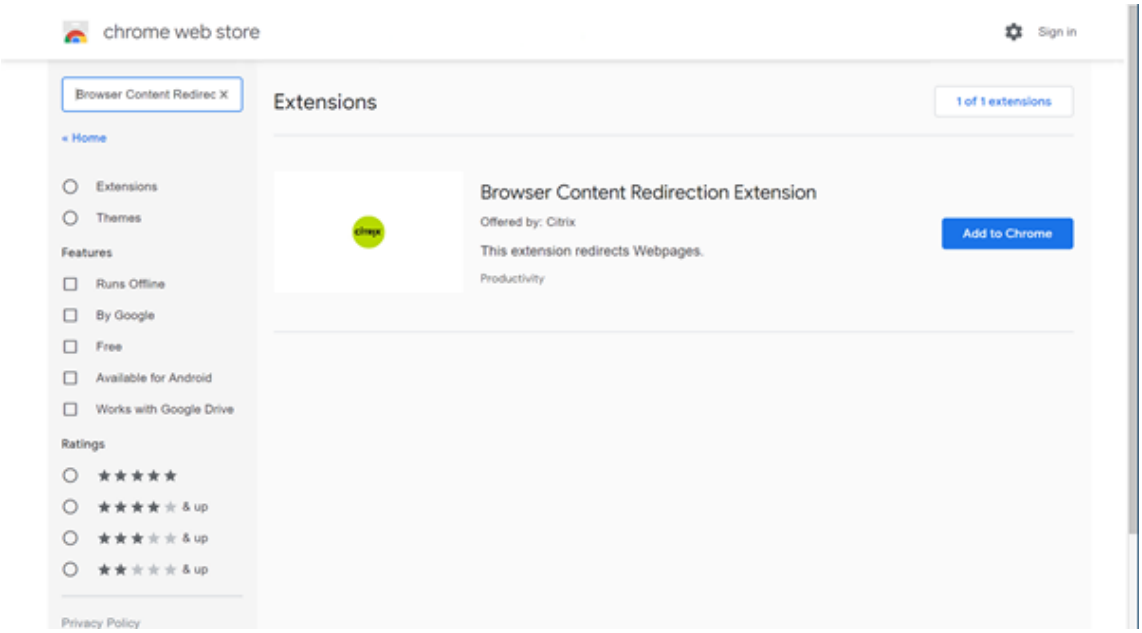
   Type the word **DIRECT** in the policy text box.

   > **Note:**
   >
   > You can also set a proxy by editing the registry value `HKLM\Software\Citrix\HdxMediastream\WebBrowserRedirectionProxyAddress`. Besides, the `HKLM\Software\Citrix\HdxMediastream\AllowNonTlsPacUri` registry value lets you decide whether to allow PAC file downloads through HTTP. The default value is 0, which means HTTP is not allowed.
   >
   > Registries override options for policy settings. For a list of the relevant registry keys, see Browser content redirection registry key overrides.

4. Click **Add to Chrome** on the VDA to add the Citrix browser content redirection extension from the Chrome Web Store. Doing so helps the browser on the VDA to detect whether a URL (being navigated to) matches an allow list or a block list.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Important:**
>
> The extension is not required on the client. Add it only on the VDA.
>
> Chrome extensions are installed on a per-user basis. Updating a golden image to add or remove an extension is not required.



Fallback mechanism

If you enable the **Browser Content Redirection Proxy Configuration** policy, Citrix Workspace app attempts server fetch and client render. If server fetch and client render fails, it falls back to client fetch and client render. If the client machine doesn't have access to the web server, the browser on the VDA can then reload and render the page on the server (server fetch and server render).

Browser content redirection registry key overrides

> **Warning:**
>
> Editing the registry incorrectly can cause serious problems that might require you to reinstall your operating system. Citrix can't guarantee that problems resulting from the incorrect use of the Registry Editor can be solved. Use the Registry Editor at your own risk. Be sure to back up the registry before you edit it.

`HKLM\Software\Citrix\HdxMediastream`

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Name	Type	Value
WebBrowserRedirection	DWORD	1=Allowed, 0=Prohibited
WebBrowserRedirectionAcl	REG_MULTI_SZ	/
WebBrowserRedirectionProxyAddress	REG_SZ	If you set it to any of the following modes, server fetch client render is enabled: **Explicit Proxy** - If you have a single explicit proxy in your data center. **Direct or Transparent** - If you do not have proxies, or if you use transparent proxies. **PAC files** - If you rely on PAC files so browsers in the VDA can automatically choose the appropriate proxy server for fetching a specified URL.
WebBrowserRedirectionBlacklist	REG_MULTI_SZ	/
AllowNonTlsPacUri	DWORD	Determines whether to allow PAC file downloads through HTTP. The default value is 0, which means HTTP is not allowed. If you set it to 1, **HDXWebProxy.exe** can download PAC files over HTTP (not strictly over HTTPS).

HDX webcam video compression

June 20, 2024

Overview

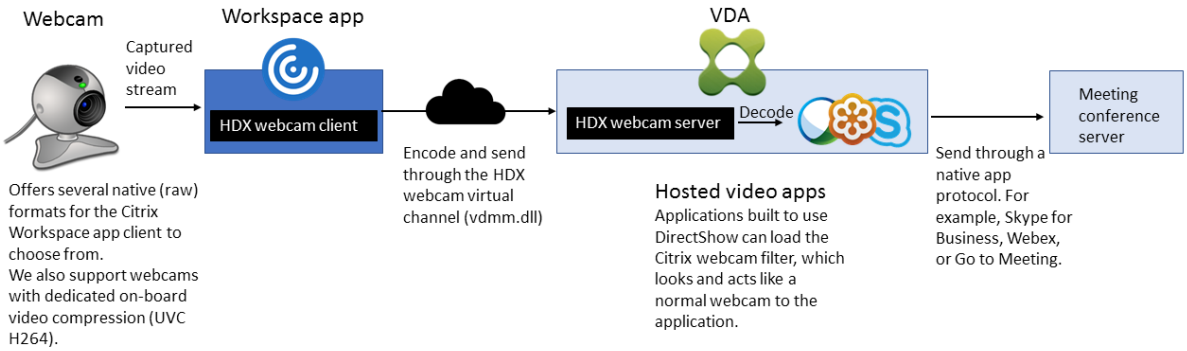
Users of video conferencing applications running in Linux VDA sessions can now use their webcams with HDX webcam video compression. The feature is enabled by default. We recommend you always

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

use HDX webcam video compression if possible.

HDX webcam video compression is also called **Optimized** webcam mode. This type of webcam video compression sends the H.264 video directly to the video conferencing application running in the virtual session. HDX webcam video compression uses the multimedia framework technology that is part of the client operating system to intercept video from capture devices, transcode, and compress it. Manufacturers of capture devices supply the drivers that plug into the OS kernel streaming architecture.

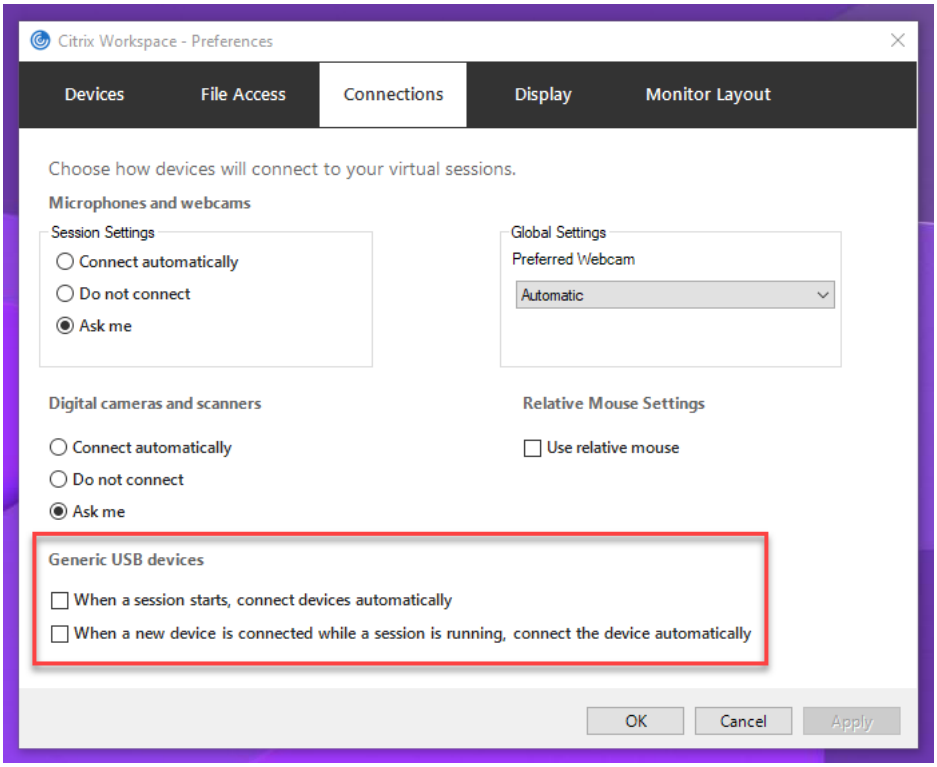
The client handles communication with the webcam. The client then sends the video only to the server that can display it properly. The server doesn't deal directly with the webcam, but its integration gives you the same experience in your desktop. Workspace app compresses the video to save bandwidth and provide better resiliency on WAN scenarios.



Note:

- The feature is not available for Azure machines because the **videodev** kernel module that the feature depends on is missing on Azure machines.
- The feature supports only H.264 videos from your Citrix Workspace app client.
- The supported webcam resolution ranges between 48x32 and 1920x1080.
- Do not choose **Generic USB devices** from your Citrix Workspace app toolbar when you are using a webcam. Otherwise, unexpected issues might occur.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



Supported Citrix Workspace app

HDX webcam video compression supports the following versions of Citrix Workspace app:

Platform	Processor
Citrix Workspace app for Windows	Citrix Workspace app for Windows supports webcam video compression for 32-bit and 64-bit apps on XenApp and XenDesktop 7.17 and later. On earlier versions, Citrix Workspace app for Windows supports only 32-bit apps.
Citrix Workspace app for Chrome	Because some ARM Chromebooks don't support H.264 encoding, only 32-bit apps can use the optimized HDX webcam video compression.

Fully tested webcams

Different webcams offer different frame rates and have different levels of brightness and contrast. Citrix uses the following webcams for initial feature validation:

- Logitech HD Webcam C270

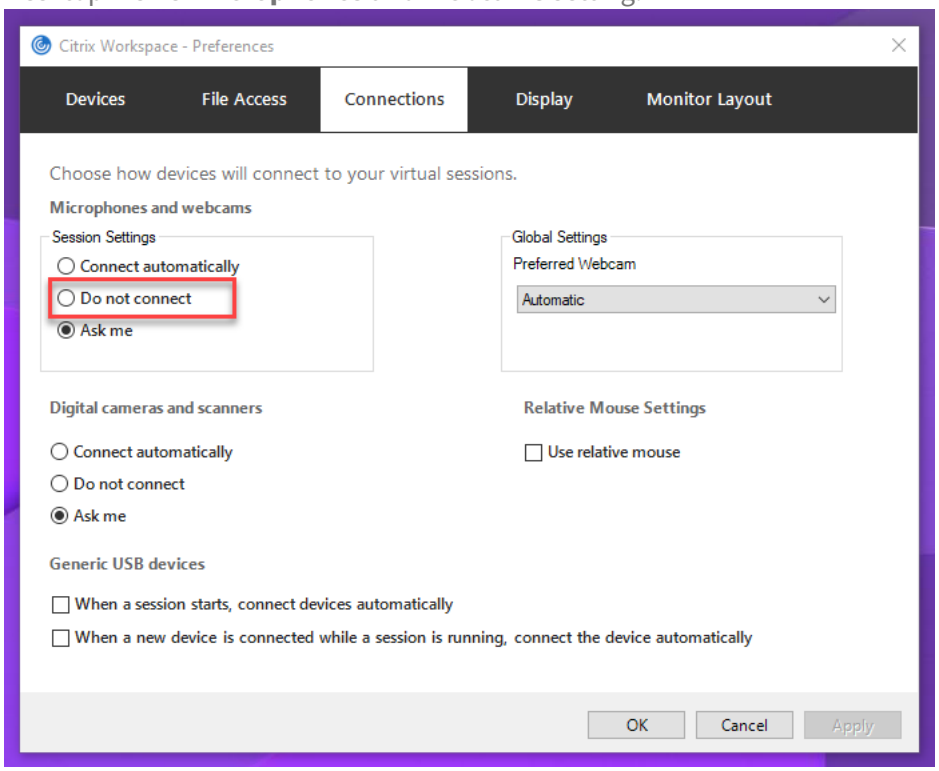
© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- Logitech Webcam C930e
- Microsoft-LifeCam-HD3000

Configuration

This feature is enabled by default. To use it, complete the following verification and configuration:

> **Tip:**
>
> Citrix Workspace app users can override the default setting by choosing **Do not connect** for the Desktop Viewer **Microphones and webcams** setting.



1. After your VDA installation completes, verify that your VDA can register with the Delivery Controller and the published Linux desktop sessions can be launched successfully using Windows credentials.
2. Ensure that your VDA has Internet access and then run the `sudo /opt/Citrix/VDA/sbin/ctxwcamcfg.sh` command to complete your webcam configurations. If your VDA does not have Internet access, go to step 3.

   > **Note:**
   >
   > Kernel mismatch might happen between `uname -r` and kernel headers. The mismatch

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> causes the ctxwcamcfg.sh script to fail. To use HDX webcam video compression properly, run **sudo apt-get dist-upgrade**, restart the VDA, and then rerun the ctxwcamcfg.sh script.

If your VDA is deployed on Debian, ensure that it is running on the latest kernel version. Otherwise, run the following commands to update to the latest kernel version:

```
sudo apt-get update
sudo apt-get dist-upgrade
sudo reboot
```

If your VDA is deployed on SUSE 15, run the following commands to update to the latest kernel version and to reboot:

```
zypper up kernel-default
reboot
```

The ctxwcamcfg.sh script helps to:

a) Install the `kernel-devel` and Dynamic Kernel Module Support (DKMS) programs on your VDA.

- `kernel-devel` is used to build a virtual webcam kernel module of the corresponding version.
- DKMS is used to dynamically manage the virtual webcam kernel module.

> **Note:**
>
> When installing the preceding programs on RHEL, Rocky Linux, and CentOS, the `ctxwcamcfg.sh` script installs and enables the following repositories on your VDA:
>
> - Extra Packages for Enterprise Linux (EPEL)
> - RPM Fusion

b) Download the `v4l2loopback` open source code from https://github.com/umlaeute/v4l2loopback and use DKMS to manage `v4l2loopback`.
`v4l2loopback` is a kernel module that allows you to create V4L2 loopback devices.

c) Run the `sudo systemctl restart ctxwcamsd` command. The Linux VDA's webcam service - `ctxwcamsd` - restarts and loads the `v4l2loopback` kernel module for the HDX webcam video compression feature.

3. If your VDA does not have Internet access, build the v4l2loopback kernel module on another machine and then copy it to your VDA.

a) Prepare a machine that has Internet access and has the same kernel version with your VDA. The `uname -r` command helps to find kernel versions.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

b) On the machine, run the `sudo mkdir -p /var/xdl` command.

c) Copy `/var/xdl/configure_*` from your VDA to the machine under `/var/xdl/`.

d) On the machine, run the `sudo /opt/Citrix/VDA/sbin/ctxwcamcfg.sh` command to build the kernel module. If the command runs successfully, it creates a v4l2loopback.ko file under the `/var/lib/dkms/v4l2loopback/1.81b8df79107d1fbf392fdcbaa051bd227a9c94c1/$(uname -r)/x86_64/module/` path. Ignore errors that might occur when you run the ctxwcamcfg.sh script.

e) Copy v4l2loopback.ko from the machine to your VDA and place it under `/opt/Citrix/VDA/lib64/`.

f) On your VDA, run the `sudo systemctl restart ctxwcamsd` command to restart the webcam service and load the v4l2loopback kernel module.

Non-domain-joined Linux VDAs

June 20, 2024

Overview

Non-domain-joined VDAs obliterate the need to join VDAs to Active Directory domains for VDA and user authentication. When creating a non-domain-joined VDA, you generate a public-private key pair for registering the VDA to the cloud control plane. Thus, joining an Active Directory domain is no longer required. When a user launches a session from the non-domain-joined VDA, the VDA creates a local mapping account using the user name that the user uses to log on to Citrix Workspace app. The VDA assigns a random password that the local mapping account uses for SSO and session reconnection. If you change the random password, SSO and session reconnection fail. To disable SSO, see Non-SSO authentication.

> **Important:**
>
> - Non-domain-joined VDAs are supported for Citrix DaaS.
>   - Your control plane must be deployed over Citrix DaaS.
>   - You can deploy non-domain-joined VDAs in a public cloud or on-premises data center. The control plane in Citrix DaaS manages non-domain-joined VDAs.
>   - To create non-domain-joined VDAs, you must enable Rendezvous V2. Cloud Connectors are required only if you plan to provision machines on on-premises hypervisors or if you want to use Active Directory as the identity provider in Workspace.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> - To create non-domain joined VDAs, you can use both Machine Creation Services (MCS) and easy install. For more information, see Create non-domain-joined Linux VDAs using MCS and Create a non-domain-joined Linux VDA using easy install.
>   - MCS doesn't support bare metal servers.

Features available for non-domain-joined Linux VDAs

Create local users with specified attributes on non-domain-joined VDAs

When you open a session hosted on a non-domain-joined VDA, the VDA automatically creates a local user with default attributes. The VDA creates the local user based on the user name that you used to log on to Citrix Workspace app. You can also specify user attributes including the user's User Identifier (UID), Group ID (GID), home directory, and log-in shell. To use this feature, complete the following steps:

1. Run the following command to enable the feature:

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\
   VirtualDesktopAgent\LocalMappedAccount" -t "REG_DWORD" -v "
   CreateWithUidGid" -d "0x00000001" --force
```

2. Specify the following attributes in the `/var/xdl/getuidgid.sh` script under the installation path of the VDA:

Attribute	Required or optional	Description
`uid`	Required	A User Identifier (UID) is a number assigned by Linux to each user on the system. It determines which system resources that the user can access.
`gid`	Required	A Group Identifier (GID) is a number used to represent a specific group.
`homedir`	Optional	The Linux home directory is a directory for a particular user.
`shell`	Optional	A login shell is a shell given to a user upon the login to their user account.

The following is an example of the `getuidgid.sh` script:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> Make sure that the attributes specified in the script are valid.

```
#!/bin/bash

################################################################################

#
# Citrix Virtual Apps & Desktops For Linux Script: Get uid and gid
    for the user
#
# Copyright (c) Citrix Systems, Inc. All Rights Reserved.
#

export LC_ALL="en_US.UTF-8"

function get_uid_gid_for_user()
{

echo "uid:12345"
echo "gid:1003"
echo "homedir:/home/$1"
echo "shell:/bin/sh"
 }


get_uid_gid_for_user $1
```

Non-SSO authentication

By default, the Linux VDA has single sign-on (SSO) enabled. Users log on to Citrix Workspace app and to VDA sessions using one set of credentials.
To have users log on to VDA sessions using a different set of credentials, disable SSO on the Linux VDA. For more information, see Non-SSO authentication.

Authentication with Azure Active Directory

The non-domain-joined VDAs that you deploy in Azure integrate with the AAD identity service to provide user authentication. For more information, see Authentication with Azure Active Directory.

Rendezvous V2

Non-domain-joined VDAs are supported for using Rendezvous V2 to bypass Citrix Cloud Connectors. For more information, see Rendezvous V2.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Policy support list

June 20, 2024

Linux VDA policy support list

Studio Policy	Key Name	Type	Module	Default Value
Use local time of client	UseLocalTimeOfClient	User	ICA\Time Zone Control	**Use server time zone**
ICA round trip calculation	IcaRoundTripCheckEnabled	Computer	ICA\End User Monitoring	Enabled (1)
ICA round trip calculation interval	IcaRoundTripCheckPeriod	Computer	ICA\End User Monitoring	15
ICA round trip calculations for idle connections	IcaRoundTripCheckWhenIdle	Computer	ICA\End User Monitoring	Disabled (0)
Overall session bandwidth limit	LimitOverallBw	User	ICA\Bandwidth	0
Audio redirection bandwidth limit	LimitAudioBw	User	ICA\Bandwidth	0
Audio redirection bandwidth limit percent	LimitAudioBwPercent	User	ICA\Bandwidth	0
Client USB device redirection bandwidth limit	LimitUSBBw	User	ICA\Bandwidth	0

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Studio Policy	Key Name	Type	Module	Default Value
Client USB device redirection bandwidth percent	LimitUSBBwPercent	User	ICA\Bandwidth	0
File redirection bandwidth limit	LimitCdmBw	User	ICA\Bandwidth	0
File redirection bandwidth limit percent	LimitCdmBwPercent	User	ICA\Bandwidth	0
Printer redirection bandwidth limit	LimitPrinterBw	User	ICA\Bandwidth	0
Printer redirection bandwidth limit percent	LimitPrinterBwPercent	User	ICA\Bandwidth	0
WebSockets connections	AcceptWebSocketsConnections	Computer	ICA\WebSockets	Prohibited
WebSockets port number	WebSocketsPort	Computer	ICA\WebSockets	8008
WebSockets trusted origin server list	WSTrustedOriginServerList	Computer	ICA\WebSockets	*
ICA keep alives	SendICAKeepAlives	Computer	**ICA keep alive**	Do not send ICA keep alive messages (0)
ICA keep alive timeout	ICAKeepAliveTimeout	Computer	**ICA keep alive**	60 seconds
ICA listener port number	IcaListenerPortNumber	Computer	ICA	1494
HDX adaptive transport	HDXoverUDP	Computer	ICA	Preferred(2)

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Studio Policy	Key Name	Type	Module	Default Value
Rendezvous protocol	RendezvousProtocol	Computer	ICA	Prohibited
Session reliability connections	AcceptSessionReliabilityConnections	Computer	ICA\Session Reliability	Allowed(1)
Reconnection UI transparency level	ReconnectionUiTransparencyLevel	Computer	ICA\Auto Client Reconnect	80%
Session reliability port number	SessionReliabilityPort	Computer	ICA\Session Reliability	2598
Session reliability timeout	SessionReliabilityTimeout	Computer	ICA\Session Reliability	180 s
Auto Client Reconnect	AllowAutoClientReconnect	User	ICA\Auto Client Reconnect	Allowed (1)
Client audio redirection	AllowAudioRedirection	User	Audio	Allowed (1)
Client printer redirection	AllowPrinterRedir	User	Printing	Allowed (1)
Auto-create PDF Universal Printer	AutoCreatePDFPrinter	User	Printing	Disabled (0)
Printer driver mapping and compatibility	DriverMappingList	User	Printing	`"Microsoft XPS Document Writer *, Deny;Send to Microsoft OneNote *, Deny"`
Client clipboard redirection	AllowClipboardRedir	User	Clipboard	Allowed (1)

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Studio Policy	Key Name	Type	Module	Default Value
Limit clipboard client to session transfer size	LimitClipboardTransferC2H	User	ICA	Disabled (0)
Limit clipboard session to client transfer size	LimitClipboardTransferH2C	User	ICA	Disabled (0)
Clipboard redirection bandwidth limit	LimitClipbdBW	User	ICA\Bandwidth	0
Clipboard redirection bandwidth limit percent	LimitClipbdBWPercent	User	ICA\Bandwidth	0
Restrict client clipboard write	RestrictClientClipboardWrite	User	Clipboard	Disabled (0)
Client clipboard write allowed formats	ClientClipboardWriteAllowedFormats	User	Clipboard	N/A
Restrict session clipboard write	RestrictSessionClipboardWrite	User	Clipboard	Disabled (0)
Session clipboard write allowed formats	SessionClipboardWriteAllowedFormats	User	Clipboard	N/A
Client USB device redirection	AllowUSBRedir	User	USB	Prohibited (0)
Client USB device redirection rules	USBDeviceRules	User	USB	“\0”
Moving image compression	MovingImageCompression	SessionConfiguration	Thinwire	Enabled (1)
Extra color compression	ExtraColorCompression	User	Thinwire	Disabled (0)

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Studio Policy	Key Name	Type	Module	Default Value
Target minimum frame rate	TargetedMinimumFramesPerSecond	User	Thinwire	10 fps
Target frame rate	FramesPerSecond	User	Thinwire	30 fps
Visual quality	VisualQuality	User	Thinwire	Medium (3)
Use video codec for compression	VideoCodec	User	Thinwire	Use when preferred (3)
Use hardware encoding for video codec	UseHardwareEncodingForVideoCodec	User	Thinwire	Enabled (1)
Allow visually lossless compression	AllowVisuallyLosslessCompression	User	Thinwire	Disabled (0)
Optimize for 3D graphics workload	OptimizeFor3dWorkload	User	Thinwire	Disabled (0)
Preferred color depth for simple graphics	PreferredColorDepth	User	Thinwire	24 bits per pixel(1)
Graphic status indicator	DisplayLosslessIndicator	User	Thinwire	Disabled (0)
Audio quality	SoundQuality	User	Audio	High –high definition audio (2)
Client microphone redirection	AllowMicrophoneRedir	User	Audio	Allowed (1)
Maximum number of sessions	MaximumNumberOfSessions	Computer	Load Management	250
Concurrent logons tolerance	ConcurrentLogonsTolerance	Computer	Load Management	2

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Studio Policy	Key Name	Type	Module	Default Value
Enable auto update of Controllers	EnableAutoUpdateOfControllers	Computer	Virtual Delivery Agent Settings	Allowed (1)
Clipboard selection update mode	ClipboardSelectionUpdateMode	User	Clipboard	3
Primary selection update mode	PrimarySelectionUpdateMode	User	Clipboard	3
Max speex quality	MaxSpeexQuality	User	Audio	5
Auto connect client drives	AutoConnectDrives	User	File redirection\CDM	Enabled (1)
Client optical drives	AllowCdromDrives	User	File redirection\CDM	Allowed (1)
Client fixed drives	AllowFixedDrives	User	File redirection\CDM	Allowed (1)
Client floppy drives	AllowFloppyDrives	User	File redirection\CDM	Allowed (1)
Client network drives	AllowNetworkDrives	User	File redirection\CDM	Allowed (1)
Client removable drives	AllowRemoveableDrives	User	File redirection\CDM	Allowed (1)
Client drive redirection	AllowDriveRedir	User	File redirection\CDM	Allowed (1)
Read-only client drive access	ReadOnlyMappedDrive	User	File redirection\CDM	Disabled (0)
Automatic keyboard display	AllowAutoKeyboardPopup	User	MRVC	Disabled (0)

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Studio Policy	Key Name	Type	Module	Default Value
Allow file transfer between desktop and client	AllowFileTransfer	User	File Transfer	Allowed
Download file from desktop	AllowFileDownload	User	File Transfer	Allowed
Upload file to desktop	AllowFileUpload	User	File Transfer	Allowed
Session idle timer	EnableSessionIdleTimer	User	Session Timers	Enabled (1)
Session idle timer interval	SessionIdleTimerInterval	User	Session Timers	1440 minutes
Disconnected session timer	EnableSessionDisconnectTimer	User	Session Timers	Disabled (0)
Disconnected session timer interval	SessionDisconnectTimerPeriod	User	Session Timers	1440 minutes
Browser Content Redirection	WebBrowserRedirection	Computer	ICA\Multimedia	Allowed
Browser Content Redirection ACL Configuration	WebBrowserRedirectionAcl	User	ICA\Multimedia	`https://www.youtube.com/*`
Browser Content Redirection Authentication Sites	WebBrowserRedirectionAuthenticationSites	User	ICA\Multimedia	Null

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Studio Policy	Key Name	Type	Module	Default Value
Browser Content Redirection Blacklist Configuration	WebBrowserRedirectionBlacklist	User	ICA\Multimedia	Null
Browser Content Redirection Proxy Configuration	WebBrowserRedirectionProxy	User	ICA\Multimedia	Null
Enable session watermark	EnableSessionWatermark	User	ICA\Session Watermark\Watermark Content	Disabled
Include client IP address	IncludeClientIPAddress	User	ICA\Session Watermark\Watermark Content	Disabled
Include connection time	IncludeConnectTime	User	ICA\Session Watermark\Watermark Content	Disabled
Include logon user name	IncludeLogonUsername	User	ICA\Session Watermark\Watermark Content	Enabled
Include VDA host name	IncludeVDAHostname	User	ICA\Session Watermark\Watermark Content	Enabled
Include VDA IP address	IncludeVDAIPAddress	User	ICA\Session Watermark\Watermark Content	Disabled

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Studio Policy	Key Name	Type	Module	Default Value
Session watermark style	WatermarkStyle	User	ICA\Session Watermark\Watermark Style	Multiple
Watermark transparency	WatermarkTransparency	User	ICA\Session Watermark\Watermark Style	17
Watermark custom text	WatermarkCustomText	User	ICA\Session Watermark\Watermark Content	A valid custom text configuration can look like: <font=Arial Unicode MS><domain><newline><fontsize=47><username><newline><fontzoom=120>Citrix<newline><style=tile>Najing <clientip><newline><rotation=45><serverip><position=bottomleft>. For more information, see Session watermark.
Session metrics collection	UsageDataCollectionThroughClient	Computer	ICA	Enabled (1)

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Studio Policy	Key Name	Type	Module	Default Value
Dynamic windows preview	EnableDynamicPreview	Computer	ICA\Graphics	This setting enables or disables the display of seamless windows. With this setting enabled, you can preview the onscreen content in the windows of published apps. For example, you can open multiple windows of Google Chrome sessions using Citrix Workspace app for Windows and then hover over the Chrome icon on the taskbar to preview the content. For more information, see Dynamic windows preview. By default, this setting is **Enabled (1)**.

> **Note:**
>
> Only the Windows Virtual Delivery Agent (VDA) supports audio over User Datagram Protocol

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

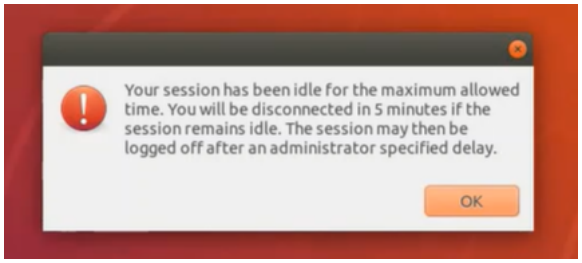
> (UDP). The Linux VDA does not. For more information, see Audio over User Datagram Protocol (UDP) Real-time Transport.

You can use the following Citrix policy settings to configure session connection timers in Citrix Studio:

- **Session idle timer**: Determines whether to enforce a time limit for idle sessions.
- **Session idle timer interval**: Sets a time limit for idle sessions. If **Session idle timer** is set to **Enabled** and an active session has not received user input during the set time, the session disconnects.
- **Disconnected session timer**: Determines whether to enforce a time limit for disconnected sessions.
- **Disconnected session timer interval**: Sets an interval before a disconnected session is logged off.

When you update any of the policy settings, ensure that they are consistent across your deployment.

A warning message appears when your time limit for idle sessions expires. See the following screen capture for an example. Pressing **OK** closes the warning message but cannot keep your session active. To keep your session active, provide user input to reset the idle timer.



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

- MaxSpeexQuality

  Value (integer): [0–10]

  Default value: 5

  Details:

  Audio redirection encodes audio data with the Speex codec when audio quality is medium or low (see the policy Audio quality). Speex is a lossy codec, which means that it achieves compression at the expense of fidelity of the input speech signal. Unlike some other speech codecs, it is possible to control the tradeoff made between quality and bit rate. The Speex encoding process is controlled most of the time by a quality parameter that ranges from 0 to 10. The higher the quality is, the higher the bit rate.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

The max Speex quality chooses the best Speex quality to encode audio data according to audio quality and bandwidth limit (see the policy Audio redirection bandwidth limit). If the audio quality is medium, the encoder is in wide band mode, which means a higher sampling rate. If the audio quality is low, the encoder is in narrow band mode, which means a lower sampling rate. The same Speex quality has different bit rates in different modes. The best Speex quality is when the largest value meets the following conditions:

- It is equal to or less than the max Speex quality.
- Its bit rate is equal to or less than the bandwidth limit.

Related Settings: Audio quality, Audio redirection bandwidth limit

- PrimarySelectionUpdateMode

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

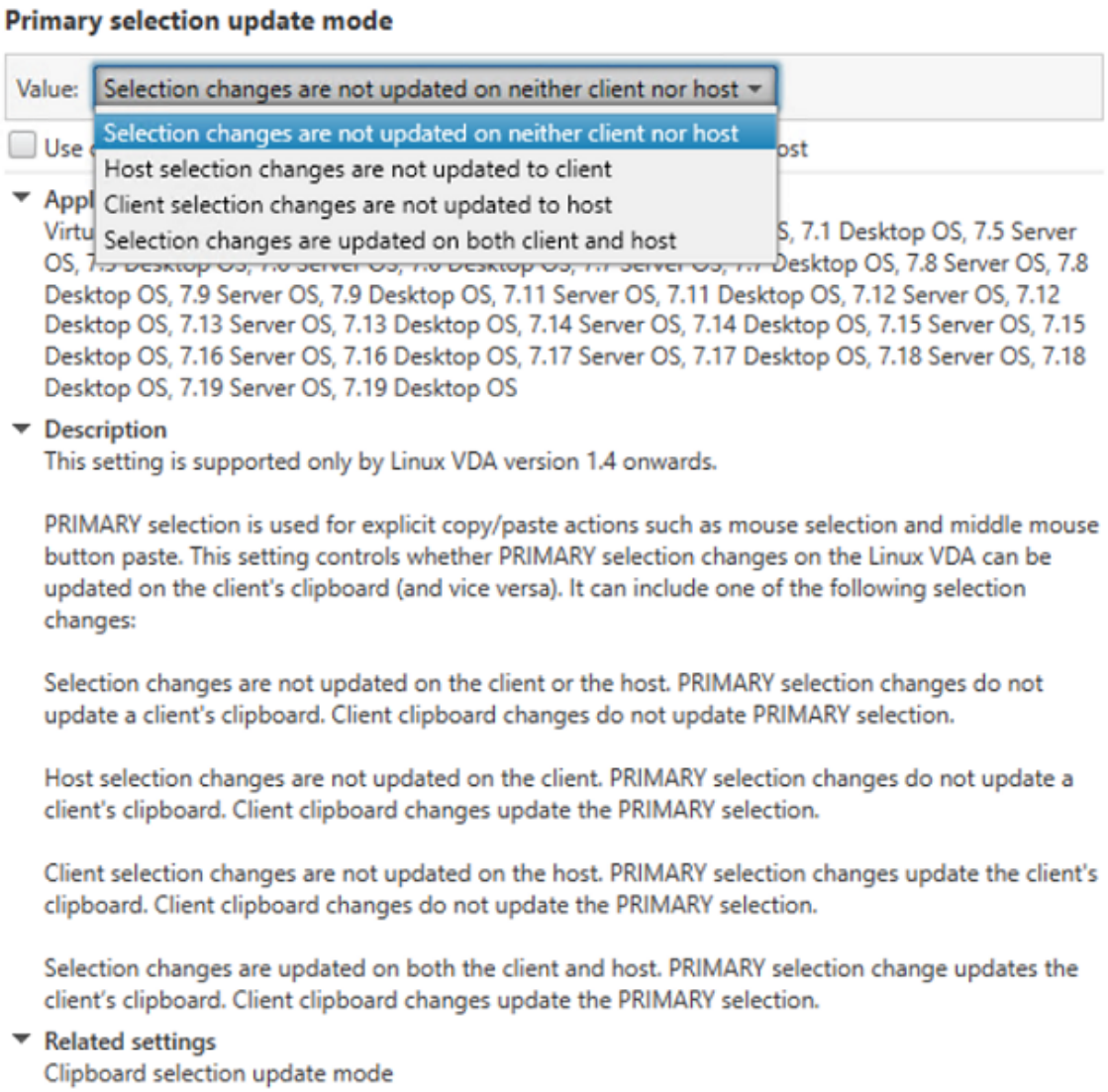
  Default value: 3

  Details:

  Primary selection is used when you select data and paste it by pressing the middle mouse button.

  This policy controls whether primary selection changes on the Linux VDA and the client can update the clipboard on each other. There are four value options:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



- **Selection changes are not updated on neither client nor host**
  Primary selection changes on the Linux VDA do not update the clipboard on the client. Primary selection changes on the client do not update the clipboard on the Linux VDA.
- **Host selection changes are not updated to client**
  Primary selection changes on the Linux VDA do not update the clipboard on the client. Primary selection changes on the client update the clipboard on the Linux VDA.
- **Client selection changes are not updated to host**
  Primary selection changes on the Linux VDA update the clipboard on the client. Primary selection changes on the client do not update the clipboard on the Linux VDA.
- **Selection changes are updated on both client and host**
  Primary selection changes on the Linux VDA update the clipboard on the client. Primary selection changes on the client update the clipboard on the Linux VDA. This option is the

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

default value.

Related Setting: Clipboard selection update mode

- ClipboardSelectionUpdateMode

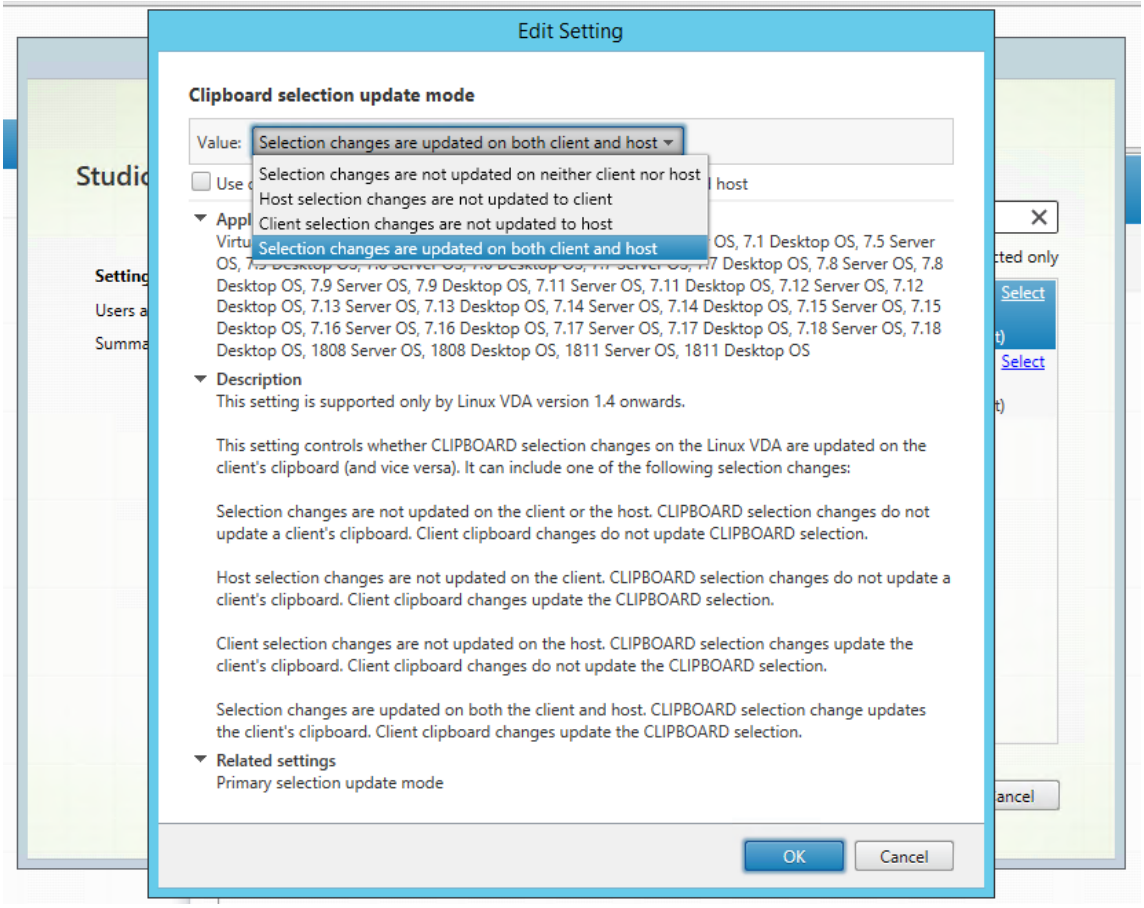
  Value (enum): [0, 1, 2, 3]

  Default value: 3

  Details:

  Clipboard selection is used when you select some data and explicitly request it to be "copied" to the clipboard, such as by selecting "Copy"from the shortcut menu. Clipboard selection is primarily used in connection with Microsoft Windows clipboard operations while primary selection is unique to Linux.

  This policy controls whether clipboard selection changes on the Linux VDA and the client can update the clipboard on each other. There are four value options:



  - **Selection changes are not updated on neither client nor host**
    Clipboard selection changes on the Linux VDA do not update the clipboard on the client. Clipboard selection changes on the client do not update the clipboard on the Linux VDA.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- **Host selection changes are not updated to client**
  Clipboard selection changes on the Linux VDA do not update the clipboard on the client. Clipboard selection changes on the client update the clipboard on the Linux VDA.
- **Client selection changes are not updated to host**
  Clipboard selection changes on the Linux VDA update the clipboard on the client. Clipboard selection changes on the client do not update the clipboard on the Linux VDA.
- **Selection changes are updated on both client and host**
  Clipboard selection changes on the Linux VDA update the clipboard on the client. Clipboard selection changes on the client update the clipboard on the Linux VDA. This option is the default value.

Related Setting: Primary selection update mode

> **Note:**
>
> The Linux VDA supports both clipboard selection and primary selection. To control the copy and paste behaviors between the Linux VDA and the client, we recommend that you set both clipboard selection update mode and primary selection update mode to the same value.

Printing

June 20, 2024

This section contains the following topics:

- Printing best practices
- PDF printing

Printing best practices

July 23, 2024

This article provides information about printing best practices.

Installation

The Linux VDA requires both **cups** and **foomatic** filters. The filters are installed when you install the VDA. You can also install the filters manually based on the distribution.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Printing policy settings

Client Printer Redirection

This setting allows you to determine whether client printers are mapped to a VDA session. By default, client printer mapping is allowed.

Auto-create client printers

This setting specifies client printers that can be mapped into VDA sessions. By default, it is set to **Auto-create all client printers**, which means all client printers are mapped to VDA sessions. For more information about this setting, see Auto-create client printers in the Citrix Virtual Apps and Desktops documentation.

Auto-create PDF Universal Printer

To use the PDF printing feature, set this policy to **Enabled**.

Printer driver mapping and compatibility

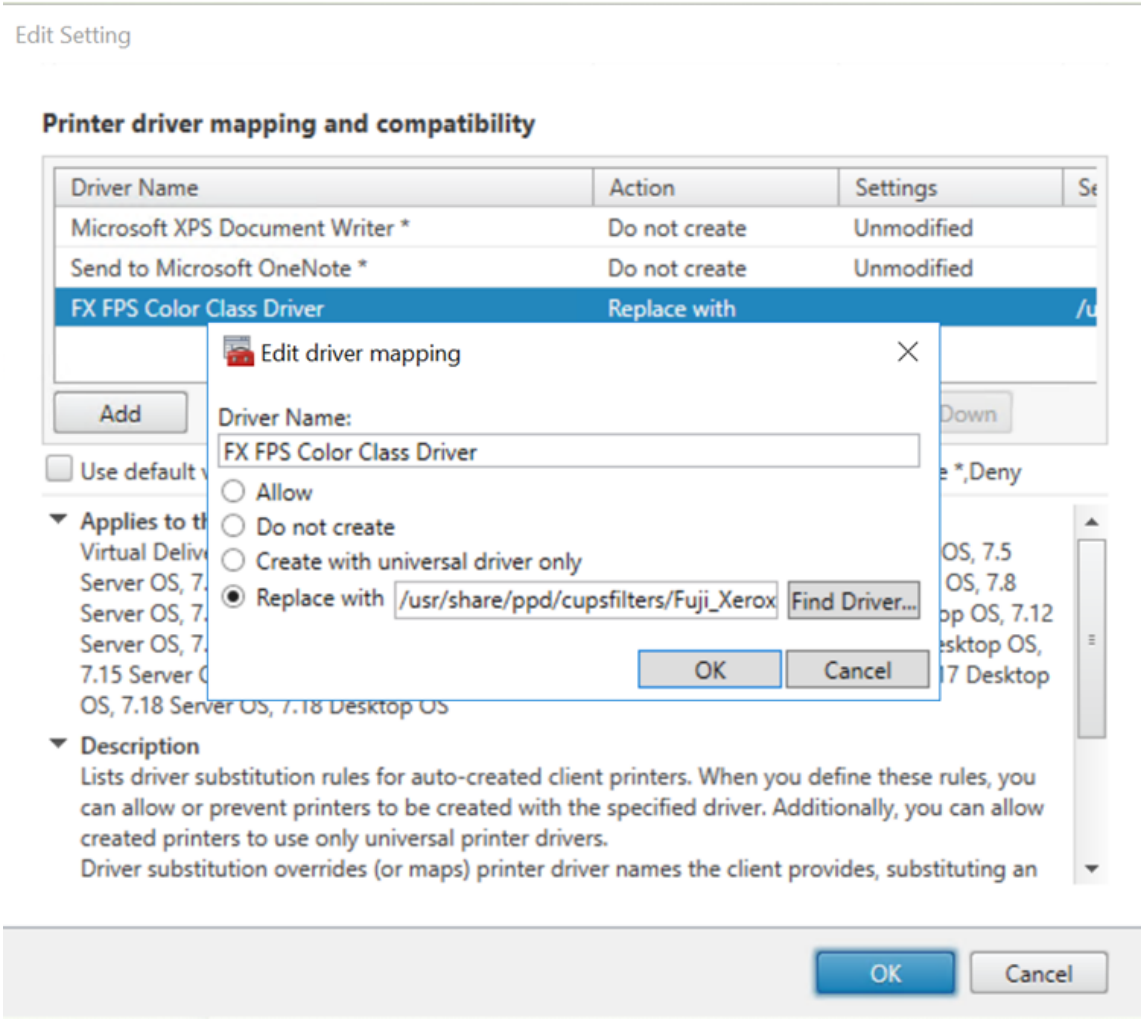
Citrix supplies three types of Universal Printer Drivers (postscript, pcl5, and pcl6). However, the Universal Printer Driver might not be compatible with your client printer. In this case, your only option in earlier releases was to edit the ~/.CtxlpProfile$CLIENT_NAME configuration file. Starting with Version 1906, you can choose to configure the **Printer driver mapping and compatibility** policy in Citrix Studio instead.

To configure the **Printer driver mapping and compatibility** policy in Citrix Studio:

1. Select the **Printer driver mapping and compatibility** policy.
2. Click **Add**.
3. Fill in **Driver name** with the driver name of the client printer. If you are using Citrix Workspace app for Linux, fill in the printer name instead.
4. Perform the following actions as needed:
    - To allow the client printer to be redirected to the VDA session and use only universal print drivers, select **Allow** or **Create with universal driver only**.
    - To prevent the client printer with the specified driver name from being redirected to the VDA session, select **Do not create**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- To give server applications access to the client printer that has the same driver as the server but has a different driver name, select **Replace with** and type in the absolute path of the driver file on the VDA.



> **Note:**
>
> Only PPD driver files are supported.

Usage

You can print from both published desktops and published applications. All client printers can be mapped to a VDA session. The printer names are different for desktops and applications:

- For published desktops:
  `<client printer name>:$CLIENT_NAME:dsk$SESSION_ID`
- For published applications:
  `<client printer name>:$CLIENT_NAME:app$SESSION_ID`

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> If the same user opens both a published desktop and a published application, both printers are available to the session. Printing on a desktop printer in a published application session, or printing on an application printer in a published desktop fails.

Troubleshooting

Unable to print

When printing is not working correctly, check the print daemon **ctxlpmngt** and the **CUPs framework**.

The print daemon **ctxlpmngt** is a per-session process and must be running for the length of the session. Run the following command to verify that the printing daemon is running. If **ctxlpmngt** is not running, start **ctxlpmngt** manually from a command line.

```
ps –ef | grep ctxlpmngt
```

If printing is still not working, check the **CUPS** framework. The **ctxcups** service is used for printer management and communicates with the Linux CUPS framework. It is a single process per machine and can be checked by running the following command:

```
systemctl status ctxcups
```

Extra steps for collecting CUPS logs

To collect CUPS logs, run the following commands to configure the CUPS service file. Otherwise, CUPS logs cannot be recorded in **hdx.log**:

```
sudo systemctl stop cups

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

PrivateTmp=false

sudo systemctl start cups

sudo systemctl daemon-reload
```

> **Note:**
>
> This configuration is made only for collecting the full printing log when an issue arises. Under normal circumstances, this configuration is not recommended because it breaks CUPS security.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Print output is garbled

An incompatible printer driver can cause garbled output. A per-user driver configuration is available and can be configured by editing the **~/.CtxlpProfile$CLIENT_NAME** configuration file:

```
[DEFAULT_PRINTER]

printername=

model=

ppdpath=

drivertype=
```

> **Important:**
>
> The **printername** is a field containing the name of the current client-side default printer. It is a read-only value. Do not edit it.
>
> The fields **ppdpath**, **model**, and **drivertype** cannot be set at the same time because only one takes effect for the **mapped printer**.

- If the Universal Printer driver is not compatible with the client printer, configure the model of the native printer driver using the **model**= option. You can find the current model name of the printer by using the **lpinfo** command:

  ```
  lpinfo  – m

  …

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

  xerox/ph3115fr.ppd.gz Xerox Phaser 3115, SpliX V. 2.0.0
  xerox/ph3115pt.ppd.gz Xerox Phaser 3115, SpliX V. 2.0.0
  ```

  You can then set the model to match the printer:

  ```
  model=xerox/ph3115.ppd.gz
  ```

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

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

  ```
  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 the driver type based on your printer properties.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

For example, if the client default printer driver type is PCL5, set **drivertype** to:

```
drivertype=pcl5
```

Output size is zero

Try different types of printers. And try a virtual printer like CutePDF and PDFCreator to find out whether this issue is related to the printer driver.

The print job depends on the printer driver of the client default printer. It's important to identify the type of the current active driver type. If the client printer is using a PCL5 driver but the Linux VDA chooses a Postscript driver, an issue can occur.

If the printer driver type is correct, you can identify the problem by performing the following steps:

1. Log on to a published desktop session.
2. Run the **vi ~/.CtxlpProfile$CLIENT_NAME** command.
3. Add the following field to save the spool 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. Check whether the spool is empty. If the spool file is zero, it represents an issue. Contact Citrix Support (and provide the printing log) for more guidance.
7. If the spool size is not zero, copy the file to the client. The spool file content depends on the printer driver type of the client default printer. If the mapped printer (native) driver is postscript, the spool file can be opened in the Linux OS directly. Check 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 on the client-side printer by using a different printer driver.
8. Change the **mapped printer** to use a different printer driver. The following example uses the postscript client printer as an example:

   a) Log on to an active session and open a browser on the client desktop.

   b) Open the printing management portal:

      ```
      localhost:631
      ```

   c) Choose the **mapped printer** `CitrixUniversalPrinter:$ClientName:app/dsk$SESSION_ID` and **Modify Printer**. This operation requires administrator privileges.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

d) Retain the **cups-ctx** connection, then click **Continue** to change the printer driver.

e) In the **Make** and **Model** fields, choose a different printer driver from the Citrix UPD driver. For example, if the CUPS-PDF virtual printer is installed, select the Generic CUPS-PDF Printer driver. Save the change.

f) If this process succeeds, configure the PPD file path of the driver in `.CtxlpProfile$CLIENT_NAME` to allow the mapped printer to use the newly selected driver.

Known issues

The following issues have been identified during printing on the Linux VDA:

CTXPS driver is not compatible with some PLC printers

If you encounter printing output corruption, set the printer driver to the native one provided by the manufacturer.

Slow printing performance for large documents

When you print a large document on a local client printer, the document is transferred over the server connection. On slow connections, the transfer can take a long time.

Printer and print job notifications seen from other sessions

Linux does not have the same session concept as the Windows operating system. Therefore, all users get system-wide notifications. You can disable these notifications by changing the CUPS configuration file: **/etc/cups/cupsd.conf**.

Locate the current policy name configured in the file:

```
DefaultPolicy default
```

If the policy name is *default*, add the following lines to the default policy XML block:

```
<Policy default>

     # Job/subscription privacy...

     JobPrivateAccess default

     JobPrivateValues default

     SubscriptionPrivateAccess default

```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
11      SubscriptionPrivateValues default
12
13      ... ...
14
15      <Limit Create-Printer-Subscription>
16
17           Require user @OWNER
18
19           Order deny,allow
20
21      </Limit>
22
23      <Limit All>
24
25           Order deny,allow
26
27      </Limit>
28
29 </Policy>
```

PDF printing

June 20, 2024

Using a version of Citrix Workspace app that supports PDF printing, you can print PDFs converted from within the Linux VDA sessions. Session print jobs are sent to the local machine where Citrix Workspace app is installed. On the local machine, you can open PDFs using your PDF viewer of choice and print them on your printer of choice.

The Linux VDA supports PDF printing on the following versions of Citrix Workspace app:

- Citrix Receiver for HTML5 Versions 2.4 through 2.6.9, Citrix Workspace app 1808 for HTML5 and later
- Citrix Receiver for Chrome Versions 2.4 through 2.6.9, Citrix Workspace app 1808 for Chrome and later
- Citrix Workspace app 1905 for Windows and later

Configuration

Apart from using a version of Citrix Workspace app that supports PDF printing, set the following policies in Citrix Studio:

- Set **Client Printer Redirection** to **Allowed** (**Allowed** by default)
- Set **Auto-create PDF Universal Printer** to **Enabled** (**Disabled** by default)

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- Set **Auto-create client printers** to **Auto-create all client printers**.

With these policies enabled, a print preview appears on the local machine for you to select a printer when you click **Print** within your launched session. See the Citrix Workspace app documentation for information about setting default printers.

Remote PC Access

June 20, 2024

Overview

Remote PC Access is an extension of Citrix Virtual Apps and Desktops. It enables organizations to easily allow employees to access their physical office PCs remotely in a secure manner. If users can access their office PCs, they can access all the applications, data, and resources they need to do their work.

Remote PC Access uses the same Citrix Virtual Apps and Desktops components that deliver virtual desktops and applications. The requirements and process of deploying and configuring Remote PC Access are the same as the requirements and process required for deploying Citrix Virtual Apps and Desktops. This uniformity provides a consistent and unified administrative experience. Users receive the best user experience by using Citrix HDX to deliver their remote office PC sessions.

For more information, see Remote PC Access in the Citrix Virtual Apps and Desktops documentation.

Considerations

These considerations are specific to the Linux VDA:

- On physical machines, use the Linux VDA only in non-3D mode. Due to limitations on NVIDIA's driver, the local screen of the PC cannot be blacked out when HDX 3D mode is enabled. Showing this screen is a potential security risk.
- Use machine catalogs of type single-session OS for physical Linux machines.
- Automatic user assignment is not available for Linux machines. With automatic user assignment, users are assigned to their machines automatically when they log on locally to the PCs. This logon occurs without administrator intervention. Citrix Workspace app on the client helps users access the applications and data on the office PC within the Remote PC Access desktop session.
- If users are already logged on to their PCs locally, attempts to launch the PCs from StoreFront fail.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- Power saving options are not available for Linux machines.

Configuration

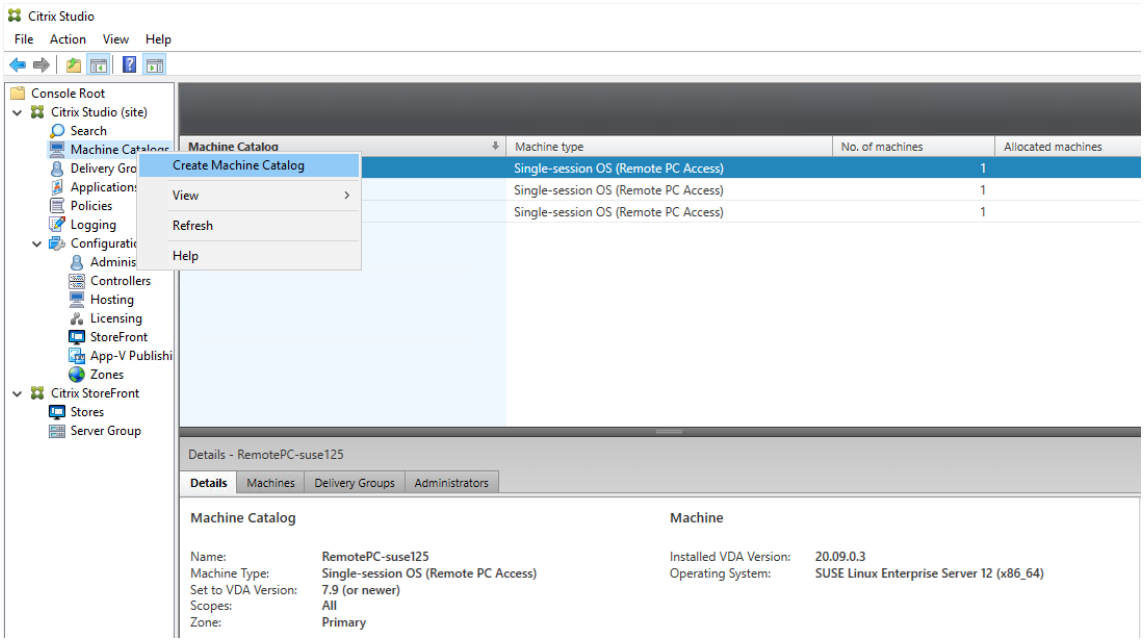
To deliver Linux PC sessions, install the Linux VDA on target PCs, create a machine catalog of the **Remote PC Access** type, and create a Delivery Group to make the PCs in the machine catalog available for users who request access. The following section details the procedure:

Step 1 - Install the Linux VDA on target PCs

We recommend you use easy install to install the Linux VDA. During the installation, set the value of the `CTX_XDL_VDI_MODE` variable to `Y`.

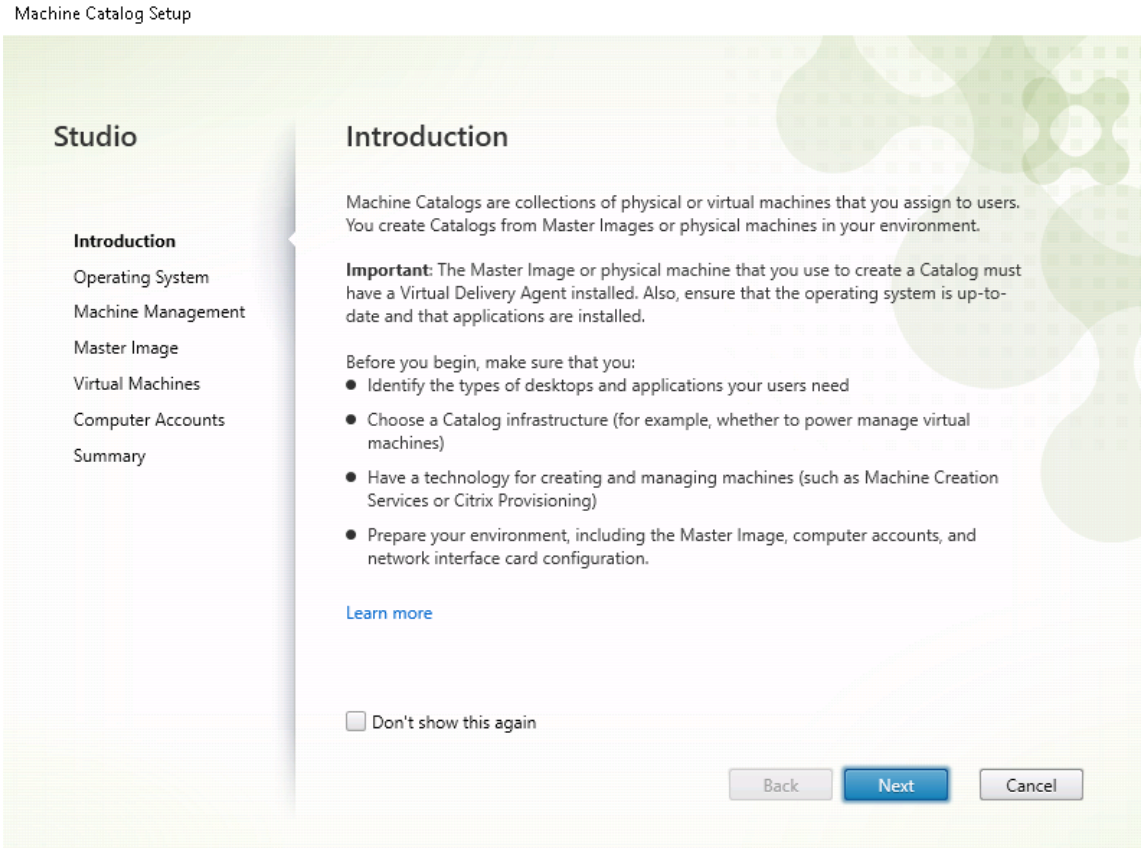
Step 2 - Create a machine catalog of the Remote PC Access type

1. In Citrix Studio, right-click **Machine Catalogs** and select **Create Machine Catalog** from the shortcut menu.



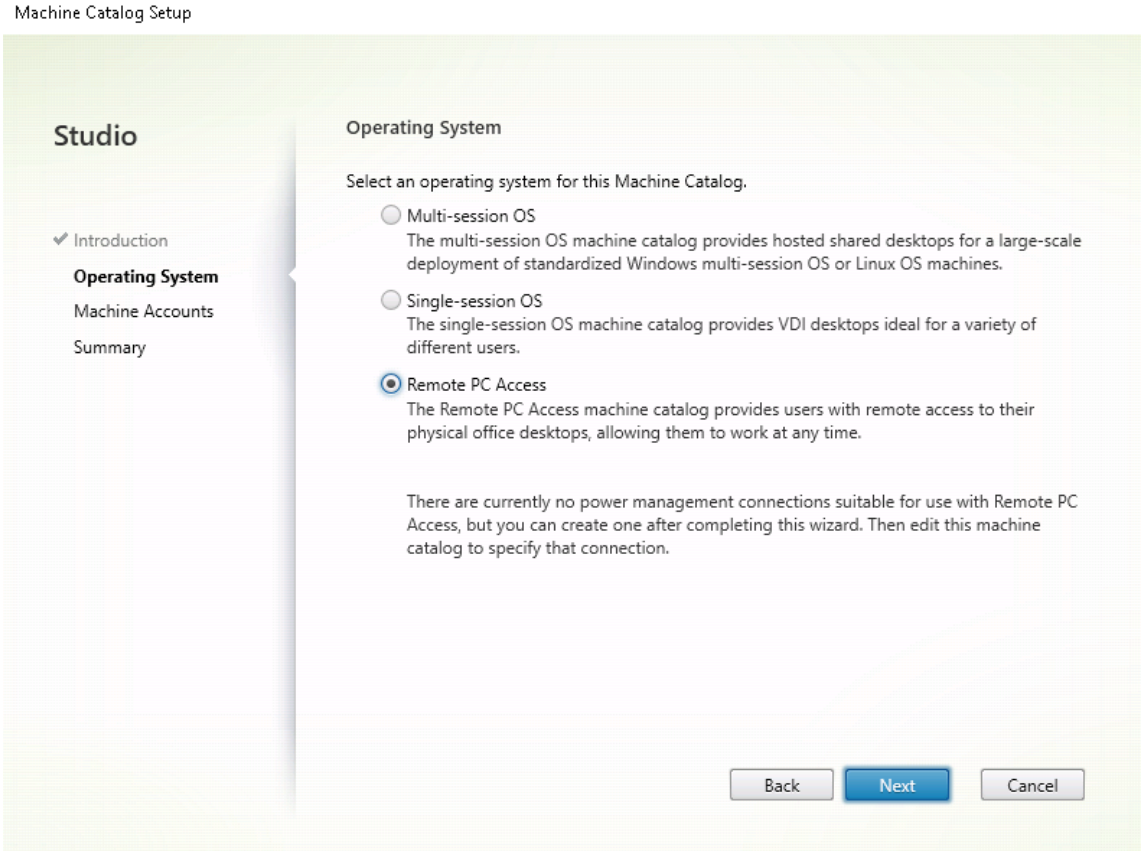
2. Click **Next** on the **Introduction** page.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



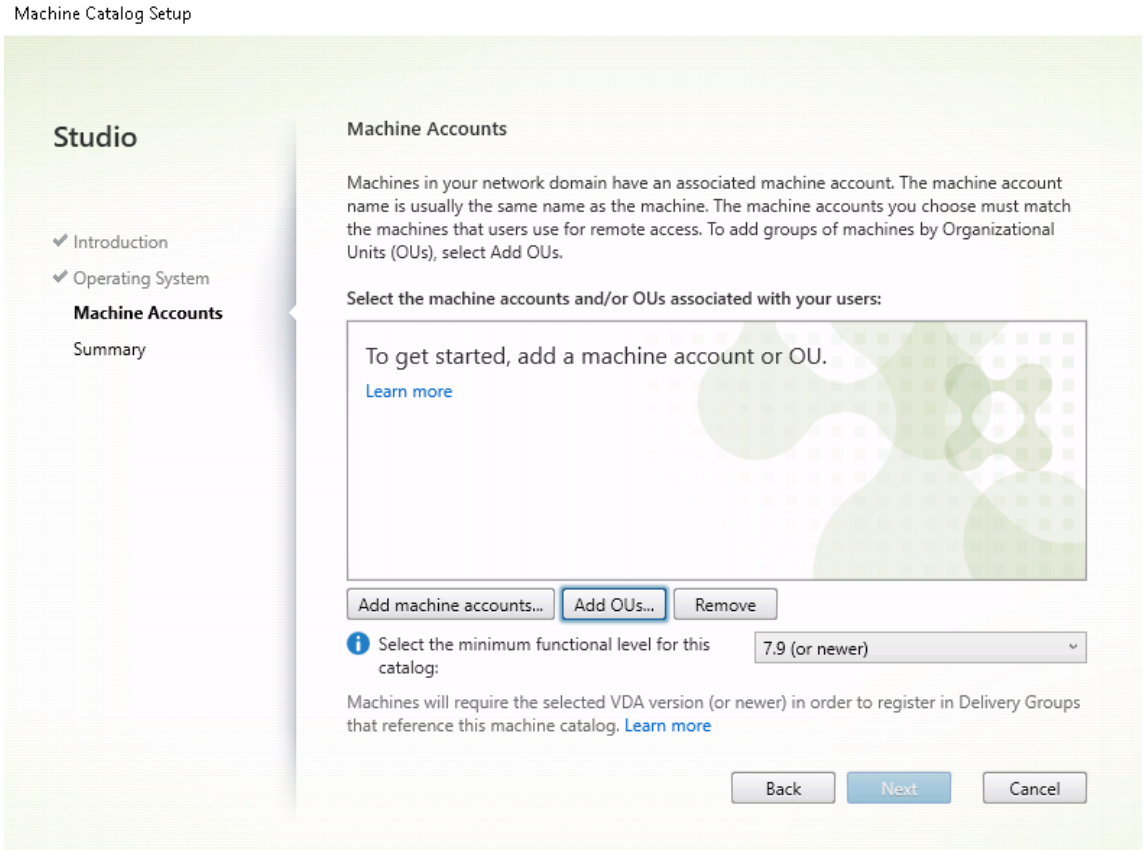
3. Select **Remote PC Access** on the **Operating System** page.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



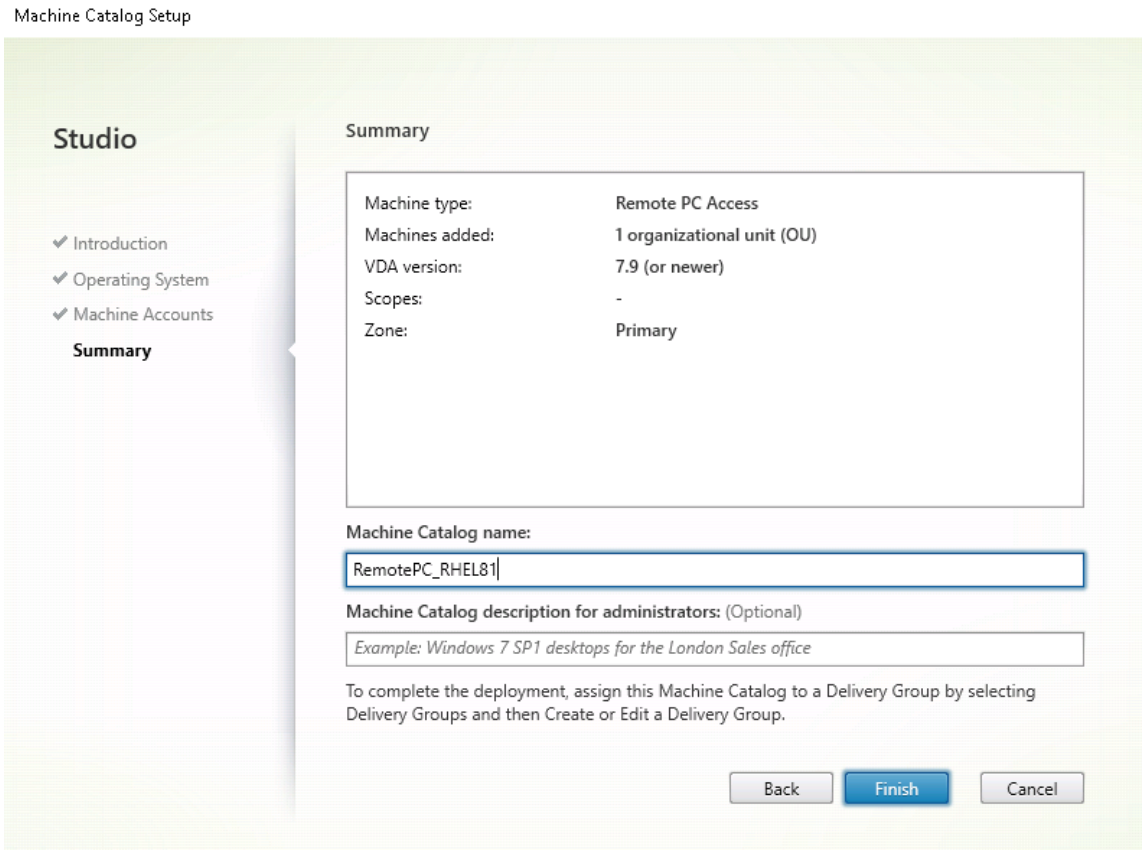
4. Click **Add OUs** to select OUs that contain the target PCs, or click **Add machine accounts** to add individual machines to the machine catalog.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

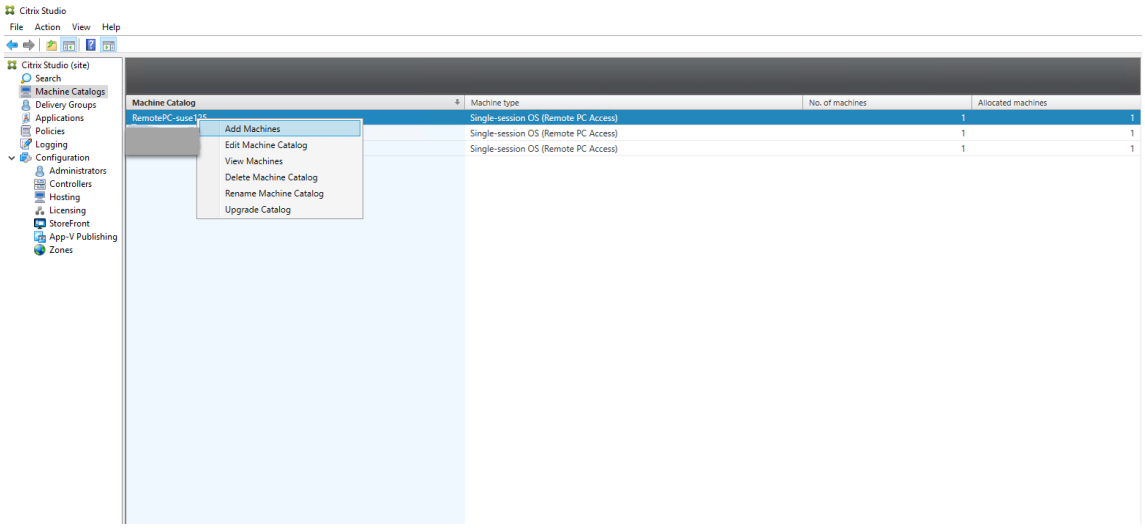


5. Name the machine catalog.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



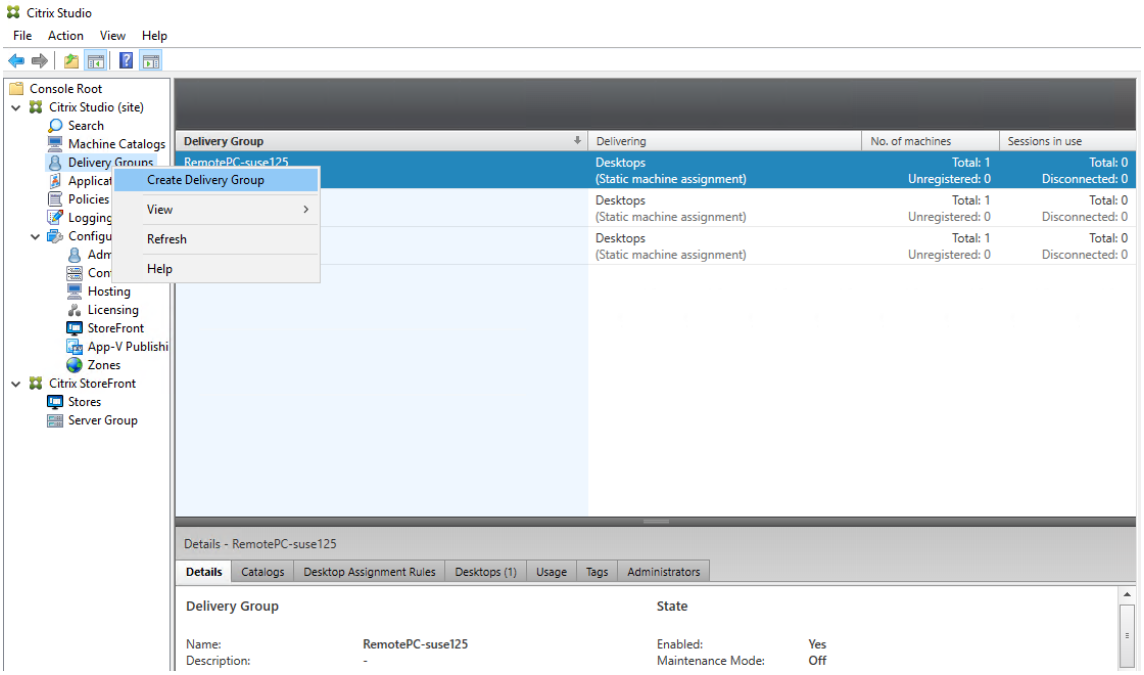
6. (Optional) Right-click the machine catalog to perform relevant operations.



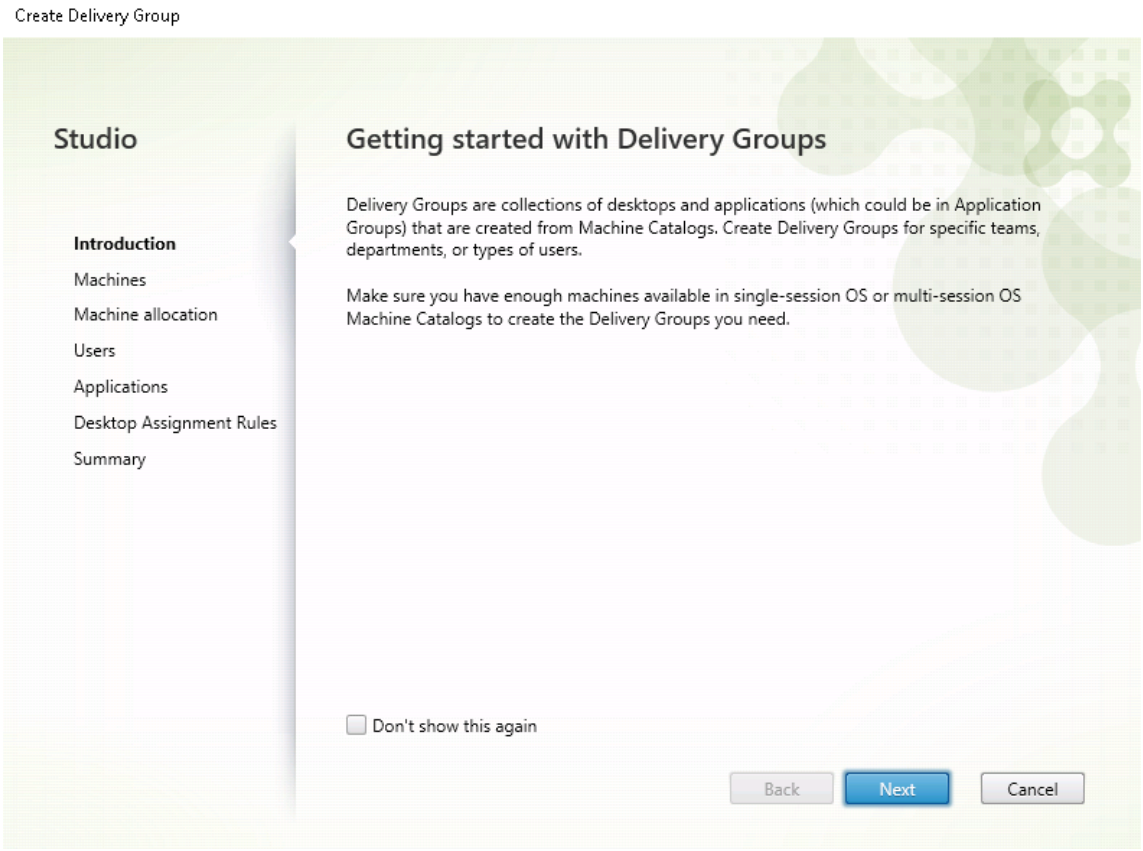
Step 3 - Create a Delivery Group to make the PCs in the machine catalog available for users who request access

1. In Citrix Studio, right-click **Delivery Groups** and select **Create Delivery Group** from the shortcut menu.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

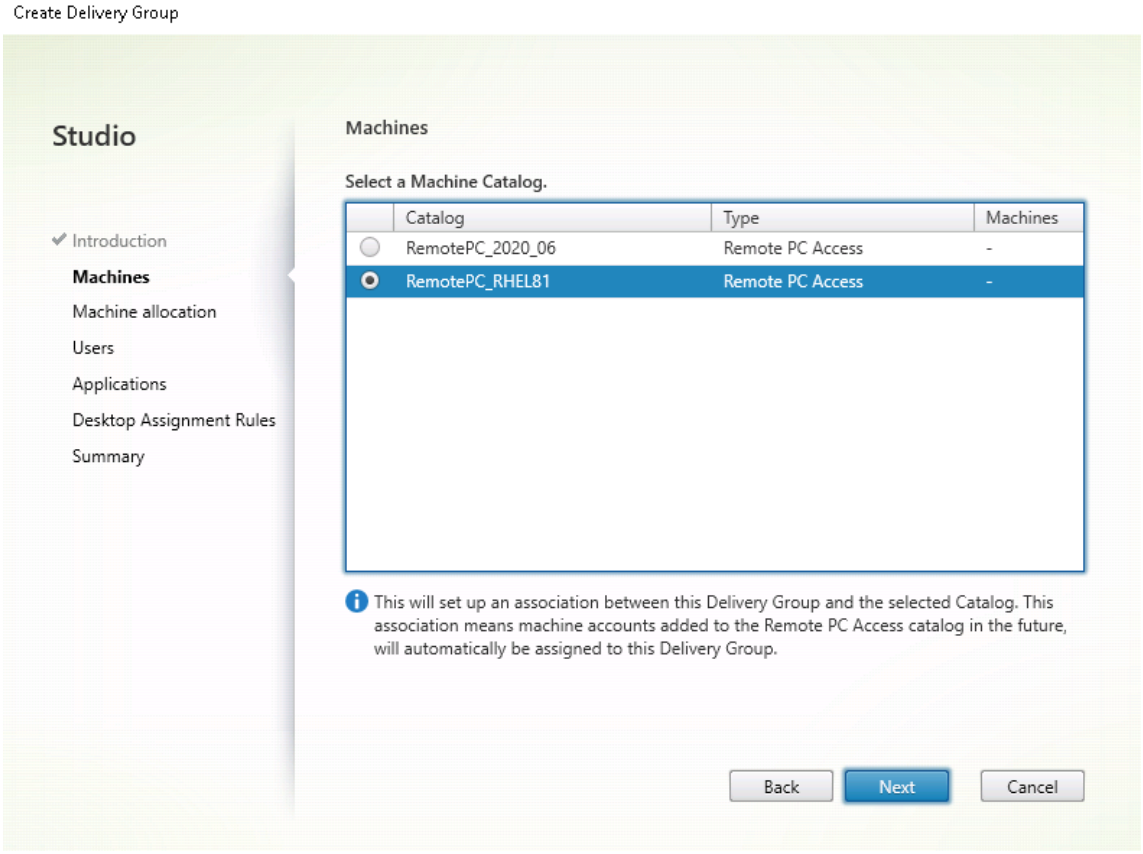


2. Click **Next** on the **Getting started with Delivery Groups** page.



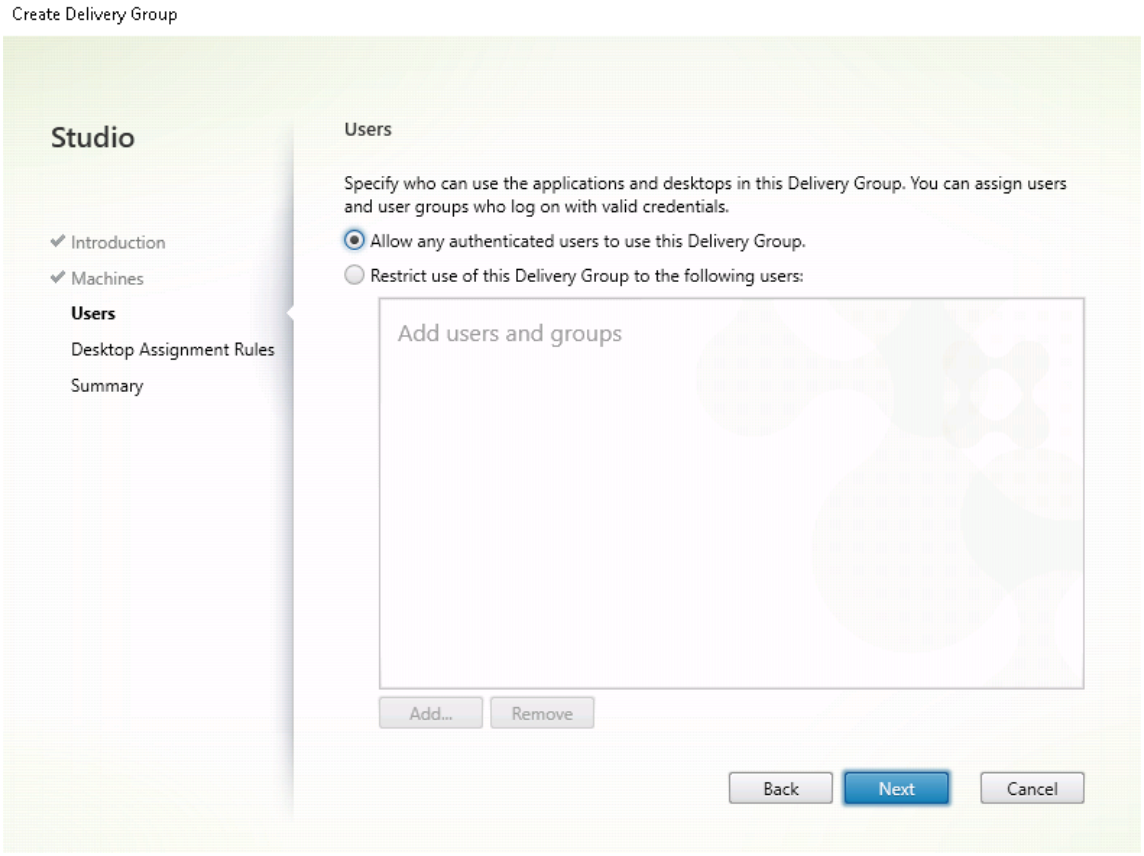
3. Select the machine catalog created in Step 2 to associate it with the Delivery Group.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



4. Add users who can access the PCs in the machine catalog. The users you add can use Citrix Workspace app on a client device to access the PCs remotely.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



Wake on LAN

Remote PC Access supports Wake on LAN, which gives users the ability to turn on physical PCs remotely. This feature enables users to keep their office PCs turned off when not in use to save energy costs. It also enables remote access when a machine has been turned off inadvertently.

With the Wake on LAN feature, the magic packets are sent directly from the VDA running on the PC to the subnet in which the PC resides when instructed by the delivery controller. This allows the feature to work without dependencies on extra infrastructure components or third-party solutions for delivery of magic packets.

The Wake on LAN feature differs from the legacy SCCM-based Wake on LAN feature. For information on the SCCM-based Wake on LAN, see Wake on LAN –SCCM-integrated.

System requirements

The following are the system requirements for using the Wake on LAN feature:

- Control plane:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

  - Citrix DaaS (formerly Citrix Virtual Apps and Desktops service)
  - Citrix Virtual Apps and Desktops 2012 or later

- Physical PCs:
  - VDA version 2012 or later
  - Wake on LAN enabled in the BIOS and on the NIC

Configure Wake on LAN

Currently, the configuration of integrated Wake on LAN is only supported using PowerShell.

To configure Wake on LAN:

1. Create the Remote PC Access machine catalog if you do not have one already.
2. Create the Wake on LAN host connection if you do not have one already.

   > **Note:**
   >
   > To use the Wake on LAN feature, if you have a host connection of the "Microsoft Configuration Manager Wake on LAN"type, create a host connection.

3. Retrieve the Wake on LAN host connection's unique identifier.
4. Associate the Wake on LAN host connection with a machine catalog.

   To create the Wake on LAN host connection:

```
# Load Citrix SnapIns
Add-PSSnapIn -Name "*citrix*"

# Provide the name of the Wake on LAN host connection
[string]$connectionName = "Remote PC Access Wake on LAN"

# Create the hypervisor connection
$hypHc = New-Item -Path xdhyp:\Connections `
            -Name $connectionName `
            -HypervisorAddress "N/A" `
            -UserName "woluser" `
            -Password "wolpwd" `
            -ConnectionType Custom `
            -PluginId VdaWOLMachineManagerFactory `
            -CustomProperties "<CustomProperties></
               CustomProperties>" `
            -Persist

$bhc = New-BrokerHypervisorConnection -HypHypervisorConnectionUid
   $hypHc.HypervisorConnectionUid

# Wait for the connection to be ready before trying to use it
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
while (-not $bhc.IsReady)
{

    Start-Sleep -s 5
    $bhc = Get-BrokerHypervisorConnection -
       HypHypervisorConnectionUid $hypHc.HypervisorConnectionUid
 }
```

When the host connection is ready, run the following commands to retrieve the host connection's unique identifier:

```
$bhc = Get-BrokerHypervisorConnection -Name "<WoL Connection Name>
    "
$hypUid = $bhc.Uid
```

After you retrieve the connection's unique identifier, run the following commands to associate the connection with the Remote PC Access machine catalog:

```
Get-BrokerCatalog -Name "<Catalog Name>" | Set-BrokerCatalog -
    RemotePCHypervisorConnectionUid $hypUid
```

5. Enable Wake on LAN in the BIOS and on the NIC on each VM in the machine catalog.

   Note: The method for enabling Wake on LAN varies with different machine configurations.

   - To enable Wake on LAN in the BIOS:
     a) Enter the BIOS and enable the Wake on LAN feature.
        The method for accessing the BIOS depends on the manufacturer of your motherboard and the BIOS vendor the manufacturer has selected.
     b) Save your settings and restart the machine.
   - To enable Wake on LAN on the NIC:
     a) Run the `sudo ethtool <NIC>` command to check whether your NIC supports magic packets.
        `<NIC>` is the device name of your NIC, for example, `eth0`. The `sudo ethtool <NIC>` command provides an output about the capabilities of your NIC:
        - If the output contains a line similar to `Supports Wake-on: <letters>` where `<letters>` contains the letter `g`, your NIC supports the Wake on LAN magic packet method.
        - If the output contains a line similar to `Wake-on: <letters>` where `<letters>` contains the letter `g` and does not contain the letter `d`, the Wake on LAN magic packet method is enabled. However, if `<letters>` contains the letter `d`, it indicates that the Wake on LAN feature is disabled. In this case, enable Wake on LAN by running the `sudo ethtool -s <NIC> wol g` command.

     b) On most distributions, the `sudo ethtool -s <NIC> wol g` command is required after each startup. To persistently set this option, complete the following

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

steps based on your distributions:

Ubuntu:

Add the `up ethtool -s <NIC> wol g` line to the interface configuration file `/etc/network/interfaces`. For example:

```
# ifupdown has been replaced by netplan(5) on this system.
    See
# /etc/netplan for current configuration.
# To re-enable ifupdown on this system, you can run:
# sudo apt install ifupdown
auto eth0
iface eth0 inet static
           address 10.0.0.1
           netmask 255.255.240.0
           gateway 10.0.0.1
           up ethtool -s eth0 wol g
```

RHEL/SUSE:

Add the following `ETHTOOL_OPTS` parameter to the interface configuration file `/etc/sysconfig/network-scripts/ifcfg-<NIC>`:

```
ETHTOOL_OPTS="-s ${
 DEVICE }
  wol g"
```

Design considerations

When you are planning to use Wake on LAN with Remote PC Access, consider the following:

- Multiple machine catalogs can use the same Wake on LAN host connection.
- For a PC to wake up another PC, both PCs must be in the same subnet and use the same Wake on LAN host connection. It does not matter if the PCs are in the same or different machine catalogs.
- Host connections are assigned to specific zones. If your deployment contains more than one zone, you need a Wake on LAN host connection in each zone. The same applies to machine catalogs.
- Magic packets are broadcasted using the global broadcast address 255.255.255.255. Ensure that the address is not blocked.
- There must be at least one PC turned on in the subnet - for every Wake on LAN connection - to be able to wake up machines in that subnet.

Operational considerations

The following are considerations for using the Wake on LAN feature:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- The VDA must register at least once before the PC can be woken up using the integrated Wake on LAN feature.
- Wake on LAN can only be used to wake up PCs. It does not support other power actions, such as restart or shut down.
- After the Wake on LAN connection is created, it is visible in Studio. However, editing its properties within Studio is not supported.
- Magic packets are sent in one of the two ways:
  - When a user tries to launch a session to their PC and the VDA is unregistered
  - When an administrator sends a power on command manually from Studio or PowerShell
- Because the delivery controller is unaware of a PC's power state, Studio displays **Not Supported** under power state. The delivery controller uses the VDA registration state to determine whether a PC is on or off.

More resources

The following are other resources for Remote PC Access:

- Solution design guidance: Remote PC Access Design Decisions.
- Examples of Remote PC Access architectures: Reference Architecture for Citrix Remote PC Access Solution.

Session

June 20, 2024

This section contains the following topics:

- Adaptive transport
- Logon with a temp home directory
- Publish applications
- Session reliability
- Rendezvous V1
- Rendezvous V2
- Secure user sessions using TLS
- Secure user sessions using DTLS

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Adaptive transport

June 20, 2024

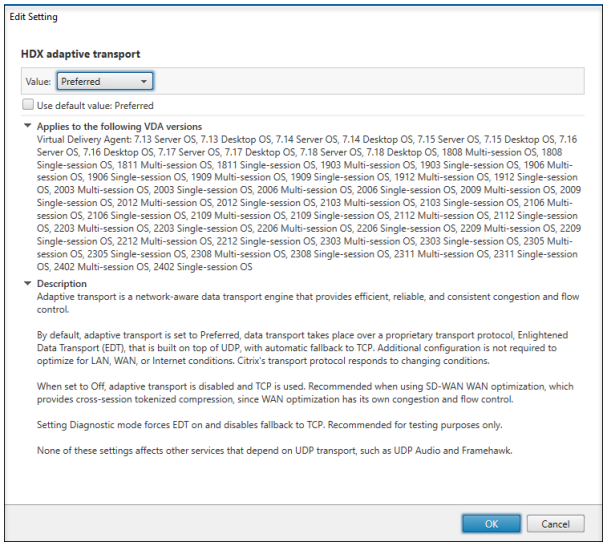
Adaptive transport is a mechanism in Citrix Virtual Apps and Desktops that can use Enlightened Data Transport (EDT) as the transport protocol for ICA connections. Adaptive transport switches to TCP when EDT is not available.

EDT is a Citrix-proprietary transport protocol built on top of the User Datagram Protocol (UDP). It delivers a superior user experience on challenging long-haul connections while maintaining server scalability. EDT improves data throughput for all ICA virtual channels on unreliable networks, providing a better and more consistent user experience.

For more information, see Adaptive transport in the Citrix Virtual Apps and Desktops documentation.

Enable or disable adaptive transport

Adaptive transport is enabled by default. You can configure the following options using the **HDX Adaptive Transport** policy setting:



- **Preferred**: Adaptive transport is enabled, and it uses Enlightened Data Transport (EDT) as the preferred transport protocol, with fallback to TCP.
- **Diagnostic mode**: Adaptive transport is enabled, and it forces the use of EDT. Fallback to TCP is disabled. This setting is recommended for testing and troubleshooting only.
- **Off**. Adaptive transport is disabled, and only TCP is used for transport.

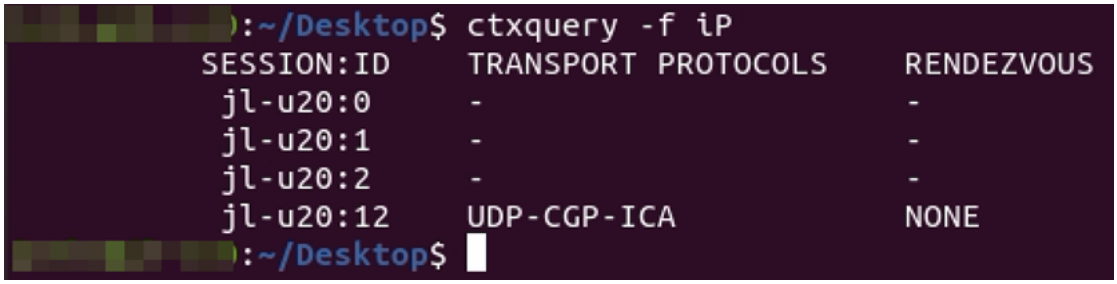
© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Check whether adaptive transport is being used

To check whether EDT is being used as the transport protocol for the current session, run the following command.

```
/opt/Citrix/VDA/bin/ctxquery -f iP
```

If EDT is being used, the transport protocols displayed include UDP, for example:



EDT MTU discovery

MTU discovery allows EDT to automatically determine the Maximum Transmission Unit (MTU) when establishing a session. Doing so prevents EDT packet fragmentation that might result in performance degradation or failure to establish a session.

System requirements:

- Linux VDA minimum version 2012
- Citrix Workspace app:
  - Windows: 1911 or later
- Citrix ADC:
  - 13.0.52.24 or later
  - 12.1.56.22 or later
- Session reliability must be enabled

If you use client platforms or versions that don't support this feature, see CTX231821 for details about configuring a custom EDT MTU that is appropriate for your environment.

Control EDT MTU discovery on the VDA

EDT MTU discovery is enabled on the VDA by default. You can enable or disable it by setting the `MtuDiscovery` registry key as follows:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- To enable EDT MTU discovery, set the `MtuDiscovery` registry key using the following command, restart the VDA, and wait for the VDA to register:

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\
   CurrentControlSet\Control\Terminal Server\Wds\icawd" -t "
   REG_DWORD" -v "MtuDiscovery" -d "0x00000001" --force
```

- To disable EDT MTU discovery, delete the `MtuDiscovery` registry value.

> **Warning:**
>
> Editing the registry incorrectly can cause serious problems that might require you to reinstall your operating system. Citrix cannot guarantee that problems resulting from the incorrect use of **Registry Editor** can be solved. Use **Registry Editor** at your own risk. Be sure to back up the registry before you edit it.

Control EDT MTU discovery on the client

You can control EDT MTU discovery selectively on the client by adding the `MtuDiscovery` parameter in the ICA file. To disable the feature, set the following under the `Application` section:

```
MtuDiscovery=Off
```

To re-enable the feature, remove the `MtuDiscovery` parameter from the ICA file.

> **Important:**
>
> For this ICA file parameter to work, enable EDT MTU discovery on the VDA. If EDT MTU discovery is not enabled on the VDA, the ICA file parameter has no effect.

Enhanced EDT congestion control

A congestion control algorithm is introduced to optimize the EDT protocol. This implementation allows EDT to achieve higher throughput and reduce latency for an enhanced user experience.

This feature is enabled by default. To disable and re-enable it, run the following commands, respectively and then restart the `ctxhdx` service:

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet\
   Control\Terminal Server\Wds\icawd\Tds\udp\UDPStackParameters" -t "
   REG_DWORD" -v "edtBBR" -d "0x00000000" --force
```

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet\
   Control\Terminal Server\Wds\icawd\Tds\udp\UDPStackParameters" -t "
   REG_DWORD" -v "edtBBR" -d "0x00000001" --force
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

HDX adaptive throughput

July 17, 2024

HDX adaptive throughput intelligently fine-tunes the peak throughput of ICA sessions by adjusting output buffers. The number of output buffers is initially set at a high value. This high value allows data to be transmitted to the client more quickly and efficiently, especially in high latency networks. This feature brings an enhanced user experience. It provides better interactivity, faster file transfers, smoother video playback, and higher framerate and resolution. Session interactivity is constantly measured to determine whether any data streams within an ICA session are adversely affecting interactivity. If that occurs, the throughput is decreased to reduce the impact of the large data stream on the session and allow interactivity to recover.

> **Important:**
>
> HDX adaptive throughput changes the way of setting output buffers by moving the mechanism from the client to the VDA. No manual configuration is necessary.
>
> The feature requires the VDA minimum version 2311. It is enabled by default. If it is disabled, you can enable it by running the following command on the VDA:
>
> ```
> 1 /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet
>     \Control\Terminal Server\Wds\icawd" -t "REG_DWORD" -v "
>     AdaptiveScalingEnabled" -d "0x00000001" --force
> ```
>
> The feature applies to only the sessions that are launched after the feature is enabled.

Custom backgrounds and banner messages on session logon screens

June 20, 2024

Add a custom background or banner message to session logon screens

> **Tip:**
>
> To use the feature on SUSE 15.5, install `imlib2` from http://download.opensuse.org/distribution/leap/15.3/repo/oss/.

You can use the following commands to add a custom background or banner message to session **logon** screens. To add both a background and a banner message to session **logon** screens, you can

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

embed the banner message into the background image. After you open a session, the banner message page is displayed first and then the authentication dialog appears.

To set the title of a custom banner message, run:

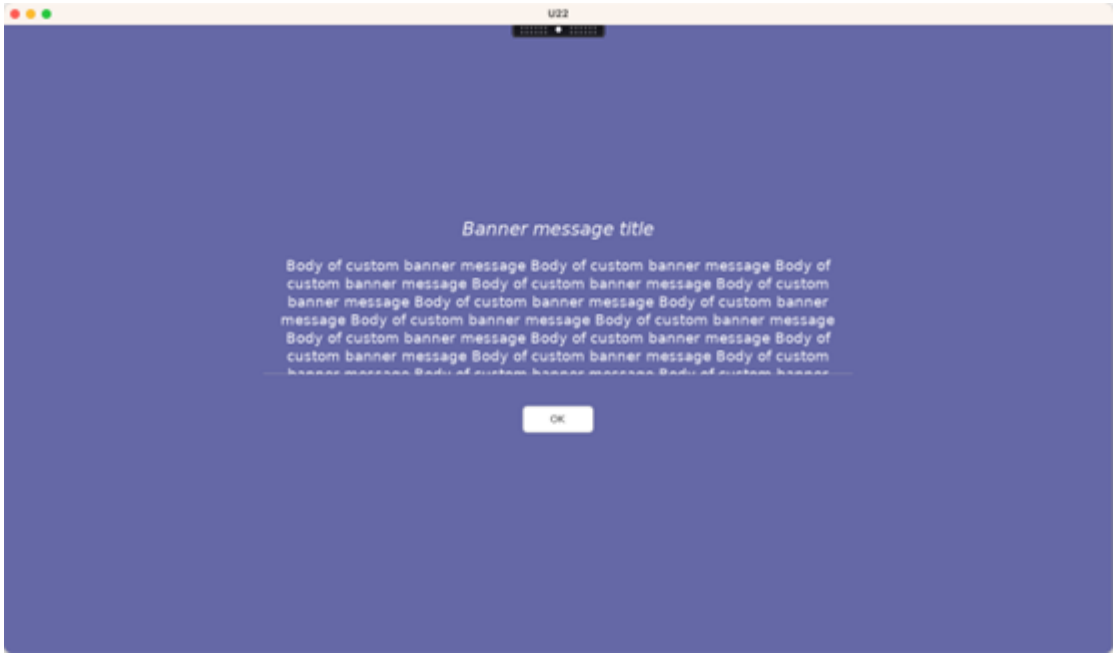
```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet\
   Control\Citrix" -t "REG_SZ" -v "LogonDisplayStringTitle" -d "<Banner
    message title>" --force
```

The maximum length of a banner message title is 64 bytes.

To set the body text of a custom banner message, run:

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet\
   Control\Citrix" -t "REG_SZ" -v "LogonDisplayString" -d "Body of
   custom banner message\nBody of custom banner message\nBody of custom
    banner message\n" --force
```

The maximum length of a banner message body is 1,024 bytes.

> **Tip:**
>
> The \n element creates a line break. In this example, the banner message screen looks like this:
>
> 
>
>
> To ensure that your custom banner message displays as expected, you might need to use escape characters. For example, use **&amp\;** to represent **&** in HTML and put a backslash (\) before the character **!** to escape it in Bash commands.

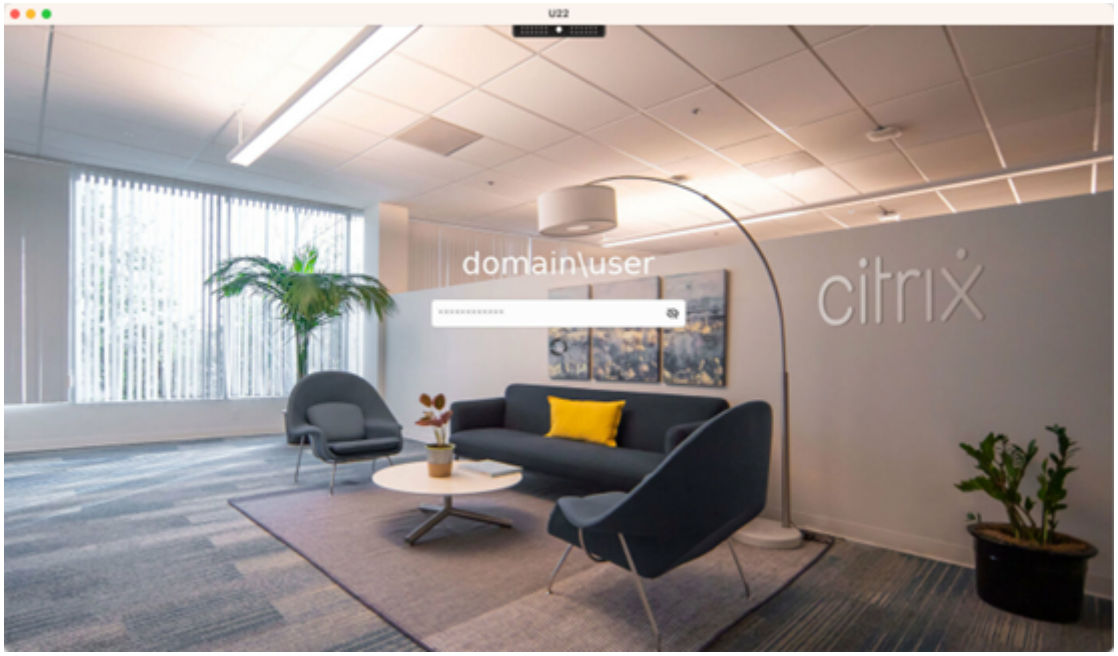
To add a custom background to session **logon** screens, run:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet\
   Control\Citrix" -t "REG_SZ" -v "BackgroundImagePath" -d "<path to
   the background image>" --force
```

To see the custom background, session users must have access to the background image path.

For example:



Example session logon screens

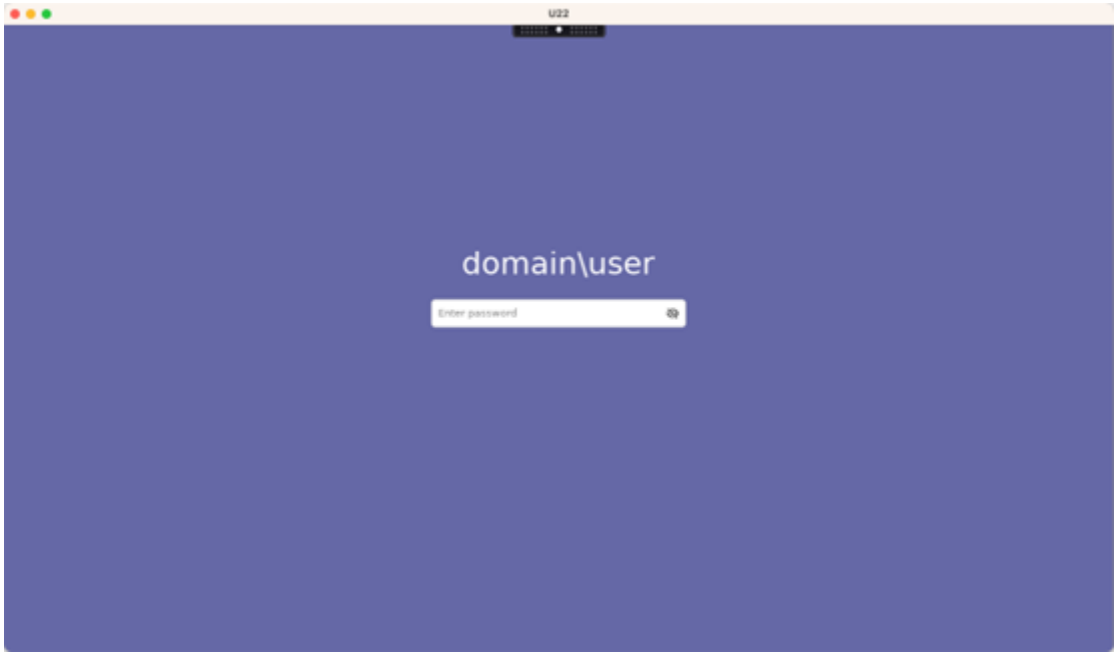
The following are example session logon screens in different scenarios:

- Session **logon** in single sign-on (SSO) scenarios:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

The **logon** process is displayed.

- Session **logon** in typical non-SSO scenarios:



  - A password or PIN code is required in non-SSO scenarios.
  - Users can toggle the visibility of passwords and PIN codes, making it easy for users to find out incorrect inputs.

- Session logon in non-SSO scenarios when users log on to VDA sessions with credentials different from the credentials used to log on to Citrix Workspace app:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

User name and password used for session logon:

Smart card used for session logon:

For the combinations of user authentication methods that are supported in non-SSO scenarios, see Non-SSO authentication.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Custom desktop environments by session users

July 15, 2024

You can specify a desktop environment for session users by using the **CTX_XDL_DESKTOP_ENVIRONMENT** variable. Starting with the 2209 release, session users can customize their own desktop environments. To let session users use this feature, you must install desktop environments on the VDA in advance.

The following table shows a matrix of the Linux distributions and the desktop environments that support custom desktop environments by session users.

Linux distribution	Supported desktop
Debian 12.5	MATE, GNOME, GNOME-Classic, KDE, **Xfce**
Debian 11.9	MATE, GNOME, GNOME-Classic, KDE
RHEL 8.10/8.9/8.8	MATE, GNOME, GNOME-Classic, **Xfce**
Rocky Linux 8.10/8.9/8.8	MATE, GNOME, GNOME-Classic, KDE, **Xfce**
SUSE 15.5	MATE, GNOME, GNOME-Classic
Ubuntu 22.04/20.04	MATE, GNOME, GNOME-Classic, KDE, **Xfce**

Desktop switching commands

> **Note:**
>
> You can switch between desktop environments from both the Terminal and the system tray.

To switch to a target desktop environment from the Terminal, run the corresponding command within the session:

If the target desktop environment is	Run the command
GNOME	`/opt/Citrix/VDA/bin/ctxdesktopswitch.sh GNOME`
GNOME Classic	`/opt/Citrix/VDA/bin/ctxdesktopswitch.sh GNOME-CLASSIC`
MATE	`/opt/Citrix/VDA/bin/ctxdesktopswitch.sh MATE`

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

If the target desktop environment is	Run the command
KDE	`/opt/Citrix/VDA/bin/ctxdesktopswitch.sh KDE`

KDE tips

- Magnus might load at startup in KDE. As a workaround, you can remove the Magnus package by running `sudo apt remove magnus`.
- To disable QT warnings that occur during KDE startup, edit `/usr/share/qt5/qtlogging.ini` as a root user by adding the following entries:

```
qt.qpa.xcb.xcberror.error=false
qt.qpa.xcb.warning=false
qt.qpa.xcb.error=false
```

- Screen unlock might fail for KDE. As a workaround, we recommend you disable the auto-lock feature of your desktop.

Logon with a temp home directory

June 20, 2024

You can specify a temp home directory for cases where the mount point on the Linux VDA fails. With a temp home directory specified, a prompt shows during a session logon when the mount point fails. User data is then stored under the temp home directory.

The following table describes registry keys that help with your home directory settings.

Registry key	Description	Command
`LogNoHome`	Controls whether users can log on to sessions without a home directory. The default value is 1 and it means yes. If the value is set to 0, session logons without a home directory are disabled.	`create -k "HKLM\System\CurrentControlSet\Control\Citrix"-t "REG_DWORD"-v "LogNoHome"-d "0x00000001"--force`

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Registry key	Description	Command
HomeMountPoint	Sets a local mount point on the Linux VDA. For example, if /mnt/home is the mount point, a user's home directory is /mnt/home/domain/<user_name>. Make sure that the mount point is the same as the user home directory in your environment. This setting takes effect only when CheckUserHomeMountPoint is set to 0.	create -k "HKLM\System\CurrentControlSet\Control\Citrix"-t "REG_SZ"-v "HomeMountPoint"-d "<A directory where the NFS share is to be mounted>"--force
CheckUserHomeMountPoint	Controls whether to set user-specific home directories as the mount point on the Linux VDA. If you want to set user-specific home directories as the mount point, set the value to **1**. The default value is **0**.	ctxreg create -k "HKLM\System\CurrentControlSet\Control\Citrix"-t "REG_DWORD"-v "CheckUserHomeMountPoint"-d "0x00000001"--force
TempHomeDirectoryPath	Sets a temp home directory on the Linux VDA in case the mount point fails. The default value is /tmp. The temp home directory setting takes effect only when the mount point determined by HomeMountPoint and CheckUserHomeMountPoint is unavailable. A temp home directory for a user is /tmp/CTXSmf_user_id.	create -k "HKLM\System\CurrentControlSet\Control\Citrix"-t "REG_SZ"-v "TempHomeDirectoryPath"-d "</tmp by default>"--force

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Registry key	Description	Command
CheckMountPointRetryTime	Sets the number of checks, at a frequency of once per second, on whether mounting is successful. The default value is 5.	`ctxreg create -k "HKLM\System\CurrentControlSet\Control\Citrix"-t "REG_DWORD"-v "CheckMountPointRetryTime"-d "0x000000010"--force`
RemoveHomeOnLogoff	Controls whether to remove temp home directories on user logoffs. 1 means yes. 0 means no.	`create -k "HKLM\System\CurrentControlSet\Control\Citrix"-t "REG_DWORD"-v "RemoveHomeOnLogoff"-d "0x00000000"--force`

Publish applications

June 20, 2024

With Linux VDA Version 7.13, Citrix added the seamless applications feature to all the supported Linux platforms. No specific installation procedures are required to use this feature.

> **Tip:**
>
> With Linux VDA version 1.4, Citrix added support for non-seamless published applications and session sharing.

Publish applications using Citrix Studio

You can publish applications installed on a Linux VDA when you create a delivery group or add applications to an existing delivery group. The process is similar to publishing applications installed on a Windows VDA. For more information, see the Citrix Virtual Apps and Desktops documentation (based on the version of Citrix Virtual Apps and Desktops being used).

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> - When configuring delivery groups, ensure that the delivery type is set to **Desktop and applications** or **Applications**.
> - We recommend that you create separate VDAs and delivery groups for app and desktop deliveries.
> - To use seamless applications, do not disable the seamless mode on StoreFront. The seamless mode is enabled by default. If you have already disabled it by setting "TWIMode=Off," remove this setting instead of changing it to "TWIMode=On."Otherwise you might not be able to launch a published desktop.

Limitation

The Linux VDA does not support the launch of multiple concurrent instances of the same application by a single user.

In an app session, only shortcuts that are specific to the app work as expected.

Known issues

The following known issues are identified during publishing applications:

- 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.
- When you run multiple LibreOffice applications, only the one launched first shows on Citrix Studio because these applications share the process.
- Published Qt5-based applications like "Dolphin"might not show icons. To resolve the issue, see the article at https://wiki.archlinux.org/title/Qt.
- Linux applications often have an About window containing information about the application in use, and web links for more information are commonly found in those windows. Clicking the web links in the About window can launch a web browser from within published applications such as **calc**, **gedit**, **calendar**, and **LibreOffice Suite**. The unintentional launch of a web browser bypasses application isolation and can compromise security. To address the issue, change the default web browser by completing the following steps:

  1. Create a none.sh file in a custom location, for example:

```
sudo mkdir /home/none

sudo touch /home/none/none.sh
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
4
5  sudo chmod +x /home/none/none.sh
```

2. Add the following lines to the none.sh file:

```
1  #!/bin/bash
2
3  echo "NONE"
```

3. Create the /etc/xdg/mimeapps.list file with sudo, and then add the following lines to the mimeapps.list file:

```
1  [Default Applications]
2
3  text/html=none.desktop
4
5  x-scheme-handler/http=none.desktop
6
7  x-scheme-handler/https=none.desktop
8
9  x-scheme-handler/about=none.desktop
10
11 x-scheme-handler/unknown=none.desktop
```

4. Create the /usr/share/applications/none.desktop file with sudo, and then add the following lines to the none.desktop file:

```
1  [Desktop Entry]
2
3  Encoding=UTF-8
4
5  Version=1.0
6
7  Type=Application
8
9  Terminal=false
10
11 Exec=/home/none/none.sh
12
13 Name=None
14
15 Icon=/home/none/none.sh
```

Because you can put the none.sh file in a custom location, make sure that none.desktop can reference the none.sh file correctly.

Rendezvous V1

June 20, 2024

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

When using the Citrix Gateway service, the Rendezvous protocol allows traffic to bypass the Citrix Cloud Connectors and connect directly and securely with the Citrix Cloud control plane.

There are two types of traffic to consider: 1) control traffic for VDA registration and session brokering; 2) HDX session traffic.

Rendezvous V1 allows for HDX session traffic to bypass Cloud Connectors, but it still requires Cloud Connectors to proxy all control traffic for VDA registration and session brokering.

Requirements

- Access to the environment using Citrix Workspace and Citrix Gateway service.
- Control Plane: Citrix DaaS (formerly Citrix Virtual Apps and Desktops service).
- Linux VDA Version 2112 or later.
  - Version 2112 is the minimum required for no-transparent HTTP proxies.
  - Version 2204 is the minimum required for transparent and SOCKS5 proxies.
- Enable the Rendezvous protocol in the Citrix policy. For more information, see Rendezvous protocol policy setting.
- The VDAs must have access to `https://*.nssvc.net`, including all subdomains. If you can't add all subdomains to the allow list in that manner, use `https://*.c.nssvc.net` and `https://*.g.nssvc.net` instead. For more information, see the Internet Connectivity Requirements section of the Citrix Cloud documentation (under Virtual Apps and Desktop service) and the Knowledge Center article CTX270584.
- Cloud Connectors must obtain the VDAs' FQDNs when brokering a session. Accomplish this task in one of these two ways:
  - **Enable DNS resolution for the site.** Navigate to **Full Configuration > Settings** and turn on the **Enable DNS resolution** setting. Alternatively, use the Citrix Virtual Apps and Desktops Remote PowerShell SDK and run the command `Set-BrokerSite -DnsResolutionEnabled $true`. For more information about the Citrix Virtual Apps and Desktops Remote PowerShell SDK, see SDKs and APIs.
  - **DNS Reverse Lookup Zone with PTR records for the VDAs.** If you choose this option, we recommend that you configure VDAs to always attempt to register PTR records. To do so, use the Group Policy Editor or Group Policy Object, navigate to **Computer Configuration > Administrative Templates > Network > DNS Client**, and set **Register PTR Records** to **Enabled and Register**. If the connection's DNS suffix does not match the domain's DNS suffix, you must also configure the **Connection-specific DNS suffix** setting for the machines to register PTR records successfully.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> **Note:**
>
> If using the DNS resolution option, the Cloud Connectors must be able to resolve the fully qualified domain names (FQDNs) of the VDA machines. In the case that internal users connect directly to the VDA machines, the client devices also must be able to resolve the VDA machines' FQDNs.
>
> If using a DNS reverse lookup zone, the FQDNs in the PTR records must match the FQDNs of the VDA machines. If the PTR record contains a different FQDN, the Rendezvous connection fails. For example, if the machine's FQDN is `vda01.domain.net`, the PTR record must contain `vda01.domain.net`. A different FQDN such as `vda01.sub.domain.net` does not work.

Proxy configuration

The VDA supports establishing Rendezvous connections through HTTP and SOCKS5 proxies.

Proxy considerations

Consider the following when using proxies with Rendezvous:

- Non-transparent HTTP proxies and SOCKS5 proxies are supported.
- Packet decryption and inspection are not supported. Configure an exception so that the ICA traffic between the VDA and the Gateway Service is not intercepted, decrypted, or inspected. Otherwise, the connection breaks.
- HTTP proxies support machine-based authentication by using the Negotiate and Kerberos authentication protocols. When you connect to the proxy server, the **Negotiate authentication** scheme automatically selects the Kerberos protocol. Kerberos is the only scheme that the Linux VDA supports.

  > **Note:**
  >
  > To use Kerberos, you must create the service principal name (SPN) for the proxy server and associate it with the proxy's Active Directory account. The VDA generates the SPN in the format `HTTP/<proxyURL>` when establishing a session, where the proxy URL is retrieved from the **Rendezvous proxy** policy setting. If you don't create an SPN, authentication fails.

- Authentication with a SOCKS5 proxy is not currently supported. If using a SOCKS5 proxy, you must configure an exception so that traffic destined to Gateway Service addresses (specified in the requirements) can bypass authentication.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- Only SOCKS5 proxies support data transport through EDT. For an HTTP proxy, use TCP as the transport protocol for ICA.

Transparent proxy

Transparent HTTP proxy is supported for Rendezvous. If using a transparent proxy in your network, no additional configuration is required on the VDA.

Non-transparent proxy

When using a non-transparent proxy in your network, configure the Rendezvous proxy configuration setting. When the setting is enabled, specify the HTTP or SOCKS5 proxy address for the VDA to know which proxy to use. For example:

- Proxy address: `http://<URL or IP>:<port>` or `socks5://<URL or IP>:<port>`

Rendezvous validation

If you meet all requirements, follow these steps to validate if Rendezvous is in use:

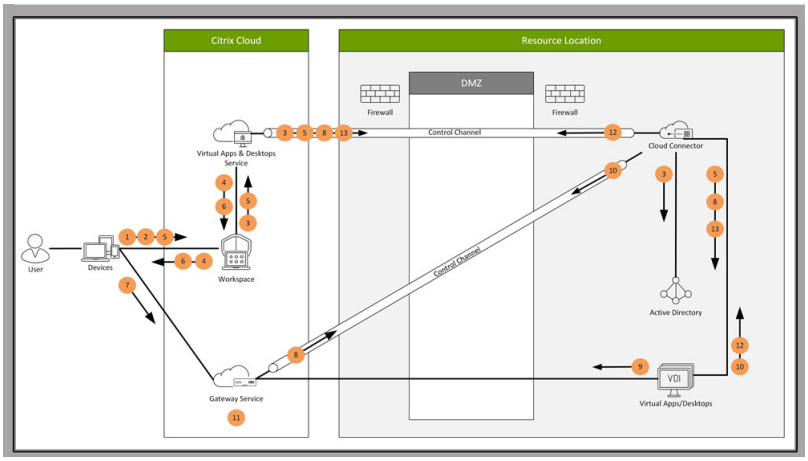
1. Launch a terminal on the VDA.
2. Run `/opt/Citrix/VDA/bin/ctxquery -f iP`.
3. The TRANSPORT PROTOCOLS indicates the type of connection:
    - TCP Rendezvous: TCP - TLS - CGP - ICA
    - EDT Rendezvous: UDP - DTLS - CGP - ICA
    - Proxy through Cloud Connector: TCP - PROXY - SSL - CGP - ICA or UDP - PROXY - DTLS - CGP - ICA

> **Tip:**
>
> If the VDA cannot reach the Citrix Gateway service directly with Rendezvous enabled, the VDA falls back to proxy the HDX session through the Cloud Connector.

How Rendezvous works

This diagram is an overview of the Rendezvous connection flow.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



Follow the steps to understand the flow.

1. Navigate to Citrix Workspace.
2. Enter credentials in Citrix Workspace.
3. If using on-premises Active Directory, Citrix DaaS authenticates credentials with Active Directory using the Cloud Connector channel.
4. Citrix Workspace displays enumerated resources from Citrix DaaS.
5. Select resources from Citrix Workspace. Citrix DaaS sends a message to the VDA to prepare for an incoming session.
6. Citrix Workspace sends an ICA file to the endpoint that contains an STA ticket generated by Citrix Cloud.
7. The endpoint connects to the Citrix Gateway service, provides the ticket to connect to the VDA, and Citrix Cloud validates the ticket.
8. The Citrix Gateway service sends connection information to the Cloud Connector. The Cloud Connector determines if the connection is a Rendezvous connection and sends the information to the VDA.
9. The VDA establishes a direct connection to the Citrix Gateway service.
10. If a direct connection between the VDA and the Citrix Gateway service isn't possible, the VDA proxies its connection over the Cloud Connector.
11. The Citrix Gateway service establishes a connection between the endpoint device and the VDA.
12. The VDA verifies its license with Citrix DaaS through the Cloud Connector.
13. Citrix DaaS sends session policies to the VDA through the Cloud Connector. Those policies are applied.

Rendezvous V2

June 20, 2024

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

When using the Citrix Gateway service, the Rendezvous protocol allows traffic to bypass the Citrix Cloud Connectors and connect directly and securely with the Citrix Cloud control plane.

There are two types of traffic to consider: 1) control traffic for VDA registration and session brokering; 2) HDX session traffic.

Rendezvous V1 allows for HDX session traffic to bypass Cloud Connectors, but it still requires Cloud Connectors to proxy all control traffic for VDA registration and session brokering.

Standard AD domain joined machines and non-domain joined machines are supported for using Rendezvous V2 with single-session and multi-session Linux VDAs. With non-domain joined machines, Rendezvous V2 allows for both HDX traffic and control traffic to bypass the Cloud Connectors.

Requirements

The requirements for using Rendezvous V2 are:

- Access to the environment using Citrix Workspace and Citrix Gateway service.
- Control Plane: Citrix DaaS (formerly Citrix Virtual Apps and Desktops service).
- VDA version 2201 or later.
  - Version 2204 is the minimum required for HTTP and SOCKS5 proxies.
- Enable the Rendezvous protocol in the Citrix policy. For more information, see Rendezvous protocol policy setting.
- The VDAs must have access to:
  - `https://*.xendesktop.net` on `TCP 443`. If you can't allow all subdomains in that manner, you can use `https://<customer_ID>.xendesktop.net`, where `<customer_ID>` is your Citrix Cloud customer ID as shown in the Citrix Cloud administrator portal.
  - `https://*.nssvc.net`, including all subdomains. If you cannot whitelist all subdomains in that manner, use `https://*.c.nssvc.net` and `https://*.g.nssvc.net` instead. For more information, see the Internet Connectivity Requirements section of the Citrix Cloud documentation (under Virtual Apps and Desktop service) and the Knowledge Center article CTX270584.
- The VDAs must be able to connect to the addresses mentioned previously:
  - On TCP 443, for TCP Rendezvous.
  - On UDP 443, for EDT Rendezvous.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Proxy configuration

The VDA supports connecting through proxies for both control traffic and HDX session traffic when using Rendezvous. The requirements and considerations for both types of traffic are different, so review them carefully.

Control traffic proxy considerations

- Only HTTP proxies are supported.
- Packet decryption and inspection are not supported. Configure an exception so the control traffic between the VDA and the Citrix Cloud control plane is not intercepted, decrypted, or inspected. Otherwise, the connection fails.
- Proxy authentication is not supported.
- To configure a proxy for control traffic, edit the registry as follows:

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\
   VirtualDesktopAgent" -t "REG_SZ" -v "ProxySettings" -d "http
   ://<URL or IP>:<port>" --force
```

HDX traffic proxy considerations

- HTTP and SOCKS5 proxies are supported.
- EDT can only be used with SOCKS5 proxies.
- To configure a proxy for HDX traffic, use the Rendezvous proxy configuration policy setting.
- Packet decryption and inspection are not supported. Configure an exception so the HDX traffic between the VDA and the Citrix Cloud control plane is not intercepted, decrypted, or inspected. Otherwise, the connection fails.
- HTTP proxies support machine-based authentication by using the Negotiate and Kerberos authentication protocols. When you connect to the proxy server, the **Negotiate authentication scheme** automatically selects the Kerberos protocol. Kerberos is the only scheme that the Linux VDA supports.

  > **Note:**
  >
  > To use Kerberos, you must create the service principal name (SPN) for the proxy server and associate it with the proxy's Active Directory account. The VDA generates the SPN in the format `HTTP/<proxyURL>` when establishing a session, where the proxy URL is retrieved from the **Rendezvous proxy** policy setting. If you don't create an SPN, authentication fails.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- Authentication with a SOCKS5 proxy is not currently supported. If using a SOCKS5 proxy, you must configure an exception so that traffic destined to Gateway Service addresses (specified in the requirements) can bypass authentication.
- Only SOCKS5 proxies support data transport through EDT. For an HTTP proxy, use TCP as the transport protocol for ICA.

Transparent proxy

Transparent HTTP proxy is supported for Rendezvous. If using a transparent proxy in your network, no additional configuration is required on the VDA.

How to configure Rendezvous V2

Following are the steps for configuring Rendezvous in your environment:

1. Make sure that all requirements are met.
2. After the VDA is installed, run the following command to set the required registry key:

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

3. Restart the VDA machine.
4. Create a Citrix policy, or edit an existing one:
   - Set the Rendezvous Protocol setting to **Allowed**. The Rendezvous Protocol is disabled by default. When the Rendezvous Protocol is enabled (**Allowed**), Rendezvous V2 instead of V1 takes effect.
   - Ensure that the Citrix policy filters are set properly. The policy applies to the machines that need Rendezvous to be enabled.
   - Ensure that the Citrix policy has the correct priority so that it does not overwrite another one.

Rendezvous validation

To check whether a session is using the Rendezvous protocol, run the `/opt/Citrix/VDA/bin/ctxquery -f iP` command in the terminal.

The transport protocols displayed indicate the type of connection:

- TCP Rendezvous: TCP - TLS - CGP - ICA

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

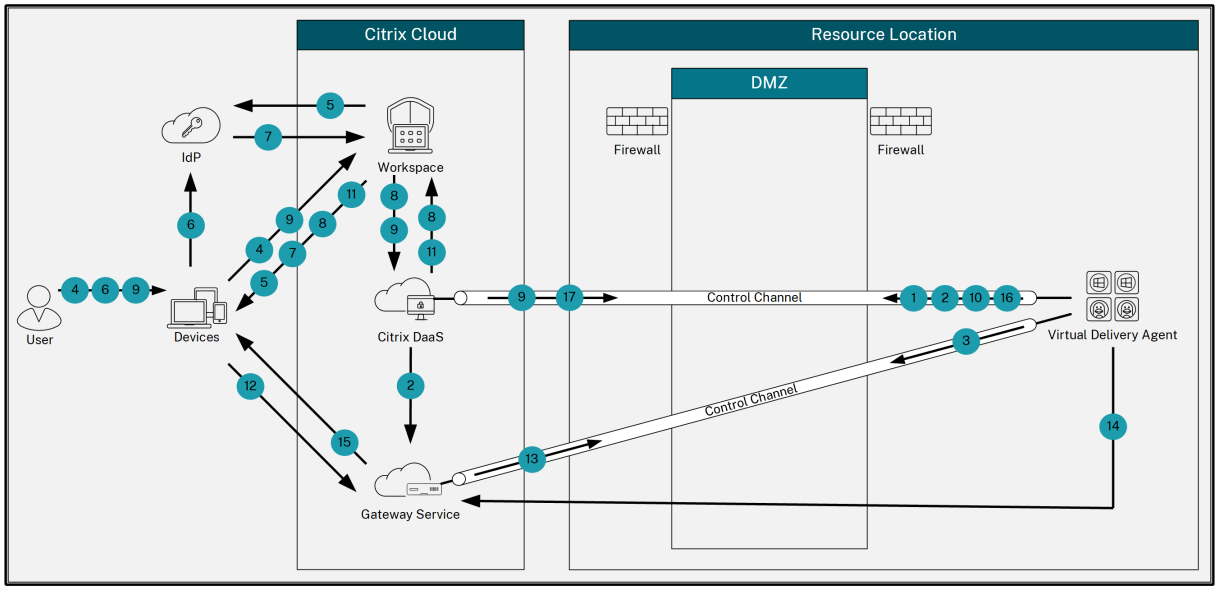
- EDT Rendezvous: UDP - DTLS - CGP - ICA
- Proxy through Cloud Connector: TCP - PROXY - SSL - CGP - ICA or UDP - PROXY - DTLS - CGP - ICA

If Rendezvous V2 is in use, the protocol version shows 2.0.

> **Tip:**
>
> If the VDA cannot reach the Citrix Gateway service directly with Rendezvous enabled, the VDA falls back to proxy the HDX session through the Cloud Connector.

Rendezvous traffic flow

The following diagram illustrates the sequence of steps about Rendezvous traffic flow.



1. The VDA establishes a WebSocket connection with Citrix Cloud and registers.
2. The VDA registers with Citrix Gateway Service and obtains a dedicated token.
3. The VDA establishes a persistent control connection with the Gateway Service.
4. The user navigates to Citrix Workspace.
5. Workspace evaluates authentication configuration and redirects users to the appropriate IdP for authentication.
6. The user enters their credentials.
7. After successfully validating the user credentials, the user is redirected to Workspace.
8. Workspace counts resources for the user and displays them.
9. The user selects a desktop or application from Workspace. Workspace sends the request to Citrix DaaS, which brokers the connection and instructs the VDA to prepare for the session.
10. The VDA responds with the Rendezvous capability and its identity.
11. Citrix DaaS generates a launch ticket and sends it to the user device through Workspace.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

12. The user's endpoint connects to the Gateway Service and provides the launch ticket to authenticate and identify the resource to connect to.
13. The Gateway Service sends the connection information to the VDA.
14. The VDA establishes a direct connection for the session with the Gateway Service.
15. The Gateway Service completes the connection between the endpoint and the VDA.
16. The VDA verifies licensing for the session.
17. Citrix DaaS sends applicable policies to the VDA.

Secure HDX (preview)

July 16, 2024

You can encrypt ICA sessions end-to-end between the Citrix Workspace app (client) and the VDA (session host).

The end-to-end encryption (E2EE) feature allows no intermediate network elements including the Citrix Gateway to decrypt the ICA traffic. It uplifts the secure posture of your environment and is easy to configure and manage.

System requirements

- Linux VDA minimum version 2311
- Delivery Controller minimum version 2308
- StoreFront minimum version 2308
- Citrix Workspace app for Windows minimum version 2308

Configuration

Enable end-to-end encryption

The end-to-end encryption (E2EE) feature is disabled by default. To enable it, set the **Secure HDX** policy to **Enabled** in Citrix Studio.

Schedule certificate renewals

The end-to-end encryption (E2EE) feature requires a self-signed certificate and its private key that the **ctxcertmgr** service on the Linux VDA manages.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

A new self-signed certificate is created when the **ctxcertmgr** service starts or restarts. By default, the **ctxcertmgr** service renews the certificate (including its private key) every 7 days at the time 2:00 AM. You can also schedule certificate renewals with registry settings similar to the following:

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\SecureHDX" -t "REG_SZ" -v "CaRotationStartDate" -d "2023-10-19" --force

/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\SecureHDX" -t "REG_SZ" -v "CaRotationTime" -d "00:45:30" --force

/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\SecureHDX" -t "REG_DWORD" -v "CaRotationPeriod" -d "0x00000005" --force
```

In the above example, the first certificate renewal time is set at 00:45:30 on 2023-10-19. After that, the **ctxcermgr** service renews the certificate every 5 days at 00:45:30. The scheduled date and time are the date and time on the Linux VDA.

Secure user sessions using DTLS

June 20, 2024

DTLS encryption is a fully supported feature starting with the 7.18 release. By default, this feature is enabled on the Linux VDA. For more information, see Transport Layer Security.

Enable DTLS encryption

Verify that adaptive transport is enabled

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

Enable SSL encryption on the Linux VDA

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

> **Note:**
>
> The Linux VDA supports DTLS 1.0, DTLS 1.2, and DTLS 1.3 and uses DTLS 1.2 by default. Check which version of DTLS is in use on your Citrix Workspace app. Ensure that the same version of DTLS is used on both the Linux VDA and your Citrix Workspace app. If your Citrix Workspace app

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> supports only DTLS 1.0 (for example, Citrix Receiver for Windows 4.11), set **SSLMinVersion** to **TLS_1.0** and **SSLCipherSuite** to **COM** or **ALL** using the **enable_vdassl.sh** tool.

Secure user sessions using TLS

June 20, 2024

Starting with Version 7.16, the Linux VDA supports TLS encryption for secure user sessions. TLS encryption is disabled by default.

Enable TLS encryption

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

Install certificates on the Linux VDA

Obtain server certificates in PEM format and root certificates in CRT format. A server certificate contains the following sections:

- Certificate
- Unencrypted private key
- Intermediate certificates (optional)

An example of a server certificate:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
-----BEGIN CERTIFICATE-----
MIIDTTCCAragAwIBAgIJALluncpiqGXCMA0GCSqGSIb3DQEBBQUAMGcxCzAJBgNV
BAYTAlVLMRIwEAYDVQQIEwlDYW1icmlkZ2UxEjAQBgNVBAcTCUNhbWJvdXJuZTEU
MBIGA1UEChMLQ2l0cml4IFRlc3QxGjAYBgNVBAMTEWNhMDAxLmNpdHJpdGUubmV0
MB4XDTA4MDkzMDEwNTk1M1oXDTI4MDkyNTEwNTk1M1owgYoxCzAJBgNVBAYTAlVL
MRIwEAYDVQQIEwlDYW1icmlkZ2UxEjAQBgNVBAcTCUNhbWJvdXJuZTEUMBIGA1UE
ChMLQ2l0cml4IFRlc3QxGzAZBgNVBAsTElNlcnZlciBDZXJ0aWZpY2F0ZTEgMB4G
A1UEAxMXY2EwMDEtc2MwMDEuY2l0cml0ZS5uZXQwgZ8wDQYJKoZIhvcNAQEBBQAD
gY0AMIGJAoGBALCTTOdxcivbIOLOF66xgO5gkNeIGKVP+37pSKV8B66lWCVzr6p9
t72Fa+9oCcf2x/ue274NXFcg4fqGRDsrEwl3YxM6COyBf7L6psrsCDNnBPlq8TJH
4xoPIXUeaW4MVk/3PVyfhHKs4fz8yy1I4VDnXVHhw+OFQ2Bq3NhwsRhnAgMBAAGj
gdwwgdkwCQYDVR0TBAIwADAdBgNVHQ4EFgQUrLidzYot+CUXSh9xMfp1M+/O8yOw
gZkGA1UdIwSBkTCBjoAU85kN1EPJOcVhcOss1slseDQwGsKha6RpMGcxCzAJBgNV
BAYTAlVLMRIwEAYDVQQIEwlDYW1icmlkZ2UxEjAQBgNVBAcTCUNhbWJvdXJuZTEU
MBIGA1UEChMLQ2l0cml4IFRlc3QxGjAYBgNVBAMTEWNhMDAxLmNpdHJpdGUubmV0
ggkAy8nC8dcB32EwEQYJYIZIAYb4QgEBBAQDAgVgMA0GCSqGSIb3DQEBBQUAA4GB
AD5axBYHwIxJCJzNt2zdXnbp2O0yUToWElBwQe/9cGaP6CpjoxJ7FJa2/8IpaT68
Vel8u1SEYY1GKCGCw93pc7sPKqb8pGBRI5/dygb+geFk1Q7KyVbuOIjOtr3pkxAe
b6CFJtNLudHUrwF6lOrB72zbyz3PiIx+HEwt1jOj8z4K
-----END CERTIFICATE-----
-----BEGIN RSA PRIVATE KEY-----
MIICXgIBAAKBgQCwk0zncXIr2yNC9BeusYDuYJDXiBilT/t+6Uilf AeupVglc6+q
fbe9hWvvaAnH9sf7ntu+DVxXIOH6hkQ7KxMNd2MTOgjsgX+y+qbK7AgzZwT9avEy
R+MaDyF1HmluDFZP9z1cn4RyrOH8/MstSOFQ511R4cPtBUNgatzYcLEYZwIDAQAB
AoGBAKwBgZu/bkl8edgB8YPyU7diiBX89IOs4b/aPjM+JDmjxb8N96RsPO24p9Ea
FtUc9+iL8mEroLUbSicCXjsJFc+cxg9vVaNa6EEkkBj735oCUERqSxOYb/lAdck/
FXzU0tqytUe/KHgcSgjtjrSeqLJqMm+yxzBAatVRTTzGdwAhAkEA3l1kRZjIN5uz
Enmi2RTI3ngBhBP/S3GEbvJfKsD5n2Ri9O+OoEPxclvvp5ne8QOzUpshbjFEPbOC
ykZ6UassFwJBAMtI5yPnV9ewPzJoaNjZIJcMtNXDchSlxXiJiyzv+Qmr8RuQz9Pv
fIenmTrfZ+Wo4DaKg+8ar2OvOnKFOHFAmDECQQDEWRlH6cE3WyCfN1u942M9XkhR
GvSpR7+b///vL6Nwwv3CwPV9n8DTpL+wuDkJZ9nCvRteil9MlaMTYjs3alNvAkEA
qy5JzZcbBnrYzMbVO32jju7ZPISnhTGO1xDjzMSLLpTGpNLN34b0k3sTclr8L42E
uQjtTqRm+wdsrVF3lFazkQJANudmsUVv3gZKhMGaV2hzIdXIfHyOIYv+3leZhQY6
h5eEmxSZS5OTvyNGt2e6m2ZgaZmjTagH59TCBHvR5nof2g==
-----END RSA PRIVATE KEY-----
-----BEGIN CERTIFICATE-----
MIIDGTCCAoKgAwIBAgIJAMvJwvHXAd9hMA0GCSqGSIb3DQEBBQUAMGcxCzAJBgNV
BAYTAlVLMRIwEAYDVQQIEwlDYW1icmlkZ2UxEjAQBgNVBAcTCUNhbWJvdXJuZTEU
MBIGA1UEChMLQ2l0cml4IFRlc3QxGjAYBgNVBAMTEWNhMDAxLmNpdHJpdGUubmV0
MB4XDTA4MDkzMDEwNDExMVoXDTI4MDkyNTEwNDExMVowZzELMAkGA1UEBhMCVUsx
EjAQBgNVBAgTCUNhbWJyaWRnZTESMBAGA1UEBxMJQ2FtYm9lcm5lMRQwEgYDVQQK
EwtDaXRyaXggVGVzdDEaMBgGA1UEAxMRY2EwMDEuY2l0cml0ZS5uZXQwgZ8wDQYJ
KoZIhvcNAQEBBQADgY0AMIGJAoGBAKVZmF7Uj7uOnvO3Qwdfi0nr3QkNH2DXpWrZ
Zh8cI9Vv+UFRUiC6oB7izLtBMFn3fOUP7i2CfkHN3ZGJ17p89pdyjket1MslVeJw
acOqrYvD+fNNSvJjunTbaCywVtALjmFSfMHeZJXVSckrpEhnk0nkMS16tcrya/K/
osSlzvI3AgMBAAGjgcwwgckwDAYDVR0TBAUwAwEB/zAdBgNVHQ4EFgQU85kN1EPJ
OcVhcOss1slseDQwGsIwgZkGA1UdIwSBkTCBjoAU85kN1EPJOcVhcOss1slseDQw
GsKha6RpMGcxCzAJBgNVBAYTAlVLMRIwEAYDVQQIEwlDYW1icmlkZ2UxEjAQBgNV
BAcTCUNhbWJvdXJuZTEUMBIGA1UEChMLQ2l0cml4IFRlc3QxGjAYBgNVBAMTEWNh
MDAxLmNpdHJpdGUubmV0ggkAy8nC8dcB32EwDQYJKoZIhvcNAQEFBQADgYEAIZ4Z
gXLLXf12RNqh/awtSbd41Ugv8BIKAsg5zhNAiTiXbzz8Cl3ec53Fb6nigMWc5Tli
UNCLXwnxRUiD4O0tESLX9ACUNH3I94yxOgujkSOSBni21jjZTvfBB32Rmr5DByJg
UmKORn/hdqMlcqpe5wO6as6+HN4WUOi+hEtUMME=
-----END CERTIFICATE-----
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Enable TLS encryption

Enable TLS encryption on the Linux VDA On the Linux VDA, use the `enable_vdassl.sh` script in the **/opt/Citrix/VDA/sbin** directory to enable (or disable) TLS encryption. For information about the options available in the script, run the `/opt/Citrix/VDA/sbin/enable_vdassl.sh -help` command.

```
root@lxu1804:~# /opt/Citrix/VDA/sbin/enable_vdassl.sh
===Enable/Disable SSL on Linux VDA===
To enable SSL, a certificate file must be specified, otherwise the local certificate file under
/etc/xdl/.sslkeystore/ is used, If the local certificate file does not exist, the command
fails. You can specify the SSL port number, version and cipher suite, otherwise, their default
values are used!

Usage: enable_vdassl.sh -Disable
       Disable Linux VDA SSL.

Usage: enable_vdassl.sh -Enable [-Certificate <CERT-FILE>] [-SSLPort <SSL-PORT-NUMBER>]
       [-SSLMinVersion <SSL-MIN-VERSION>] [-SSLCipherSuite <SSL-CIPHER-SUITE>]
       Enable Linux VDA SSL.

Options:
       -Certificate <CERT-FILE>
       Specify a certificate file, where <CERT-FILE> must include the full file path. Only one format
       is currently supported, that is PEM.

       -RootCertificate <ROOT-CERT-FILE>
       Specify a root certificate file, where <ROOT-CERT-FILE> must include the full file path, The root certificate will be put in the local keystore(under /etc/xdl/.sslkeystore/cacerts).

       -SSLPort <SSL-PORT-NUMBER>
       Specify an SSL port number. Unless otherwise specified, the default port 443 used.

       -SSLMinVersion <TLS_1.0|TLS_1.1|TLS_1.2|TLS_1.3>
       Specify SSL version. Unless otherwise specified, the default value TLS_1.2 is used.

       -SSLCipherSuite <GOV|COM|ALL>
       Specify an SSL Cipher suite. Unless otherwise specified, the default value GOV is used.

Examples:
       enable-vdassl.sh -Enable -Certificate "/home/cert001.pem"
       Enable Linux VDA SSL using Certificate cert001.pem.

       enable-vdassl.sh -Enable -RootCertificate "/home/rootCR.cer"
       Enable Linux VDA SSL using Root Certificate rootCR.cer with local certificate(under /etc/xdl/.sslkeystore).

       enable-vdassl.sh -Enable -SSLPort 445
       Enable Linux VDA SSL on port 445 using local certificate(under /etc/xdl/.sslkeystore).

       enable-vdassl.sh -Enable -Certificate "/home/cert001.pem" -SSLPort 445
       Enable Linux VDA SSL using Certificate cert001.pem on port 445, with default SSLMinVersion and SSLCipherSuite.

       enable-vdassl.sh -Enable -Certificate "/home/cert001.pem" -SSLPort 445 -SSLMinVersion "TLS_1.2"
       Enable Linux VDA SSL using Certificate cert001.pem on port 445 with SSLMinVersion TLS_1.2 and default SSLCipherSuite..

       enable-vdassl.sh -Enable -Certificate "/home/cert001.pem" -SSLPort 445 -SSLMinVersion "TLS_1.2" -SSLCipherSuite "GOV"
       Enable Linux VDA SSL using Certificate cert001.pem on port 445 with SSLMinVersion TLS_1.2 and SSLCipherSuite GOV.
```

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 TLS encryption on the Controller

> **Note:**
>
> You can enable TLS encryption only for entire delivery groups. You cannot enable TLS encryption for specific applications.

In a PowerShell window on the Controller, run the following commands in sequence to enable TLS encryption for the target delivery group.

1. `Add-PSSnapin citrix.*`
2. `Get-BrokerAccessPolicyRule –DesktopGroupName 'GROUPNAME'| Set-BrokerAccessPolicyRule –HdxSslEnabled $true`

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

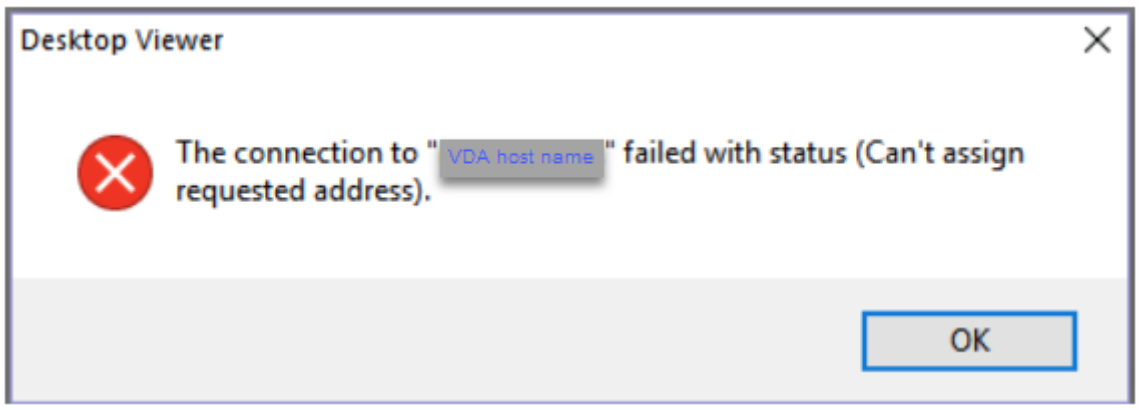
> **Note:**
>
> To ensure that only VDA FQDNs are contained in an ICA session file, you can also run the `Set-BrokerSite –DnsResolutionEnabled $true` command. The command enables DNS resolution. If you disable DNS resolution, an ICA session file discloses VDA IP addresses and provides FQDNs only for the TLS-related items such as `SSLProxyHost` and `UDPDTLSPort`.

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

1. `Add-PSSnapin citrix.*`
2. `Get-BrokerAccessPolicyRule –DesktopGroupName 'GROUPNAME'| Set-BrokerAccessPolicyRule –HdxSslEnabled $false`
3. `Set-BrokerSite –DnsResolutionEnabled $false`

Troubleshooting

The following "Can't assign requested address" error might occur in Citrix Workspace app for Windows when you try to access a published desktop session:



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

```
<IP address of the Linux VDA>      <FQDN of the Linux VDA>
```

On Windows machines, the **hosts** file typically locates at `C:\Windows\System32\drivers\etc\hosts`.

Session reliability

June 20, 2024

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Citrix introduces the session reliability feature to all supported Linux platforms. Session reliability is enabled by default.

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

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

Configuration

Policy settings in Citrix Studio

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

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

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

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

Settings on the Linux VDA

- **Enable/disable the session reliability TCP listener**

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

```
1 /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

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

> session reliability. Session reliability is still available through other listeners (for example, SSL) if the feature is enabled through the **Session reliability connections** policy.

- **Session reliability port number**

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

```
/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**

Keep-alive messages are sent between the Linux VDA and the client when there's no activity (for example, no mouse movement or screen update) in a session. 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:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

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. On the VDA, run the following command to verify that the session reliability TCP listener is running (using port 2598 as an example).

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

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

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

Session recording (preview)

June 20, 2024

You can record and replay sessions hosted on a Linux VDA.

> **Note:**
>
> This feature is in preview. Preview features might not be fully localized and are recommended for use in non-production environments. Issues found with preview features are not supported by Citrix Technical Support.

Enable or disable session recording

To enable or disable session recording for a Linux VDA, set **SmAudAllowed** to **1** or **0**, respectively. You can use the following commands:

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

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\
   CurrentControlSet\Control\Citrix\SmartAuditor" -t "REG_DWORD" -v "
   SmAudAllowed" -d "0x00000000" --force
```

> **Note:**
>
> After you enable session recording on a Linux VDA, users are notified about their sessions being recorded when they log on to their sessions.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Specify file size for recordings

As recordings grow in size, recording files take longer to download and respond more slowly when you use the seek slider to navigate during playback. To control file size, specify a threshold limit for a file. When the recording reaches this limit, the current file is closed, and an extra file is created to continue recording. This action is called a rollover.

Using the following commands, you can specify two thresholds for a rollover:

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\
    CurrentControlSet\Control\Citrix\SmartAuditor" -t "REG_DWORD" -v "
    RolloverFileSizeInMB" -d "0x00000032" --force

sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\
    CurrentControlSet\Control\Citrix\SmartAuditor" -t "REG_DWORD" -v "
    RolloverTimeInHours" -d "0x0000000c" --force
```

- **RolloverFileSizeInMB**. The current file closes when it reaches the size, and a new file opens. By default, the rollover occurs when the size exceeds 50 MB. Supported values: 10–300.
- **RolloverTimeInHours**. When the duration is reached, the current file closes and a new file opens. By default, the rollover occurs when the session records for 12 hours. Supported values: 1–24.

Rollovers occur when the first of the two conditions above is met. For example, you specify 17 MB for the file size and 6 hours for the duration. When your recording reaches 17 MB in 3 hours, session recording closes the file and opens a new one.

To prevent the creation of many small files, rollover doesn't happen until at least one hour elapses regardless of the value specified for the file size. The exception to this rule is if the file size surpasses 300 MB.

Specify where recordings are stored

Recording files are stored under `/var/xdl/session_recordings` by default. To specify a different path, run the following command:

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\
    CurrentControlSet\Control\Citrix\SmartAuditor" -t "REG_SZ" -v "Path"
     -d "<your custom storage path>" --force
```

You can store recordings on a local drive or a mount point that points to a network path. Configure proper access permissions to the storage path that you set and grant the user `ctxsrvr` the write permission to the path.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

View recordings

To view recordings, complete the following steps to install the Session Recording player or the Session Recording web player:

1. Use your Citrix account credentials to access the Citrix Virtual Apps and Desktops download page and download the product file. Unzip the file.
2. Double-click SessionRecordingPlayer.msi and SessionRecordingWebPlayer.msi and follow the instructions to complete the installation.

> **Tip:**
>
> To use the Session Recording web player, install it on the Session Recording server only and ensure that recordings are available on the Session Recording server. For more information, see the Citrix Session Recording documentation.

Limitations

- For virtual app sessions, recording notifications might not be centered.

Virtual Channel SDK (preview)

June 20, 2024

With the Virtual Channel Software Development Kit (SDK) for the Linux VDA, you can write server-side applications to run on the VDA. For more information, see the Citrix Virtual Channel SDK for the Linux VDA documentation.

The Citrix Virtual Channel SDK for the Linux VDA is available for download at the Citrix Virtual Apps and Desktops download page. Expand the appropriate version of Citrix Virtual Apps and Desktops and click **Components** to select the Linux VDA download.

> **Note:**
>
> This feature is in preview. Preview features might not be fully localized and are recommended for use in non-production environments. Citrix Technical Support doesn't support issues found with preview features.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Wayland (preview)

June 20, 2024

The Linux VDA supports Wayland in GNOME on RHEL 9.4/9.3/9.2, Rocky Linux 9.4/9.3/9.2, and Ubuntu 22.04. The following capabilities are fully tested in Wayland:

- Audio
- Clipboard
- Client drive mapping (CDM)
- Printing
- USB device redirection

> **Note:**
>
> - This feature is in preview. Preview features might not be fully localized and are recommended for use in non-production environments. Issues found with preview features **are not supported by** Citrix Technical Support.
> - HDX 3D Pro is not supported.
> - Linux virtual app sessions are not supported.

Enable Wayland

To use Wayland, set the registry key **EnableWayland** to **1** by running the following command:

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

By default, the registry key **EnableWayland** is set to **0**, which means X11 is used.

Check whether Wayland is in use

1. Open a Terminal window in Linux.
2. Run the **echo $XDG_SESSION_TYPE** command.

If Wayland is in use, you get '**wayland**' in the output.

Limitations

With Wayland in use, the following limitations are identified:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

- The keyboard layout of the client device is not synchronized with the keyboard layout of the VDA.

Best practices

June 20, 2024

This section guides you through the following procedures:

- Create Linux VDAs on Google Cloud Platform (GCP) using Machine Creation Services (MCS)

Create Linux VDAs on Google Cloud Platform (GCP) using Machine Creation Services (MCS)

June 20, 2024

To create Linux VDAs on GCP using MCS, complete the following steps:

Step 1: Create a Linux Virtual Machine (VM) on GCP

Step 2: Create a GCP service account

Step 3: Create a host connection to GCP in Citrix Studio

Step 4: Prepare a Linux VDA master image

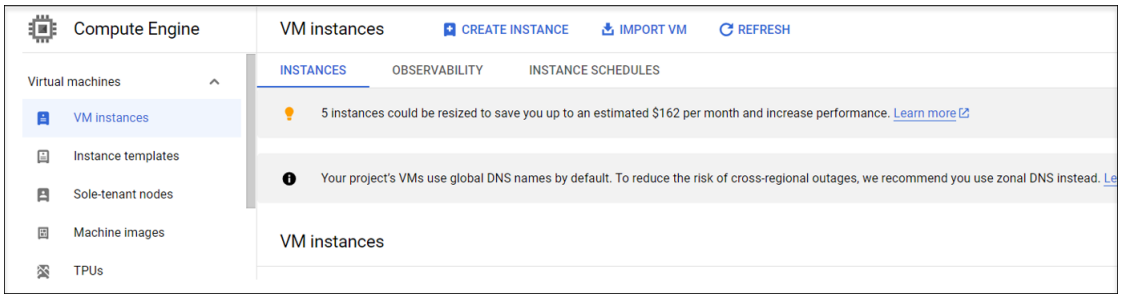
Step 5: Create a machine catalog

Step 6: Create a delivery group

Step 1: Create a Linux VM on GCP

Step 1a: In the Google Cloud console, navigate to **Compute Engine > VM instances**.

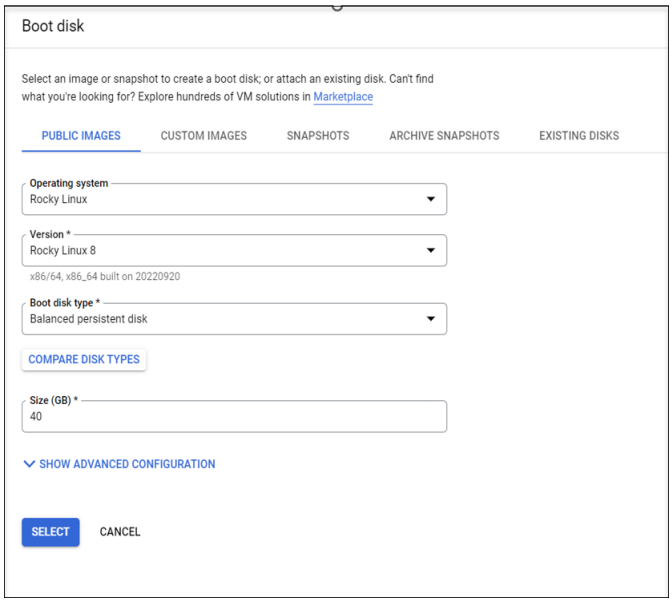
Step 1b: On the VM instances page, click **CREATE INSTANCE** to create a VM instance.



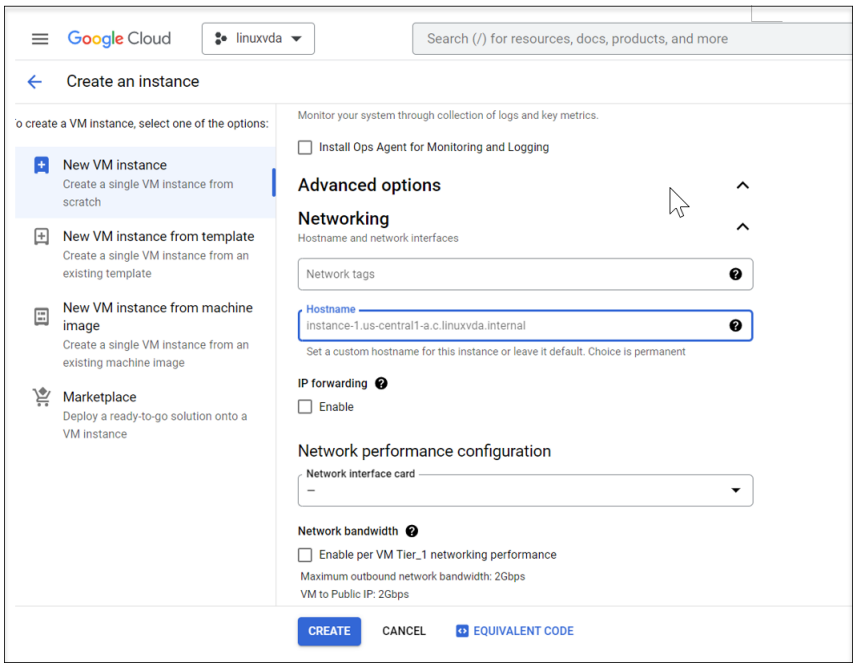
© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 1c: Make the following configurations and keep other configurations as default:

- Enter a name for your VM instance.
- Select a region and zone to host your VM.
- (Optional) Add a GPU to your VM. For more information, see **Step 4c** later in this article.
- In the **Boot disk** section, select the operating system and disk size for your VM. For example:



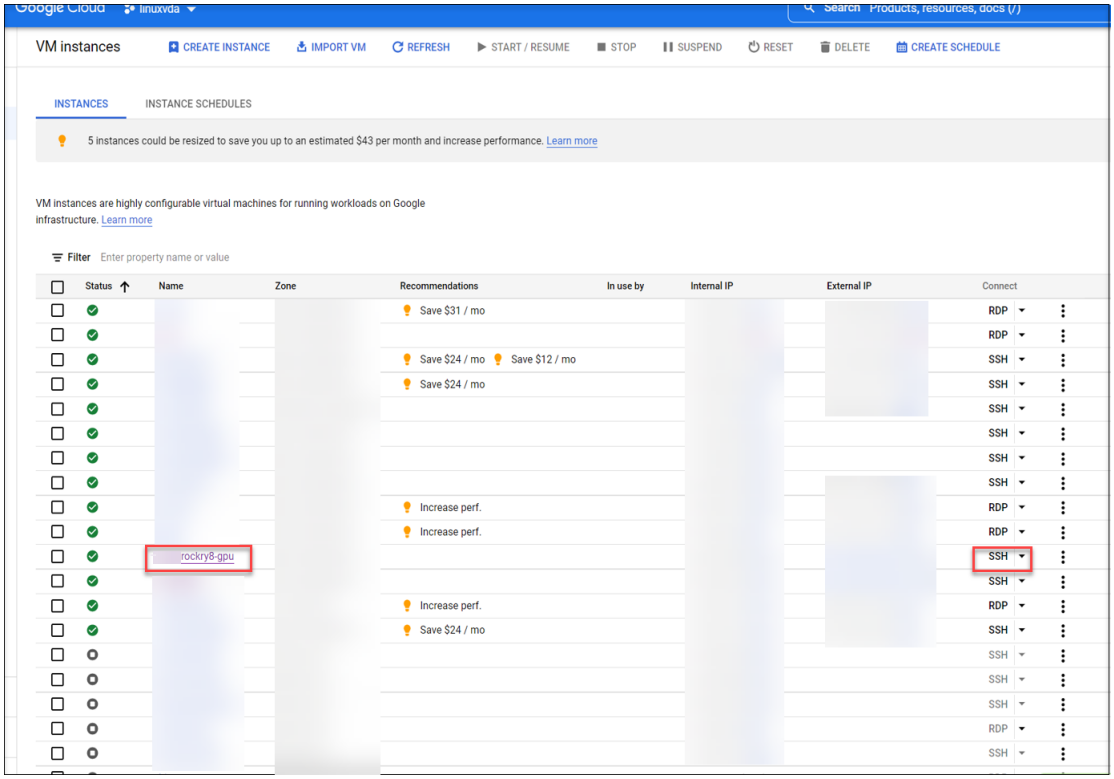
- Navigate to **Advanced options > Networking** and set the **Hostname** field to an FQDN.



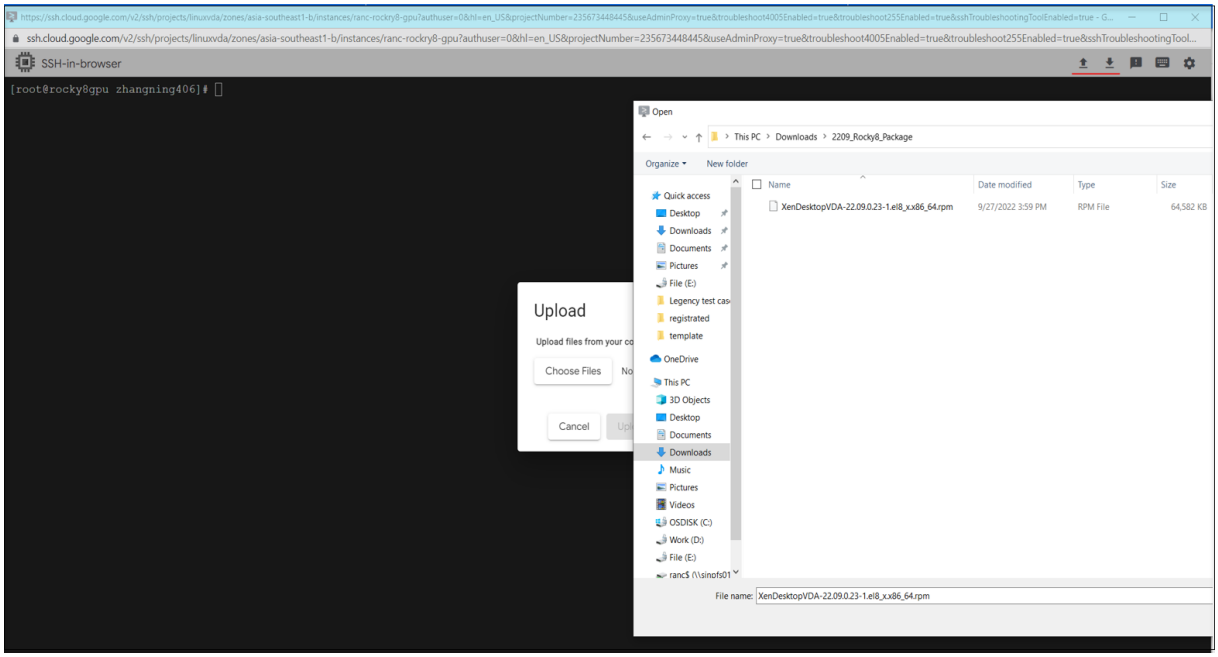
Step 1d: Click **Create** to create the VM instance.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Step 1e: After the VM is created, return to the **Compute Engine** dashboard, find your VM instance in the list, and click the SSH button to connect to your VM.



Step 1f: Upload the Linux VDA package to your VM through the web-based SSH client.



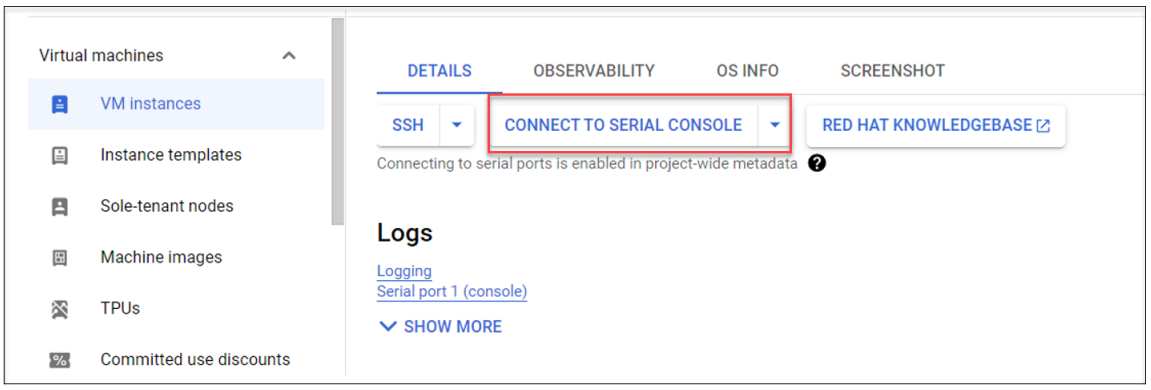
Step 1g: Prevent failures to access the VM using SSH.

The VM might be unreachable after a restart. To work around the issue, set a root password when

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

logging on to the VM for the first time and make sure that you can log on to the VM as root. Then, run the following commands in the console after restarting the VM:

```
nmcli dev connect eth0
systemctl restart NetworkManager
```



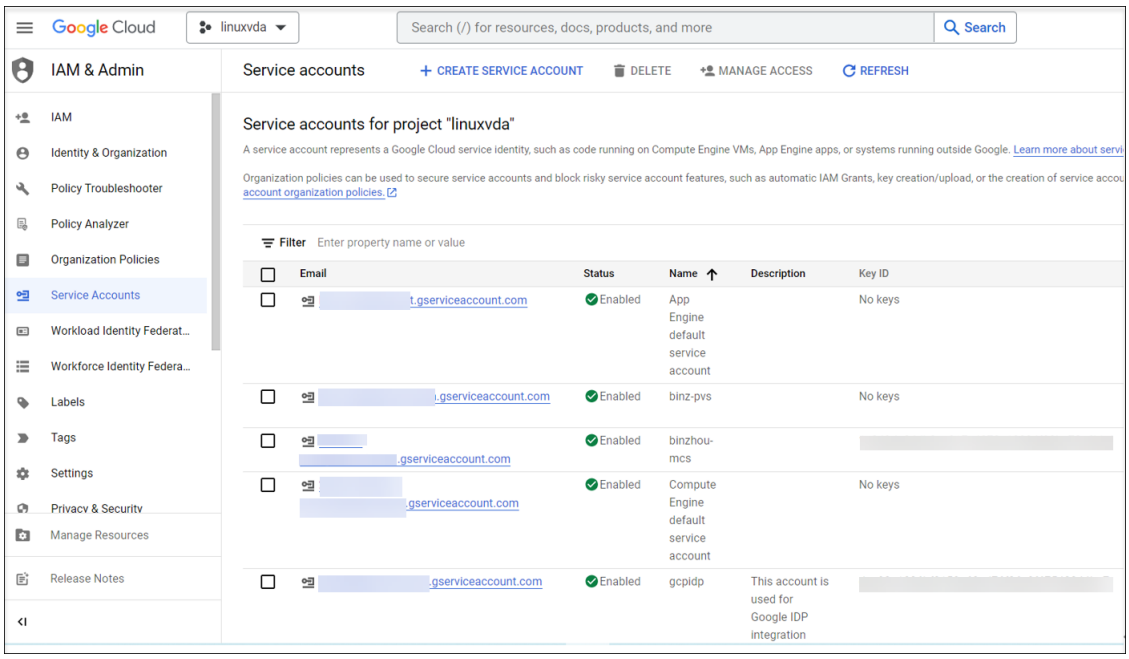
Step 2: Create a GCP service account

This section guides you through creating a GCP service account, including creating a service account key and granting the required roles to the service account.

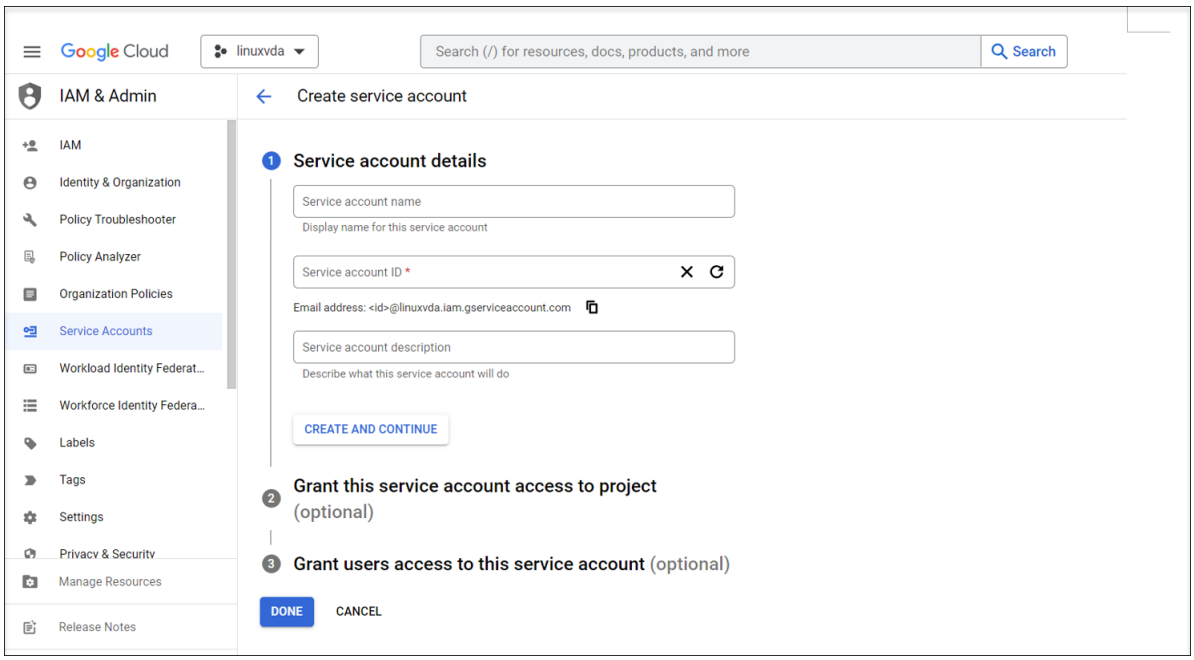
> **Note:**
>
> To create a GCP service account, ensure that you are granted the Service Account Admin (roles/iam.serviceAccountAdmin) IAM role.

Step 2a: In the Google Cloud console, navigate to **IAM & Admin > Service Accounts** and then Click the **Create Service Account** tab.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

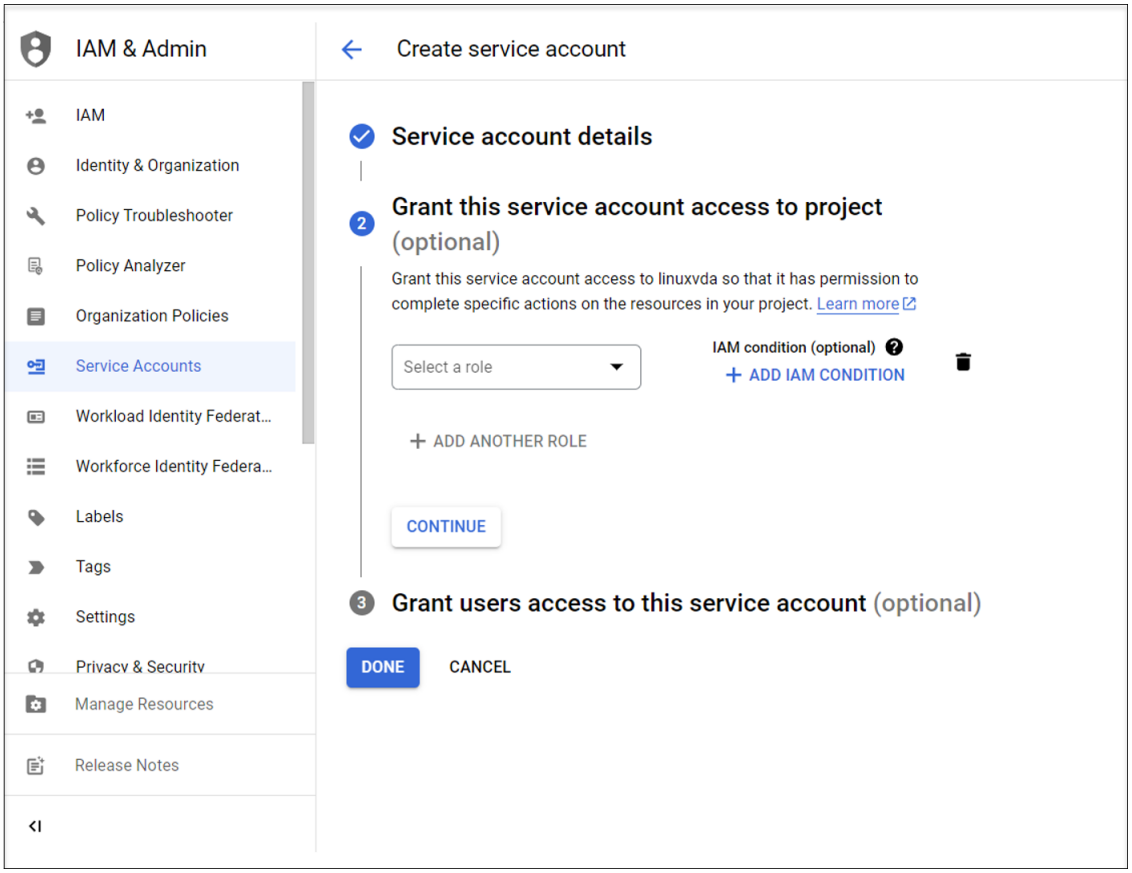


Step 2b: In the **Service account details** step, enter the values in the fields and then click **Create and continue**.



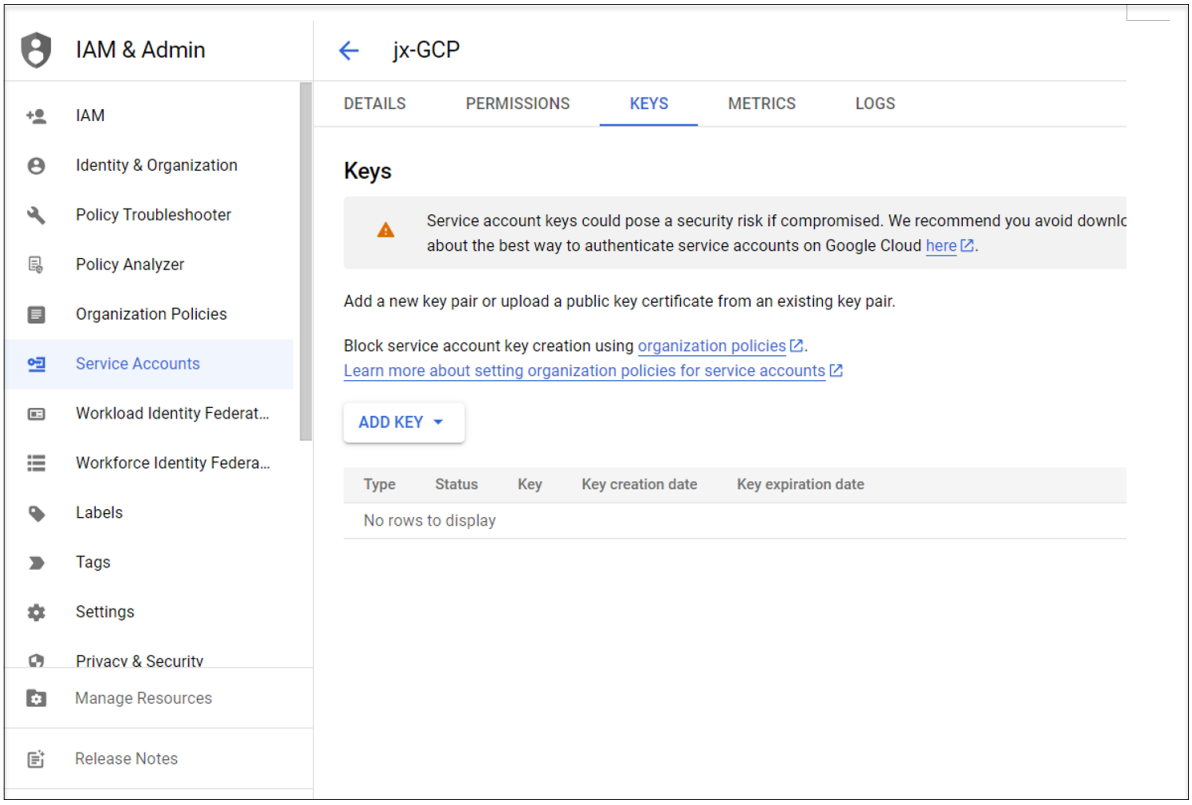
Step 2c: Skip the optional steps and click **Done** at the bottom.

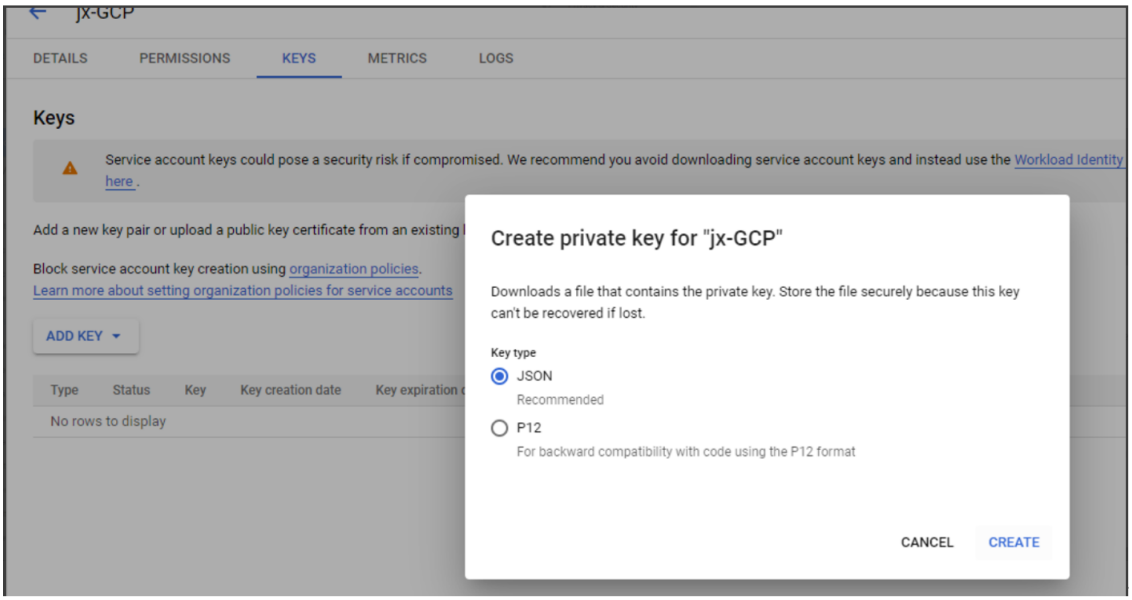
© 1999–2024 Cloud Software Group, Inc. All rights reserved.



Step 2d: Navigate to **IAM & Admin > Service Accounts** again and then click the **Service accounts** tab. Find the newly created service account, navigate to the **Keys** tab, and then click **Add key > Create new key > JSON > Create**.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.





> **Note:**
>
> After you download the key file, you can't download it again.

Step 2e: In the Google Cloud console, navigate to **IAM & Admin > IAM** and then click **Add**. Search for and select the newly created service account in the **New members** field and then select a role for the service account to grant it access to your resources. Keep granting roles by clicking **Add another role**

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

to ensure that you grant all the following roles to the newly created service account.

- **Compute Admin**
- **Storage Admin**
- **Cloud Build Editor**
- **Service Account User**
- **Cloud Datastore User**
- **Compute Instance Admin (beta)**
- **Owner**

For example:

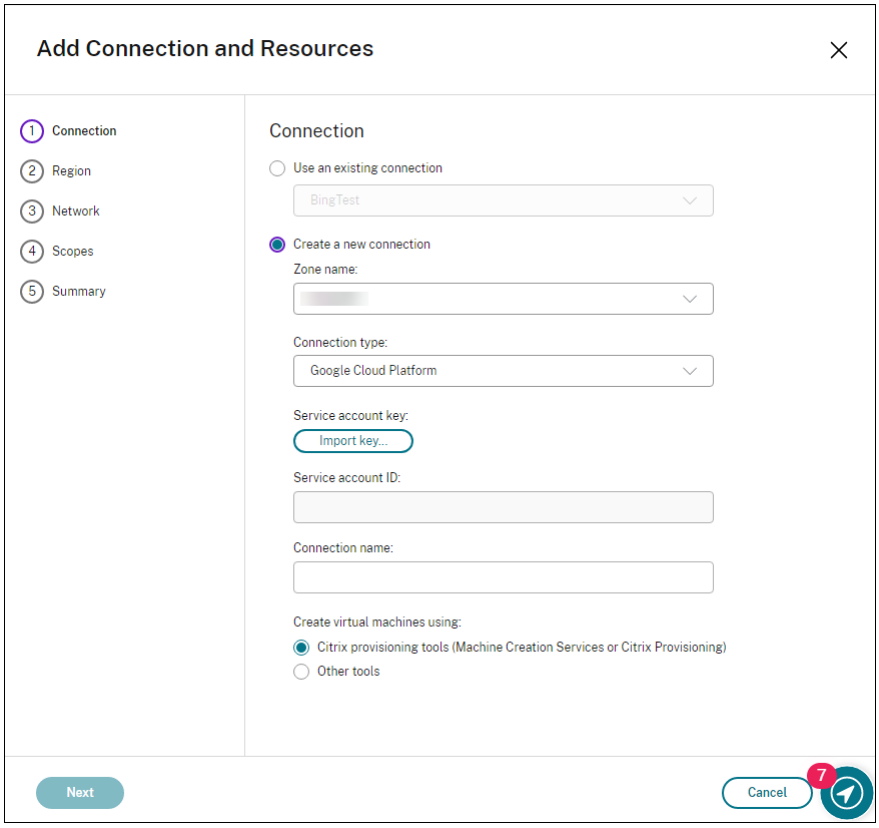
Step 3: Create a host connection to GCP in Citrix Studio

Set up your GCP environment according to Google Cloud Platform virtualization environments and then complete the following steps to create a host connection to GCP.

1. For on-premises Delivery Controllers, choose **Configuration > Hosting > Add Connection and Resources** in the on-premises Citrix Studio to create a host connection. For cloud Delivery Controllers, choose **Manage > Hosting > Add Connection and Resources** in the web-based Studio console on Citrix Cloud to create a host connection.
2. In the **Add Connection and Resources** wizard, select **Google Cloud Platform** as the connection type.

   For example, in the web-based Studio console on Citrix Cloud:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



3. Import the service account key of your GCP account and type your connection name.
4. The wizard guides you through the pages. Specific page content depends on the selected connection type. After completing each page, select **Next** until you reach the **Summary** page. For more information, see Step 2: Create a host connection in the Create non-domain-joined Linux VDAs using MCS article.

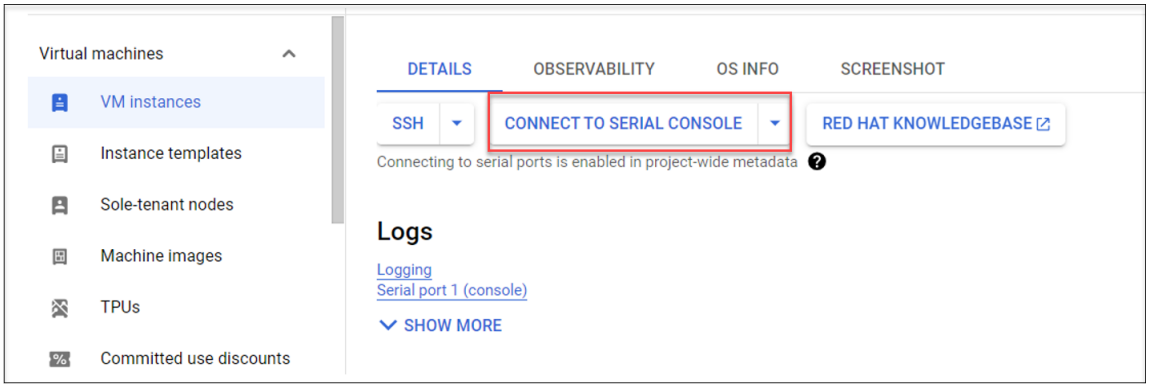
Step 4: Prepare a Linux VDA master image

Step 4a: (For RHEL 8.x/9.x and Rocky Linux 8.x/9.x) Configure Ethernet connection.

After you install the Linux VDA on RHEL 8.x/9.x and Rocky Linux 8.x/9.x hosted on GCP, the Ethernet connection might be lost and the Linux VDA might be unreachable after a VM restart. To work around the issue, set a root password when logging on to the VM for the first time and make sure that you can log on to the VM as root. Then, run the following commands in the console after restarting the VM:

```
1 nmcli dev connect eth0
2 systemctl restart NetworkManager
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



Step 4b: Install the Linux VDA package on the template VM.

On the template VM, do the following to install the Linux VDA package:

1. Install .NET.

   In addition to the .NET Runtime, you must install .ASP.NET Core Runtime on all supported Linux distributions before you install or upgrade the Linux VDA. Version 6 is required for Amazon Linux 2. Version 8 is required for other distributions.

   If your Linux distribution contains the .NET version that you require, install it from the built-in feed. Otherwise, install .NET from the Microsoft package feed. For more information, see https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers.

2. Install the Linux VDA package:

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

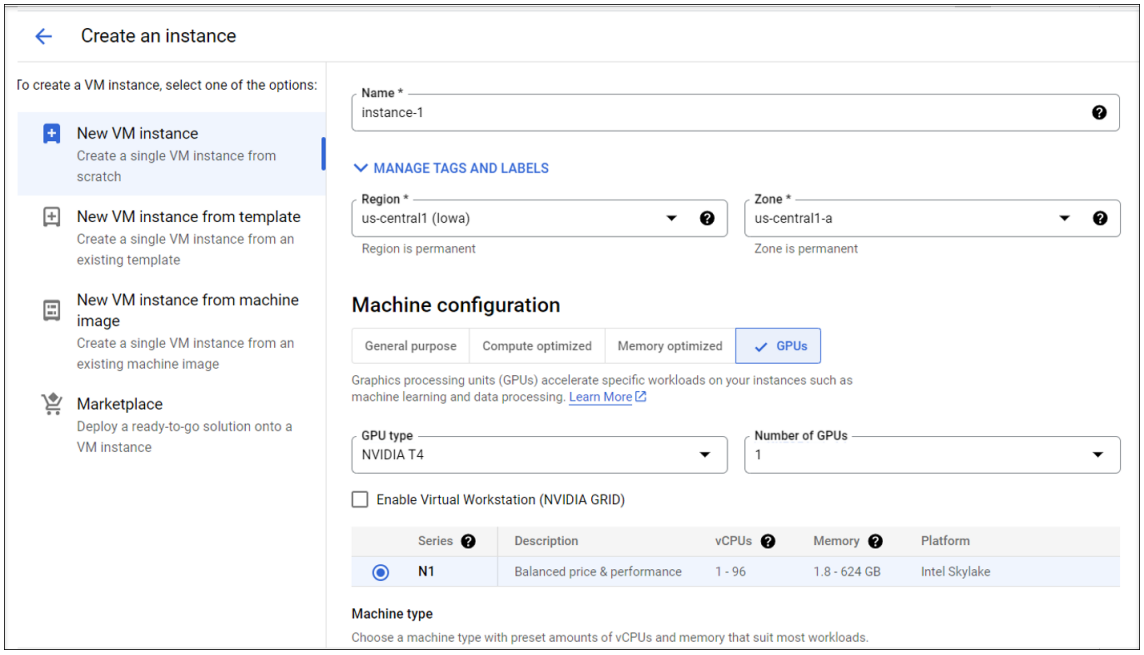
3. Enable the EPEL repository:

   ```
   sudo rpm -ivh https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
   ```

Step 4c: Add a Graphics Processing Unit (GPU) to the template VM on GCP (optional).

1. In the Google Cloud console, add one or more GPUs to the template VM. For information about adding and removing GPUs on GCP, see https://cloud.google.com/compute/docs/gpus/add-remove-gpus.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



2. Install the appropriate GPU driver on the template VM. For more information, see https://cloud.google.com/compute/docs/gpus/install-drivers-gpu.

   Required NVIDIA driver versions:

   NVIDIA GPUs running on Compute Engine must use the following NVIDIA driver versions:

   - For L4 GPUs:
     - Linux : 525.60.13 or later
   - For A100 GPUs:
     - Linux : 450.80.02 or later
   - For T4, P4, P100, and V100 GPUs:
     - Linux : 410.79 or later
   - For K80 GPUs (End-of-life):
     - Linux : 410.79 - latest R470 version

   For K80 GPUs, NVIDIA has announced that the R470 driver branch is the final driver version to receive debug support. To review this update, see NVIDIA Software Support Matrix.

   Installation script:

   You can use the following script to automate the installation process:

   ```
   https://raw.githubusercontent.com/GoogleCloudPlatform/compute-gpu-installation/main/linux/install_gpu_driver.py --output install_gpu_driver.py
   ```

   Supported operating systems:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

The installation script was tested on the following Linux distributions:

- Debian 11
- Red Hat Enterprise Linux (RHEL) 8
- Rocky Linux 8
- Ubuntu 20/22

If you use this script on other Linux distributions, the installation fails. For Linux VMs, this script installs only the NVIDIA driver.

a) Download the installation script.

```
curl https://raw.githubusercontent.com/GoogleCloudPlatform/
    compute-gpu-installation/main/linux/install_gpu_driver.py
    --output install_gpu_driver.py
```

b) Grant complete access to the script.

```
chmod 777 install_gpu_driver.py
```

c) Run the installation script.

```
python3 install_gpu_driver.py
```

d) Disable Wayland in gdm3.

- Locate your distribution's Wayland configuration file in one of the following locations:
  - /etc/gdm3/custom.conf (Ubuntu)
  - /etc/gdm/custom.conf (CentOS, RHEL, Rocky Linux)
- Open the file with sudo/root privileges.
- Uncomment **WaylandEnable=false** by deleting the **#** at the beginning of the line.
- Restart the VM.

e) If you installed an NVIDIA 510 or later driver, disable the GSP firmware.

If GSP firmware is enabled, disable it by setting the NVIDIA module parameter **NVreg_EnableGpuFirmware** to 0.

Set this parameter by adding the following entry to the /etc/modprobe.d/nvidia.conf file:

- options nvidia NVreg_EnableGpuFirmware=0
- If the /etc/modprobe.d/nvidia.conf file does not exist, create it.

When completing this step, keep the following in mind:

- Use sudo to run the commands to create and update the configuration file.
- To restart the VM, you can use **sudo reboot** in the Linux terminal or stop and start the VM.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

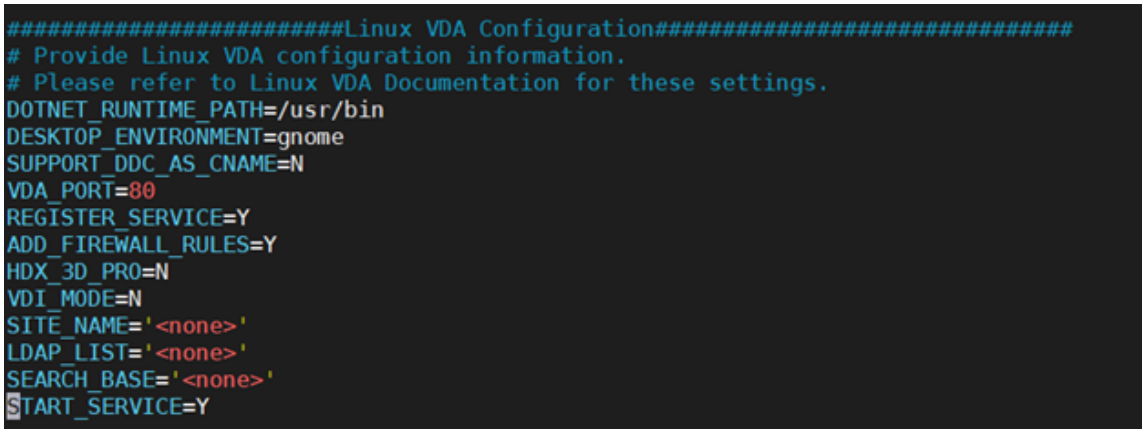
Step 4d: Configure MCS variables.

Configure MCS variables by editing the /etc/xdl/mcs/mcs.conf file. The following are MCS variables that you can configure for non-domain-joined and domain-joined scenarios:

- **For non-domain-joined scenarios**

  You can use the default variable values or customize the variables as required (optional):

  DOTNET_RUNTIME_PATH=**path-to-install-dotnet-runtime **
  DESKTOP_ENVIRONMENT= **gnome | mate **
  REGISTER_SERVICE=**Y | N**
  ADD_FIREWALL_RULES=**Y | N**
  VDI_MODE=**Y | N**
  START_SERVICE=**Y | N**



- **For domain-joined scenarios**

  - `Use_AD_Configuration_Files_Of_Current_VDA`: Determines whether to use the existing AD-related configuration files (/etc/krb5.conf, /etc/sssd.conf, and /etc/samba/smb.conf) of the currently running VDA. If set to Y, the configuration files on MCS-created machines are the same as the equivalents on the currently running VDA. However, you still must configure the `dns` and `AD_INTEGRATION` variables. The default value is N, which means the configuration templates on the master image determine the configuration files on MCS-created machines. To use a currently running VDA as the template VM, set the value to Y. Otherwise, set it to N.
  - `dns`: Sets the IP address for each DNS server. You can set up to four DNS servers.
  - `NTP_SERVER`: Sets the IP address for your NTP server. Unless otherwise specified, it's the IP address of your domain controller.
  - `WORKGROUP`: Sets the workgroup name to the NetBIOS name (case-sensitive) that you configured in AD. Otherwise, MCS uses the part of the domain name that immediately follows the machine hostname as the workgroup name. For example, if the machine account

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

is **user1.lvda.citrix.com**, MCS uses **lvda** as the workgroup name while **citrix** is the correct choice. Ensure that you set the workgroup name correctly.

- `AD_INTEGRATION`: Sets Winbind, SSSD, PBIS, or Centrify. For a matrix of the Linux distributions and domain joining methods that MSC supports, see Supported distributions.
- `CENTRIFY_DOWNLOAD_PATH`: Sets the path for downloading the Server Suite Free (formerly Centrify Express) package. The value takes effect only when you set the `AD_INTEGRATION` variable to Centrify.
- `CENTRIFY_SAMBA_DOWNLOAD_PATH`: Sets the path for downloading the Centrify Samba package. The value takes effect only when you set the `AD_INTEGRATION` variable to Centrify.
- `PBIS_DOWNLOAD_PATH`: Sets the path for downloading the PBIS package. The value takes effect only when you set the `AD_INTEGRATION` variable to PBIS.
- `UPDATE_MACHINE_PW`: Enables or disables automating machine account password updates. For more information, see Automate machine account password updates.
- Linux VDA configuration variables:

  `DOTNET_RUNTIME_PATH`=**path-to-install-dotnet-runtime **
  `DESKTOP_ENVIRONMENT`= **gnome | mate **
  `SUPPORT_DDC_AS_CNAME`=**Y | N**
  `VDA_PORT`=**port-number**
  `REGISTER_SERVICE`=**Y | N**
  `ADD_FIREWALL_RULES`=**Y | N**
  `HDX_3D_PRO`=**Y | N**
  `VDI_MODE`=**Y | N**
  `SITE_NAME`=**dns-site-name | '<none>'**
  `LDAP_LIST`=**'list-ldap-servers'| '<none>'**
  `SEARCH_BASE`=**search-base-set | '<none>'**
  `FAS_LIST`=**'list-fas-servers'| '<none>'**
  `START_SERVICE`=**Y | N**
  `TELEMETRY_SOCKET_PORT`=**port-number**
  `TELEMETRY_PORT`=**port-number**

Step 4e: Create a master image

1. (For SSSD + RHEL 8.x/9.x or Rocky Linux 8.x/9.x only) Run the `update-crypto-policies --set DEFAULT:AD-SUPPORT` command and then restart the template VM.
2. Run `/opt/Citrix/VDA/sbin/deploymcs.sh`.

   For non-domain-joined scenarios, the following error is normal and does not block you from proceeding.

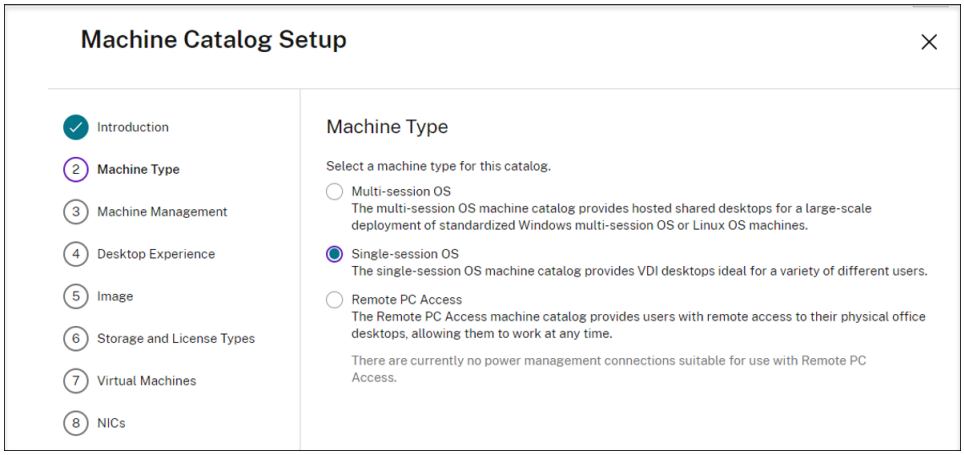
© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
[root@localhost ~]#
[root@localhost ~]# /opt/Citrix/VDA/sbin/deploymcs.sh
Installing Linux VDA dependency packages ...
Installing package redhat-lsb-core
Installing package nautilus
Installing package totem-nautilus
Installing package brasero-nautilus
Installing package pulseaudio
Installing package pulseaudio-module-x11
Installing package pulseaudio-gdm-hooks
Installing package pulseaudio-module-bluetooth
Installing package alsa-plugins-pulseaudio
Installing package pciutils
Installing package openssh
Installing package openssh-clients
Installing package java-11-openjdk
Installing package chrony
Installing package krb5-workstation
Installing package oddjob-mkhomedir
Starting PostgreSQL database ...
Installing package tdb-tools
Installing package ntfs-3g
/opt/Citrix/VDA/lib64/mcs ~
installing mcs systemd unit file: ad_join.service ...
~
/opt/Citrix/VDA/lib64/mcs ~
~
Installing winbind dependency packages ...
Installing package samba-winbind
Installing package samba-winbind-clients
ERROR: Exit funtion conf_dns, dns not configured, please do it manually.
[root@localhost ~]#
```

3. Install applications on the template VM and shut down the template VM. Create and name a snapshot of your master image.

Step 5: Create a machine catalog

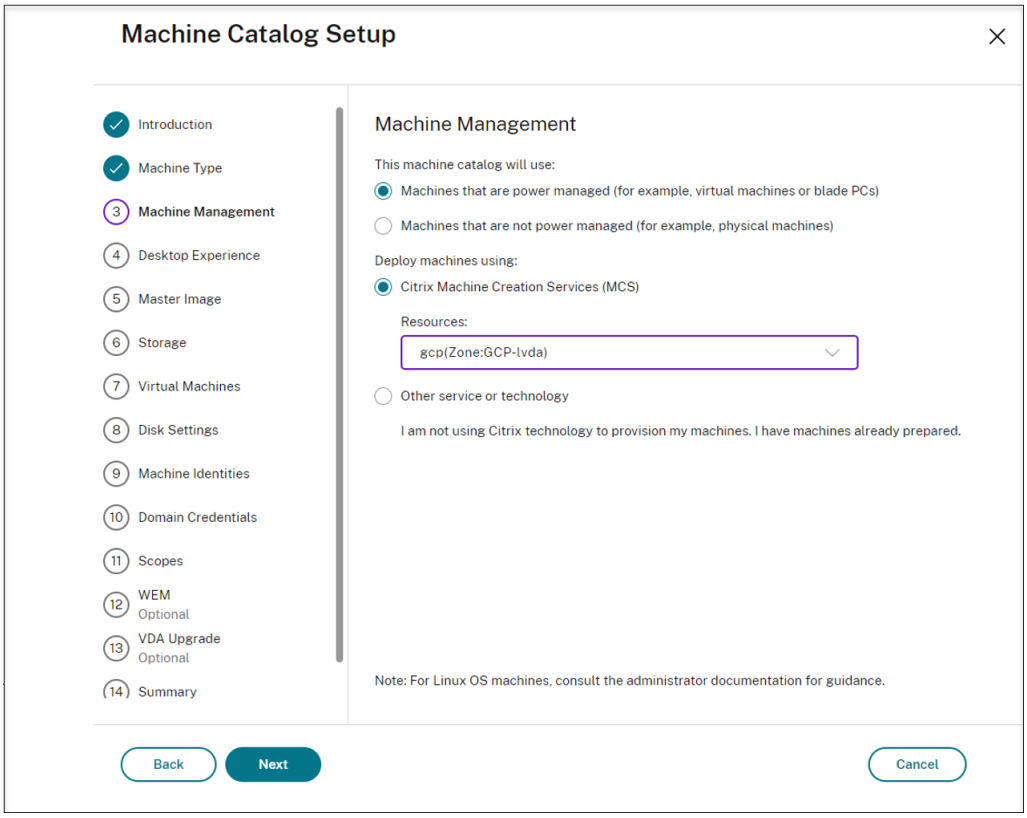
1. Sign in to Citrix Cloud. In the upper left menu, select **My Services > DaaS**.
2. From **Manage > Full Configuration**, select **Machine Catalogs** in the left pane.
3. Select **Create Machine Catalog** in the action bar.
4. On the **Machine Type** page, select **Multi-session OS** or **Single-session OS** and then select **Next**.



5. On the **Machine Management** page, select the **Machines that are power managed** and the **Citrix Machine Creation Services** options and then select **Next**. If there are multiple resources,

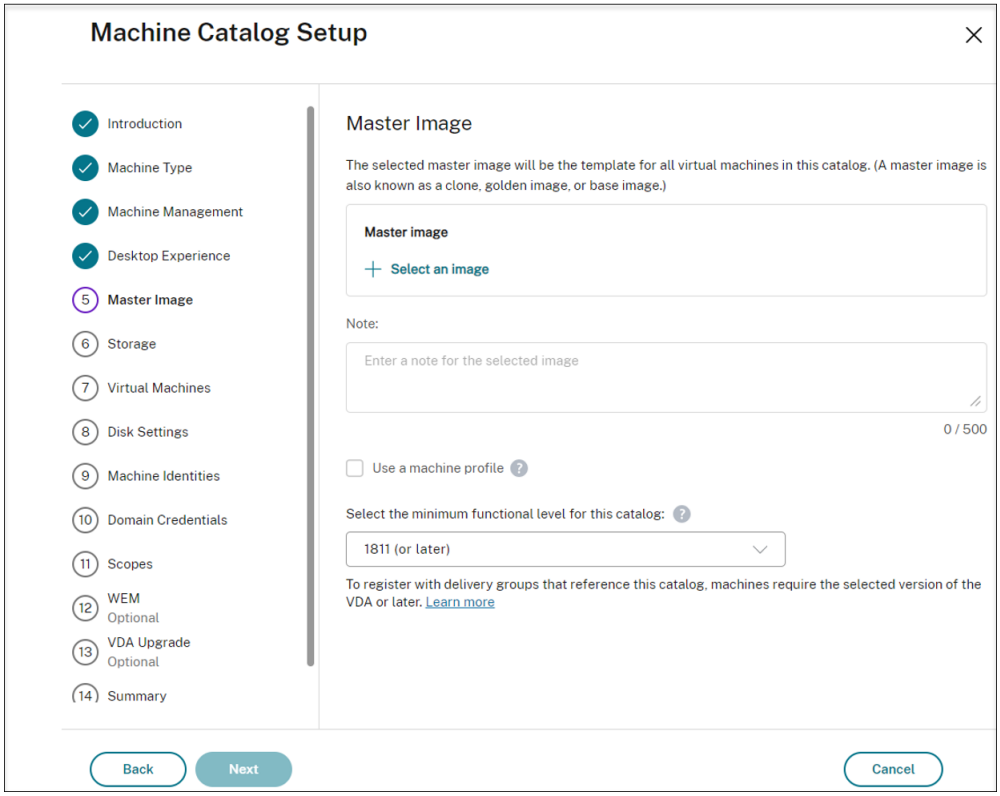
© 1999–2024 Cloud Software Group, Inc. All rights reserved.

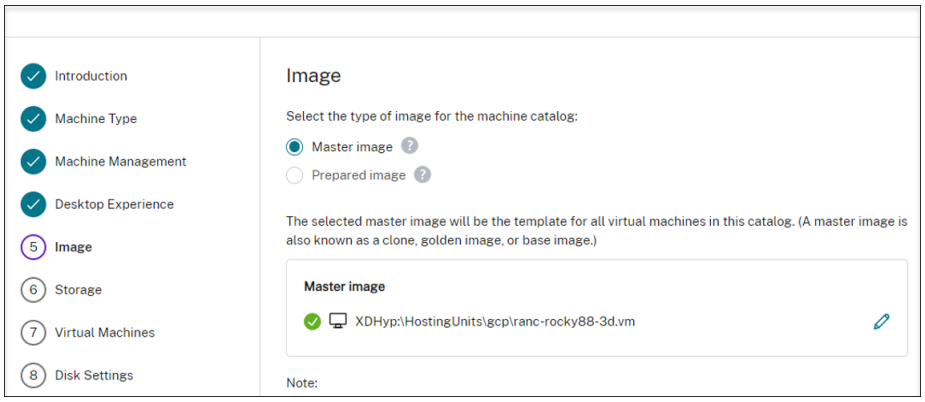
select one from the menu.



6. On the **Master Image** page, select the master image created earlier.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

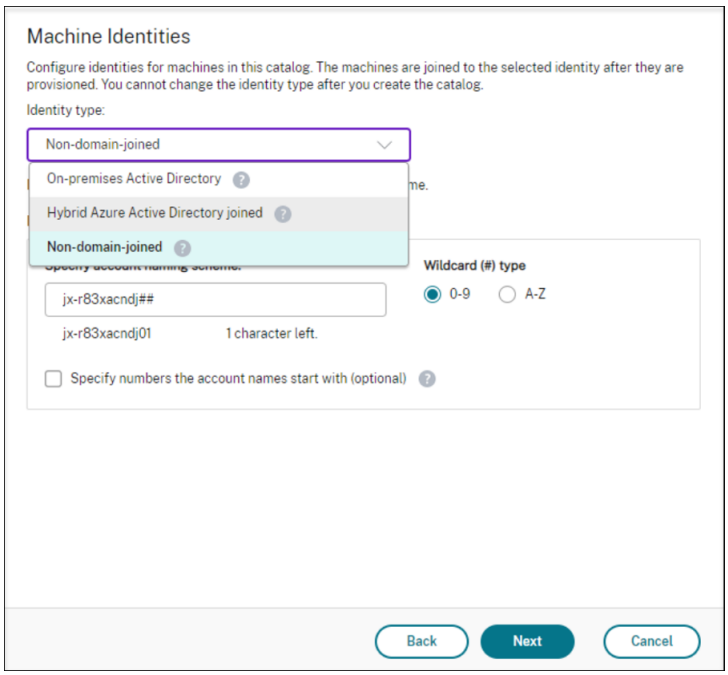




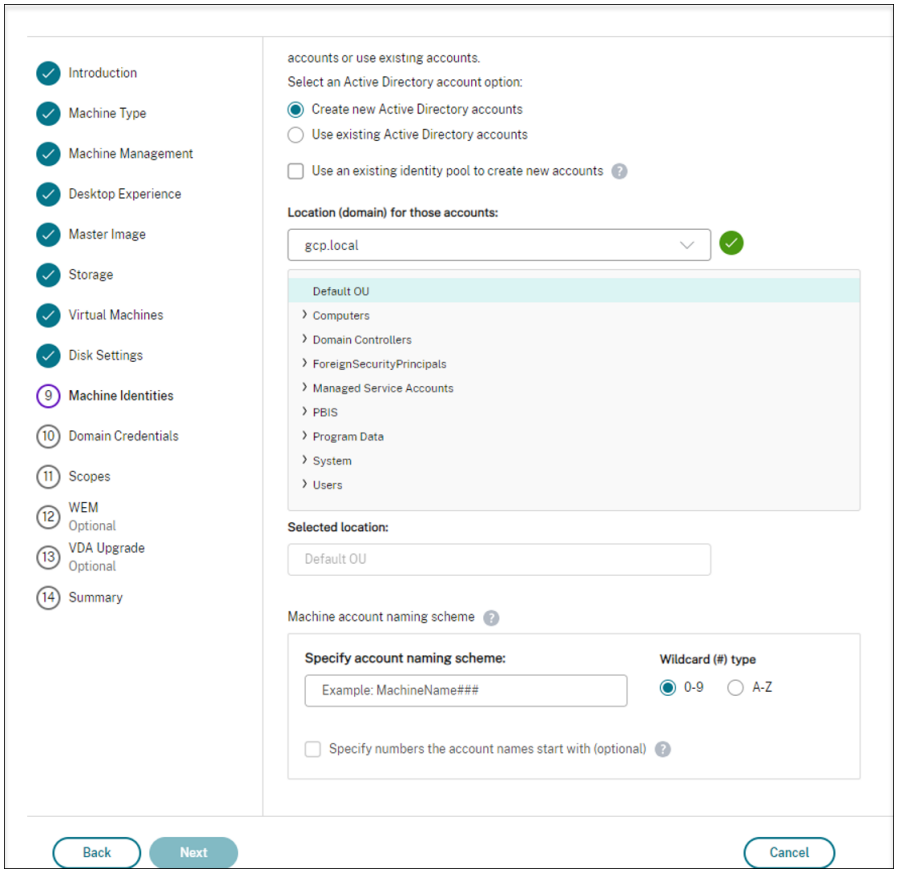
7. On the **Machine Identities** page, select **Non-domain-joined** if the master image wasn't joined to any domain or select an Active Directory account if you joined the master image to a domain.

   For non-domain-joined scenarios:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



For domain-joined scenarios:



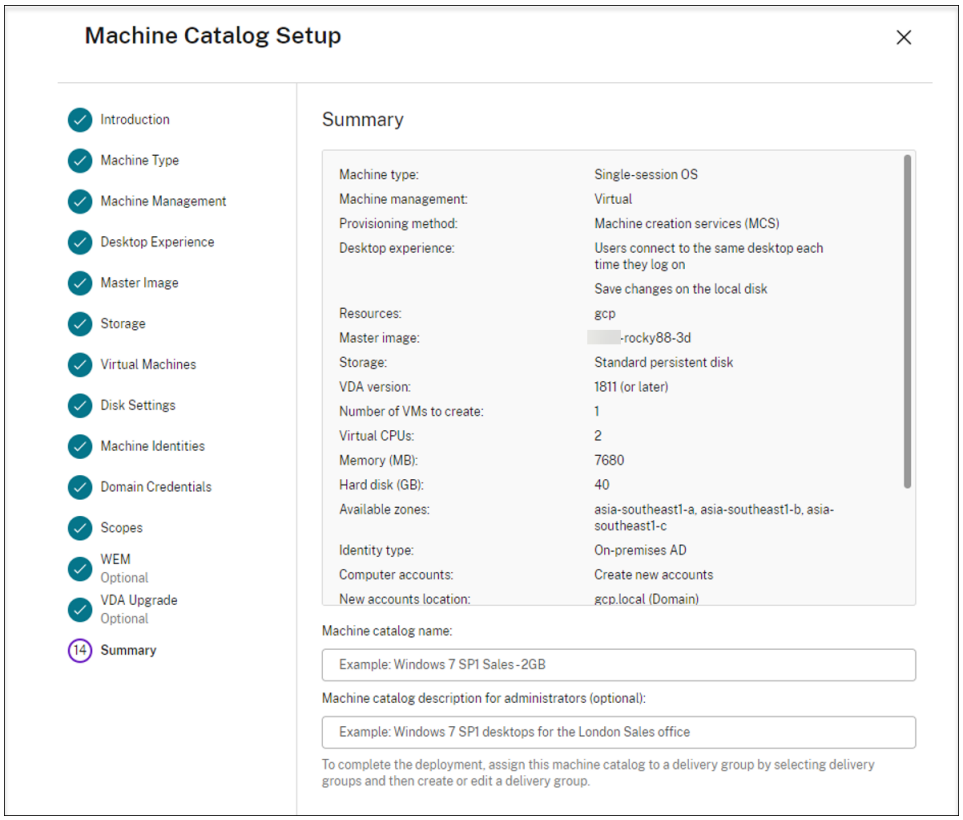
- If you select **Create new Active Directory accounts**, select a domain and then enter the sequence of characters representing the naming scheme for the provisioned VM computer

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

accounts created in the Active Directory. The account naming scheme can contain 1–64 characters, and cannot contain blank spaces, or non-ASCII or special characters.

- If you select **Use existing Active Directory accounts**, select **Browse** to navigate to the existing Active Directory computer accounts for the selected machines.
- On the **Domain Credentials** page, select **Enter credentials**, type the user name and password, select **Save**, and then select **Next**. The credential you type must have permission to perform Active Directory account operations.

8. Configure additional settings on the other pages. For more information, see Create a Google Cloud Platform catalog.
9. On the **Summary** page, confirm the information, specify a name for the catalog, and then select **Finish**.



Machine catalog creation might take a long time to complete. When it completes, the catalog is listed. You can verify that the machines are created on the target node groups in the Google Cloud console.

Step 6: Create a delivery group

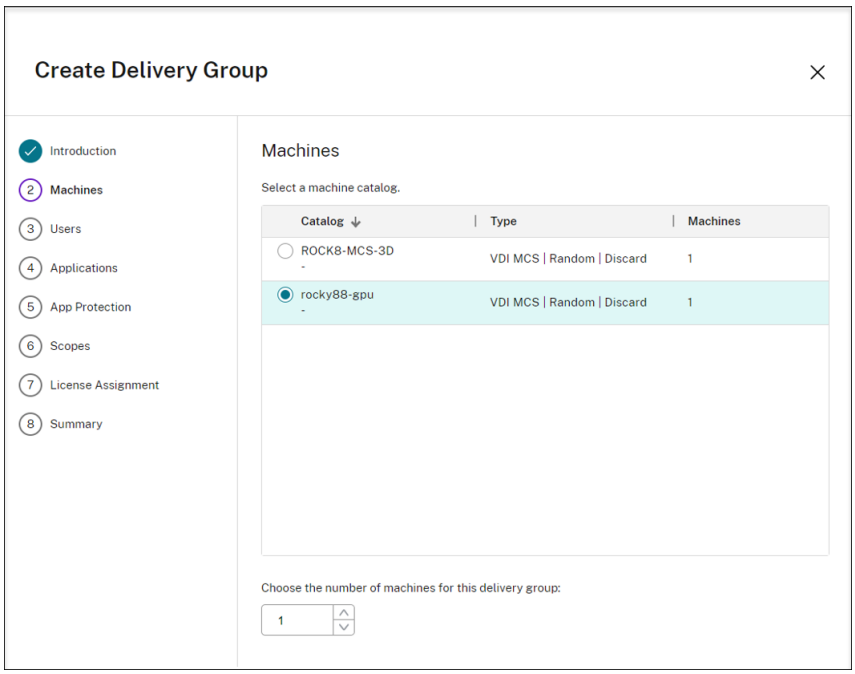
A delivery group is a collection of machines selected from one or more machine catalogs. It specifies which users can use those machines, and the applications and desktops available to those users.

For more information, see delivery group creation in the Citrix DaaS documentation.

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

1. Sign in to Citrix Cloud. In the upper left menu, select **My Services > DaaS**.
2. From **Manage > Full Configuration**, select **Delivery Groups** in the left pane.
3. Select **Create Delivery Group** in the action bar. The delivery group creation wizard opens.

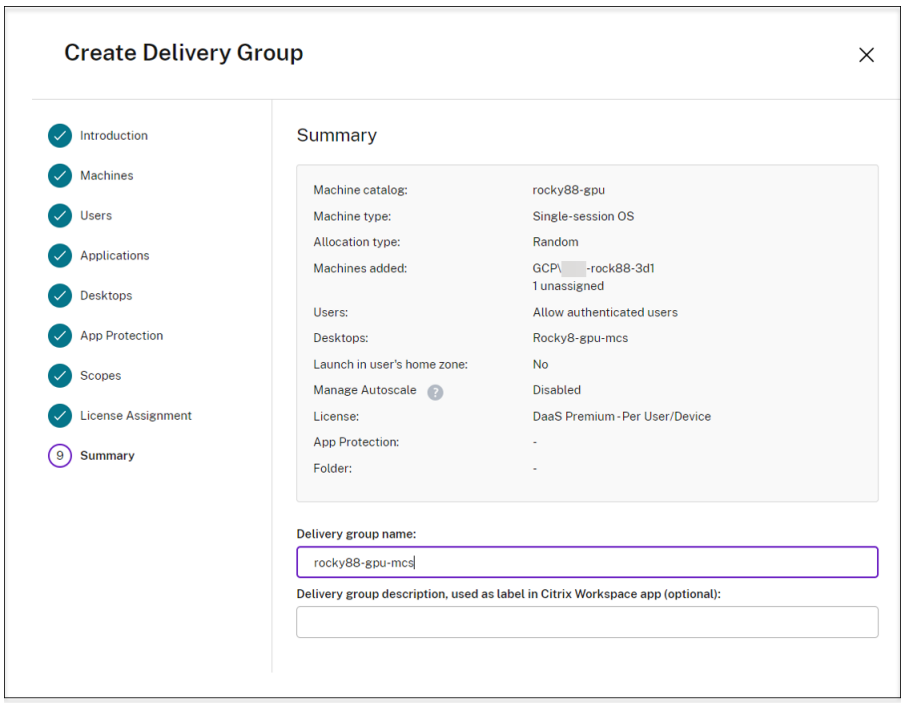
   The wizard pages that you see might be different, depending on the selections you make.
4. On the **Machines** page, select a machine catalog and select the number of machines you want to use from that catalog.



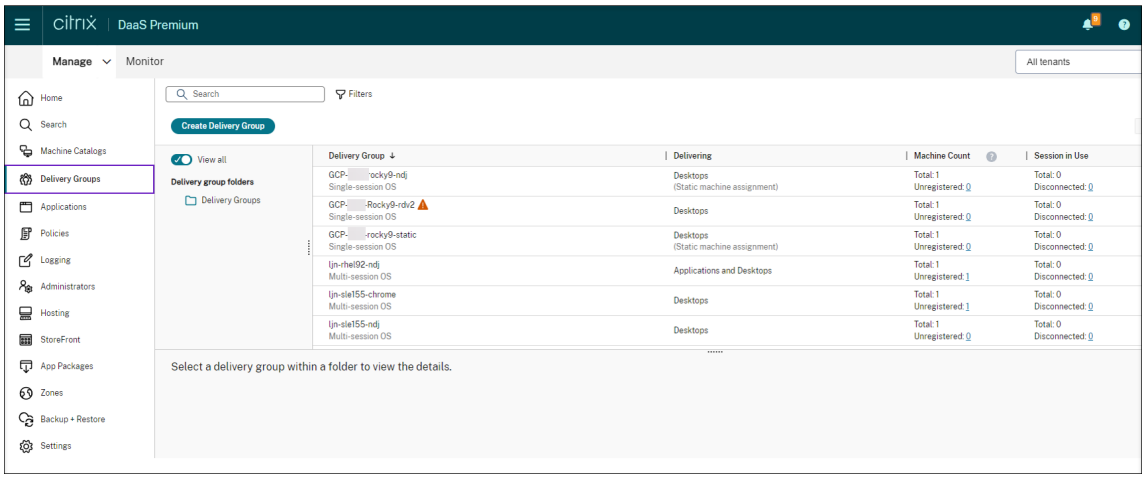
5. Configure additional settings on the other pages. For more information, see Create delivery groups.
6. On the **Summary** page, enter a name for the delivery group. You can also (optionally) enter a description, which appears in Citrix Workspace app and in the **Full Configuration** management interface. For example:

   Enter a name for the delivery group:

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

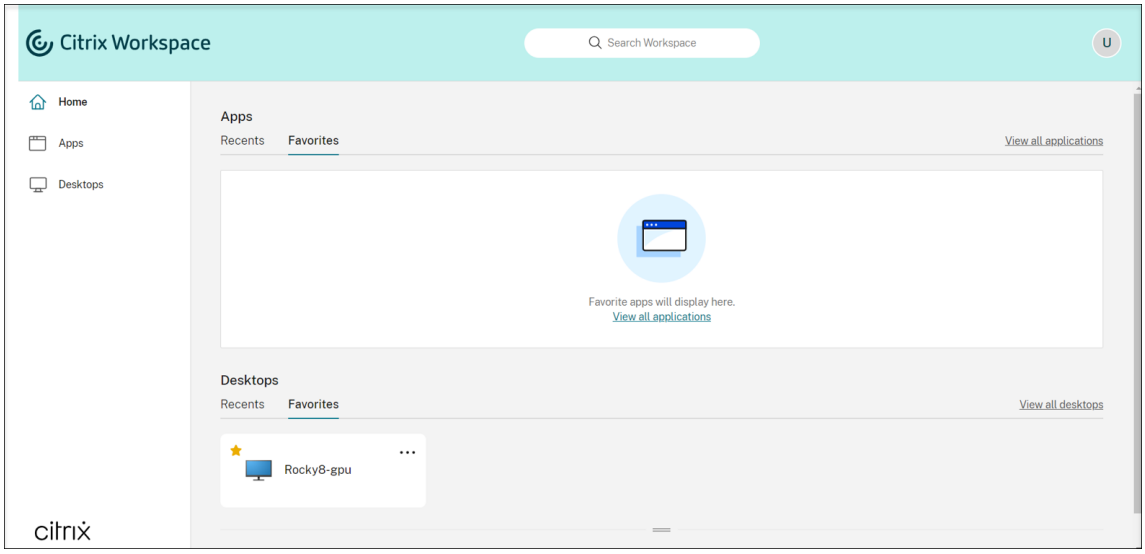


List of delivery groups in the **Full Configuration** management interface:



Delivered machine in Citrix Workspace app

© 1999–2024 Cloud Software Group, Inc. All rights reserved.



Manage your deployment using Ansible

July 16, 2024

Ansible helps to automate the process of deploying applications, configurations, and updates across your deployment. This article offers step-by-step instructions on using Ansible for managing your deployment with efficiency.

Step 1: Determine what to deploy

Before you start, identify what you need to deploy, such as applications, services, configurations, and environment variables.

Step 2: Set up your Ansible project

Create a directory structure for your Ansible project. One crucial way to organize your playbook content is Ansible's "roles"feature. For more information, see Roles in the Ansible documentation.

The following are two example directory structures for your reference:

Example directory structure #1

```
production                # inventory file for production servers
staging                   # inventory file for staging environment

```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
group_vars/
   group1.yml             # here we assign variables to particular
      groups
   group2.yml
host_vars/
   hostname1.yml          # here we assign variables to particular
      systems
   hostname2.yml

library/                  # if any custom modules, put them here (
   optional)
module_utils/             # if any custom module_utils to support
   modules, put them here (optional)
filter_plugins/           # if any custom filter plugins, put them here
    (optional)

site.yml                  # master playbook
webservers.yml            # playbook for webserver tier
dbservers.yml             # playbook for dbserver tier

roles/
    common/               # this hierarchy represents a "role"
        tasks/            #
            main.yml      #  <-- tasks file can include smaller files
               if warranted
        handlers/         #
            main.yml      #  <-- handlers file
        templates/        #  <-- files for use with the template
           resource
            ntp.conf.j2   #  <------- templates end in .j2
        files/            #
            bar.txt       #  <-- files for use with the copy resource
            foo.sh        #  <-- script files for use with the script
               resource
        vars/             #
            main.yml      #  <-- variables associated with this role
        defaults/         #
            main.yml      #  <-- default lower priority variables for
               this role
        meta/             #
            main.yml      #  <-- role dependencies
        library/          # roles can also include custom modules
        module_utils/     # roles can also include custom module_utils
        lookup_plugins/   # or other types of plugins, like lookup in
           this case

    webtier/              # same kind of structure as "common" was
       above, done for the webtier role
    monitoring/           # ""
    fooapp/               # ""
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Example directory structure #2

```
inventories/
   production/
      hosts               # inventory file for production servers
      group_vars/
         group1.yml       # here we assign variables to particular
            groups
         group2.yml
      host_vars/
         hostname1.yml    # here we assign variables to particular
            systems
         hostname2.yml

   staging/
      hosts               # inventory file for staging environment
      group_vars/
         group1.yml       # here we assign variables to particular
            groups
         group2.yml
      host_vars/
         stagehost1.yml   # here we assign variables to particular
            systems
         stagehost2.yml

library/
module_utils/
filter_plugins/

site.yml
webservers.yml
dbservers.yml

roles/
    common/
    webtier/
    monitoring/
    fooapp/
```

Step 3: Configure your inventory

Define your inventory file (inventory.ini). An inventory file typically lists the hosts that you want to manage by using Ansible, along with necessary details such as host names, IP addresses, and group memberships. For example:

```
# Hostname and ip address
[UBUNTU2004]
<ip address>
[UBUNTU2204]
<ip address>
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
[RHEL8]
<ip address>
[RHEL9]
<ip address>
[DEBIAN11]
<ip address>
[DEBIAN12]
<ip address>
[SUSE15]
<ip address>

[all:children]
UBUNTU2004
UBUNTU2204
RHEL8
RHEL9
DEBIAN11
DEBIAN12
SUSE15

[all:vars]
ansible_user=<ansible execute user e.g root>
ansible_password=<>
ansible_ssh_common_args='-o StrictHostKeyChecking=no'
```

Step 4: Create Ansible playbooks

Create playbooks (.yml files) to automate deployment tasks. The section provides example playbooks for automating different deployment tasks.

Example playbook for patching Linux distributions

To patch various Linux distributions using Ansible, you can create a YAML playbook file named patch-for-different-distribution.yml and populate the file with tasks similar to the following. The hosts directive defines the target hosts from the inventory (inventory.ini in this context) that are targeted to execute the playbook tasks.

```
- name: Upgrade and Reboot RHEL & Debian family Linux distros
  hosts: <host1,host2,host3>  # replace with your actual hosts in the
     inventory file.
  vars:
    reboot_connect_timeout: 5
    reboot_post_reboot_delay: 15
    reboot_timeout: 600
  tasks:
    # Upgrade RHEL family OS packages
    - name: Upgrade RHEL Family OS packages
      ansible.builtin.yum:
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
11          name: '*'
12          state: latest
13        when:
14          - ansible_facts['distribution'] == "RedHat"
15          - ansible_facts['distribution_major_version'] == "7"
16
17      # Upgrade RHEL family OS packages
18      - name: Upgrade RHEL Family OS packages
19        ansible.builtin.yum:
20          name: '*'
21          state: latest
22        when:
23          - ansible_facts['distribution'] == "RedHat"
24          - ansible_facts['distribution_major_version'] == "8"
25
26      # Upgrade RHEL family OS packages
27      - name: Upgrade RHEL Family OS packages
28        ansible.builtin.yum:
29          name: '*'
30          state: latest
31        when:
32          - ansible_facts['distribution'] == "RedHat"
33          - ansible_facts['distribution_major_version'] == "9"
34
35      # Ubuntu Family upgrade
36      - name: Update repositories cache
37        apt:
38          update_cache: yes
39        when:
40          - ansible_facts['distribution'] == "Ubuntu"
41          - ansible_facts['distribution_major_version'] == "20"
42
43      - name: Update all packages to their latest version
44        apt:
45          name: "*"
46          state: latest
47        when:
48          - ansible_facts['distribution'] == "Ubuntu"
49          - ansible_facts['distribution_major_version'] == "22"
50
51      # Debian Family upgrade
52      - name: Upgrade the OS (apt-get dist-upgrade)
53        apt:
54          upgrade: dist
55        when:
56          - ansible_facts['distribution'] == "Debian"
57          - ansible_facts['distribution_major_version'] == "11"
58
59      - name: Upgrade the OS (apt-get dist-upgrade)
60        apt:
61          upgrade: dist
62        when:
63          - ansible_facts['distribution'] == "Debian"
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
64          - ansible_facts['distribution_major_version'] == "12"
65
66      # Reboot after upgrade
67      - name: Reboot host
68        reboot:
69          connect_timeout: ""
70          post_reboot_delay: ""
71          reboot_timeout: ""
```

Example playbook for installing .Net environments

The following example playbook installs different versions of .Net environments on the specific Linux distributions.

```
1  - name: Install dotnet runtime environment on Linux distros
2    hosts: <host1,host2,host3>  # replace with your actual hosts in the
        inventory file.
3    tasks:
4      # Install dotnet runtime environment on RHEL7
5      - name: Enable the rhel-7-server-dotnet-rpms repository
6        command: subscription-manager repos --enable=rhel-7-server-dotnet
            -rpms
7        when:
8          - ansible_facts['distribution'] == "RedHat"
9          - ansible_facts['distribution_major_version'] == "7"
10
11     - name: Install dotnet runtime environment on RHEL7
12       ansible.builtin.yum:
13         name: rh-dotnet60-aspnetcore-runtime-6.0
14         state: present
15       when:
16         - ansible_facts['distribution'] == "RedHat"
17         - ansible_facts['distribution_major_version'] == "7"
18
19     - name: Remove /usr/bin/dotnet if it exists
20       file:
21         path: /usr/bin/dotnet
22         state: absent
23
24     - name: Create a symbolic link
25       file:
26         src: /opt/rh/rh-dotnet60/root/usr/lib64/dotnet/dotnet
27         dest: /usr/bin/dotnet
28         state: link
29
30     # RHEL8 linux vda install dotnet runtime environment
31     - name: Install dotnet-runtime-8.0
32       ansible.builtin.dnf:
33         name: dotnet-runtime-8.0
34         state: present
35       when:
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
36          - ansible_facts['distribution'] == "RedHat"
37          - ansible_facts['distribution_major_version'] == "8"
38
39      - name: Install aspnetcore-runtime-8.0
40        ansible.builtin.dnf:
41          name: aspnetcore-runtime-8.0
42          state: present
43        when:
44          - ansible_facts['distribution'] == "RedHat"
45          - ansible_facts['distribution_major_version'] == "8"
46
47      # RHEL9 linux vda install dotnet runtime environment
48      - name: Install dotnet-runtime-8.0
49        ansible.builtin.dnf:
50          name: dotnet-runtime-8.0
51          state: present
52        when:
53          - ansible_facts['distribution'] == "RedHat"
54          - ansible_facts['distribution_major_version'] == "9"
55
56      - name: Install aspnetcore-runtime-8.0
57        ansible.builtin.dnf:
58          name: aspnetcore-runtime-8.0
59          state: present
60        when:
61          - ansible_facts['distribution'] == "RedHat"
62          - ansible_facts['distribution_major_version'] == "9"
63
64      # Ubuntu20.04 linux vda install dotnet runtime environment
65      - name: Register Microsoft key and feed
66        shell: |
67          wget https://packages.microsoft.com/config/ubuntu/20.04/
              packages-microsoft-prod.deb -O packages-microsoft-prod.deb
68          dpkg -i packages-microsoft-prod.deb
69          rm packages-microsoft-prod.deb
70        when:
71          - ansible_facts['distribution'] == "Ubuntu"
72          - ansible_facts['distribution_major_version'] == "20"
73
74      - name: Install dotnet-runtime-8.0
75        ansible.builtin.apt:
76          name: dotnet-runtime-8.0
77          state: present
78          update_cache: yes
79        when:
80          - ansible_facts['distribution'] == "Ubuntu"
81          - ansible_facts['distribution_major_version'] == "20"
82
83      - name: Install aspnetcore-runtime-8.0
84        ansible.builtin.apt:
85          name: aspnetcore-runtime-8.0
86          state: present
87          update_cache: yes
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
88          when:
89            - ansible_facts['distribution'] == "Ubuntu"
90            - ansible_facts['distribution_major_version'] == "20"
91
92        # Ubuntu22.04 linux vda install dotnet runtime environment
93        - name: Install dotnet-runtime-8.0
94          ansible.builtin.apt:
95            name: dotnet-runtime-8.0
96            state: present
97            update_cache: yes
98          when:
99            - ansible_facts['distribution'] == "Ubuntu"
100           - ansible_facts['distribution_major_version'] == "22"
101
102       - name: Install aspnetcore-runtime-8.0
103         ansible.builtin.apt:
104           name: aspnetcore-runtime-8.0
105           state: present
106           update_cache: yes
107         when:
108           - ansible_facts['distribution'] == "Ubuntu"
109           - ansible_facts['distribution_major_version'] == "22"
110
111       # Debian11 linux vda install dotnet runtime environment
112       - name: Register Microsoft key and feed
113         shell: |
114           wget https://packages.microsoft.com/config/debian/11/packages-
                microsoft-prod.deb -O packages-microsoft-prod.deb
115           dpkg -i packages-microsoft-prod.deb
116           rm packages-microsoft-prod.deb
117         when:
118           - ansible_facts['distribution'] == "Debian"
119           - ansible_facts['distribution_major_version'] == "11"
120
121       - name: Install dotnet-runtime-8.0
122         ansible.builtin.apt:
123           name: dotnet-runtime-8.0
124           state: present
125           update_cache: yes
126         when:
127           - ansible_facts['distribution'] == "Debian"
128           - ansible_facts['distribution_major_version'] == "11"
129
130       - name: Install aspnetcore-runtime-8.0
131         ansible.builtin.apt:
132           name: aspnetcore-runtime-8.0
133           state: present
134           update_cache: yes
135         when:
136           - ansible_facts['distribution'] == "Debian"
137           - ansible_facts['distribution_major_version'] == "11"
138
139       # Debian12 linux vda install dotnet runtime environment
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
140     - name: Register Microsoft key and feed
141       shell: |
142         wget https://packages.microsoft.com/config/debian/12/packages-
                microsoft-prod.deb -O packages-microsoft-prod.deb
143         dpkg -i packages-microsoft-prod.deb
144         rm packages-microsoft-prod.deb
145       when:
146         - ansible_facts['distribution'] == "Debian"
147         - ansible_facts['distribution_major_version'] == "12"
148
149     - name: Install dotnet-runtime-8.0
150       ansible.builtin.apt:
151         name: dotnet-runtime-8.0
152         state: present
153         update_cache: yes
154       when:
155         - ansible_facts['distribution'] == "Debian"
156         - ansible_facts['distribution_major_version'] == "12"
157
158     - name: Install aspnetcore-runtime-8.0
159       ansible.builtin.apt:
160         name: aspnetcore-runtime-8.0
161         state: present
162         update_cache: yes
163       when:
164         - ansible_facts['distribution'] == "Debian"
165         - ansible_facts['distribution_major_version'] == "12"
166
167     # Sles15 linux vda install dotnet runtime environment
168     - name: Register Microsoft key and feed
169       shell: |
170         sudo rpm -Uvh https://packages.microsoft.com/config/sles/15/
                packages-microsoft-prod.rpm
171         sudo ln -s /etc/yum.repos.d/microsoft-prod.repo /etc/zypp/repos
                .d/microsoft-prod.repo
172       when:
173         - ansible_facts['distribution'] == "SLES"
174         - ansible_facts['distribution_major_version'] == "15"
175
176     - name: Install dotnet-runtime-8.0
177       community.general.zypper:
178         name: dotnet-runtime-8.0
179         state: present
180         update_cache: yes
181       when:
182         - ansible_facts['distribution'] == "SLES"
183         - ansible_facts['distribution_major_version'] == "15"
184
185     - name: Install aspnetcore-runtime-8.0
186       community.general.zypper:
187         name: aspnetcore-runtime-8.0
188         state: present
189         update_cache: yes
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
      when:
        - ansible_facts['distribution'] == "SLES"
        - ansible_facts['distribution_major_version'] == "15"

    # Amazon2 linux vda install dotnet runtime environment
    - name: Install dotnet-runtime-8.0
      ansible.builtin.yum:
        name: dotnet-runtime-8.0
        state: present
      when:
        - ansible_facts['distribution'] == "Amazon"
        - ansible_facts['distribution_major_version'] == "2"

    - name: Install aspnetcore-runtime-8.0
      ansible.builtin.yum:
        name: aspnetcore-runtime-8.0
        state: present
      when:
        - ansible_facts['distribution'] == "Amazon"
        - ansible_facts['distribution_major_version'] == "2"
```

Example playbooks for upgrading the Linux VDA

To automate the Linux VDA upgrades using Ansible, you can create two separate playbooks. One playbook, such as get_the_build.yml, is dedicated to downloading and transferring the Linux VDA package to the target machines (hosts). The other playbook, for example linux_upgrade.yml, contains tasks designed to upgrade the Linux VDA on the target machines using the previously downloaded package.

Example playbook get_the_build.yml

```
- hosts: localhost
  name: Get the latest release build to local
  vars:
    build_url: <linux vda download link>  # replace with your actual
       value.
    local_tmp: "/tmp/"  # replace with your actual value.
    remote_tmp: "/tmp/"  # replace with your actual value.
    linuxvda_file_name : "linux vda rpm/deb file name" # replace with
       your actual value.
  tasks:
  - name: Download the file
    get_url:
      url: ""
      dest: ""
    tags:
      - get


- hosts: <host1,host2,host3>  # replace with your actual hosts in the
    inventory file.
  name: Copy a file to remote location
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
  tasks:
  - name: Copy vda to the remote machine
    ansible.builtin.copy:
      src: ""
      dest: ""
      remote_src: no
    tags:
      - copy
```

Example playbook linux_upgrade.yml

```
- name: Upgrade Linux VDA and Reboot RHEL & Debian Linux distros
  hosts: <host1,host2,host3>  # replace with your actual hosts in the
     inventory file.
  vars:
    remote_tmp: "/path/to/remote/tmp"  # replace with your actual path
    rhel7_file_name: "rhel7_file.rpm"  # replace with your actual file
       name
    rhel8_file_name: "rhel8_file.rpm"  # replace with your actual file
       name
    rhel9_file_name: "rhel9_file.rpm"  # replace with your actual file
       name
    ubuntu2004_file_name: "ubuntu2004_file.deb"  # replace with your
       actual file name
    ubuntu2204_file_name: "ubuntu2204_file.deb"  # replace with your
       actual file name
    debian11_file_name: "debian11_file.deb" # replace with your actual
       file name
    debian12_file_name: "debian12_file.deb" # replace with your actual
       file name
    suse15_file_name: "suse15_file.deb" # replace with your actual file
        name
    amazon2_file_name: "amazon2_file.rpm" # replace with your actual
       file name
  tasks:
    # Upgrade RHEL linux vda packages
    - name: Upgrade RHEL7 linux vda packages
      ansible.builtin.yum:
        name: ""
        state: present
      when:
        - ansible_facts['distribution'] == "RedHat"
        - ansible_facts['distribution_major_version'] == "7"

    # Upgrade RHEL linux vda packages
    - name: Upgrade RHEL8 linux vda packages
      ansible.builtin.yum:
        name: ""
        state: present
      when:
        - ansible_facts['distribution'] == "RedHat"
        - ansible_facts['distribution_major_version'] == "8"

    # Upgrade RHEL linux vda packages
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
34      - name: Upgrade RHEL9 linux vda packages
35        ansible.builtin.yum:
36          name: ""
37          state: present
38        when:
39          - ansible_facts['distribution'] == "RedHat"
40          - ansible_facts['distribution_major_version'] == "9"
41
42      # Ubuntu20.04 linux vda upgrade
43      - name: Ubuntu20.04 linux vda upgrade
44        ansible.builtin.apt:
45          deb: ""
46        when:
47          - ansible_facts['distribution'] == "Ubuntu"
48          - ansible_facts['distribution_major_version'] == "20"
49
50      - name: Ubuntu22.04 linux vda upgrade
51        ansible.builtin.apt:
52          deb: ""
53        when:
54          - ansible_facts['distribution'] == "Ubuntu"
55          - ansible_facts['distribution_major_version'] == "22"
56
57      # Debian Linux VDA upgrade
58      - name: Debian11 Linux VDA upgrade
59        ansible.builtin.apt:
60          deb: ""
61        when:
62          - ansible_facts['distribution'] == "Debian"
63          - ansible_facts['distribution_major_version'] == "11"
64
65      - name: Debian12 Linux VDA upgrade
66        ansible.builtin.apt:
67          deb: ""
68        when:
69          - ansible_facts['distribution'] == "Debian"
70          - ansible_facts['distribution_major_version'] == "12"
71
72      # Sles15 Linux VDA upgrade
73      - name: Sles15 Linux VDA upgrade
74        community.general.zypper:
75          name: ""
76          state: present
77        when:
78          - ansible_facts['distribution'] == "SLES"
79          - ansible_facts['distribution_major_version'] == "15"
80
81      # Amazon2 Linux VDA upgrade
82      - name: Amazon2 Linux VDA upgrade
83        ansible.builtin.yum:
84          name: ""
85        when:
86          - ansible_facts['distribution'] == "Amazon"
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

```
        - ansible_facts['distribution_major_version'] == "2"
    # Reboot after upgrade
    - name: Reboot host
      reboot:
        connect_timeout: ""
        post_reboot_delay: ""
        reboot_timeout: ""
```

Example playbook for mounting a Network File System (NFS) server as the home directory

The following example playbook mounts an NFS server as the home directory on the target hosts:

```
- hosts: <host1,host2,host3>  # replace with your actual hosts in the
    inventory file.
  vars:
    nfs_server = <nfsserver ip address> # replace with your actual
       values
    mount_points = /home/<domain realm>/user1,/home/<domain realm>user2
         # replace with your actual values
    nfs_shares = user1,user2  # replace with your actual values
    owners = user1,user2   # replace with your actual values
    groups = group1,group2  # replace with your actual values
  tasks:
    - name: Enable NFS as home directory
      ansible.builtin.command:
        cmd: "/opt/Citrix/VDA/bin/ctxreg create -k 'HKLM\\System\\
           CurrentControlSet\\Control\\Citrix' -t 'REG_DWORD' -v '
           CheckUserHomeMountPoint' -d '0x00000001' --force"
      register: result
      failed_when: result.rc != 0
      check_mode: no

    - name: Mount NFS shares
      ansible.builtin.mount:
        path: ""
        src: ":"
        fstype: nfs
        opts: rw,nolock
        state: mounted
      loop: ""

    - name: Set owner, group and mode for NFS client paths
      ansible.builtin.file:
        path: ""
        owner: ""
        group: ""
        mode: ""
      loop: ""
```

© 1999–2024 Cloud Software Group, Inc. All rights reserved.

Example playbooks for remote command execution

Example playbook for modifying registry settings

```
- hosts: <host1,host2,host3>  # replace with your actual hosts in the inventory file.
  vars:
    registry_key: "your_registry_key"      #  E.g. registry_key = HKLM\System\CurrentControlSet\Control\Terminal Server\Wds\icawd
    registry_type: "your_registry_type"    #  E.g. registry_type = REG_DWORD
    registry_value: "your_registry_value"  #  E.g. registry_value = AdaptiveScalingEnabled
    registry_data: "your_registry_data"    #  E.g. registry_data = 0x00000000
  tasks:
  - name: Execute AdaptiveScaling redirection script
    ansible.builtin.command:
      cmd: "/opt/Citrix/VDA/bin/ctxreg create -k \"\" -t \"\" -v \"\" -d \"\" --force"
    register: result
    failed_when: result.rc != 0
    check_mode: no
```

Example playbook for locking the RHEL minor version

```
- hosts: <host1,host2,host3>  # replace with your actual hosts in the inventory file.
  vars:
    rhel_minor_version: "9.3"  # replace with your actual minor version such as 9.3, 8.8
  tasks:
    - name: Lock system to a specific minor version
      ansible.builtin.command:
        cmd: "subscription-manager release --set="
      register: result
      failed_when: "'Error' in result.stderr"
```

© 2024 Cloud Software Group, Inc. All rights reserved. Cloud Software Group, the Cloud Software Group logo, and other marks appearing herein are property of Cloud Software Group, Inc. and/or one or more of its subsidiaries, and may be registered with the U.S. Patent and Trademark Office and in other countries. All other marks are the property of their respective owner(s).