# Linux Virtual Delivery Agent 2103

# Contents



# 新增功能

November 8, 2021

## **2103** 中的新增功能

Linux VDA 2103 版包括以下新增功能和增强功能：

### 支持 **Debian 10.7** 和 **CentOS 8.3**

我们添加了 Debian 10.7 和 CentOS 8.3 作为支持的发行版。有关更多信息，请参阅 系统要求、安装 Linux Virtual Delivery Agent for Debian 和安装 Linux Virtual Delivery Agent for RHEL/CentOS。

### **SSSD** 支持将 **SUSE** 计算机加入 **Windows** 域

我们为将 SUSE 计算机加入 Windows 域添加了 SSSD 支持。

### 加入域和未加入域的用例的单个 **Linux VDA** 映像

我们现在提供单个映像，用于在适用于 Azure 的 Citrix Virtual Apps and Desktops Standard 中创建已加入域和未加入域的 Linux VDA。此功能简化了映像准备和维护过程。有关详细信息，请参阅在适用于 Azure 的 Citrix Virtual Apps and Desktops Standard 中创建 Linux VDA。

### 文件传输增强功能

我们通过重新设计进度条、加快下载速度和解决一些缺陷增强了文件传输功能。我们还将您随时可以传输的文件数量从 10 个增加到 100 个。有关详细信息，请参阅文件传输。

### 支持在 **Citrix Studio** 中配置会话连接计时器

以前，您只能通过 `/opt/Citrix/VDA/bin/ctxcfg` 分别在每个 VDA 上和为每个 VDA 配置空闲会话和断开连接的会话的时间限制。此版本添加了以下策略，供您在 Citrix Studio 中配置会话连接计时器时使用：

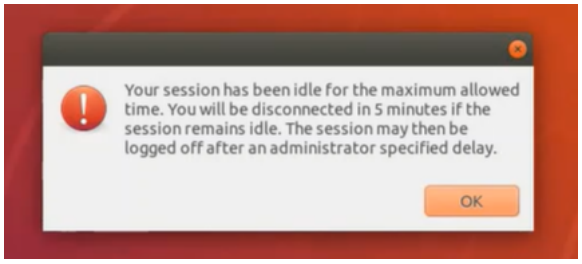
- 会话空闲计时器：确定是否对空闲会话强制实施时间限制。
- 会话空闲计时器时隔：设置空闲会话的时间限制。如果会话空闲计时器已启用，并且活动会话在设定的时间内未收到用户输入，会话将断开连接。
- 断开连接的会话计时器：确定是否对断开连接的会话强制实施时间限制。

- 断开连接的会话计时器间隔：设置断开连接的会话注销之前的时间间隔。

更新任何策略设置时，请确保这些设置在部署中保持一致。有关策略的详细信息，请参阅策略支持列表。

空闲会话的时间限制到期时，将显示一条警告消息。有关示例，请参见以下屏幕截图：按确定关闭警告消息，但无法使会话保持活动状态。要使会话保持活动状态，请提供用户输入以重置空闲计时器。




### **Citrix Director** 中提供了 **Linux VM** 和 **Linux** 会话的新指标

此版本为 Citrix Director 中的 Linux VM 和 Linux 会话添加了新的指标。

每个 Linux VM 的新指标：

- CPU 核心的数量
- 内存大小
- 硬盘容量
- 当前和历史 CPU 和内存利用率

每个 Linux 会话的新指标：

- 空闲时间

有关详细信息，请参阅在 Citrix Director 中监视 Linux VM 和 Linux 会话。

### **Linux VDA** 的 **FAS** 增强功能

我们现在提供了更有见地的日志输出，并允许您在运行 ctxfascfg.sh 脚本时指定包含根证书和所有中间证书的路径。有关配置信息，请参阅配置联合身份验证服务。

### **RHEL 8.3** 和 **Ubuntu 18.04.5** 的 **Linux** 流技术推送支持 **-** 实验性功能

将 Linux 流技术推送功能与 Citrix Provisioning 结合使用，您可以直接在 Citrix Virtual Apps and Desktops 环境中预配 Linux 虚拟桌面。有关详细信息，请参阅 Linux 流技术推送。

## 已修复的问题

### January 26, 2022

自 Linux Virtual Delivery Agent 2012 起，以下问题已修复：

- 启用通道绑定后，Linux VDA 可能无法向 Delivery Controller 注册。用于身份验证的 Kerberos 票证过期时会出现此问题。[LNXVDA-9076]
- 在小米 Mi 10 手机上尝试选择图片传输协议 (PTP) 和媒体传输协议 (MTP) USB 选项可能会失败。[CVADHELP-16188]
- 在 Linux VDA 中，某些应用程序可能无法识别出现在设备选项中的网络摄像机设备。[CVADHELP-16247]
- 在 Linux VDA 会话中为网络摄像机使用 USB 重定向时，`ctxusbsd` 进程可能会意外退出，并提示 `segfault` 错误。[CVADHELP-16366]

## 已知问题

### July 14, 2023

在本版本中确定了以下问题：

- 在 GNOME 桌面会话中，尝试更改键盘布局可能会失败。[CVADHELP-15639]
- 已发布的非无缝应用程序可能会在启动后不久退出。Mutter 升级高于 mutter-3.28.3-4 的版本后会出现此问题。要解决此问题，请使用 mutter-3.28.3-4 或更早版本。[LNXVDA-6967]
- 当您使用 NVIDIA GRID 3D 卡但不启用 HDX 3D Pro 时，Linux VDA 不按预期运行。RHEL 7.7 及更早版本、SUSE 12.5 及更早版本和 Ubuntu 16.04 上会出现此问题。原因是多个 OpenGL 库不能在这些 Linux 发行版的图形系统中共存。
- 文件下载过程中出现意外窗口。该窗口不会影响文件下载功能，并且在一段时间后会自动消失。[LNXVDA-5646]
- PulseAudio 的默认设置会导致正常运行的服务器程序在处于不活动状态 20 秒后退出。当 PulseAudio 退出时，音频将不起作用。要解决此问题，请在 /etc/pulse/daemon.conf 文件中设置 exit-idle-time=-1。[LNXVDA-5464]
- SUSE 12.5 中的 `libtcmalloc 4.3.0` 可能会导致进程意外退出。
- 在 Ubuntu 16.04 和 SUSE 12.5 VDA 上，`ctxhdx` 服务可能会意外退出。此问题会在 GNU C 库 (`glibc`) 2.22 到 2.24 版中出现。该问题已在 `glibc 2.25` 中修复。如果要使用 SUSE 12.5 发行版，可以安装 SUSE 提供的修补程序以修复此问题。发布 Linux VDA 时，没有适用于 Ubuntu 16.04 的修补程序。[LNXVDA-4481]

- 启用了 SSL 加密并且禁用了会话可靠性时，无法在适用于 Linux 的 Citrix Workspace 应用程序中启动会话。[RFLNX-1557]
- `indicator-datetime-service` 进程不使用 $TZ 环境变量。当客户端和会话位于不同的时区时，Ubuntu 16.04 Unity Desktop 上的 unity 面板不显示客户端的时间。[LNXVDA-2128]
- Ubuntu 图形：在 HDX 3D Pro 中，调整 Desktop Viewer 的大小后，应用程序周围可能会显示一个黑框，或者有时背景会显示为黑色。
- 注销会话后，可能不会删除 Linux VDA 打印重定向创建的打印机。
- 目录中包含大量文件和子目录时会遗失 CDM 文件如果客户端有太多文件或目录，则可能会出现此问题。
- 在本版本中，仅支持对非英语语言使用 UTF-8 编码。
- 适用于 Android 的 Citrix Workspace 应用程序的 CapsLock 键状态可能会在会话漫游期间反转。漫游与适用于 Android 的 Citrix Workspace 应用程序的现有连接时，Caps Lock 键状态可能会丢失。解决方法：使用扩展键盘上的 Shift 键在大写与小写之间切换。
- 使用适用于 Mac 的 Citrix Workspace 应用程序连接至 Linux VDA 时，含 Alt 的快捷键有时不起作用。默认情况下，对于左侧和右侧 Options/Alt 键，适用于 Mac 的 Citrix Workspace 应用程序都会发送 AltGr。您可以在 Citrix Workspace 应用程序设置中修改此行为，但是结果因不同的应用程序而异。
- Linux VDA 重新加入域时注册失败。重新加入将生成一组全新的 Kerberos 密钥。但是，Broker 可能会根据先前的一组 Kerberos 密钥使用缓存的过时 VDA 服务票据。VDA 尝试连接到 Broker 时，Broker 可能无法与 VDA 建立返回安全上下文。常见症状是 VDA 注册失败。

  VDA 服务票据过期并续订后，此问题最终会自行解决。但是，由于服务票据的有效期很长，因此可能需要很长时间。

  解决方法：清除 Broker 的票据缓存。重新启动 Broker 或在 Broker 上以管理员身份从命令提示窗口运行以下命令：

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

  此命令会清除 Citrix Broker Service 运行所在的网络服务主体持有的 LSA 缓存中的所有服务票据。此命令也会删除其他 VDA 的服务票据，因而可能会影响其他服务。但是，此操作不会造成负面影响，因为这些服务可在需要时从 KDC 重新获取这些服务票据。
- 不支持音频即插即用。您可以先将音频捕获设备连接到客户端计算机，然后开始在 ICA 会话中录制音频。如果在启动了音频录制应用程序后连接捕获设备，应用程序可能会无响应，因此您必须将其重新启动。如果在录制期间拔下捕获设备，可能会出现类似的问题。
- 适用于 Windows 的 Citrix Workspace 应用程序可能会在音频录制期间遇到音频失真问题。

## 第三方声明

August 11, 2022

Linux Virtual Delivery Agent 2103（PDF 下载）

此 Linux VDA 版本可能包含根据该文档中定义的条款许可使用的第三方软件。

## 弃用

November 8, 2021

本文声明旨在提前通知您正在逐渐淘汰的平台、Citrix 产品和功能，以便您能够及时制定业务决策。Citrix 将监视客户使用情况和反馈以确定其退出时间。在后续版本中声明可能会有更改，可能不会包括每个弃用的特性或功能。
有关产品生命周期支持的详细信息，请参阅 Product Lifecycle Support Policy（产品生命周期支持策略）一文。

### 弃用和删除

下表显示了已弃用或删除的平台、Citrix 产品和功能。
已弃用的项目不会立即删除。Citrix 在此版本中继续支持这些项目，但将在未来的当前版本中将其删除。在 Linux VDA 中，
已删除的项目已被删除或不再受支持。

| 项目 | 宣布弃用的版本 | 删除版本 |
| --- | --- | --- |
| 支持 RHEL 7.7、CentOS 7.7 | 2006 | 2009 |
| 支持 SUSE 12.3 | 2006 | 2006 |
| 支持 RHEL 6.10、CentOS 6.10 | 2003 | 2003 |
| 支持 RHEL 6.9、CentOS 6.9 | 1909 | 1909 |
| 支持 RHEL 7.5、CentOS 7.5 | 1903 | 1903 |
| 支持 RHEL 7.4、CentOS 7.4 | 1811 | 1811 |
| 支持 RHEL 6.8、CentOS 6.8 | 1811 | 1811 |
| 支持 RHEL 7.3、CentOS 7.3 | 7.18 | 7.18 |
| 支持 RHEL 6.6、CentOS 6.6 | 7.16 | 7.16 |
| SUSE 11.4 | 7.16 | 7.16 |

# 系统要求

November 8, 2021

## Linux 发行版

> 注意：
>
> 本文档中未涉及的组件（例如 Citrix Workspace 应用程序）的系统要求在其各自的文档集中进行说明。
>
> 在安装 Linux VDA 之前，请按照 https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers 上的说明安装.NET Core 运行时 3.1。

有关在长期服务版本 (LTSR) 环境中使用此当前版本 (CR) 以及其他常见问题解答的详细信息，请参阅知识中心文章。

Linux VDA 支持以下 Linux 发行版：

- SUSE Linux Enterprise：
  - Server 12 Service Pack 5 + SUSE Linux Enterprise Workstation Extension 12 SP5
  - Server 12 Service Pack 5
- Red Hat Enterprise Linux
  - Workstation 8.3
  - 工作站 8.2
  - Workstation 8.1
  - 工作站 7.9
  - 工作站 7.8
  - Server 8.3
  - Server 8.2
  - Server 8.1
  - Server 7.9
  - Server 7.8
- CentOS Linux
  - CentOS 8.3
  - CentOS 8.2
  - CentOS 8.1
  - CentOS 7.9
  - CentOS 7.8
- Ubuntu Linux

  - Ubuntu Desktop 20.04
  - Ubuntu Server 20.04
  - Ubuntu Desktop 18.04
  - Ubuntu Server 18.04
  - Ubuntu Live Server 18.04
  - Ubuntu Desktop 16.04
  - Ubuntu Server 16.04

- Debian Linux
  - Debian 10.7

> 注意：
>
> CentOS 项目将关注的焦点切换到 CentOS Stream。作为 RHEL 8 的重建，CentOS Linux 8 将于 2021 年底终止提供。之后，CentOS Stream 将继续用作 Red Hat Enterprise Linux 的上游(开发)分支。有关详细信息，请参阅 https://www.redhat.com/en/blog/centos-stream-building-innovative-future-enterprise-linux。

Linux VDA 支持多种将 Linux 计算机与 Microsoft Active Directory (AD) 集成的方法：

| | Winbind | SSSD | Centrify | PBIS | Quest |
|---|---|---|---|---|---|
| RHEL 8.3 | 是 | 是 | 是 | 否 | 否 |
| CentOS 8.3 | 是 | 是 | 是 | 否 | 否 |
| RHEL 8.2 | 是 | 是 | 是 | 否 | 否 |
| CentOS 8.2 | 是 | 是 | 是 | 否 | 否 |
| RHEL 8.1 | 是 | 是 | 是 | 否 | 否 |
| CentOS 8.1 | 是 | 是 | 是 | 否 | 否 |
| RHEL 7.9 | 是 | 是 | 是 | 是 | 是 |
| CentOS 7.9 | 是 | 是 | 是 | 是 | 是 |
| RHEL 7.8 | 是 | 是 | 是 | 是 | 是 |
| CentOS 7.8 | 是 | 是 | 是 | 是 | 是 |
| Ubuntu 20.04 | 是 | 是 | 是 | 是 | 是 |
| Ubuntu 18.04 | 是 | 是 | 是 | 是 | 是 |
| Ubuntu 16.04 | 是 | 是 | 是 | 是 | 是 |
| Debian 10.7 | 是 | 是 | 是 | 是 | 否 |
| SUSE 12.5 | 是 | 是 | 是 | 是 | 是 |

有关此 Linux VDA 版本支持的 Linux 发行版和 Xorg 版本列表，请参阅下表。有关详细信息，请参阅 XorgModuleABIVersions。

| Linux 发行版 | Xorg 版本 |
| --- | --- |
| RHEL 8.3、CentOS 8.3 | 1.20.8 |
| RHEL 8.2、CentOS 8.2 | 1.20.8 |
| RHEL 8.1、CentOS 8.1 | 1.20.8 |
| RHEL 7.9、CentOS 7.9 | 1.20 |
| RHEL 7.8、CentOS 7.8 | 1.20 |
| Ubuntu 20.04 | 1.20 |
| Ubuntu 18.04 | 1.19 |
| Ubuntu 16.04 | 1.18 |
| Debian 10.7 | 1.20 |
| SUSE 12.5 | 1.19 |

请勿在 Ubuntu 上使用 HWE 内核或 HWE Xorg。

在 RHEL 8.x 和 CentOS 8.x 上使用 PulseAudio 13.99。

在所有情况下，受支持的处理器架构均为 x86-64。

> 注意：
>
> 当 Linux 操作系统供应商提供的支持过期时，Citrix 对该操作系统平台和版本的支持也将过期。
>
> 重要：
>
> GNOME 和 KDE 桌面在 SUSE 12、RHEL 7、CentOS 7、RHEL 8 和 CentOS 8 中受支持。只有 Ubuntu 16.04 支持 Unity 桌面。GNOME 桌面在 Ubuntu 20.04 和 Ubuntu 18.04 中受支持。至少必须安装一种桌面。

## Citrix Virtual Desktops

Linux VDA 与当前所有受支持的 Citrix Virtual Apps and Desktops 版本兼容。要获取有关 Citrix 产品生命周期的信息，以及了解何时 Citrix 会停止支持特定的产品版本，请参阅 Citrix 产品生命周期表。

Linux VDA 与 Windows VDA 的配置过程略有差别。但是，所有 Delivery Controller 场都能为 Windows 和 Linux 桌面提供代理服务。

支持的主机平台和虚拟化环境

- 裸机服务器
- Citrix Hypervisor
- VMware ESX 和 ESXi
- Microsoft Hyper-V
- Nutanix AHV
- Microsoft Azure Resource Manager
- Amazon Web Services (AWS)
- Google 云端平台 (GCP)

> 提示：
>
> 有关受支持平台的列表，请参阅供应商的文档。
>
> 注意：
>
> Azure、AWS 和 GCP 仅与 Citrix Virtual Apps and Desktops 服务兼容。使用 MCS 创建虚拟机时，裸机服务器不受支持。

## Active Directory 集成软件包

Linux VDA 支持以下 Active Directory 集成软件包或产品：

- Samba Winbind
- Quest Authentication Services v4.1 或更高版本
- Centrify DirectControl
- SSSD
- PBIS（与 RHEL 7、Ubuntu 和 Debian 兼容）

> 提示：
>
> 有关受支持平台的列表，请参阅 Active Directory 集成软件包供应商提供的文档。

## HDX 3D Pro

使用以下虚拟机管理程序和 NVIDIA GRID™ GPU 才能支持 HDX 3D Pro。

虚拟机管理程序

- Citrix Hypervisor
- VMware ESX 和 ESXi
- Nutanix AHV

> 注意
>
> ：虚拟机管理程序与某些 Linux 发行版兼容。

**GPU**

Linux VDA 支持以下 GPU 用于 GPU 直通：

- NVIDIA GRID - Tesla T4
- NVIDIA GTX750Ti
- NVIDIA GRID - Tesla M60
- NVIDIA GRID - K2
- NVIDIA GRID - Tesla P40
- NVIDIA GRID - Tesla P4
- NVIDIA GRID - Tesla P100

Linux VDA 支持以下 GPU 用于 vGPU：

- NVIDIA GRID - Tesla T4
- NVIDIA GRID - Tesla V100
- NVIDIA GRID - Tesla M60
- NVIDIA GRID - Tesla M10
- NVIDIA GRID - Tesla P40
- NVIDIA GRID - Tesla P4
- NVIDIA GRID - Tesla P100

## 安装概述

November 8, 2021

有多个选项可供您用来安装 Linux VDA。可以执行全新安装或从先前的两个版本和 LTSR 版本升级现有安装。

- 轻松安装。在计算机上安装 Linux VDA 软件包后，可以使用 ctxinstall.sh 脚本来配置正在运行的环境。有关详细信息，请参阅轻松安装。
- 在适用于 Azure 的 Citrix Virtual Apps and Desktops Standard 中创建 Linux VDA：可以在适用于 Azure 的 Citrix Virtual Apps and Desktops Standard 中创建已加入域和未加入域的 Linux VDA，以将虚拟应用程序和桌面从 Microsoft Azure 交付到任何设备。有关详细信息，请参阅在适用于 Azure 的 Citrix Virtual Apps and Desktops Standard 中创建 Linux VDA。
- MCS。可以使用 MCS 来批量创建也安装了 Linux VDA 软件包的 Linux VM。有关详细信息，请参阅使用 MCS 创建 Linux VM。

- 手动安装。您可以使用以下常规步骤来安装 Linux VDA。各种变体和特定命令按发行版进行记录。有关详细信息，请参阅安装 Linux Virtual Delivery Agent for RHEL/CentOS、安装 Linux Virtual Delivery Agent for SUSE 和安装 Linux Virtual Delivery Agent for Ubuntu 和安装 Linux Virtual Delivery Agent for Debian。

  1. 准备安装。
  2. 准备虚拟机管理程序。
  3. 向 Windows 域中添加 Linux 虚拟机 (VM)。
  4. 安装 Linux VDA。
  5. 配置 Linux VDA。
  6. 在 Citrix Virtual Apps 或 Citrix Virtual Desktops 中创建计算机目录。
  7. 在 Citrix Virtual Apps 或 Citrix Virtual Desktops 中创建交付组。

### XDPing

可以使用 Linux XDPing 工具检查 Linux VDA 环境中的常见配置问题。有关详细信息，请参阅 XDPing。

### 安装.NET Core 运行时 3.1 作为必备项

在安装 Linux VDA 之前，请按照 https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers 上的说明安装.NET Core 运行时 3.1。

安装.NET Core 运行时 3.1 后，运行 **which dotnet** 命令查找您的运行时路径。

根据命令输出，设置.NET Core 运行时二进制文件路径。例如，如果命令输出为 /aa/bb/dotnet，请使用 /aa/bb 作为.NET 二进制文件路径。

## 配置 Delivery Controller

March 29, 2022

XenDesktop 7.6 及更早版本需要更改才能支持 Linux VDA。对于这些版本，需要运行修补程序或更新脚本。安装和验证信息在本文中提供。

### 更新 Delivery Controller 配置

对于 XenDesktop 7.6 SP2，请应用 Hotfix Update 2 更新 Linux 虚拟桌面的 Broker。Hotfix Update 2 可在以下位置找到：

CTX142438：Hotfix Update 2 - 适用于 Delivery Controller 7.6（32 位）- 英文版

对于早于 XenDesktop 7.6 SP2 的版本，可使用名为 **Update-BrokerServiceConfig.ps1** 的 PowerShell 脚本来更新 Broker Service 配置。以下软件包中提供此脚本：

- citrix-linuxvda-scripts.zip

对场内的每个 Delivery Controller 重复以下步骤：

1. 将 **Update-BrokerServiceConfig.ps1** 脚本复制到 Delivery Controller 计算机。
2. 在本地管理员上下文中打开 Windows PowerShell 控制台。
3. 浏览到包含 **Update-BrokerServiceConfig.ps1** 脚本的文件夹。
4. 运行 **Update-BrokerServiceConfig.ps1** 脚本：

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

> 提示：
>
> PowerShell 的默认配置是禁止执行 PowerShell 脚本。如果脚本运行失败，请先更改 PowerShell 执行策略，然后再重试：
>
> ```
> Set-ExecutionPolicy Unrestricted
> ```

**Update-BrokerServiceConfig.ps1** 脚本会使用 Linux VDA 所需的新 WCF 端点更新 Broker Service 配置文件，然后重新启动 Broker Service。该脚本会自动确定 Broker Service 配置文件的位置。系统会在同一个目录中为原始配置文件创建备份，并向文件名附加 **.prelinux**。

这些更改不会影响配置为使用同一个 Delivery Controller 场的 Windows VDA 的代理。一个 Controller 场可同时无缝管理和代理 Windows 和 Linux VDA 的会话。

> 注意：
>
> Linux VDA 不支持使用 Secure ICA 进行加密。在 Linux VDA 上启用 Secure ICA 会导致会话启动失败。

### 验证 **Delivery Controller** 配置

当所需的配置更改已应用于 Delivery Controller 时，**EndpointLinux** 字符串会在 **%PROGRAMFILES%\Citrix\Broker\Service\BrokerService.exe.config** 文件中出现五次。

在 Windows 命令提示窗口中，以本地管理员身份登录进行检查：

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

# 轻松安装

June 21, 2022

从 Linux VDA 7.13 版起支持轻松安装。此功能通过自动安装必需的软件包并自定义配置文件来帮助您设置 Linux VDA 的运行环境。

## 使用轻松安装

要使用此功能，请执行以下操作：

1. 准备配置信息和 Linux 计算机。
2. 安装 Linux VDA 软件包。
   转至 Citrix Virtual Apps and Desktops 下载页面。展开相应版本的 Citrix Virtual Apps and Desktops，然后单击组件以下载与 Linux 发行版匹配的 Linux VDA 包。
3. 设置运行时环境以完成 Linux VDA 安装。

### 步骤 **1**：准备配置信息和 **Linux** 计算机

收集轻松安装所需的以下配置信息：

- 主机名 - 要安装 Linux VDA 的计算机的主机名
- 域名服务器的 IP 地址
- NTP 服务器的 IP 地址或字符串名称
- 域名 - 域的 NetBIOS 名称
- 领域名称 - Kerberos 领域名称
- 活动域的 FQDN - 完全限定域名

> 重要：
>
> - 要安装 Linux VDA，请确认是否在 Linux 计算机上正确添加了存储库。
> - 要启动会话，请确认是否安装了 X Windows 系统和桌面环境。

#### 注意事项

- 默认情况下，工作组名称是域名。要在您的环境中自定义工作组，请执行以下操作：

  a. 在 Linux VDA 计算机上创建 tmp/ctxinstall.conf 文件。
  b. 将“wworkgroup=<your workgroup>”行添加到文件中并保存您所做的更改。

- Centrify 不支持纯 IPv6 DNS 配置。/etc/resolv.conf 中至少需要有一个使用 IPv4 的 DNS 服务器，`adclient` 才能正确查找 AD 服务。

日志：

```
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.
```

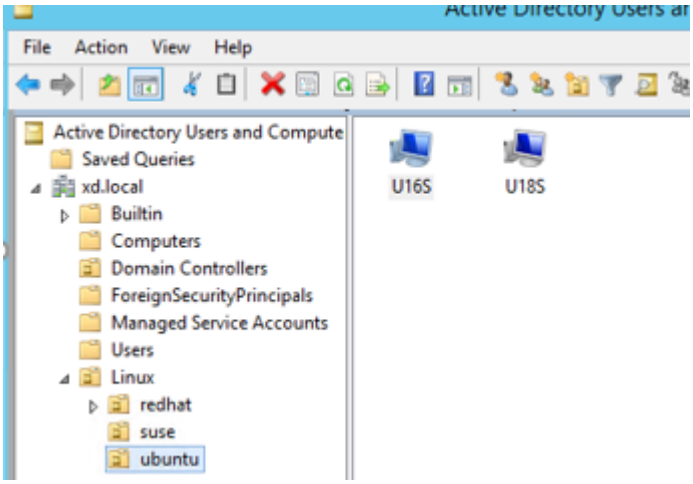
此问题是 Centrify 及其配置独有的。要解决此问题，请执行以下操作：

a. 在域控制器上打开管理工具。
b. 选择 **Active Directory** 站点和服务。
c. 为子网添加正确的子网地址。

- 要将 VDA 连接到特定 OU，请执行以下操作：

  1. 确保域控制器上存在特定的 OU。

     有关示例 OU，请参阅下面的屏幕截图。




  2. 在 VDA 上创建 /tmp/ctxinstall.conf 文件。
  3. 将 ou=<your ou> 行添加到 /tmp/ctxinstall.conf 文件中。

     OU 值因 AD 方法的不同而异。请参见下表。

| 操作系统 | Winbind | SSSD | Centrify | PBIS |
|---|---|---|---|---|
| RHEL 8 | ou=” OU=redhat,OU=Linux” | ou=” OU=redhat,OU=Linux” | ou=” XD.LOCAL/Linux/redhat” | 不支持 |
| RHEL 7 | ou=” Linux/redhat” | ou=” Linux/redhat” | ou=” XD.LOCAL/Linux/redhat” | ou=” Linux/redhat” |
| Ubuntu | ou=” Linux/ubuntu” | ou=” Linux/ubuntu” | ou=” XD.LOCAL/Linux/ubuntu” | ou=” Linux/ubuntu” |
| SUSE 12.5 | ou=” Linux/suse” | ou=” Linux/suse” | ou=” XD.LOCAL/Linux/suse” | 不支持 |

| 操作系统 | Winbind | SSSD | Centrify | PBIS |
|---|---|---|---|---|
| Debian | ou=” Linux/debian” | ou=” Linux/debian” | ou=” XD.LOCAL/Linux/debian” | ou=” Linux/debian” |

- 自 Linux VDA 7.16 起，轻松安装支持纯 IPv6。以下先决条件和限制适用：
  - 必须配置您的 Linux 存储库，以确保您的计算机可以通过纯 IPv6 网络下载所需软件包。
  - 在纯 IPv6 网络中不支持 Centrify。

> 注意：
>
> 如果您的网络是纯 IPv6，并且所有输入均采用恰当的 IPv6 格式，VDA 将通过 IPv6 向 Delivery Controller 注册。如果您的网络具有混合 IPv4 和 IPv6 配置，第一个 DNS IP 地址的类型将决定是使用 IPv4 还是使用 IPv6 来注册。

- 如果选择 Centrify 作为加入域的方法，ctxinstall.sh 脚本需要 Centrify 软件包。ctxinstall.sh 获取 Centrify 软件包的方法有两种：
  - 轻松安装可帮助自动从 Internet 下载 Centrify 软件包。下面是每个发行版的 URL：

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

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

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

    Ubuntu/Debian：wget https://downloads.centrify.com/products/infrastructure-services/19.9/centrify-infrastructure-services-19.9-deb8-x86_64.tgz?_ga=2.151462329.1042350071.1592881996-604509155.1572850145
  - 从本地目录中提取 Centrify 软件包。要指定 Centrify 软件包的目录，请执行以下操作：

    a. 在 Linux VDA 服务器上创建 tmp/ctxinstall.conf 文件（如果该文件不存在）。
    b. 在文件中添加“centrifypkgpath=<path name>”行。

    例如：

```
cat /tmp/ctxinstall.conf
set  “centrifypkgpath=/home/mydir”
ls  -ls  /home/mydir
    9548 -r-xr-xr-x. 1 root root  9776688 May 13  2016
       adcheck-rhel4-x86_64
    4140 -r--r--r--. 1 root root  4236714 Apr 21  2016
   centrifyda-3.3.1-rhel4-x86_64.rpm
```

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

- 如果选择 PBIS 作为加入域的方法，ctxinstall.sh 脚本需要 PBIS 软件包。ctxinstall.sh 获取 PBIS 软件包的方法有两种：
  - 轻松安装可帮助自动从 Internet 下载 PBIS 软件包。下面是每个发行版的 URL：

    RHEL 7/CentOS 7：wget https://github.com/BeyondTrust/pbis-open/releases/download/8.8.0/pbis-open-8.8.0.506.linux.x86_64.rpm.sh

    Ubuntu/Debian：wget https://github.com/BeyondTrust/pbis-open/releases/download/8.8.0/pbis-open-8.8.0.506.linux.x86_64.deb.sh
  - 从 Internet 获取 PBIS 软件包的特定版本。为此，请更改 /opt/Citrix/VDA/sbin/ctxinstall.sh 文件中的行“pbisDownloadPath”，以指定 PBIS 软件包的 URL。

    有关示例，请参阅以下屏幕截图：

```
pbisDownloadPath_RHEL="https://github.com/BeyondTrust/pbis-open/releases/download/8.8.0/pbis-open-8.8.0.506.linux.x86_64.rpm.sh"
pbisDownloadPath_Ubuntu="https://github.com/BeyondTrust/pbis-open/releases/download/8.8.0/pbis-open-8.8.0.506.linux.x86_64.deb.sh"
```

步骤 **2**：安装 **Linux VDA** 软件包

要为 Linux VDA 设置环境，请运行以下命令。

对于 RHEL 和 CentOS 发行版：

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

对于 Ubuntu/Debian 发行版：

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

> 注意：
>
> 要为 Debian 发行版安装必要的依赖项，请将 `deb http://deb.debian.org/debian/ oldstable main` 行添加到 /etc/apt/sources.list 文件。

对于 SUSE 发行版：

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

步骤 **3**：设置运行时环境以完成安装

> 注意：
>
> 在设置运行时环境之前，请确保已在操作系统中安装了 `en_US.UTF-8` 区域设置。如果该区域设置在您的操作系统中不可用，请运行 `sudo locale-gen en_US.UTF-8` 命令。对于 Debian，请通过取消批注 `# en_US.UTF-8 UTF-8` 行来编辑 `/etc/locale.gen` 文件，然后运行 `sudo locale-gen` 命令。

安装 Linux VDA 软件包后，使用 ctxinstall.sh 脚本来配置运行环境。可以在交互模式或无提示模式下运行该脚本。

> 注意：
>
> 下载.NET Core Runtime（该运行时大小超过 27 MB）时，轻松安装可能看起来没有响应。有关下载进度，请查看 /var/log/ctxinstall.log。

交互模式：

要执行手动配置，请运行以下命令并在每个提示处键入相关参数。

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

无提示模式：

要在无提示模式下使用轻松安装，请先设置以下环境变量，然后再运行 ctxinstall.sh。

- **CTX_EASYINSTALL_HOSTNAME=host-name** –表示 Linux VDA 服务器的主机名。
- **CTX_EASYINSTALL_DNS=ip-address-of-dns** –DNS 的 IP 地址。
- **CTX_EASYINSTALL_NTPS=address-of-ntps** –NTP 服务器的 IP 地址或字符串名称。
- **CTX_EASYINSTALL_DOMAIN=domain-name** –域的 NetBIOS 名称
- **CTX_EASYINSTALL_REALM=realm-name** –Kerberos 领域名称。
- **CTX_EASYINSTALL_FQDN=ad-fqdn-name**

- **CTX_EASYINSTALL_ADINTEGRATIONWAY=winbind | sssd | centrify | pbis** –表示 Active Directory 集成方法。
- **CTX_EASYINSTALL_USERNAME=domain-user-name** –表示域用户的名称；用于加入域。
- **CTX_EASYINSTALL_PASSWORD=password** –指定域用户的密码；用于加入域。

ctxsetup.sh 脚本使用以下变量：

- **CTX_XDL_SUPPORT_DDC_AS_CNAME = Y | N** –Linux VDA 支持使用 DNS CNAME 记录指定 Delivery Controller 名称。
- **CTX_XDL_DDC_LIST=** ‘**list-ddc-fqdns**’ –Linux VDA 要求提供由空格分隔的 Delivery Controller 完全限定域名 (FQDN) 列表以用于向 Delivery Controller 注册。必须至少指定一个 FQDN 或 CNAME。
- **CTX_XDL_VDA_PORT=port-number** –Linux VDA 将通过 TCP/IP 端口与 Delivery Controller 通信。
- **CTX_XDL_REGISTER_SERVICE=Y | N** - 在启动计算机后启动 Linux VDA 服务。
- **CTX_XDL_ADD_FIREWALL_RULES=Y | N** –Linux VDA 服务要求允许传入网络连接通过系统防火墙。您可以在系统防火墙中自动为 Linux Virtual Desktop 打开所需端口（默认端口 80 和 1494）。
- **CTX_XDL_HDX_3D_PRO=Y | N** –Linux VDA 支持 HDX 3D Pro，这是一组 GPU 加速技术，旨在优化富图形应用程序的虚拟化水平。如果选择了 HDX 3D Pro，则要为 VDI 桌面（单会话）模式配置 VDA -（即 CTX_XDL_VDI_MODE=Y）。
- **CTX_XDL_VDI_MODE=Y | N** –将计算机配置为专用桌面交付模型 (VDI) 还是托管共享桌面交付模型。对于 HDX 3D Pro 环境，将该值设置为 Y。
- **CTX_XDL_SITE_NAME=dns-name** –Linux VDA 通过 DNS 发现 LDAP 服务器。要将 DNS 搜索结果限制为本地站点，应指定 DNS 站点名称。如果不需要，可以将其设置为 **<none>**。
- **CTX_XDL_LDAP_LIST=** ‘**list-ldap-servers**’ –Linux VDA 查询 DNS 来发现 LDAP 服务器。如果 DNS 无法提供 LDAP 服务记录，您可以提供以空格分隔的 LDAP FQDN（带有 LDAP 端口）列表。例如 ad1.mycompany.com:389。如果不需要，可以将其设置为 **<none>**。
- **CTX_XDL_SEARCH_BASE=search-base-set** –Linux VDA 通过设置为 Active Directory 域根的搜索库来查询 LDAP（例如，DC=mycompany,DC=com）。为提高搜索性能，可以指定搜索基础（例如 OU=VDI,DC=mycompany,DC=com）。如果不需要，可以将其设置为 **<none>**。
- **CTX_XDL_FAS_LIST=** ‘**list-fas-servers**’ –联合身份验证服务 (FAS) 服务器是通过 AD 组策略配置的。Linux VDA 不支持 AD 组策略，但您可以改为提供以分号分隔的 FAS 服务器的列表。顺序必须与在 AD 组策略中配置的顺序相同。如果删除了任何服务器地址，请使用 **<none>** 文本字符串填充其空白，并且不要修改服务器地址的顺序。
- **CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime** - 安装.NET Core Runtime 3.1 以支持新的 Broker 代理服务 (`ctxvda`) 的路径。默认路径为 /usr/bin。
- **CTX_XDL_START_SERVICE=Y | N** –在完成配置后，是否启动 Linux VDA 服务。
- **CTX_XDL_TELEMETRY_SOCKET_PORT** - 用于侦听 Citrix Scout 的套接字端口。默认端口为 7503。
- **CTX_XDL_TELEMETRY_PORT** - 用于与 Citrix Scout 通信的端口。默认端口为 7502。

如果未设置任何参数，安装将回滚到交互模式，提示用户输入。通过环境变量设置了所有参数时，ctxinstall.sh 脚本仍会提示用户输入安装.NET Core Runtime 3.1 的路径。

在无提示模式下，必须运行以下命令以设置环境变量，然后运行 ctxinstall.sh 脚本。

```
export CTX_EASYINSTALL_HOSTNAME=host-name

export CTX_EASYINSTALL_DNS=ip-address-of-dns

export CTX_EASYINSTALL_NTPS=address-of-ntps

export CTX_EASYINSTALL_DOMAIN=domain-name

export CTX_EASYINSTALL_REALM=realm-name

export CTX_EASYINSTALL_FQDN=ad-fqdn-name

export CTX_EASYINSTALL_ADINTEGRATIONWAY=winbind | sssd | centrify |
    pbis

export CTX_EASYINSTALL_USERNAME=domain-user-name

export CTX_EASYINSTALL_PASSWORD=password

export CTX_XDL_SUPPORT_DDC_AS_CNAME=Y | N

export CTX_XDL_DDC_LIST= ‘list-ddc-fqdns’

export CTX_XDL_VDA_PORT=port-number

export CTX_XDL_REGISTER_SERVICE=Y | N

export CTX_XDL_ADD_FIREWALL_RULES=Y | N

export CTX_XDL_HDX_3D_PRO=Y | N

export CTX_XDL_VDI_MODE=Y | N

export CTX_XDL_SITE_NAME=dns-site-name | '<none>'

export CTX_XDL_LDAP_LIST= ‘list-ldap-servers’  | '<none>'

export CTX_XDL_SEARCH_BASE=search-base-set | '<none>'

export CTX_XDL_FAS_LIST= ‘list-fas-servers’  | '<none>'

export CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime

export CTX_XDL_TELEMETRY_SOCKET_PORT=port-number

export CTX_XDL_TELEMETRY_PORT=port-number

export CTX_XDL_START_SERVICE=Y | N

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

运行 sudo 命令时，键入 -E 选项以将现有环境变量传递给它创建的新 shell。我们建议您使用前面的命令并加上

**#!/bin/bash** 作为第一行来创建 shell 脚本文件。

或者，您可以使用单个命令指定所有参数：

```
sudo CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N \

CTX_XDL_DDC_LIST= ‘list-ddc-fqdns’  \

CTX_XDL_VDA_PORT=port-number \

CTX_XDL_REGISTER_SERVICE=Y|N \

CTX_XDL_ADD_FIREWALL_RULES=Y|N \

CTX_XDL_AD_INTEGRATION=1|2|3|4 \

CTX_XDL_HDX_3D_PRO=Y|N \

CTX_XDL_VDI_MODE=Y|N \

CTX_XDL_SITE_NAME=dns-name \

CTX_XDL_LDAP_LIST= ‘list-ldap-servers’  \

CTX_XDL_SEARCH_BASE=search-base-set \

CTX_XDL_FAS_LIST= ‘list-fas-servers’  \

CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime \

CTX_XDL_TELEMETRY_SOCKET_PORT=port-number \

CTX_XDL_TELEMETRY_PORT=port-number \

CTX_XDL_START_SERVICE=Y|N \

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

## 故障排除

请使用此部分中的信息对可能因使用此功能而引发的问题进行故障排除。

### 使用 **SSSD** 加入域失败

尝试加入域时可能会出现错误，输出类似如下（要进行屏幕打印，请验证日志）：

```
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$@CITRIXLAB.LOCAL'not found in Kerberos database. Unable to create GSSAPI-encrypted LDAP connection.Nov 4 02:15:27 RH-WS-68 [sssd[ldap_child[14867]]]: Client 'RH-WS-68$@CITRIXLAB.LOCAL'not found in Kerberos database
```

要解决此问题，请执行以下操作：

1. 运行 `rm -f /etc/krb5.keytab` 命令。
2. 运行 `net ads leave $REALM -U $domain-administrator` 命令。
3. 在 Delivery Controller 上删除计算机目录和交付组。
4. 运行 /opt/Citrix/VDA/sbin/ctxinstall.sh。

5. 在 Delivery Controller 上创建计算机目录和交付组。

### **Ubuntu** 桌面会话显示灰屏

启动会话时会出现此问题，随后将在空桌面中阻止启动会话功能。此外，使用本地用户帐户登录时，计算机的控制台也显示灰屏。

要解决此问题，请执行以下操作：

1. 运行 `sudo apt-get update` 命令。
2. 运行 `sudo apt-get install unity lightdm` 命令。
3. 向 `/etc/lightdm/lightdm.conf` 中添加以下行：
   `greeter-show-manual-login=true`

### 由于缺少主目录，尝试启动 **Ubuntu** 桌面会话失败

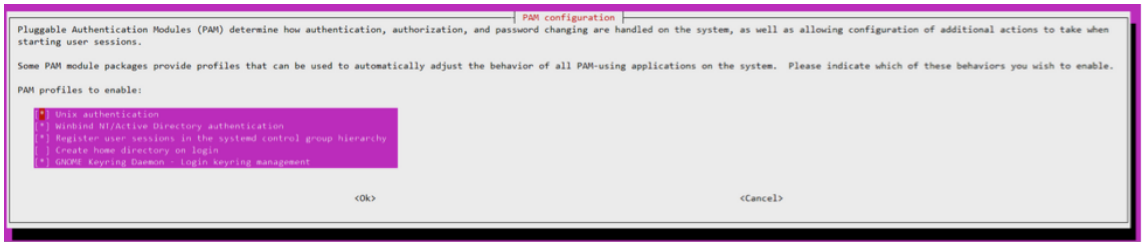
/var/log/xdl/hdx.log:

```
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.
```

> 提示：
>
> 此问题的根本原因是没有为域管理员创建主目录。

要解决此问题，请执行以下操作：

1. 在命令行中，键入 **pam-auth-update**。
2. 在生成的对话框中，确认是否已选中 **Create home directory login** （创建主目录登录信息）。



会话不启动，也不快速结束并显示 **dbus** 错误

/var/log/messages（适用于 RHEL 或 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.
```

或者，对于 Ubuntu 发行版，请使用日志 /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.

Nov  3 11:03:52 user01-HVM-domU pulseaudio[25352]: [pulseaudio] core-util.c: Failed to connect to system bus: Did not receive a reply. Possible causes include: the remote application did not send a reply, the message bus security policy blocked the reply, the reply timeout expired, or the network connection was broken.

Nov  3 11:03:52 user01-HVM-domU pulseaudio[25352]: message repeated 10 times: [ [pulseaudio] core-util.c: Failed to connect to system bus: Did not receive a reply. Possible causes include: the remote application did not send a reply, the message bus security policy blocked the reply, the reply timeout expired, or the network
```

```
    connection was broken.]

Nov  3 11:03:52 user01-HVM-domU pulseaudio[25352]: [pulseaudio] pid.c:
    Daemon already running.Nov  3 11:03:58 user01-HVM-domU citrix-ctxgfx
    [24693]: Exiting normally
```

某些组或模块在重新启动后才会生效。如果日志中出现 **dbus** 错误消息，我们建议您重新启动系统并重试。

### SELinux 可以防止 SSHD 访问主目录

用户可以启动会话，但不能登录。

/var/log/ctxinstall.log:

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

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

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

If you want to allow polyinstantiation to enabled

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

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

    Do

       setsebool -P polyinstantiation_enabled 1

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

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

Then you should report this as a bug.

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

      Do

       allow this access for now by executing:

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

```

```
# semodule -i mypol.pp
```

要解决此问题，请执行以下操作：

1. 通过对 /etc/selinux/config 进行以下更改来禁用 SELinux。

   SELINUX=disabled

2. 重新启动 VDA。

## 在适用于 Azure 的 Citrix Virtual Apps and Desktops Standard 中创建 Linux VDA

June 16, 2023

可以在适用于 Azure 的 Citrix Virtual Apps and Desktops Standard 中创建已加入域和未加入域的 Linux VDA，以将虚拟应用程序和桌面从 Microsoft Azure 交付到任何设备。有关详细信息，请参阅适用于 Azure 的 Citrix Virtual Apps and Desktops Standard。

### 支持的 Linux 发行版

以下 Linux 发行版支持此功能：

- RHEL 8.3
- RHEL 8.2
- RHEL 7.8
- Ubuntu 20.04
- Ubuntu 18.04
- Ubuntu 16.04

### 步骤

要在适用于 Azure 的 Citrix Virtual Apps and Desktops Standard 中创建 Linux VDA，请完成以下步骤：

1. 在 Azure 中准备主映像：

   a) 在 Azure 中，创建支持的发行版的 Linux VM。

   b) 如有必要，请在 Linux VM 上安装桌面环境。

   c) 在 VM 上，根据 https://docs.microsoft.com/en-us/dotnet/core/install/linux-package-managers 上的说明安装.NET Core 运行时 3.1。

d) （仅限 Ubuntu）在 `/etc/network/interfaces` 文件中添加 `source /etc/network/interfaces.d/*` 行。

e) （仅限 Ubuntu）请将 `/etc/resolv.conf` 指向 `/run/systemd/resolve/resolv.conf`，而非将其指向 `/run/systemd/resolve/stub-resolv.conf`：

```
unlink /etc/resolv.conf

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

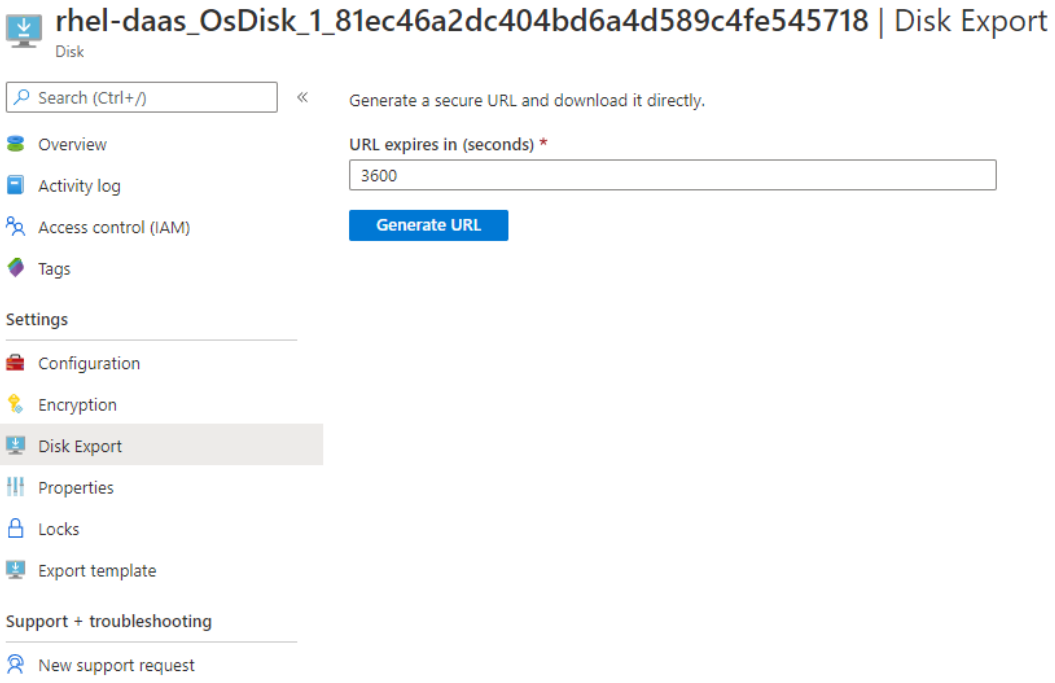
f) 安装 Linux VDA 软件包。

g) 更改 /etc/xdl/mcs/mcs.conf 中的变量。`mcs.conf` 配置文件中包含用于设置 MCS 和 Linux VDA 的变量。

> 注意：
>
> 请将 dns 变量保留为未指定。
>
> 如果在创建计算机目录时选择静态或随机类型，请设置 `VDI_MODE=Y`。

h) 运行 `/opt/Citrix/VDA/sbin/deploymcs.sh`。

i) 在 Azure 中，停止（或取消分配）VM。单击磁盘导出为虚拟硬盘 (VHD) 文件生成 SAS URL，您可以将该文件用作主映像来创建其他 VM。




j) （可选）在主映像上配置组策略设置。

```
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.
```

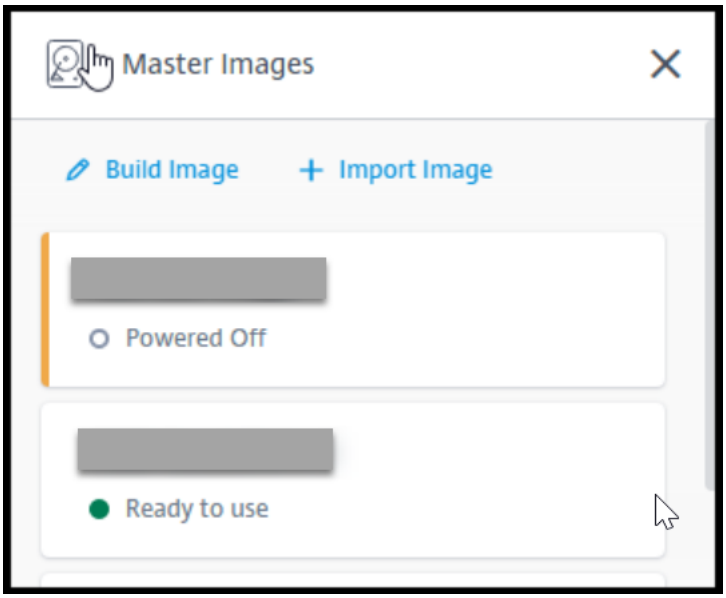
 

```
2
3 ```
4 /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\
    GroupPolicy\Defaults\PrintingPolicies" -t "REG_DWORD" -v "
    AutoCreatePDFPrinter" -d "0x00000001"  - force
5 ```
```

2. 从 Azure 导入主映像。

   a) 在管理控制板中，展开右侧的主映像。显示内容将列出 Citrix 提供的主映像以及您创建和导入的映像。

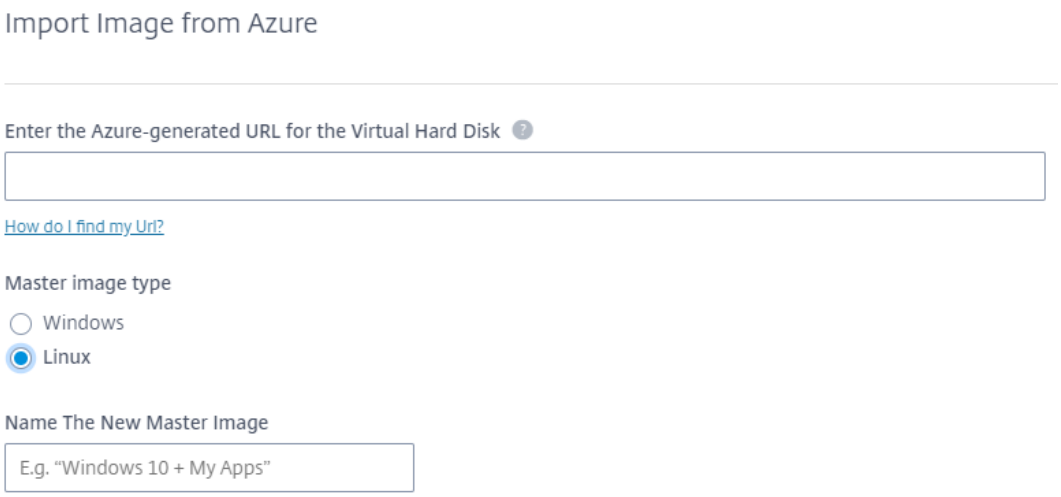
      提示：此服务的大多数管理员活动都通过管理和监视控制板进行管理。创建第一个目录后，管理控制板将在登录到 Citrix Cloud 并选择 **Managed Desktops**（托管桌面）服务后自动启动。




   b) 单击导入映像。

   c) 输入在 Azure 中生成的 VHD 文件的 SAS URL。选择 **Linux** 作为主映像类型。




d) 按照向导中的说明完成导入主映像的操作。

3. 创建计算机目录。

   访问管理控制板，然后单击创建目录。创建计算机目录时，请选择之前创建的主映像。

   注意：只能在 Citrix 托管的 Azure 订阅中创建未加入域的 Linux 计算机目录。

## 使用 **Machine Creation Services (MCS)** 创建 **Linux VM**

May 8, 2023

要使用 MCS 创建 Linux VM，请在您的虚拟机管理程序中准备主映像。此过程包括在模板 VM 上安装 VDA、在 Citrix Studio 中创建计算机目录、创建交付组以及执行某些配置任务。

> 注意：
>
> 如果您尝试在 Citrix Hypervisor、Microsoft Azure、VMware vSphere、AWS、GCP 或 Nutanix AHV 以外的其他虚拟机管理程序上准备主映像，可能会出现意外结果。
>
> 自 Citrix Virtual Apps and Desktops 7 2003 起，Microsoft Azure、AWS 和 GCP 不受支持。但是，您可以继续使用 Citrix Virtual Apps and Desktops 服务中的主机。

支持的发行版

| | Winbind | SSSD | Centrify | PBIS |
|---|---|---|---|---|
| RHEL 8.3 | 是 | 否 | 否 | 否 |
| CentOS 8.3 | 是 | 否 | 否 | 否 |
| RHEL 8.2 | 是 | 否 | 否 | 否 |
| CentOS 8.2 | 是 | 否 | 否 | 否 |
| RHEL 8.1 | 是 | 否 | 否 | 否 |
| CentOS 8.1 | 是 | 否 | 否 | 否 |
| RHEL 7.9 | 是 | 是 | 否 | 否 |
| CentOS 7.9 | 是 | 是 | 否 | 否 |
| RHEL 7.8 | 是 | 是 | 否 | 否 |
| CentOS 7.8 | 是 | 是 | 否 | 否 |
| Ubuntu 20.04 | 是 | 是 | 否 | 否 |
| Ubuntu 18.04 | 是 | 是 | 否 | 否 |
| Ubuntu 16.04 | 是 | 是 | 否 | 否 |
| Debian 10.7 | 是 | 是 | 否 | 否 |
| SUSE 12.5 | 是 | 是 | 否 | 否 |

### 使用 **MCS** 在 **Citrix Hypervisor** 上创建 **Linux VM**

步骤 **1**：准备主映像

主映像包含操作系统、非虚拟化应用程序、VDA 和其他软件。要准备主映像，请执行以下操作：

步骤 **1a**：安装 **Citrix VM Tools**　必须在模板 VM 上安装 Citrix VM Tools，每个 VM 才能使用 xe CLI 或 XenCenter。除非安装这些工具，否则 VM 性能会较低。如果没有这些工具，无法执行以下任何操作：

- 彻底关闭、重新启动或挂起 VM。
- 在 XenCenter 中查看 VM 性能数据。
- 迁移正在运行的 VM（通过 `XenMotion`）。
- 创建快照或带有内存（检查点）的快照，以及还原到快照。
- 在正在运行的 Linux VM 上调整 vCPU 数。

1. 运行以下命令装载名为 guest-tools.iso 的 Citrix VM Tools。

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

2. 根据您的 Linux 发行版，运行以下命令安装 `xe-guest-utilities` 软件包。

   对于 **RHEL/CentOS**：

```
sudo rpm -i /mnt/Linux/xe-guest-utilities_{
 package-version }
 _all.rpm
```

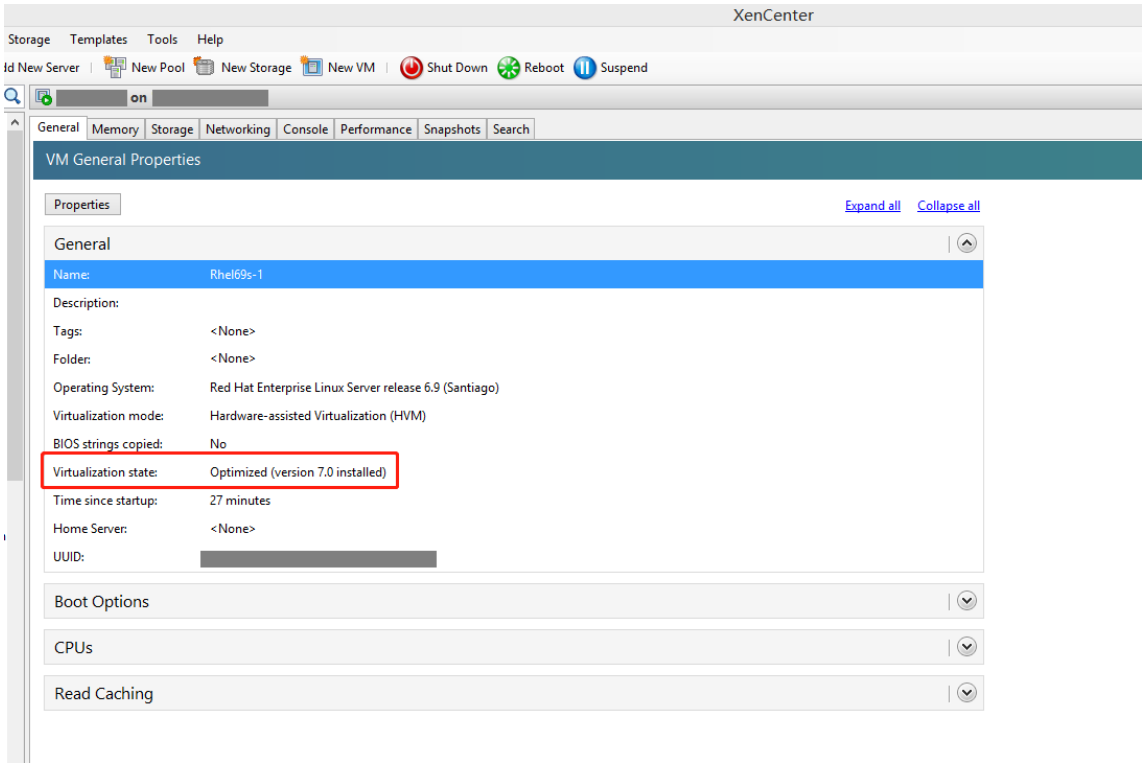
   对于 **Ubuntu/Debian**：

```
sudo dpkg -i /mnt/Linux/xe-guest-utilities_{
 package-version }
 _all.deb
```

   对于 **SUSE 12**：

```
sudo rpm -i /mnt/Linux/xe-guest-utilities_{
 package-version }
 _all.rpm
```

3. 在 XenCenter 中的常规选项卡上检查模板 VM 的虚拟化状态。如果正确安装了 Citrix VM Tools，则虚拟化状态为已优化：




步骤 **1b**：在模板 **VM** 上安装 **Linux VDA** 软件包

> 注意：
>
> 要使用当前运行的 VDA 作为模板 VM，请省略此步骤。

> 在模板 VM 上安装 Linux VDA 软件包之前，请安装.NET Core Runtime 3.1。有关详细信息，请参阅安装概述。

根据您的 Linux 发行版，运行以下命令为 Linux VDA 设置环境：

对于 **RHEL/CentOS**：

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

对于 **Ubuntu/Debian**：

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

apt-get install -f
```

对于 **SUSE 12**：

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

步骤 **1c**：启用存储库以安装 **tdb-tools** 软件包　对于 **RHEL 7** 服务器：

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

对于 **RHEL 7** 工作站：

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

步骤 **1d**：安装包含 **ntfs-3g** 的 **EPEL** 存储库　在 RHEL 8/CentOS 8、RHEL 7/CentOS 7 上安装 EPEL 存储库，以便稍后运行 deploymcs.sh 会安装其中包含的 ntfs-3g 软件包。

步骤 **1e**：在 **SUSE 12** 上手动安装 **ntfs-3g**　在 SUSE 12 平台上，没有提供 ntfs-3g 的存储库。下载源代码，编译并手动安装 ntfs-3g：

1. 安装 GNU Compiler Collection (GCC) 编译器系统并将软件包设置为：

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

2. 下载 ntfs-3g 软件包。

3. 解压缩 ntfs-3g 软件包：

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

4. 输入 ntfs-3g 软件包的路径：

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

5. 安装 ntfs-3g：

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

步骤 **1f**：设置运行时环境　在运行 `deploymcs.sh` 之前，请执行以下操作：

- 更改 `/etc/xdl/mcs/mcs.conf` 中的变量。`mcs.conf` 配置文件中包含用于设置 MCS 和 Linux VDA 的变量。以下是您可以根据需要设置的变量：
  - `Use_Existing_Configurations_Of_Current_VDA`：确定是否使用当前正在运行的 VDA 的现有配置。如果设置为 Y，MCS 创建的计算机上的配置文件将与当前正在运行的 VDA 上的等效文件相同。但是，您仍然必须配置 `dns` 和 `AD_INTEGRATION` 变量。默认值为 N，这意味着 MCS 创建的计算机上的配置文件由主映像上的配置模板确定。
  - `dns`：设置 DNS IP 地址。
  - `AD_INTEGRATION`：设置 Winbind 或 SSSD。有关 MSC 支持的 Linux 发行版和域加入方法的列表，请参阅本文中的支持的发行版。
  - `WORKGROUP`：如果在 AD 中配置了工作组，则设置工作组名称（区分大小写）。
- 在模板计算机上，将命令行添加到 `/etc/xdl/mcs/mcs_local_setting.reg` 文件以根据需要写入或更新注册表值。此操作可防止数据和设置在 MCS 预配的计算机每次重新启动时丢失。

  `/etc/xdl/mcs/mcs_local_setting.reg` 文件中的每一行就是用于设置或更新注册表值的一个命令。

  例如，您可以将以下命令行添加到 `/etc/xdl/mcs/mcs_local_setting.reg` 文件中，以分别写入或更新注册表值：

```
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 "0
    x00000003"
```

步骤 **1g**：创建主映像

1. 运行 `/opt/Citrix/VDA/sbin/deploymcs.sh`。
2. （可选）在模板 VM 上，更新配置模板以自定义创建的所有 VM 上的相关 `/etc/krb5.conf`、`/etc/samba/smb.conf` 和 `/etc/sssd/sssd.conf` 文件。

   对于 Winbind 用户，请更新 `/etc/xdl/mcs/winbind_krb5.conf.tmpl` 和 `/etc/xdl/mcs/winbind_smb.conf.tmpl` 模板。

对于 SSSD 用户，请更新 `/etc/xdl/mcs/sssd.conf.tmpl`、`/etc/xdl/mcs/sssd_krb5.conf.tmpl` 和 `/etc/xdl/mcs/sssd_smb.conf.tmpl` 模板。

> 注意：
>
> 请保留模板文件中使用的现有格式并使用变量，例如 $WORKGROUP、$REALM、$realm 和 $AD_FQDN。

3. 在 Citrix Hypervisor 上，关闭模板 VM。创建并命名主映像的快照。

步骤 **2**：创建计算机目录

在 Citrix Studio 中，创建计算机目录并指定要在目录中创建的 VM 数量。根据需要执行其他配置任务。有关详细信息，请参阅使用 Studio 创建计算机目录。

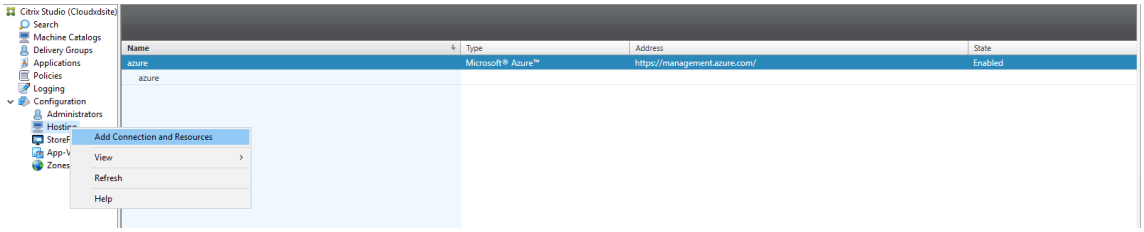
步骤 **3**：创建交付组

交付组是从一个或多个计算机目录中选择的计算机的集合。交付组指定哪些用户可以使用这些计算机，以及可供这些用户使用的应用程序和桌面。有关详细信息，请参阅创建交付组。

### 使用 **MCS** 在 **Azure** 上创建 **Linux VM**

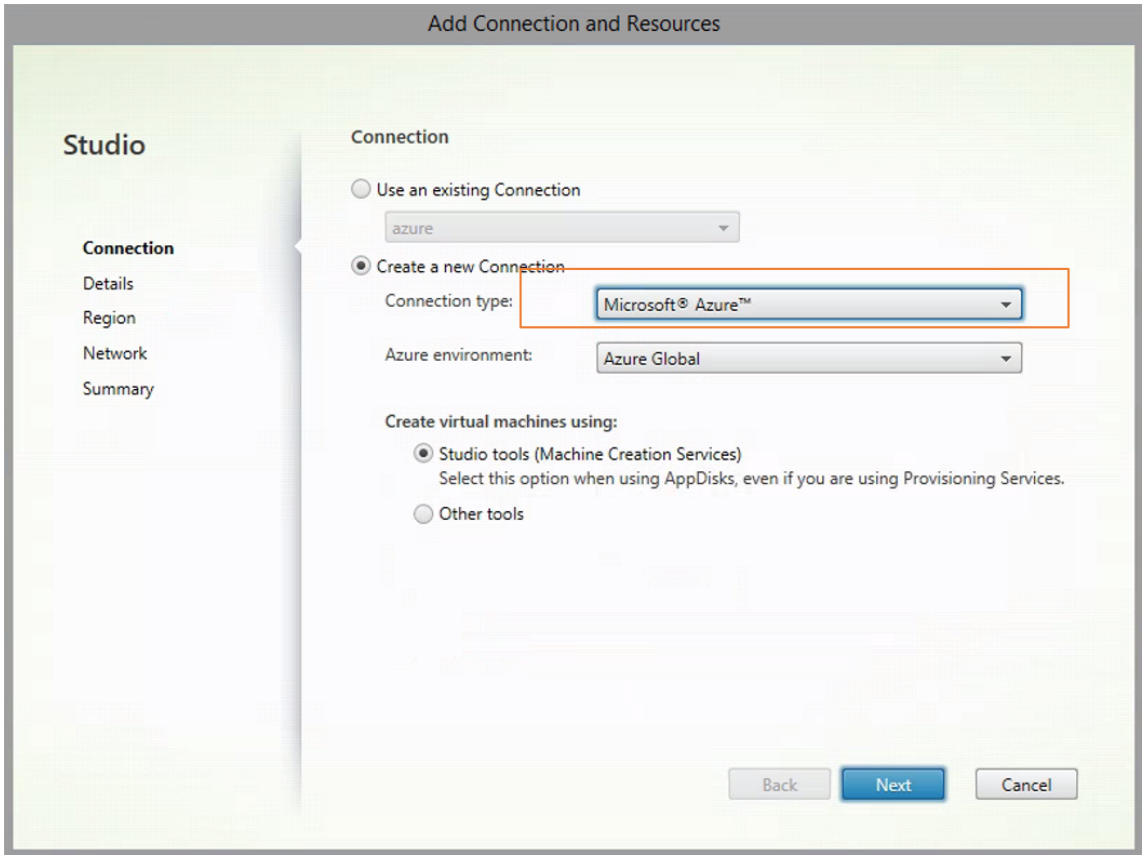
步骤 **1**：在 **Citrix Studio** 中创建与 **Azure** 的托管连接

1. 在 Citrix Cloud 上的 Citrix Studio 中，选择配置 **>** 托管 **>** 添加连接和资源以创建与 Azure 的连接。



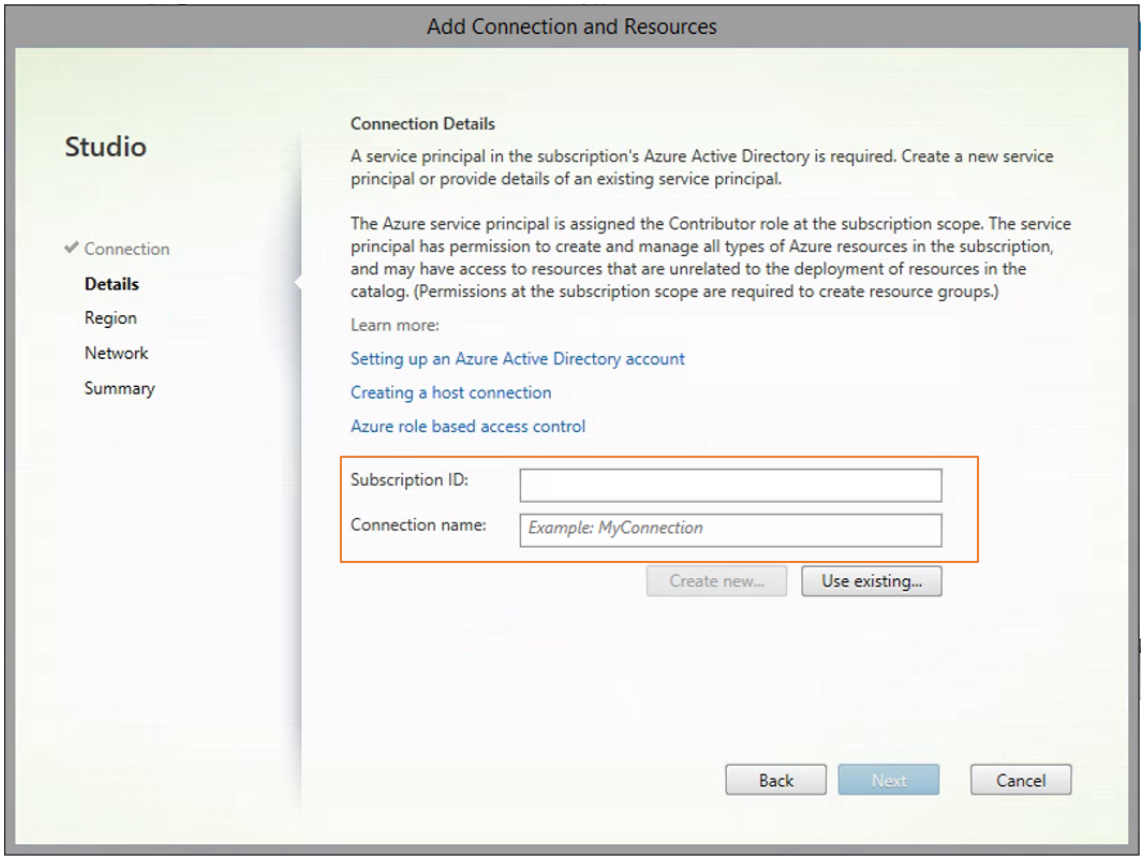

2. 选择连接类型 Microsoft Azure。

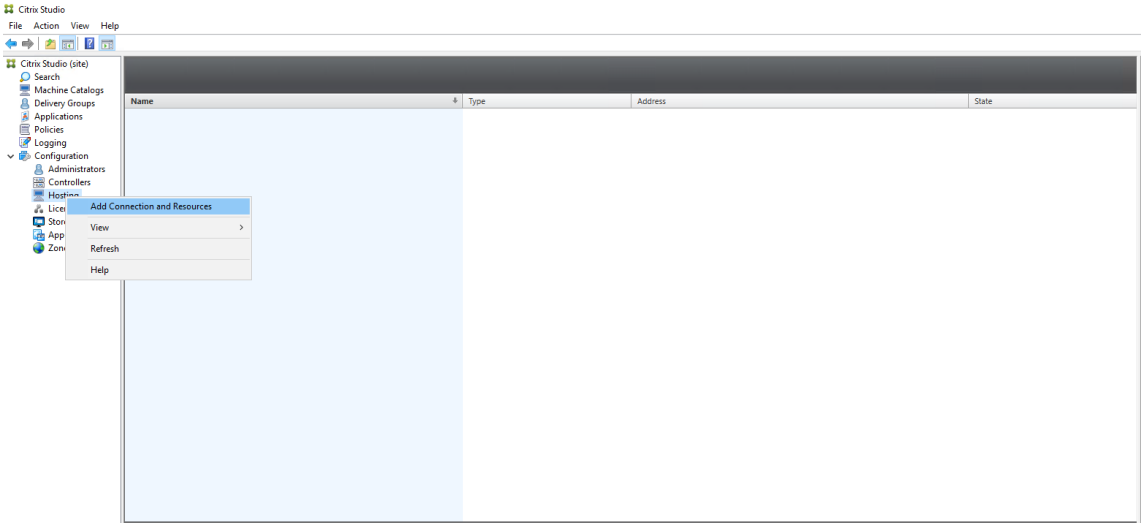



3. 键入您的 Azure 帐户的订阅 ID，然后键入您的连接名称。




将在托管窗格中显示一个新连接。




步骤 **2**：在模板 **VM** 上准备主映像

主映像包含操作系统、非虚拟化应用程序、VDA 和其他软件。要准备主映像，请执行以下操作：

步骤 **2a**：为 **Ubuntu 18.04** 配置 **cloud-init**　要确保在重新启动或停止 VM 时 VDA 主机名仍然存在，请运行以下命令。

```
echo "preserve_hostname: true" > /etc/cloud/cloud.cfg.d/99_hostname.cfg
```

确保在 /etc/cloud/cloud.cfg 文件中的 **system_info** 部分下存在以下行：

```
system_info:
   network:
       renderers: ['netplan', 'eni', 'sysconfig']
```

步骤 **2b**：在模板 **VM** 上安装 **Linux VDA** 软件包

> 注意：
>
> 要使用当前运行的 VDA 作为模板 VM，请省略此步骤。
>
> 在模板 VM 上安装 Linux VDA 软件包之前，请安装.NET Core Runtime 3.1。有关详细信息，请参阅安装概述。

根据您的 Linux 发行版，运行以下命令为 Linux VDA 设置环境：

对于 **RHEL/CentOS**：

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

对于 **Ubuntu/Debian**：

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

apt-get install -f
```

对于 **SUSE 12**：

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

步骤 **2c**：安装包含 **ntfs-3g** 的 **EPEL** 存储库　在 RHEL 8/CentOS 8、RHEL 7/CentOS 7 上安装 EPEL 存储库，以便稍后运行 deploymcs.sh 会安装其中包含的 ntfs-3g 软件包。

步骤 **2d**：在 **SUSE 12** 上手动安装 **ntfs-3g**　在 SUSE 12 平台上，没有提供 ntfs-3g 的存储库。下载源代码，编译并手动安装 ntfs-3g：

1. 安装 GNU Compiler Collection (GCC) 编译器系统并将软件包设置为：

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

2. 下载 ntfs-3g 软件包。

3. 解压缩 ntfs-3g 软件包：

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

4. 输入 ntfs-3g 软件包的路径：

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

5. 安装 ntfs-3g：

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

步骤 **2e**：设置运行时环境　在运行 `deploymcs.sh` 之前，请执行以下操作：

- 更改 `/etc/xdl/mcs/mcs.conf` 中的变量。`mcs.conf` 配置文件中包含用于设置 MCS 和 Linux VDA 的变量。下面是一些变量，其中必须设置 `dns` 和 `AD_INTEGRATION`：

  注意：如果为一个变量设置多个值，请将这些值放置在单引号内并用空格分隔。例如，LDAP_LIST=' aaa.lab:389 bbb.lab:389.'

  - `Use_Existing_Configurations_Of_Current_VDA`：确定是否使用当前正在运行的 VDA 的现有配置。如果设置为 Y，MCS 创建的计算机上的配置文件将与当前正在运行的 VDA 上的等效文件相同。但是，您仍然必须配置 `dns` 和 `AD_INTEGRATION` 变量。默认值为 N，这意味着 MCS 创建的计算机上的配置文件由主映像上的配置模板确定。
  - `dns`：设置 DNS IP 地址。
  - `AD_INTEGRATION`：设置 Winbind 或 SSSD（SUSE 不支持 SSSD）。
  - `WORKGROUP`：如果在 AD 中配置了工作组，则设置工作组名称（区分大小写）。

- 在模板计算机上，将命令行添加到 `/etc/xdl/mcs/mcs_local_setting.reg` 文件以根据需要写入或更新注册表值。此操作可防止数据和设置在 MCS 预配的计算机每次重新启动时丢失。

  `/etc/xdl/mcs/mcs_local_setting.reg` 文件中的每一行就是用于设置或更新注册表值的一个命令。

  例如，您可以将以下命令行添加到 `/etc/xdl/mcs/mcs_local_setting.reg` 文件中，以分别写入或更新注册表值：

```
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 "0
    x00000003"
```

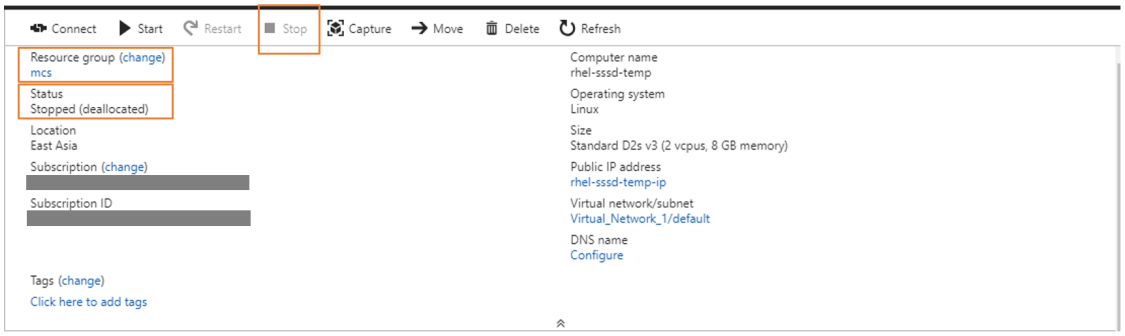
步骤 **2f**：创建主映像

1. 运行 `/opt/Citrix/VDA/sbin/deploymcs.sh`。
2. （可选）在模板 VM 上，更新配置模板以自定义创建的所有 VM 上的相关 `/etc/krb5.conf`、`/etc/samba/smb.conf` 和 `/etc/sssd/sssd.conf` 文件。

   对于 Winbind 用户，请更新 `/etc/xdl/mcs/winbind_krb5.conf.tmpl` 和 `/etc/xdl/mcs/winbind_smb.conf.tmpl` 模板。

   对于 SSSD 用户，请更新 `/etc/xdl/mcs/sssd.conf.tmpl`、`/etc/xdl/mcs/sssd_krb5.conf.tmpl` 和 `/etc/xdl/mcs/sssd_smb.conf.tmpl` 模板。

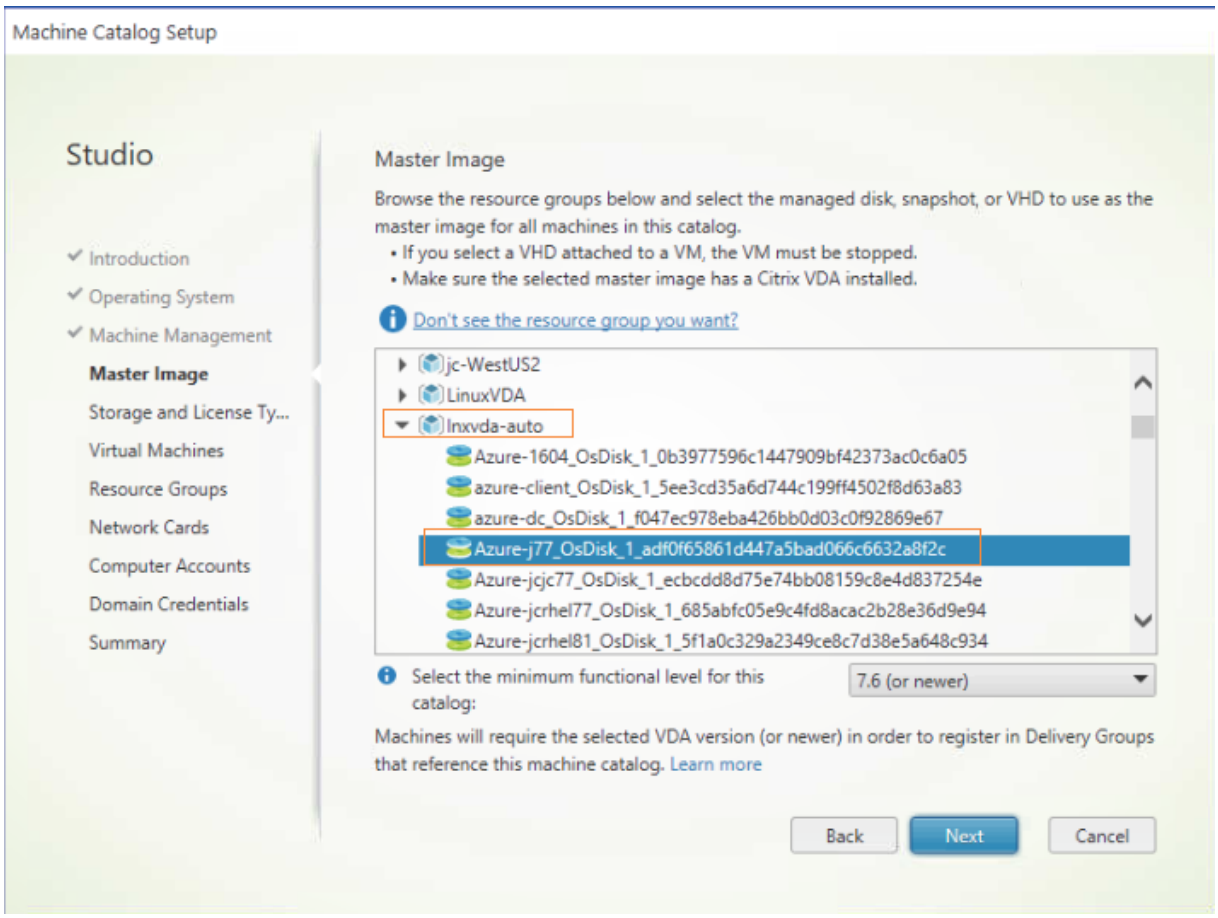
   注意：请保留模板文件中使用的现有格式并使用变量，例如 $WORKGROUP、$REALM、$realm 和 $AD_FQDN。
3. 在模板 VM 上安装应用程序后，从 Azure 门户关闭模板 VM。确保模板 VM 的电源状态为 **Stopped (deallocated)**（已停止 (已取消分配)）。记住此处的资源组名称。在 Azure 上查找您的主映像时需要该名称。




步骤 **3**：创建计算机目录

在 Citrix Studio 中，创建计算机目录并指定要在目录中创建的 VM 数量。创建计算机目录时，从模板 VM 所属的资源组中选择主映像，并查找模板 VM 的 VHD。




根据需要执行其他配置任务。有关详细信息，请参阅使用 Studio 创建计算机目录。

步骤 **4**：创建交付组

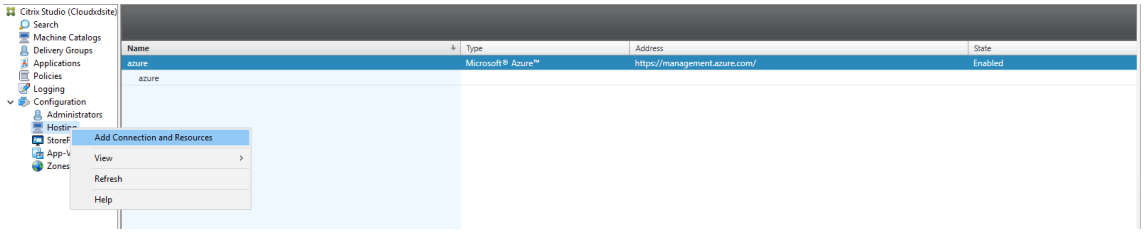
交付组是从一个或多个计算机目录中选择的计算机的集合。交付组指定哪些用户可以使用这些计算机，以及可供这些用户使用的应用程序和桌面。有关详细信息，请参阅创建交付组。

## 使用 **MCS** 在 **VMware vSphere** 上创建 **Linux VM**

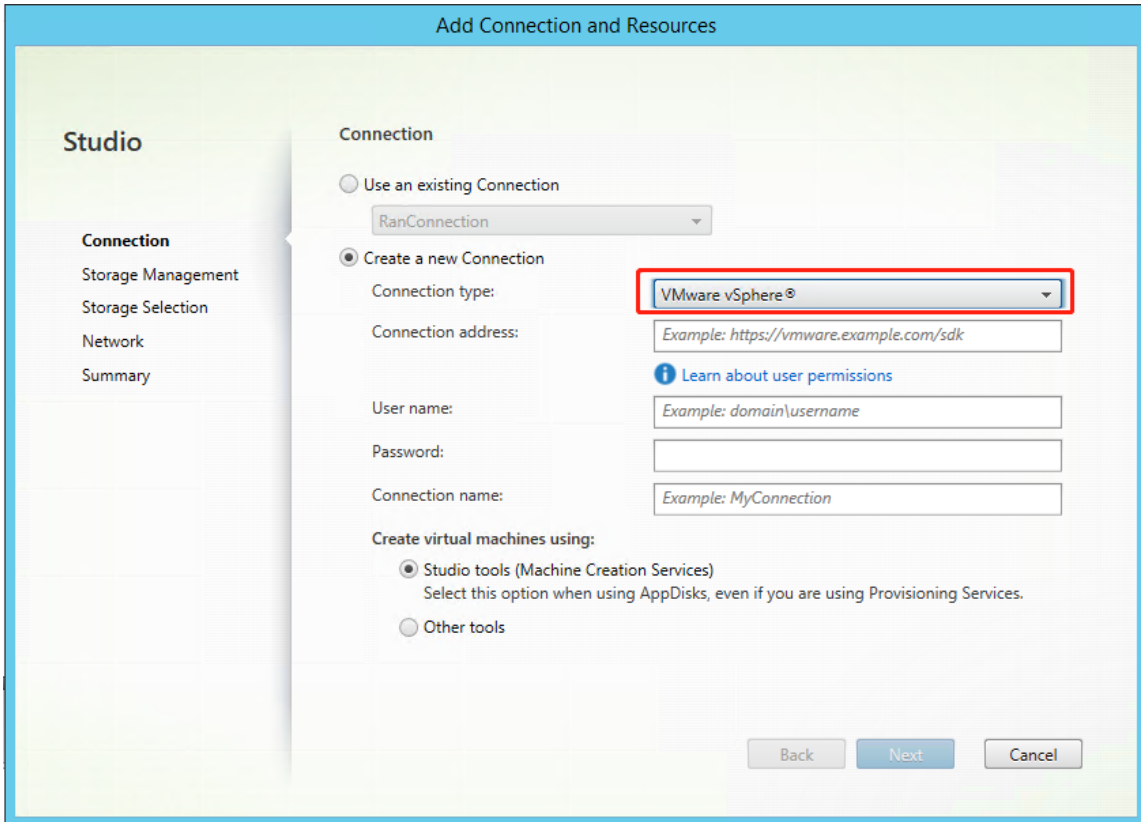
步骤 **1**：在 **Citrix Studio** 中创建与 **VMware** 的托管连接

1. 在 vSphere 环境中安装 vCenter Server。有关详细信息，请参阅 VMware vSphere。
2. 在 Citrix Studio 中，选择配置 **>** 托管 **>** 添加连接和资源以创建与 VMware vSphere 的连接。

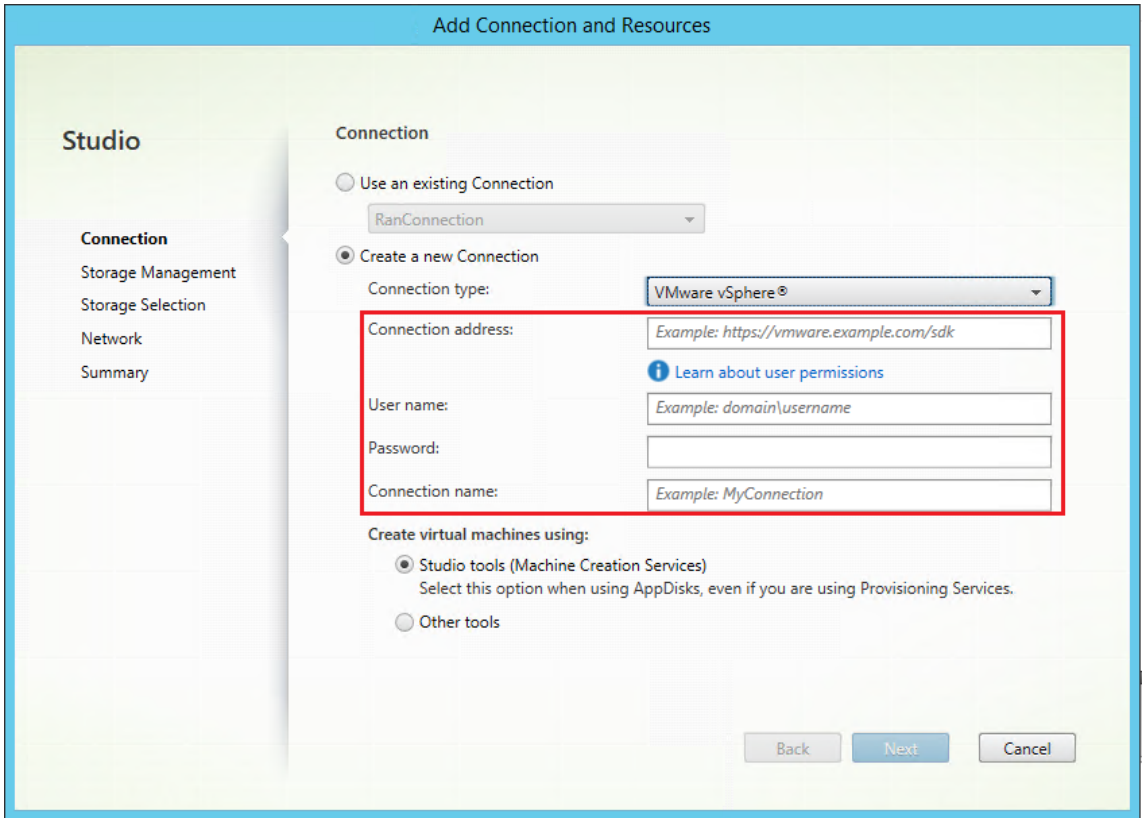



3. 选择 VMware vSphere 作为连接类型。

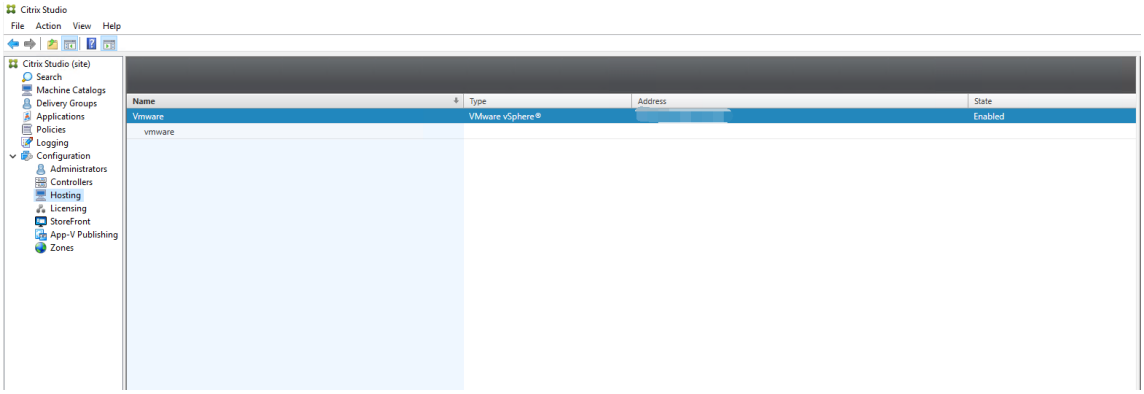



4. 键入 VMware 帐户的连接地址 (vCenter Server URL)、您的用户名和密码以及连接名称。




将在托管窗格中显示一个新连接。




步骤 **2**：准备主映像

主映像包含操作系统、非虚拟化应用程序、VDA 和其他软件。要准备主映像，请执行以下操作：

步骤 **2a**：在模板 **VM** 上安装 **Linux VDA** 软件包

> 注意：
>
> 要使用当前运行的 VDA 作为模板 VM，请省略此步骤。

> 在模板 VM 上安装 Linux VDA 软件包之前，请安装.NET Core Runtime 3.1。有关详细信息，请参阅安装概述。

根据您的 Linux 发行版，运行以下命令为 Linux VDA 设置环境：

对于 **RHEL/CentOS**：

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

对于 **Ubuntu/Debian**：

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

apt-get install -f
```

对于 **SUSE 12**：

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

步骤 **2b**：安装包含 **ntfs-3g** 的 **EPEL** 存储库　在 RHEL 8/CentOS 8、RHEL 7/CentOS 7 上安装 EPEL 存储库，以便稍后运行 deploymcs.sh 会安装其中包含的 ntfs-3g 软件包。

步骤 **2c**：在 **SUSE 12** 上手动安装 **ntfs-3g**　在 SUSE 12 平台上，没有提供 ntfs-3g 的存储库。下载源代码，编译并手动安装 ntfs-3g：

1. 安装 GNU Compiler Collection (GCC) 编译器系统并将软件包设置为：

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

2. 下载 ntfs-3g 软件包。

3. 解压缩 ntfs-3g 软件包：

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

4. 输入 ntfs-3g 软件包的路径：

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

5. 安装 ntfs-3g：

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

步骤 **2d**：设置运行时环境　在运行 `deploymcs.sh` 之前，请执行以下操作：

- 更改 /etc/xdl/mcs/mcs.conf 中的变量。mcs.conf 配置文件中包含用于设置 MCS 和 Linux VDA 的变量。下面是一些变量，其中必须设置 dns 和 AD_INTEGRATION：

  注意：如果为一个变量设置多个值，请将这些值放置在单引号内并用空格分隔。例如，LDAP_LIST=' aaa.lab:389 bbb.lab:389.'

  - Use_Existing_Configurations_Of_Current_VDA：确定是否使用当前正在运行的 VDA 的现有配置。如果设置为 Y，MCS 创建的计算机上的配置文件将与当前正在运行的 VDA 上的等效文件相同。但是，您仍然必须配置 dns 和 AD_INTEGRATION 变量。默认值为 N，这意味着 MCS 创建的计算机上的配置文件由主映像上的配置模板确定。
  - dns：设置 DNS IP 地址。
  - AD_INTEGRATION：设置 Winbind 或 SSSD（SUSE 不支持 SSSD）。
  - WORKGROUP：如果在 AD 中配置了工作组，则设置工作组名称（区分大小写）。

- 在模板计算机上，将命令行添加到 /etc/xdl/mcs/mcs_local_setting.reg 文件以根据需要写入或更新注册表值。此操作可防止数据和设置在 MCS 预配的计算机每次重新启动时丢失。

  /etc/xdl/mcs/mcs_local_setting.reg 文件中的每一行就是用于设置或更新注册表值的一个命令。

  例如，您可以将以下命令行添加到 /etc/xdl/mcs/mcs_local_setting.reg 文件中，以分别写入或更新注册表值：

```
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"
```

步骤 **2e**：创建主映像

1. 运行 /opt/Citrix/VDA/sbin/deploymcs.sh。
2. （可选）在模板 VM 上，更新配置模板以自定义创建的所有 VM 上的相关 /etc/krb5.conf、/etc/samba/smb.conf 和 /etc/sssd/sssd.conf 文件。

   对于 Winbind 用户，请更新 /etc/xdl/mcs/winbind_krb5.conf.tmpl 和 /etc/xdl/mcs/winbind_smb.conf.tmpl 模板。

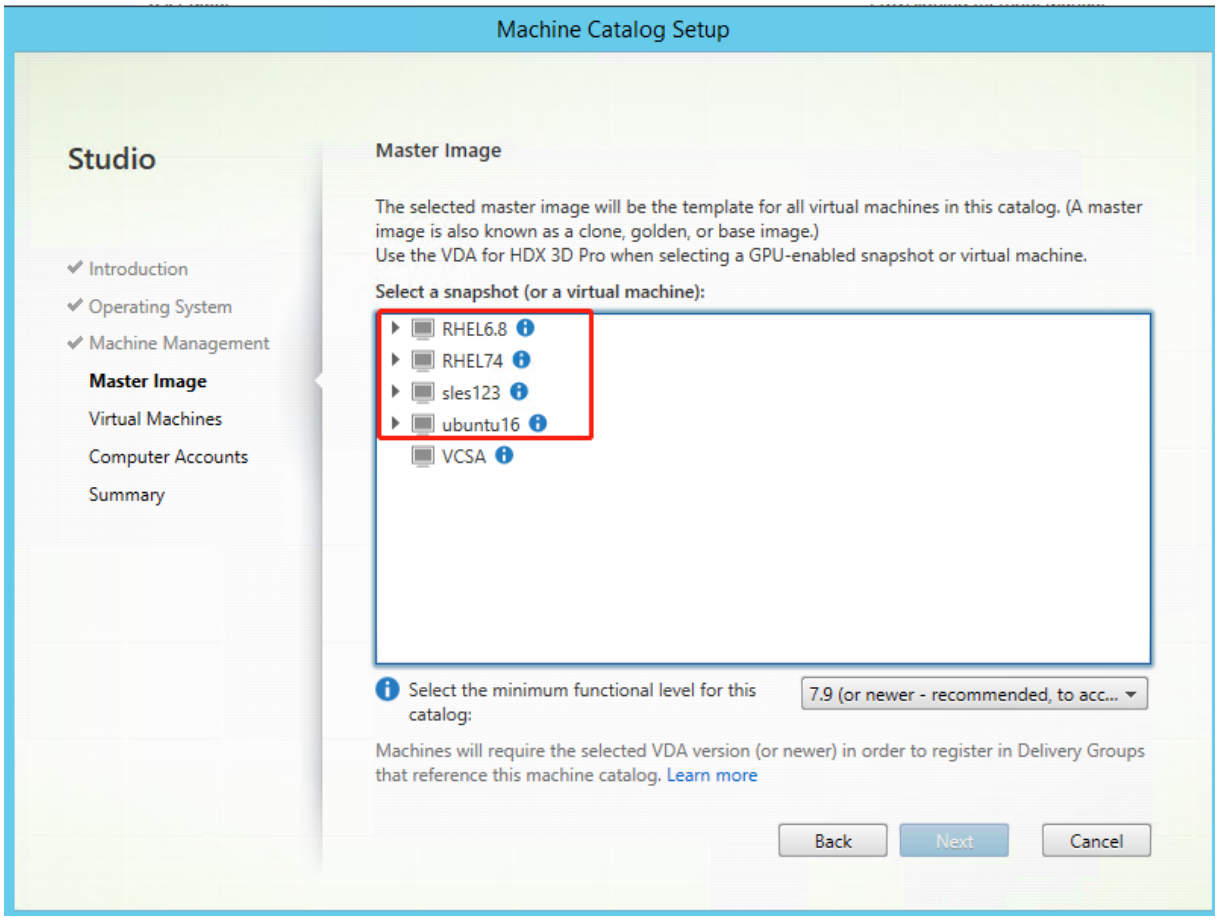
   对于 SSSD 用户，请更新 /etc/xdl/mcs/sssd.conf.tmpl、/etc/xdl/mcs/sssd_krb5.conf.tmpl 和 /etc/xdl/mcs/sssd_smb.conf.tmpl 模板。

   注意：请保留模板文件中使用的现有格式并使用变量，例如 $WORKGROUP、$REALM、$realm 和 $AD_FQDN。
3. 在模板 VM 上安装应用程序后，从 VMware 门户关闭模板 VM。创建模板 VM 的快照。

步骤 **3**：创建计算机目录

在 Citrix Studio 中，创建计算机目录并指定要在目录中创建的 VM 数量。创建计算机目录时，请从快照列表中选择主映像。




根据需要执行其他配置任务。有关详细信息，请参阅使用 Studio 创建计算机目录。

步骤 **4**：创建交付组

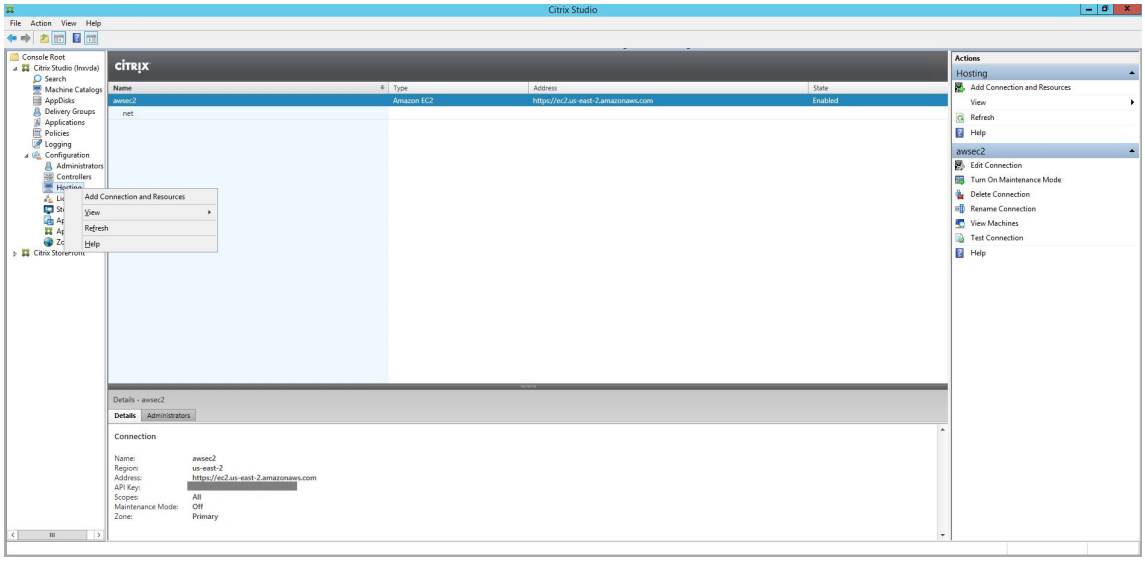
交付组是从一个或多个计算机目录中选择的计算机的集合。交付组指定哪些用户可以使用这些计算机，以及可供这些用户使用的应用程序和桌面。有关详细信息，请参阅创建交付组。

## 在 **AWS** 上使用 **MCS** 创建 **Linux VM**

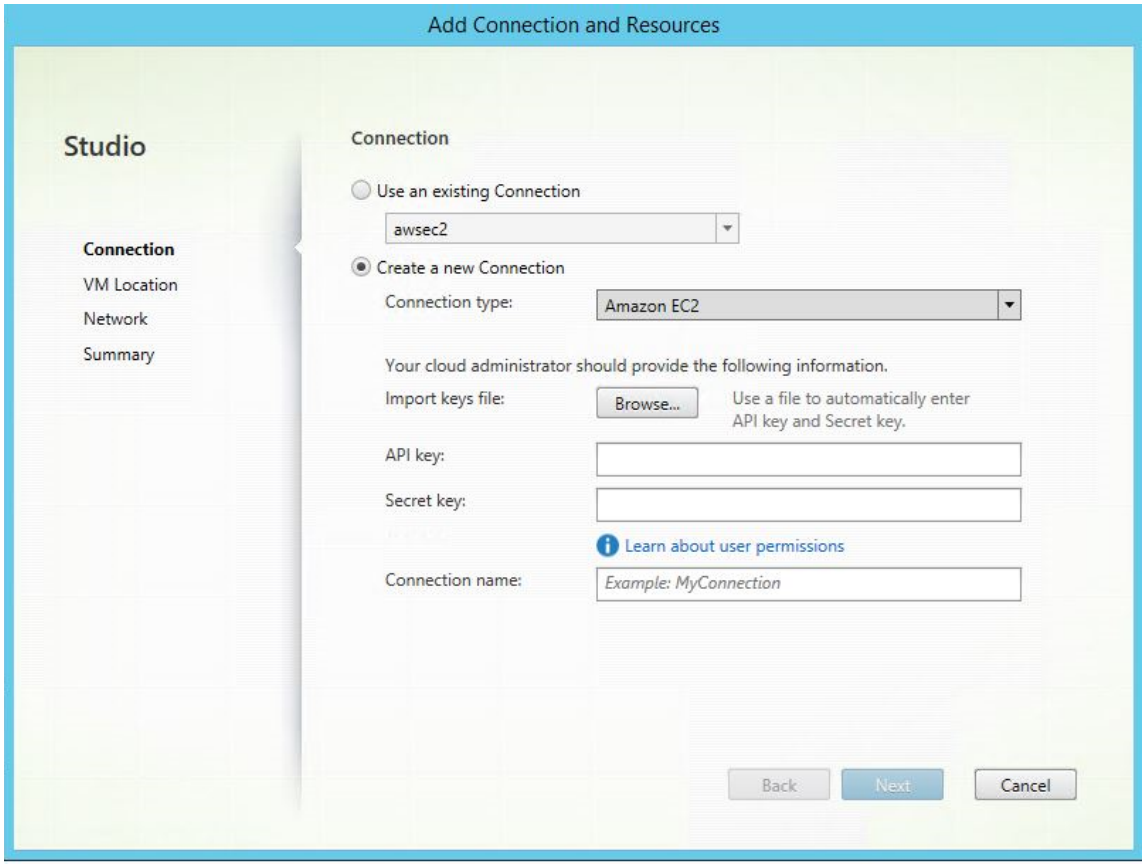
步骤 **1**：在 **Citrix Studio** 中创建与 **AWS** 的托管连接

1. 在 Citrix Cloud 上的 Citrix Studio 中，选择配置 > 托管 > 添加连接和资源以创建与 AWS 的连接。

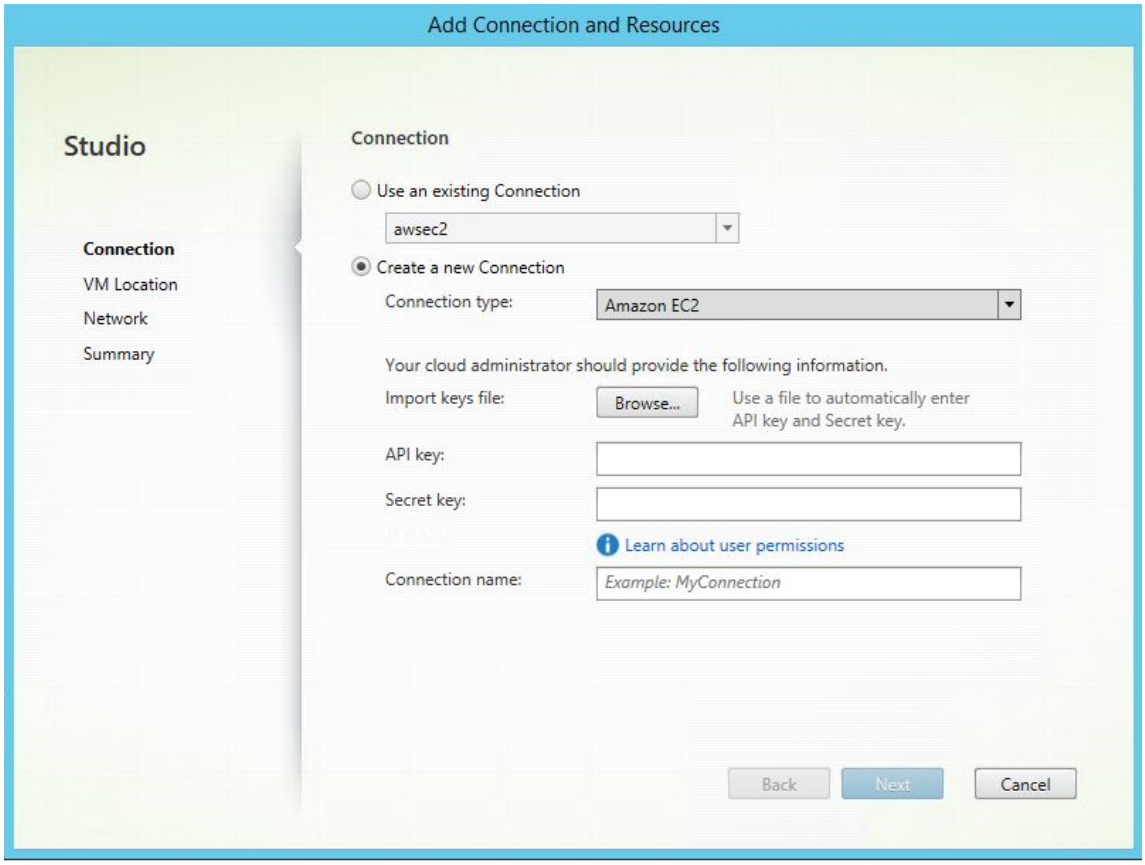



2. 选择 **Amazon EC2** 作为连接类型。




3. 键入 AWS 帐户的 API 密钥和密钥，然后键入您的连接名称。

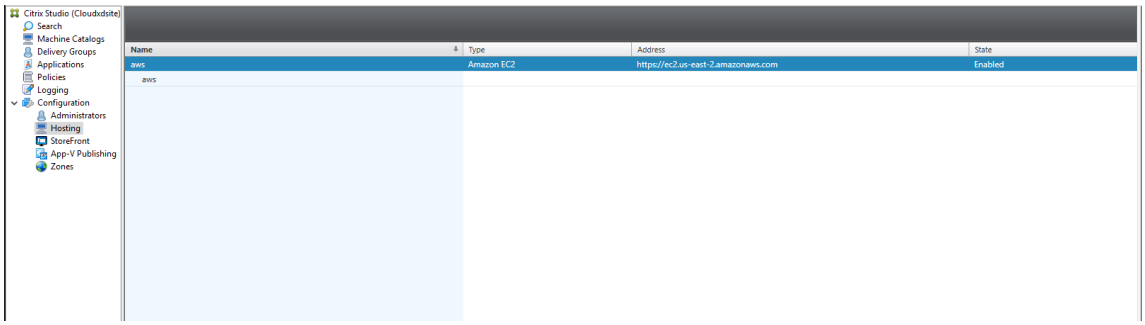
 




**API** 密钥是您的访问密钥 ID，密钥是您的秘密访问密钥。这些密钥被视为访问密钥对。如果您丢失了秘密访问密钥，则可以删除该访问密钥并创建另一个访问密钥。要创建访问密钥，请执行以下操作：

a) 登录 AWS 服务。
b) 导航到身份和访问管理 (IAM) 控制台。
c) 在左侧导航窗格中，选择用户。
d) 选择目标用户并向下滚动以选择安全凭据选项卡。
e) 向下滚动并单击创建访问密钥。此时将显示一个新窗口。
f) 单击下载**.csv** 文件并将访问密钥保存到一个安全的位置。

将在托管窗格中显示一个新连接。



步骤 **2**：准备主映像

主映像包含操作系统、非虚拟化应用程序、VDA 和其他软件。要准备主映像，请执行以下操作：

步骤 **2a**：配置 **cloud-init**

1. 要确保在重新启动或停止 EC2 实例时 VDA 主机名仍然存在，请运行以下命令以保留 VDA 主机名。

```
echo "preserve_hostname: true" > /etc/cloud/cloud.cfg.d/99_hostname.cfg
```

   对于 Ubuntu 18.04，请确保在 /etc/cloud/cloud.cfg 文件中的 system_info 部分下存在以下行：

```
system_info:
    network:
        renderers: ['netplan', 'eni', 'sysconfig']
```

2. 要使用 SSH 远程访问 AWS 上 MCS 创建的 VM，请启用密码身份验证，因为这些 VM 没有附加的密钥名称。根据需要执行以下操作。

   - 编辑 `cloud-init` 配置文件 /etc/cloud/cloud.cfg。确保 **ssh_pwauth: true** 行存在。删除或注释 **set-password** 行和以下行（如果存在）。

```
users:
- default
```

   - 如果您计划使用通过 `cloud-init` 创建的默认用户 `ec2-user` 或 `ubuntu`，则可以使用 `passwd` 命令更改用户密码。请记住新密码，以便以后用于登录 MCS 创建的 VM。

   - 编辑 `/etc/ssh/sshd_config` 文件以确保存在以下行：

```
PasswordAuthentication yes
```

     保存该文件并运行 `sudo service sshd restart` 命令。

步骤 **2b**：在模板 **VM** 上安装 **Linux VDA** 软件包

> 注意：
>
> 要使用当前运行的 VDA 作为模板 VM，请省略此步骤。
>
> 在模板 VM 上安装 Linux VDA 软件包之前，请安装.NET Core Runtime 3.1。有关详细信息，请参阅安装概述。

根据您的 Linux 发行版，运行以下命令为 Linux VDA 设置环境：

对于 **RHEL/CentOS**：

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

对于 **Ubuntu/Debian**：

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

apt-get install -f
```

对于 **SUSE 12**：

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

步骤 **2c**：安装包含 **ntfs-3g** 的 **EPEL** 存储库　在 RHEL 8/CentOS 8、RHEL 7/CentOS 7 上安装 EPEL 存储库，以便稍后运行 deploymcs.sh 会安装其中包含的 ntfs-3g 软件包。

步骤 **2d**：在 **SUSE 12** 上手动安装 **ntfs-3g**　在 SUSE 12 平台上，没有提供 ntfs-3g 的存储库。下载源代码，编译并手动安装 ntfs-3g：

1. 安装 GNU Compiler Collection (GCC) 编译器系统并将软件包设置为：

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

2. 下载 ntfs-3g 软件包。

3. 解压缩 ntfs-3g 软件包：

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

4. 输入 ntfs-3g 软件包的路径：

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

5. 安装 ntfs-3g：

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

步骤 **2e**：设置运行时环境　在运行 `deploymcs.sh` 之前，请执行以下操作：

- 更改 `/etc/xdl/mcs/mcs.conf` 中的变量。`mcs.conf` 配置文件中包含用于设置 MCS 和 Linux VDA 的变量。下面是一些变量，其中必须设置 `dns` 和 `AD_INTEGRATION`：

  注意：如果为一个变量设置多个值，请将这些值放置在单引号内并用空格分隔。例如，LDAP_LIST=' aaa.lab:389 bbb.lab:389.'

  - `Use_Existing_Configurations_Of_Current_VDA`：确定是否使用当前正在运行的 VDA 的现有配置。如果设置为 Y，MCS 创建的计算机上的配置文件将与当前正在运行的 VDA 上的等效文

件相同。但是，您仍然必须配置 dns 和 AD_INTEGRATION 变量。默认值为 N，这意味着 MCS 创建的计算机上的配置文件由主映像上的配置模板确定。
  - dns：设置 DNS IP 地址。
  - AD_INTEGRATION：设置 Winbind 或 SSSD（SUSE 不支持 SSSD）。
  - WORKGROUP：如果在 AD 中配置了工作组，则设置工作组名称（区分大小写）。

- 在模板计算机上，将命令行添加到 /etc/xdl/mcs/mcs_local_setting.reg 文件以根据需要写入或更新注册表值。此操作可防止数据和设置在 MCS 预配的计算机每次重新启动时丢失。

  /etc/xdl/mcs/mcs_local_setting.reg 文件中的每一行就是用于设置或更新注册表值的一个命令。

  例如，您可以将以下命令行添加到 /etc/xdl/mcs/mcs_local_setting.reg 文件中，以分别写入或更新注册表值：

```
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 "0
    x00000003"
```

步骤 **2f**：创建主映像

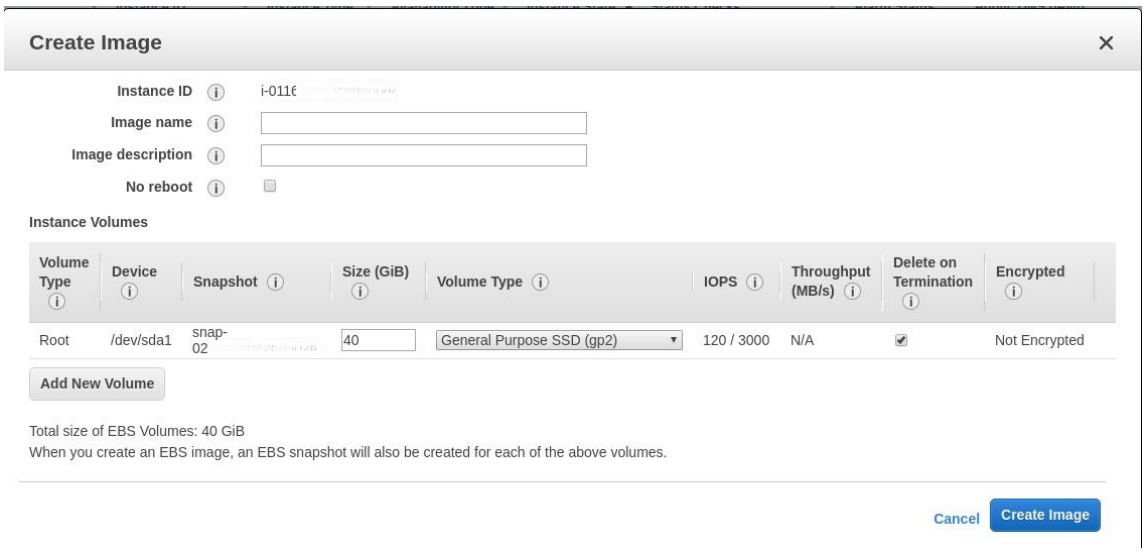
1. 运行 /opt/Citrix/VDA/sbin/deploymcs.sh。
2. （可选）在模板 VM 上，更新配置模板以自定义创建的所有 VM 上的相关 /etc/krb5.conf、/etc/samba/smb.conf 和 /etc/sssd/sssd.conf 文件。

   对于 Winbind 用户，请更新 /etc/xdl/mcs/winbind_krb5.conf.tmpl 和 /etc/xdl/mcs/winbind_smb.conf.tmpl 模板。

   对于 SSSD 用户，请更新 /etc/xdl/mcs/sssd.conf.tmpl、/etc/xdl/mcs/sssd_krb5.conf.tmpl 和 /etc/xdl/mcs/sssd_smb.conf.tmpl 模板。

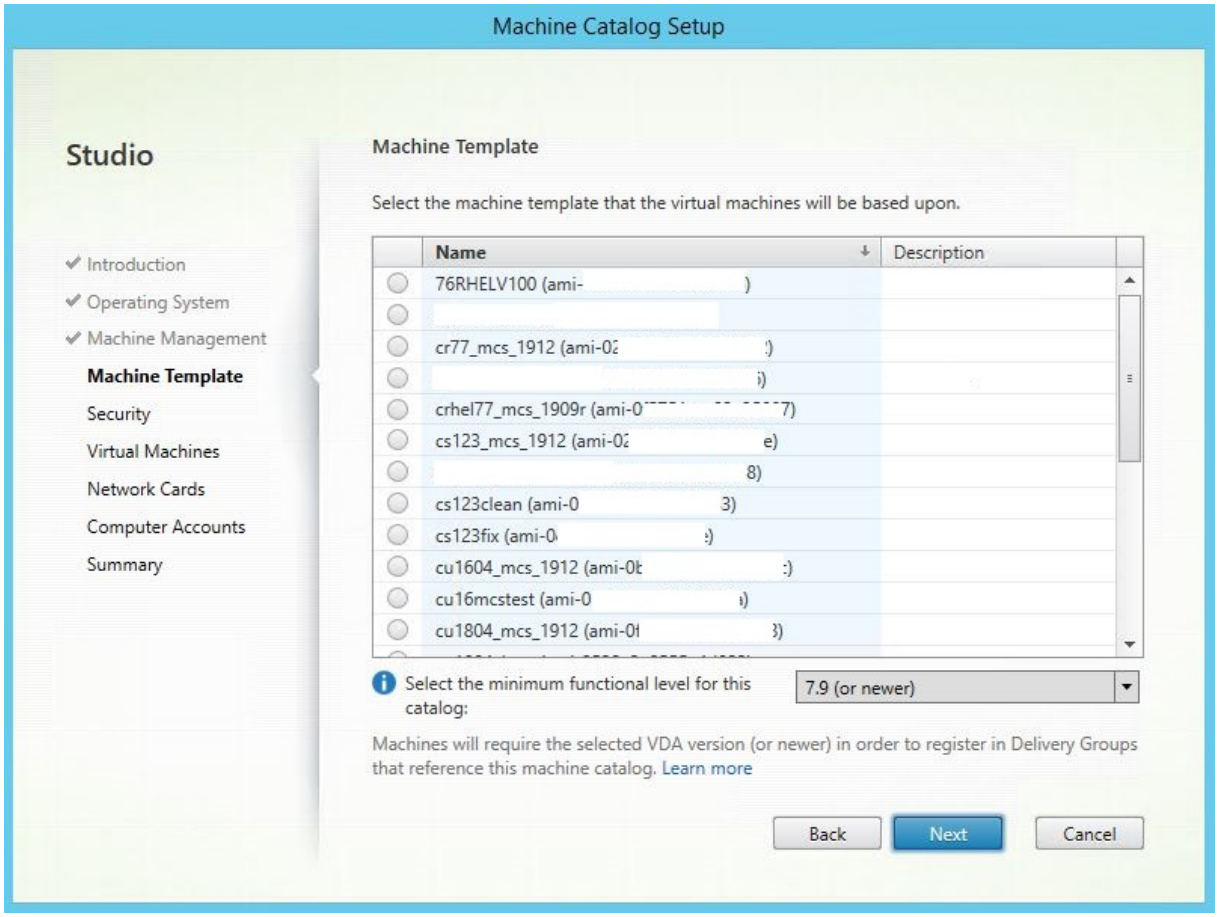
   注意：请保留模板文件中使用的现有格式并使用变量，例如 $WORKGROUP、$REALM、$realm 和 $AD_FQDN。
3. 在模板 VM 上安装应用程序后，从 AWS EC2 门户关闭模板 VM。确保模板 VM 的电源状态为已停止。
4. 右键单击模板 VM，然后选择映像 **>** 创建映像。键入信息并根据需要进行设置。单击创建映像。

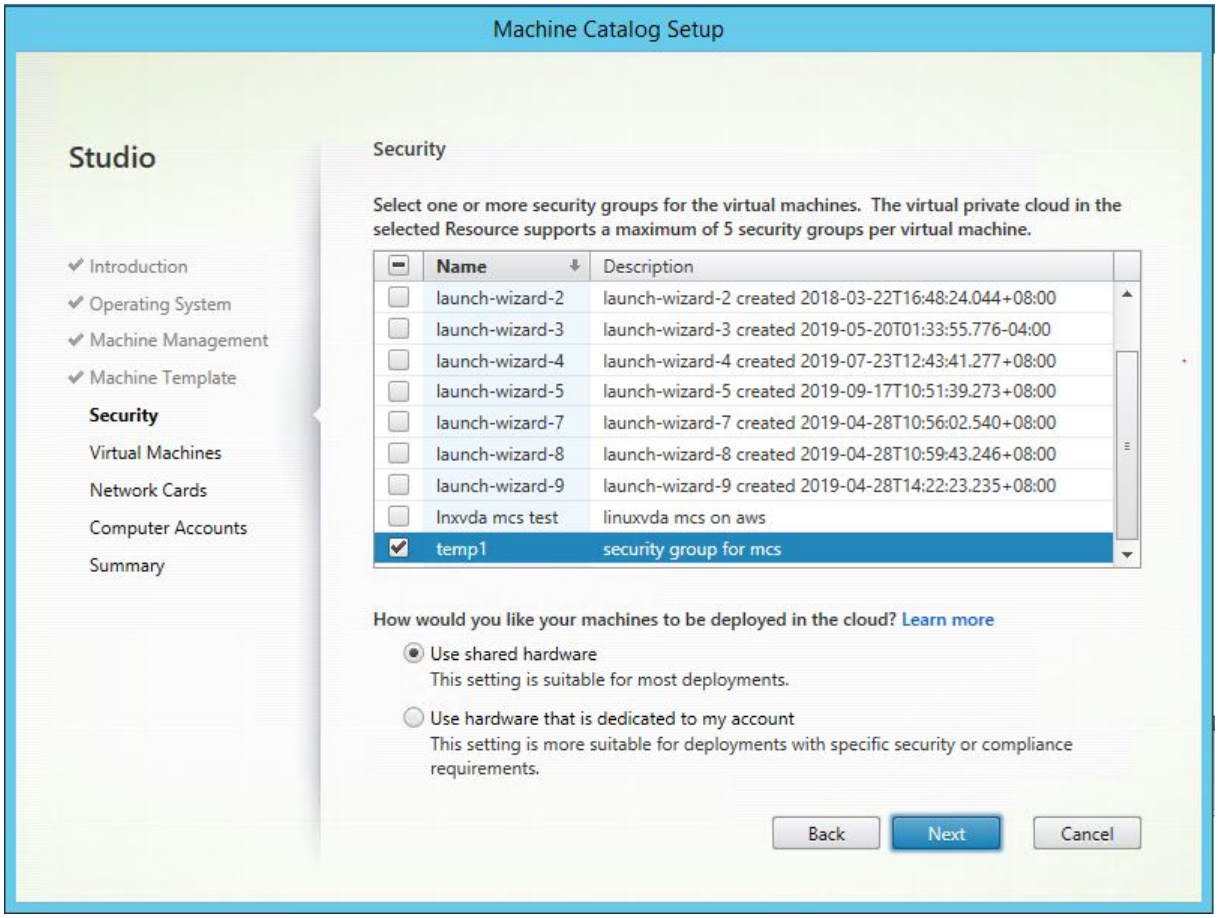



步骤 **3**：创建计算机目录

在 Citrix Studio 中，创建计算机目录并指定要在目录中创建的 VM 数量。创建计算机目录时，请选择您的计算机模板（之前创建的主映像），然后选择一个或多个安全组。






根据需要执行其他配置任务。有关详细信息，请参阅使用 Studio 创建计算机目录。

步骤 **4**：创建交付组

交付组是从一个或多个计算机目录中选择的计算机的集合。交付组指定哪些用户可以使用这些计算机，以及可供这些用户使用的应用程序和桌面。有关详细信息，请参阅创建交付组。

## 在 **GCP** 上使用 **MCS** 创建 **Linux VM**

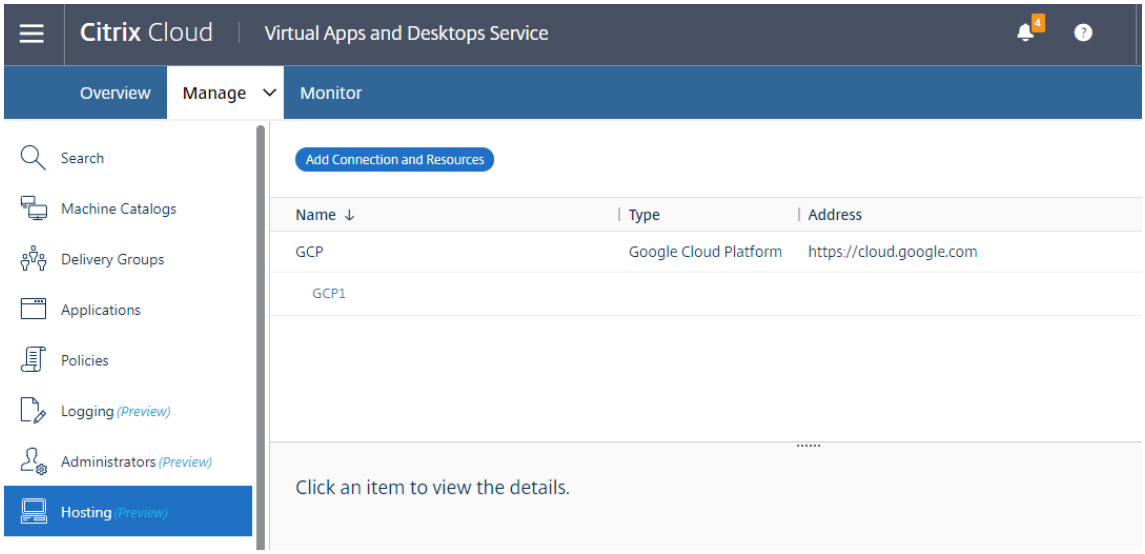
步骤 **1**：设置 **GCP** 环境

有关详细信息，请参阅 Google 云端平台虚拟化环境。

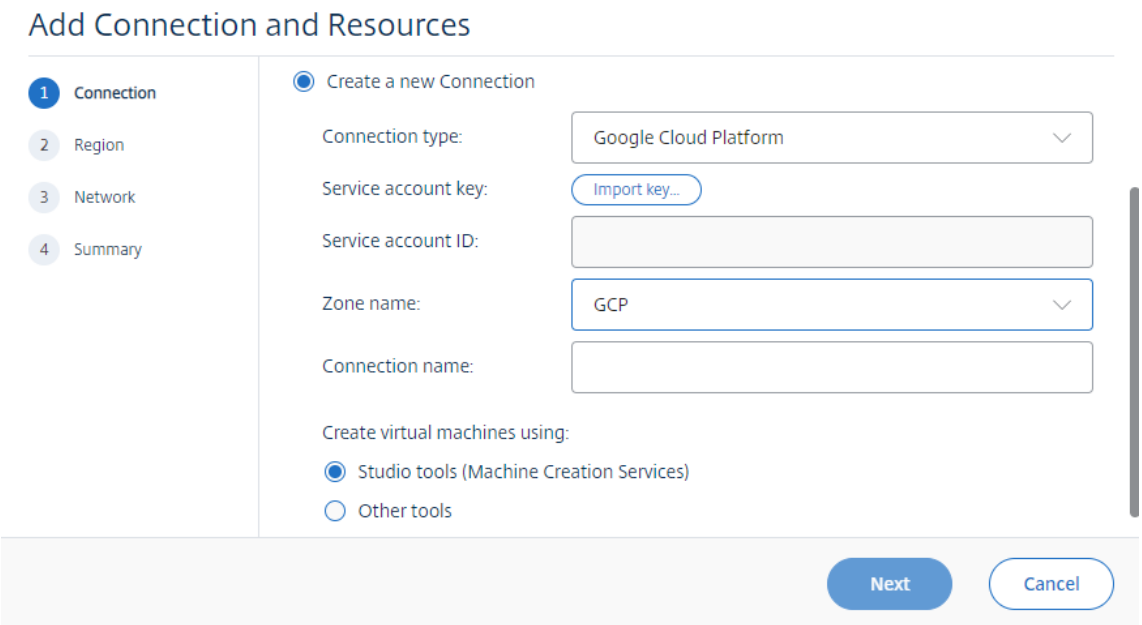
步骤 **2**：在 **Citrix Studio** 中创建与 **GCP** 的托管连接

1. 在 Citrix Cloud 上的 Citrix Studio 中，选择配置 > 托管 > 添加连接和资源以创建与 GCP 的连接。

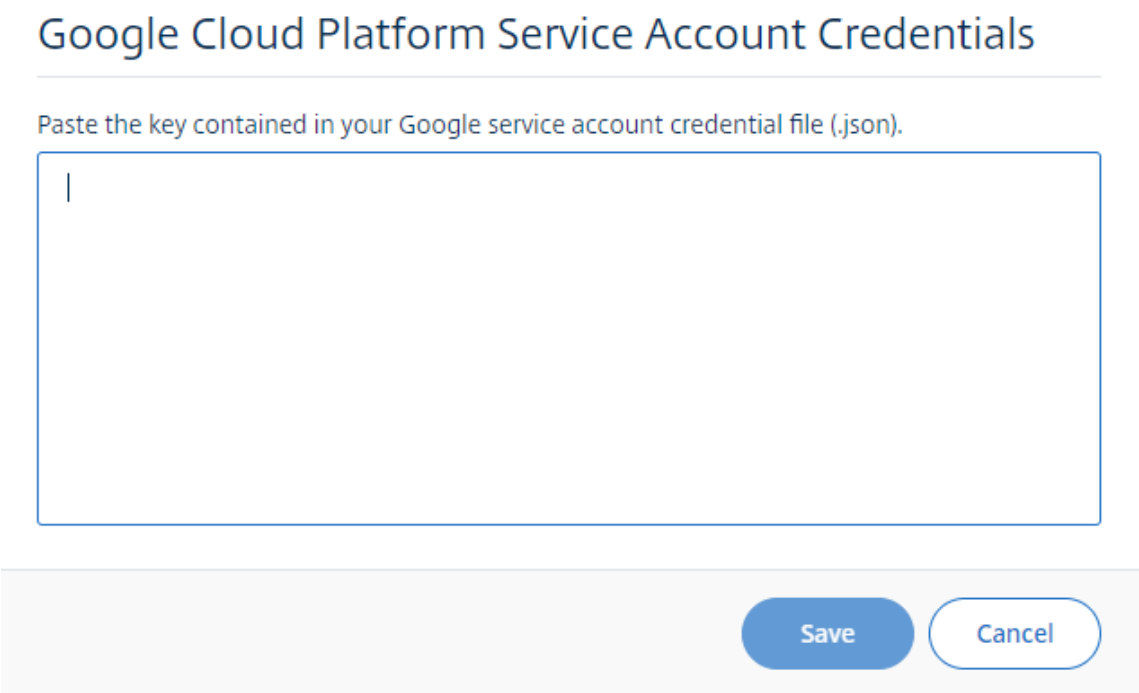



2. 选择 **Google Cloud Platform**（Google 云端平台）作为连接类型。

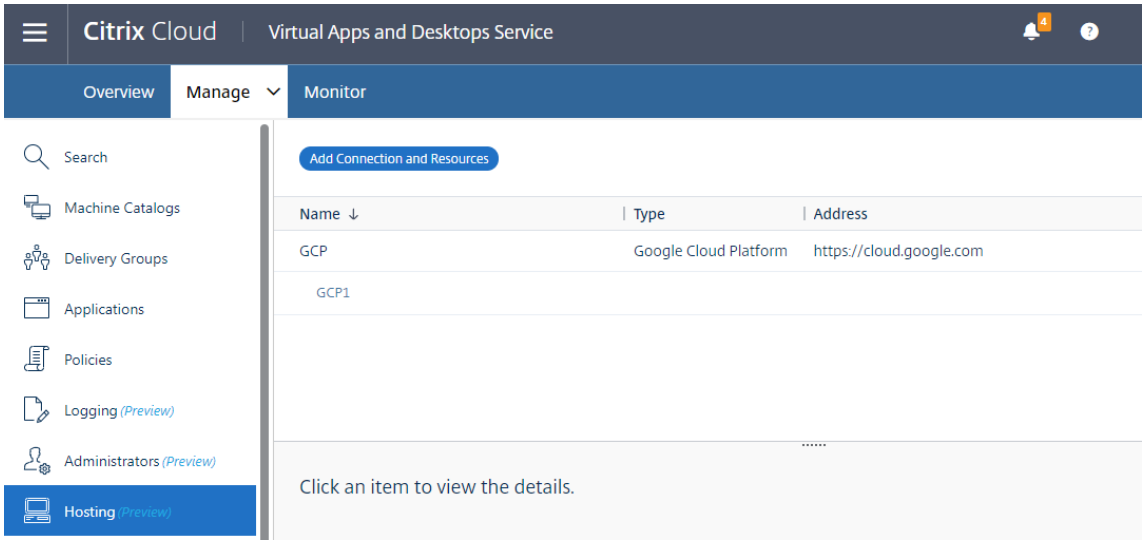



3. 导入 GCP 帐户的服务帐号密钥并键入连接名称。




将在托管窗格中显示一个新连接。




步骤 **3**：准备主映像

主映像包含操作系统、非虚拟化应用程序、VDA 和其他软件。要准备主映像，请执行以下操作：

步骤 **3a**：在模板 **VM** 上安装 **Linux VDA** 软件包

> 注意：
>
> 要使用当前运行的 VDA 作为模板 VM，请省略此步骤。
>
> 在模板 VM 上安装 Linux VDA 软件包之前，请安装.NET Core Runtime 3.1。有关详细信息，请参阅安装概述。

根据您的 Linux 发行版，运行以下命令为 Linux VDA 设置环境：

对于 **RHEL/CentOS**：

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

对于 **Ubuntu/Debian**：

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

apt-get install -f
```

对于 **SUSE 12**：

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

步骤 **3b**：安装包含 **ntfs-3g** 的 **EPEL** 存储库　在 RHEL 8/CentOS 8、RHEL 7/CentOS 7 上安装 EPEL 存储库，以便稍后运行 deploymcs.sh 会安装其中包含的 ntfs-3g 软件包。

步骤 **3c**：在 **SUSE 12** 上手动安装 **ntfs-3g**　在 SUSE 12 平台上，没有提供 ntfs-3g 的存储库。下载源代码，编译并手动安装 ntfs-3g：

1. 安装 GNU Compiler Collection (GCC) 编译器系统并将软件包设置为：

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

2. 下载 ntfs-3g 软件包。

3. 解压缩 ntfs-3g 软件包：

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

4. 输入 ntfs-3g 软件包的路径：

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

5. 安装 ntfs-3g：

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

步骤 **3d**：设置运行时环境　在运行 deploymcs.sh 之前，请执行以下操作：

- 更改 /etc/xdl/mcs/mcs.conf 中的变量。mcs.conf 配置文件中包含用于设置 MCS 和 Linux VDA 的变量。下面是一些变量，其中必须设置 dns 和 AD_INTEGRATION：

  注意：如果为一个变量设置多个值，请将这些值放置在单引号内并用空格分隔。例如，LDAP_LIST=' aaa.lab:389 bbb.lab:389.'

  - Use_Existing_Configurations_Of_Current_VDA：确定是否使用当前正在运行的 VDA 的现有配置。如果设置为 Y，MCS 创建的计算机上的配置文件将与当前正在运行的 VDA 上的等效文件相同。但是，您仍然必须配置 dns 和 AD_INTEGRATION 变量。默认值为 N，这意味着 MCS 创建的计算机上的配置文件由主映像上的配置模板确定。
  - dns：设置 DNS IP 地址。
  - AD_INTEGRATION：设置 Winbind 或 SSSD（SUSE 不支持 SSSD）。
  - WORKGROUP：如果在 AD 中配置了工作组，则设置工作组名称（区分大小写）。

- 在模板计算机上，将命令行添加到 /etc/xdl/mcs/mcs_local_setting.reg 文件以根据需要写入或更新注册表值。此操作可防止数据和设置在 MCS 预配的计算机每次重新启动时丢失。

  /etc/xdl/mcs/mcs_local_setting.reg 文件中的每一行就是用于设置或更新注册表值的一个命令。

  例如，您可以将以下命令行添加到 /etc/xdl/mcs/mcs_local_setting.reg 文件中，以分别写入或更新注册表值：

```
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"
```

步骤 **3e**：创建主映像

1. 运行 /opt/Citrix/VDA/sbin/deploymcs.sh。
2. （可选）在模板 VM 上，更新配置模板以自定义创建的所有 VM 上的相关 /etc/krb5.conf、/etc/samba/smb.conf 和 /etc/sssd/sssd.conf 文件。

   对于 Winbind 用户，请更新 /etc/xdl/mcs/winbind_krb5.conf.tmpl 和 /etc/xdl/mcs/winbind_smb.conf.tmpl 模板。

   对于 SSSD 用户，请更新 /etc/xdl/mcs/sssd.conf.tmpl、/etc/xdl/mcs/sssd_krb5.conf.tmpl 和 /etc/xdl/mcs/sssd_smb.conf.tmpl 模板。

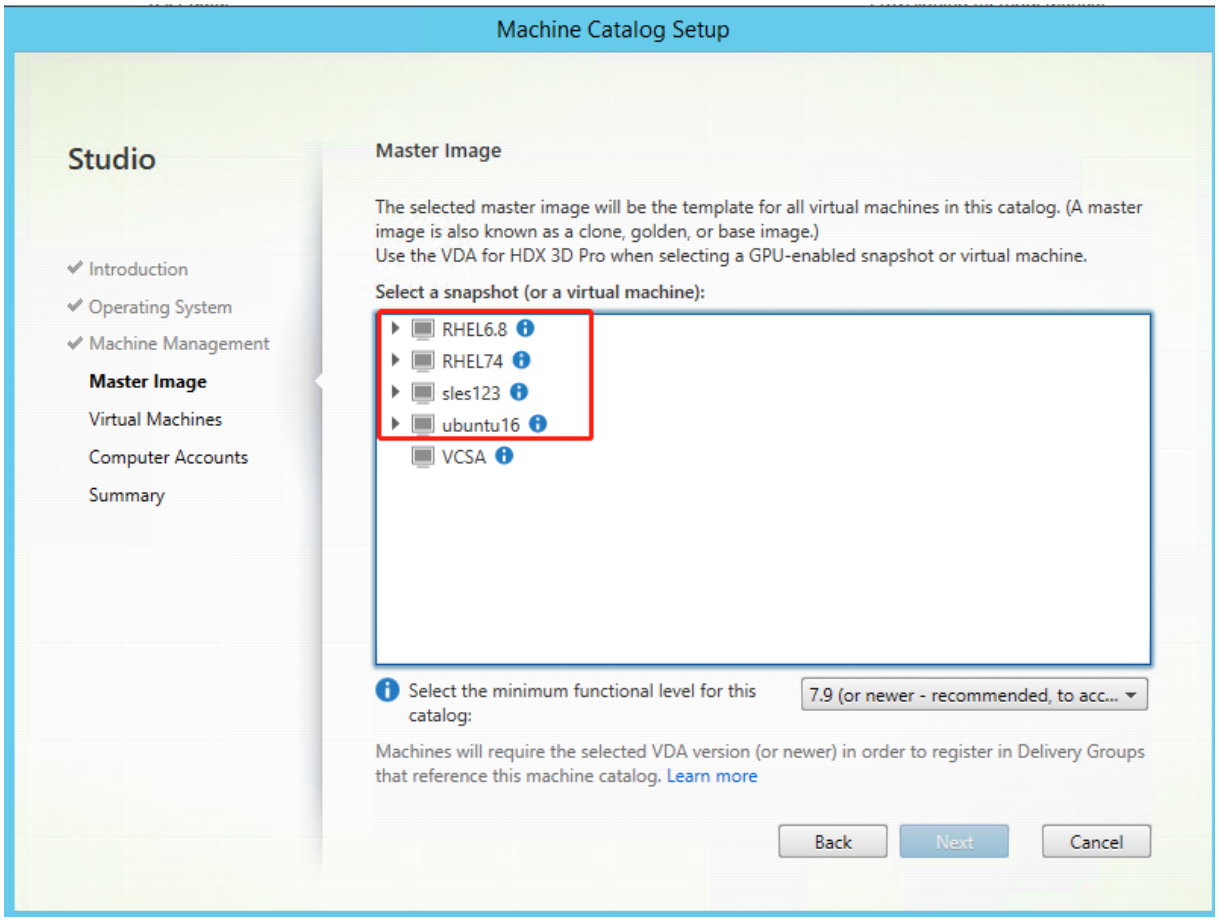
 

> 注意：
>
> 请保留模板文件中使用的现有格式并使用变量，例如 $WORKGROUP、$REALM、$realm 和 $AD_FQDN。

3. 在模板 VM 上安装应用程序后，从 VMware 门户关闭模板 VM。创建模板 VM 的快照。

步骤 **4**：创建计算机目录

在 Citrix Studio 中，创建计算机目录并指定要在目录中创建的 VM 数量。创建计算机目录时，请从快照列表中选择主映像。




根据需要执行其他配置任务。有关详细信息，请参阅使用 Studio 创建计算机目录。

步骤 **5**：创建交付组

交付组是从一个或多个计算机目录中选择的计算机的集合。交付组指定哪些用户可以使用这些计算机，以及可供这些用户使用的应用程序和桌面。有关详细信息，请参阅创建交付组。

### 使用 **MCS** 升级您的 **Linux VDA**

要使用 MCS 来升级 Linux VDA，请执行以下操作：

1. 在将 Linux VDA 升级到当前版本之前，请确保已安装.NET Core Runtime 3.1。
2. 在模板计算机上升级 Linux VDA：

   对于 **RHEL 7** 和 **CentOS 7**：

   ```
   sudo rpm -U XenDesktopVDA-<version>.el7_x.x86_64.rpm
   ```

   对于 **RHEL 8** 和 **CentOS 8**：

   ```
   sudo rpm -U XenDesktopVDA-<version>.el8_x.x86_64.rpm
   ```

   对于 **SUSE 12**：

   ```
   sudo rpm -U XenDesktopVDA-<version>.sle12_x.x86_64.rpm
   ```

   对于 **Ubuntu 16.04**：

   ```
   sudo dpkg -i xendesktopvda_<version>.ubuntu16.04_amd64.deb
   ```

   对于 **Ubuntu 18.04**：

   ```
   sudo dpkg -i xendesktopvda_<version>.ubuntu18.04_amd64.deb
   ```

   对于 **Ubuntu 20.04**：

   ```
   sudo dpkg -i xendesktopvda_<version>.ubuntu20.04_amd64.deb
   ```

3. 编辑 `/etc/xdl/mcs/mcs.conf` 和 `/etc/xdl/mcs/mcs_local_setting.reg`。
4. 生成新快照。
5. 在 Citrix Studio 中，选择用于更新计算机目录的新快照。在每台计算机重新启动之前，等待一段时间。请勿手动重新启动计算机。

### 自动更新计算机帐户密码

默认情况下，计算机帐户密码在创建计算机目录后 30 天过期。要防止密码过期以及自动更新计算机帐户密码，请执行以下操作：

1. 在运行 /opt/Citrix/VDA/sbin/deploymcs.sh 之前，将以下条目添加到 /etc/xdl/mcs/mcs.conf 中。

   ```
   UPDATE_MACHINE_PW="enabled"
   ```

2. 运行 /opt/Citrix/VDA/sbin/deploymcs.sh 后，打开 /etc/cron.d/mcs_update_password_cronjob 以设置更新时间和频率。默认设置每周星期日凌晨 2:30 更新计算机帐户密码。

每次更新计算机帐户密码后，Delivery Controller 上的票证缓存将变为无效，并且在 /var/log/xdl/jproxy.log 中可能会出现以下错误：

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

要消除错误，请定期清除票证缓存。可以在所有 Delivery Controller 或域控制器上安排缓存清理任务。

### 在 **MCS** 创建的 **VM** 上启用 **FAS**

可以在以下发行版上运行的 MCS 创建的虚拟机上启用 FAS：

| | Winbind | SSSD | Centrify |
|---|---|---|---|
| RHEL 8、CentOS 8 | 是 | 否 | 否 |
| RHEL 7、CentOS 7 | 是 | 是 | 否 |
| Ubuntu 20.04 | 是 | 否 | 否 |
| Ubuntu 18.04 | 是 | 否 | 否 |
| Ubuntu 16.04 | 是 | 否 | 否 |
| Debian 10.7 | 是 | 否 | 否 |
| SUSE 12.5 | 是 | 否 | 否 |

#### 在模板 **VM** 上准备主映像时启用 **FAS**

1. 运行脚本 `opt/Citrix/VDA/sbin/ctxinstall.sh` 并设置所有环境变量，例如 FAS 服务器列表。有关这些环境变量的详细信息，请参阅轻松安装。

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

2. 导入根 CA 证书。

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

3. 运行 ctxfascfg.sh。
4. 在 `/etc/xdl/mcs/mcs.conf` 中设置变量。

   a) 将 `Use_Existing_Configurations_Of_Current_VDA` 的值设置为 Y。
   b) 将 `FAS_LIST` 变量设置为您的 FAS 服务器地址或者由分号分隔并由双引号引起的多个 FAS 服务器地址，例如 `FAS_LIST="<FAS_SERVER_FQDN>;<FAS_SERVER_FQDN>"`。

c) 根据需要设置其他变量，例如 `VDI_MODE`。

5. 运行脚本 `/opt/Citrix/VDA/sbin/deploymcs.sh`。

#### 在 **MCS** 创建的 **VM** 上启用 **FAS**

如果未如上文所述在模板计算机上启用 FAS，则可以在每个 MCS 创建的 VM 上启用 FAS。

要在 MCS 创建的 VM 上启用 FAS，请执行以下操作：

1. 在 /etc/xdl/mcs/mcs.conf 中设置变量。

   a) 将 `Use_Existing_Configurations_Of_Current_VDA` 的值设置为 Y。
   b) 将 `FAS_LIST` 变量设置为 FAS 服务器地址。
   c) 根据需要设置其他变量，例如 `VDI_MODE`。

2. 导入根 CA 证书。

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

3. 运行脚本 `/opt/Citrix/VDA/sbin/ctxfascfg.sh`。

> 注意：
>
> 必须在 `/etc/xdl/mcs/mcs.conf` 中设置所有必需的变量，因为这些变量是在 VM 启动时调用的。

## 安装 **Linux Virtual Delivery Agent for RHEL/CentOS**

August 20, 2024

可以选择按照本文中的步骤进行手动安装，也可以使用轻松安装进行自动安装和配置。轻松安装省时又省力，与手动安装相比，更不易于出错。

> 注意：
>
> 请仅对全新安装使用轻松安装功能。请勿使用轻松安装更新现有安装。

### 步骤 **1**：为 **VDA** 安装准备 **RHEL 8/CentOS 8**、**RHEL 7/CentOS 7**

#### 步骤 **1a**：验证网络配置

我们建议您先连接并正确配置网络，然后再继续操作。

步骤 **1b**：设置主机名

为确保正确报告计算机的主机名，请更改 **/etc/hostname** 文件，使其仅包含计算机的主机名。

```
hostname
```

步骤 **1c**：为主机名分配环回地址

为确保正确报告计算机的 DNS 域名和完全限定域名 (FQDN)，请更改 `/etc/hosts` 文件中的以下行，使其前两个条目为 FQDN 和主机名：

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

例如：

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

从文件中的其他条目中删除对 **hostname-fqdn** 或 **hostname** 的任何其他引用。

> 注意：
>
> Linux VDA 当前不支持 NetBIOS 名称截断。因此，主机名不得超过 15 个字符。
>
> 提示：
>
> 只能使用字符 a-z、A-Z、0-9 和连字符 (-)。请避免使用下划线 (_)、空格和其他符号。主机名不得以数字开头和以连字符结尾。此规则也适用于 Delivery Controller 主机名。

步骤 **1d**：检查主机名

验证主机名设置是否正确无误：

```
hostname
```

此命令仅返回计算机的主机名，而不返回其完全限定域名 (FQDN)。

验证 FQDN 设置是否正确无误：

```
hostname -f
```

此命令返回计算机的 FQDN。

步骤 **1e**：检查名称解析和服务可访问性

确认可以解析 FQDN 并对域控制器和 Delivery Controller 执行 ping 操作：

```
nslookup domain-controller-fqdn

ping domain-controller-fqdn

nslookup delivery-controller-fqdn

ping delivery-controller-fqdn
```

如果无法解析 FQDN 或 Ping 不通上述任一计算机，请先检查相关步骤，然后再继续。

步骤 **1f**：配置时钟同步

确保 VDA、Delivery Controller 和域控制器之间的时钟始终精确同步至关重要。将 Linux VDA 托管为虚拟机可能会导致时钟偏差问题。出于此原因，最好使用远程时间服务来同步时间。

RHEL 8/RHEL 7 默认环境使用 Chrony 守护程序 (`chronyd`) 进行时钟同步。

配置 **Chrony** 服务　以 root 用户身份，编辑 **/etc/chrony.conf** 并为每个远程时间服务器添加一个服务器条目：

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

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

在典型部署中，时间从本地域控制器同步，而不是直接从公共 NTP 池服务器同步。为域中的每个 Active Directory 域控制器添加一个服务器条目。

删除列出的任何其他服务器条目，包括环回 IP 地址、localhost 以及公共服务器 ***.pool.ntp.org** 条目。

保存更改并重新启动 Chrony 守护程序：

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

步骤 **1g**：安装 **OpenJDK**

Linux VDA 依赖于 OpenJDK。通常，运行时环境作为操作系统安装的一部分进行安装。

确认版本是否正确：

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

预先封装的 OpenJDK 可能为早期版本。请根据需要更新为最新版本：

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

打开新的 shell，然后确认 Java 版本：

```
java -version
```

> 提示：
>
> 为避免在 Delivery Controller 中注册失败，请务必仅安装 OpenJDK 1.8.0。从系统中删除所有其他版本的 Java。

#### 步骤 **1h**：安装 **PostgreSQL**

Linux VDA 要求在 RHEL 8 上使用 PostgreSQL 10.5 或更高版本，或者在 RHEL 7 上使用 PostgreSQL 9.2 或更高版本。

安装以下软件包：

```
sudo yum -y install postgresql-server

sudo yum -y install postgresql-jdbc
```

此时需要执行一些安装后步骤，以便初始化数据库，并确保服务在计算机启动时启动。此操作会在 **/var/lib/pgsql/data** 下创建数据库文件。此命令在 PostgreSQL 10 和 9 上有所差别：

```
sudo postgresql-setup initdb
```

#### 步骤 **1i**：启动 **PostgreSQL**

在计算机启动时启动服务和立即启动服务：

```
sudo systemctl enable postgresql

sudo systemctl start postgresql
```

请使用以下命令检查 PostgreSQL 版本：

```
psql --version
```

（仅限 RHEL 7）使用 **psql** 命令行实用程序确认数据目录已设置：

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

#### 步骤 **2**：准备虚拟机管理程序

在支持的虚拟机管理程序上将 Linux VDA 当作虚拟机运行时，需要作出一些更改。根据使用的虚拟机管理程序平台作出以下更改。如果正在裸机硬件上运行 Linux 计算机，则无需作出任何更改。

### 修复 **Citrix Hypervisor** 上的时间同步问题

启用了 Citrix Hypervisor 时间同步功能时，在每个半虚拟化 Linux VM 中，您会发现 NTP 和 Citrix Hypervisor 都尝试管理系统时钟。为避免时钟与其他服务器不同步，请确保每个 Linux 客户机中的系统时钟都与 NTP 同步。这种情况要求禁用主机时间同步。无需在 HVM 模式下进行任何更改。

在某些 Linux 发行版中，如果正在运行半虚拟化 Linux 内核，并安装了 Citrix VM Tools，您可以检查 Citrix Hypervisor 时间同步功能是否存在，以及是否已在 Linux VM 中启用：

```
su -

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

此命令返回 0 或 1：

- 0 - 时间同步功能已启用，且必须禁用。
- 1 - 时间同步功能已禁用，无需采取任何操作。

如果 /proc/sys/xen/independent_wallclock 文件不存在，则不需要执行以下步骤。

如果已启用，请通过向该文件写入 1 来禁用时间同步功能：

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

要使此更改成为永久更改，并在重新启动后仍然有效，请编辑 **/etc/sysctl.conf** 文件并添加以下行：

```
xen.independent_wallclock = 1
```

要验证这些更改，请重新启动系统：

```
su -

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

此命令返回值 1。

### 修复 **Microsoft Hyper-V** 上的时间同步问题

安装了 Hyper-V Linux 集成服务的 Linux VM 可应用 Hyper-V 时间同步功能来使用主机操作系统的时间。为确保系统时钟始终精确可靠，必须一同启用此功能与 NTP 服务。

从管理操作系统中：

1. 打开 Hyper-V 管理器控制台。
2. 对于 Linux VM 的设置，请选择 **Integration Services**（集成服务）。
3. 确保已选择 **Time synchronization**（时间同步）。

> 注意：
>
> 此方法与 VMware 和 Citrix Hypervisor 不同，这两种产品会禁用主机时间同步功能，以免与 NTP 发生冲突。Hyper-V 时间同步可以与 NTP 时间同步共存并互补。

#### 修复 **ESX** 和 **ESXi** 上的时间同步问题

启用了 VMware 时间同步功能时，在每个半虚拟化 Linux VM 中，您会发现 NTP 和虚拟机管理程序都尝试同步系统时钟。为避免时钟与其他服务器不同步，请确保每个 Linux 客户机中的系统时钟都与 NTP 同步。这种情况要求禁用主机时间同步。

如果正在运行安装了 VMware Tools 的半虚拟化 Linux 内核：

1. 打开 vSphere Client。
2. 编辑 Linux VM 设置。
3. 在 **Virtual Machine Properties**（虚拟机属性）对话框中，打开 **Options**（选项）选项卡。
4. 选择 **VMware Tools**。
5. 在 **Advanced**（高级）框中，取消选中 **Synchronize guest time with host**（与主机同步客户机时间）。

### 步骤 **3**：向 **Windows** 域中添加 **Linux** 虚拟机 **(VM)**

Linux VDA 支持多种向 Active Directory (AD) 域添加 Linux 计算机的方法：

- Samba Winbind
- Quest Authentication Services
- Centrify DirectControl
- SSSD
- PBIS（仅与 RHEL 7 兼容）

根据所选的方法，按说明执行操作。

> 注意：
>
> 为 Linux VDA 中的本地帐户和 AD 中的帐户使用相同的用户名时，会话启动可能会失败。

#### Samba Winbind

安装或更新所需软件包：

对于 RHEL 8/CentOS 8：

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

对于 RHEL 7/CentOS 7：

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

在计算机启动时启用要启动的 **Winbind** 守护程序　Winbind 守护程序必须配置为在计算机启动时启动：

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

配置 **Winbind** 身份验证　通过使用 Winbind 将计算机配置为执行 Kerberos 身份验证：

1. 请运行以下命令：。

   对于 RHEL 8：

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

   对于 RHEL 7：

   ```
   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
   ```

   其中，**REALM** 是大写的 Kerberos 领域名称，而 **domain** 是域的 NetBIOS 名称。

   如果需要通过 DNS 查找 KDC 服务器和领域名称，请将以下两个选项添加至前面的命令：

   `--enablekrb5kdcdns --enablekrb5realmdns`

   请忽略 `authconfig` 命令返回的有关 `winbind` 服务无法启动的任何错误。`authconfig` 尝试在计算机尚未加入域的情况下启动 `winbind` 服务时，可能会出现这些错误。

2. 打开 **/etc/samba/smb.conf** 并将以下条目添加到 [Global] 部分下方，但要放在 `authconfig` 工具生成的部分后面：

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

3. （仅限 RHEL 8）打开 **/etc/krb5.conf** 并在 [libdefaults]、[realms] 和 [domain_realm] 部分下添加条目：

   在 [libdefaults] 部分下：

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

   在 [realms] 部分下：

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

在 [domain_realm] 部分下：

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

Linux VDA 需要使用系统 keytab 文件 /etc/krb5.keytab 以执行身份验证并向 Delivery Controller 注册。计算机首次加入域后，前面的 kerberos method 设置将强制 Winbind 创建系统 keytab 文件。

加入 **Windows** 域　您的域控制器必须可访问，并且您必须具有有权将计算机添加到域的 Active Directory 用户帐户：

RHEL 8：

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

RHEL 7：

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

**REALM** 是大写的 Kerberos 领域名称，**user** 是有权将计算机添加到域的域用户。

为 **Winbind** 配置 **PAM**　默认情况下，Winbind PAM 模块 (pam_winbind) 的配置不启用 Kerberos 票据缓存和主目录的创建。打开 **/etc/security/pam_winbind.conf**，并在 [Global] 部分下添加或更改以下条目：

```
krb5_auth = yes
krb5_ccache_type = FILE
mkhomedir = yes
```

确保删除每个设置中的任何前置分号。这些更改要求重新启动 Winbind 守护程序：

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

> 提示：
>
> 仅当计算机加入域后，`winbind` 守护程序才会始终保持运行。

打开 **/etc/krb5.conf** 并将 [libdefaults] 部分下方的以下设置从 KEYRING 更改为 FILE 类型：

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

验证域成员身份　Delivery Controller 要求所有 VDA 计算机（Windows 和 Linux VDA）都要在 Active Directory 中有一个计算机对象。

运行 Samba 的 **net ads** 命令来验证计算机是否已加入域：

```
sudo net ads testjoin
```

运行以下命令验证额外的域和计算机对象信息：

```
sudo net ads info
```

验证 **Kerberos** 配置　为了确保 Kerberos 已正确配置为可与 Linux VDA 配合使用，请验证系统 keytab 文件是否已创建并包含有效密钥：

```
sudo klist -ke
```

此命令显示各种主体名称与密码套件组合可用的密钥列表。运行 Kerberos `kinit` 命令，使用这些密钥向域控制器验证计算机的身份：

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

计算机和领域名称必须指定为大写。美元符号 ($) 必须使用反斜杠 (\) 进行转义，以免发生 shell 替换。在某些环境中，DNS 域名与 Kerberos 领域名称不同。请确保使用领域名称。如果此命令成功运行，则不会显示任何输出。

请使用以下命令验证计算机帐户的 TGT 票据已缓存：

```
sudo klist
```

请使用以下命令检查计算机的帐户详细信息：

```
sudo net ads status
```

验证用户身份验证　使用 **wbinfo** 工具验证是否可向域验证域用户的身份：

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

这里指定的域为 AD 域名，而不是 Kerberos 领域名称。对于 bash shell，必须使用另一个反斜杠对反斜杠 (\) 字符进行转义。此命令返回一条成功或失败消息。

要验证 Winbind PAM 模块是否已正确配置，请使用以前未使用的域用户帐户登录 Linux VDA。

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

验证 Kerberos 凭据缓存中的票据是否有效且未过期：

```
klist
```

退出会话。

```
exit
```

直接登录 Gnome 或 KDE 控制台也可以执行类似的测试。在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### Quest Authentication Services

在域控制器上配置 **Quest**　假定您已在 Active Directory 域控制器上安装并配置了 Quest 软件，而且已获得管理权限，有权在 Active Directory 中创建计算机对象。

允许域用户登录 **Linux VDA** 计算机　为了让域用户能够在 Linux VDA 计算机上建立 HDX 会话：

1. 在 Active Directory 用户和计算机管理控制台中，为该用户帐户打开 Active Directory 用户属性。
2. 选择 **Unix Account**（Unix 帐户）选项卡。
3. 选中 **Unix-enabled**（已启用 Unix）。
4. 将 **Primary GID Number**（首选 GID 编号）设置为实际域用户组的组 ID。

> 注意：
>
> 这些说明相当于设置域用户，以便他们可以使用控制台、RDP、SSH 或任何其他远程协议进行登录。

#### 在 **Linux VDA** 上配置 **Quest**

**SELinux** 策略强制实施解决方法　默认 RHEL 环境会强制实施 SELinux。此强制功能会影响 Quest 使用的 Unix 域套接字 IPC 机制，并阻止域用户登录。

解决此问题的最便捷的方法是禁用 SELinux。以 root 用户身份，编辑 **/etc/selinux/config** 并更改 **SELinux** 设置：

```
SELINUX=permissive
```

此更改要求重新启动计算机：

```
reboot
```

> 重要：
>
> 请谨慎使用此设置。禁用后重新启用 SELinux 策略强制实施会导致完全锁定，即便是对 root 用户和其他本地用户也是如此。

配置 **VAS** 守护程序　必须启用并断开自动续订 Kerberos 票据功能。必须禁用身份验证（脱机登录）功能。

```
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
```

此命令将续订间隔设为 9 小时（32400 秒），即比默认的 10 小时票据生命周期短 1 小时。请在票据生命周期较短的系统上设置较低的值。

配置 **PAM** 和 **NSS**　要启用通过 HDX 进行的域用户登录以及其他服务（例如 su、ssh 和 RDP），请运行以下命令以手动配置 PAM 和 NSS：

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

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

加入 **Windows** 域　使用 Quest **vastool** 命令将 Linux 计算机加入到 Active Directory 域中：

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

user 为有权将计算机加入 Active Directory 域的任何域用户。**domain-name** 为域的 DNS 名称，例如 example.com。

验证域成员身份　Delivery Controller 要求所有 VDA 计算机（Windows 和 Linux VDA）都要在 Active Directory 中有一个计算机对象。要验证加入了 Quest 的 Linux 计算机是否位于域中，请执行以下操作：

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

如果计算机已加入域，此命令会返回域名。如果计算机未加入任何域，则会显示以下错误：

```
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
```

验证用户身份验证　要验证 Quest 是否能够通过 PAM 对域用户进行身份验证，请使用以前未使用的域用户帐户登录 Linux VDA。

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

验证是否为 **id -u** 命令返回的 UID 创建了对应的 Kerberos 凭据缓存文件：

```
ls /tmp/krb5cc_uid
```

验证 Kerberos 凭据缓存中的票据是否有效且未过期：

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

退出会话。

```
exit
```

直接登录 Gnome 或 KDE 控制台也可以执行类似的测试。在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### Centrify DirectControl

加入 **Windows** 域　安装 Centrify DirectControl Agent 后，请使用 Centrify `adjoin` 命令将 Linux 计算机加入 Active Directory 域：

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

user 参数为有权将计算机加入 Active Directory 域的任何 Active Directory 域用户。**domain-name** 是将 Linux 计算机加入到的域的名称。

验证域成员身份　Delivery Controller 要求所有 VDA 计算机（Windows 和 Linux VDA）都要在 Active Directory 中有一个计算机对象。要验证加入了 Centrify 的 Linux 计算机是否位于域中，请执行以下操作：

```
su -
adinfo
```

验证 Joined to domain 值是否有效以及 CentrifyDC mode 是否返回了 connected。如果模式仍然卡在正在启动状态，则表明 Centrify 客户端遇到了服务器连接或身份验证问题。

请使用以下命令可获得更全面的系统和诊断信息：

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

测试与各种 Active Directory 和 Kerberos 服务的连接。

```
adinfo --test
```

在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### SSSD

如果您使用的是 SSSD，请按照此部分中的说明进行操作。此部分包含有关如何将 Linux VDA 计算机加入 Windows 域的说明以及如何配置 Kerberos 身份验证的指导。

要在 RHEL 和 CentOS 上设置 SSSD，请执行以下操作：

1. 加入域并创建主机 keytab
2. 设置 SSSD
3. 启用 SSSD
4. 验证 Kerberos 配置
5. 验证用户身份验证

加入域并创建主机 **keytab**　SSSD 并不提供用于加入域和管理系统 keytab 文件的 Active Directory 客户端功能。可以改为使用 `adcli`、`realmd` 或 `Samba`。

本部分内容分别介绍 RHEL 7 和 RHEL 8 的 `Samba` 和 `adcli` 方法。对于 `realmd`，请参阅 RHEL 或 CentOS 文档。必须在配置 SSSD 之前执行这些步骤。

- **Samba (RHEL 7)**：

  安装或更新所需软件包：

  ```
  sudo yum -y install krb5-workstation authconfig oddjob-mkhomedir samba-common-tools
  ```

  在正确配置了以下文件的 Linux 客户端上：

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

  将计算机配置为进行 Samba 和 Kerberos 身份验证：

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

  其中，**REALM** 是大写的 Kerberos 领域名称，**domain** 是 Active Directory 域的简短 NetBIOS 名称。

  > 注意：
  >
  > 本文中的设置适用于单域、单林模型。根据您的 AD 基础结构配置 Kerberos。

  如果需要通过 DNS 查找 KDC 服务器和领域名称，请将以下两个选项添加至前面的命令：

  `--enablekrb5kdcdns --enablekrb5realmdns`

  打开 **/etc/samba/smb.conf** 并将以下条目添加到 **[Global]** 部分下方，但要放在 **authconfig** 工具生成的部分后面：

  ```
  kerberos method = secrets and keytab
  winbind offline logon = no
  ```

  加入 Windows 域。请确保域控制器可访问，并且您具有有权将计算机添加到域的 Active Directory 用户帐户：

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

  **REALM** 是大写的 Kerberos 领域名称，**user** 是有权将计算机添加到域的域用户。

- **Adcli (RHEL 8)**：

  安装或更新所需软件包：

  ```
  sudo yum -y install samba-common samba-common-tools krb5-workstation authconfig oddjob-mkhomedir realmd oddjob authselect
  ```

将计算机配置为进行 Samba 和 Kerberos 身份验证：

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

打开 **/etc/krb5.conf** 并在 [realms] 和 [domain_realm] 部分下添加条目。

在 [realms] 部分下：

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

在 [domain_realm] 部分下：

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

加入 Windows 域。请确保域控制器可访问，并且您具有有权将计算机添加到域的 Active Directory 用户帐户：

```
sudo realm join REALM -U user
```

**REALM** 是大写的 Kerberos 领域名称，**user** 是有权将计算机添加到域的域用户。

设置 **SSSD**　设置 SSSD 的步骤如下：

- 通过运行 `sudo yum -y install sssd` 命令在 Linux VDA 上安装 **sssd-ad** 软件包。
- 对各种文件（例如 sssd.conf）进行配置更改。
- 启动 **sssd** 服务。

RHEL 7 的 **sssd.conf** 配置示例（可以根据需要添加额外的选项）：

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

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

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

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

# Kerberos settings
```

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

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

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

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

将 **ad.example.com**、**server.ad.example.com** 替换为相应的值。有关详细信息，请参阅 sssd-ad(5) - Linux 手册页。

(仅限 RHEL 8)

打开 **/etc/sssd/sssd.conf** 并在 [domain/ad.example.com] 部分下添加以下条目：

```
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
```

对 sssd.conf 设置文件所有权和权限：

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

启用 **SSSD** **RHEL 8**：

请运行以下命令以启用 SSSD：

```
sudo systemctl restart sssd
sudo systemctl enable sssd.service
sudo chkconfig sssd on
```

**RHEL 7/CentOS 7**：

使用 `authconfig` 启用 SSSD。安装 **oddjob mkhomedir** 以确保主目录创建与 SELinux 兼容：

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

sudo service sssd start

```

```
sudo chkconfig sssd on
```

验证 **Kerberos** 配置　验证系统 **keytab** 文件是否已创建并且包含有效密钥：

```
sudo klist -ke
```

此命令显示各种主体名称与密码套件组合可用的密钥列表。运行 Kerberos **kinit** 命令，以使用这些密钥向域控制器验证计算机的身份：

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

计算机和领域名称必须指定为大写。美元符号 (**$**) 必须使用反斜杠 (****) 进行转义，以免发生 shell 替换。在某些环境中，DNS 域名与 Kerberos 领域名称不同。请确保使用领域名称。如果此命令成功运行，则不会显示任何输出。

请使用以下命令验证计算机帐户的 TGT 票据已缓存：

```
sudo klist
```

验证用户身份验证　使用 **getent** 命令确认支持的登录格式以及 NSS 是否工作：

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

**DOMAIN** 参数指示简短形式的域名。如果需要使用另一种登录格式，请先使用 **getent** 命令进行验证。

支持的登录格式如下：

- 低级别登录名称：`DOMAIN\username`
- UPN：`username@domain.com`
- NetBIOS 前缀格式：`username@DOMAIN`

要验证 SSSD PAM 模块是否已正确配置，请使用以前未使用的域用户帐户登录 Linux VDA。

```
sudo ssh localhost -l DOMAIN\username

id -u
```

验证是否为以下命令返回的 **uid** 创建了对应的 Kerberos 凭据缓存文件：

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

验证用户的 Kerberos 凭据缓存中的票据是否有效且未过期。

```
klist
```

在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### PBIS

下载所需的 **PBIS** 软件包　例如：

```
wget https://github.com/BeyondTrust/pbis-open/releases/download/8.8.0/pbis-open-8.8.0.506.linux.x86_64.rpm.sh
```

使 **PBIS** 安装脚本可执行　例如：

```
chmod +x pbis-open-8.8.0.506.linux.x86_64.rpm.sh
```

运行 **PBIS** 安装脚本　例如：

```
sh pbis-open-8.8.0.506.linux.x86_64.rpm.sh
```

加入 **Windows** 域　您的域控制器必须可访问，并且您必须具有有权将计算机添加到域的 Active Directory 用户帐户：

```
/opt/pbis/bin/domainjoin-cli join domain-name user
```

**user** 为有权将计算机添加到 Active Directory 域的域用户。**domain-name** 为域的 DNS 名称，例如 example.com。

注意：要将 Bash 设置为默认 shell，请运行 **/opt/pbis/bin/config LoginShellTemplate/bin/bash** 命令。

验证域成员身份　Delivery Controller 要求所有 VDA 计算机（Windows 和 Linux VDA）都要在 Active Directory 中有一个计算机对象。要验证加入了 PBIS 的 Linux 计算机是否位于域中，请执行以下操作：

```
/opt/pbis/bin/domainjoin-cli query
```

如果计算机已加入某个域，此命令将返回有关当前加入的 AD 域和 OU 的信息。否则，仅显示主机名。

验证用户身份验证　要验证 PBIS 是否能够通过 PAM 对域用户进行身份验证，请使用以前未使用的域用户帐户登录 Linux VDA。

```
ssh localhost -l domain\user

id -u
```

验证是否为 **id -u** 命令返回的 UID 创建了对应的 Kerberos 凭据缓存文件：

```
ls /tmp/krb5cc_uid
```

退出会话。

```
exit
```

在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### 步骤 4：安装 Linux VDA

可以执行全新安装或从先前的两个版本和 LTSR 版本升级现有安装。

执行全新安装

1. （可选）卸载旧版本

   如果安装了除先前的两个版本和 LTSR 版本之外的早期版本，请在安装新版本之前将其卸载。

   a) 停止 Linux VDA 服务：

   ```
   sudo /sbin/service ctxvda stop

   sudo /sbin/service ctxhdx stop
   ```

   > 注意：
   >
   > 在停止 `ctxvda` 和 `ctxhdx` 服务之前，请运行 **service ctxmonitorservice stop** 命令以停止监视服务守护程序。否则，监视服务守护程序将重新启动您停止的服务。

   b) 卸载软件包：

   ```
   sudo rpm -e XenDesktopVDA
   ```

   > 注意：
   >
   > 要运行命令，需要提供完整路径；或者，也可以将 **/opt/Citrix/VDA/sbin** 和 **/opt/Citrix/VDA/bin** 添加到系统路径。

2. 下载 Linux VDA 软件包

   转至 Citrix Virtual Apps and Desktops 下载页面。展开相应版本的 Citrix Virtual Apps and Desktops，然后单击组件以下载与 Linux 发行版匹配的 Linux VDA 包。

3. 安装 Linux VDA

   - 使用 Yum 安装 Linux VDA 软件：

     对于 **RHEL 8/CentOS 8**：

     ```
     sudo yum install -y XenDesktopVDA-<version>.el8_x.x86_64.rpm
     ```

对于 **RHEL 7/CentOS 7**:

```
sudo yum install -y XenDesktopVDA-<version>.el7_x.x86_64.rpm
```

- 使用 RPM 软件包管理器安装 Linux VDA 软件。在此之前，您必须解决以下依赖项:

对于 **RHEL 8/CentOS 8**:

```
sudo rpm -i XenDesktopVDA-<version>.el8_x.x86_64.rpm
```

对于 **RHEL 7/CentOS 7**:

```
sudo rpm -i XenDesktopVDA-<version>.el7_x.x86_64.rpm
```

**RHEL 8.2/CentOS 8.2** 的 **RPM** 依赖项列表:

```
postgresql-jdbc >= 42.2.3

postgresql-server >= 10.6

java-1.8.0-openjdk >= 1.8.0

icoutils >= 0.32

firewalld  >= 0.8.0

policycoreutils-python-utils >= 2.9

python3-policycoreutils >= 2.9

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

motif >= 2.3.4

pam >= 1.3.1

```

```
util-linux >= 2.32.1

util-linux-user  >= 2.32.1

xorg-x11-utils  >= 7.5

bash >= 4.4

findutils >= 4.6

gawk >= 4.2

sed >= 4.5

cups >= 2.2

foomatic-filters >= 4.0.9

cups-filters >= 1.20.0

ghostscript  >= 9.25

libxml2 >= 2.9

libmspack >= 0.7
```

**RHEL 7/CentOS 7** 的 **RPM** 依赖项列表：

```
postgresql-server >= 9.2

postgresql-jdbc >= 9.2

java-1.8.0-openjdk >= 1.8.0

ImageMagick >= 6.7.8.9

firewalld >= 0.3.9

policycoreutils-python >= 2.0.83

dbus >= 1.6.12

dbus-x11 >= 1.6.12

xorg-x11-server-utils >= 7.7

xorg-x11-xinit >= 1.3.2

libXpm >= 3.5.10

libXrandr >= 1.4.1

libXtst >= 1.2.2
```

```

motif >= 2.3.4

pam >= 1.1.8

util-linux >= 2.23.2

bash >= 4.2

findutils >= 4.5

gawk >= 4.0

sed >= 4.2

cups >= 1.6.0

foomatic-filters >= 4.0.9

openldap >= 2.4

cyrus-sasl >= 2.1

cyrus-sasl-gssapi >= 2.1

libxml2 >= 2.9

python-requests >= 2.6.0

gperftools-libs >= 2.4

rpmlib(FileDigests) <= 4.6.0-1

rpmlib(PayloadFilesHavePrefix) <= 4.0-1

pmlib(CompressedFileNames) <= 3.0.4-1

rpmlib(PayloadIsXz) <= 5.2-1
```

> 注意：
>
> 有关此版本的 Linux VDA 支持的 Linux 发行版和 Xorg 版本的列表，请参阅系统要求。

> 注意：
>
> 在 RHEL 7.x 上安装 Linux VDA 后，运行 `sudo yum install -y python-websockify x11vnc` 命令。目的是手动安装 `python-websockify` 和 `x11vnc` 以使用会话重影功能。有关详细信息，请参阅重影会话。

升级现有安装

可以从先前的两个版本和 LTSR 版本升级现有安装。

- 要使用 Yum 升级您的软件，请执行以下操作：

  对于 **RHEL 7/CentOS 7**：

```
sudo yum install -y XenDesktopVDA-<version>.el7_x.x86_64.rpm
```

- 要使用 RPM 软件包管理器升级您的软件，请执行以下操作：

  对于 **RHEL 7/CentOS 7**：

```
sudo rpm -U XenDesktopVDA-<version>.el7_x.x86_64.rpm
```

> 注意：
>
> 如果您使用的是 RHEL 7，请务必在运行上述升级命令后完成以下步骤：
>
> 1. 运行 `/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\VirtualDesktopAgent"-t "REG_SZ"-v "DotNetRuntimePath"-d "/opt/rh/rh-dotnet31/root/usr/bin/"--force` 以设置正确的.NET Runtime 路径。
> 2. 重新启动 `ctxvda` 服务。
>
> 重要：
>
> 升级软件后重新启动 Linux VDA 计算机。

### 步骤 5：安装 NVIDIA GRID 驱动程序

启用 HDX 3D Pro 需要执行额外的安装步骤，以在虚拟机管理程序和 VDA 计算机上安装必备的图形驱动程序。

配置以下设置：

1. Citrix Hypervisor
2. VMware ESX

根据所选的虚拟机管理程序，按以下说明执行操作。

**Citrix Hypervisor**：

此部分将详细介绍如何在 Citrix Hypervisor 上安装和配置 NVIDIA GRID 驱动程序。

**VMware ESX**：

请按照本指南中包含的信息进行操作，为 VMware ESX 安装和配置 NVIDIA GRID 驱动程序。

**VDA** 计算机：

按照这些步骤为每个 Linux VM 客户机安装和配置驱动程序：

1. 开始前，请确保 Linux VM 已关闭。
2. 在 XenCenter 中，将处于 GPU 直通模式的 GPU 添加至 VM。
3. 启动 RHEL VM。

要准备计算机以使用 NVIDIA GRID 驱动程序，请运行以下命令：

```
yum install gcc

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

systemctl set-default multi-user.target
```

请按照 Red Hat Enterprise Linux 文档中的步骤安装 NVIDIA GRID 驱动程序。

> 注意：
>
> 安装 GPU 驱动程序期间，为每个问题选择默认答案 (no)。
>
> 重要：
>
> 在启用 GPU 直通后，无法再通过 XenCenter 访问 Linux VM。使用 SSH 进行连接。

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

为显卡设置正确的配置：

```
etc/X11/ctx-nvidia.sh
```

要利用高分辨率和多监视器功能，您需要有效的 NVIDIA 许可证。要申请许可证，请按照 “GRID Licensing Guide.pdf - DU-07757-001 September 2015” 产品文档执行操作。

### 步骤 6：配置 Linux VDA

安装软件包后，必须运行 ctxsetup.sh 脚本来配置 Linux VDA。执行任何更改之前，该脚本都会验证环境，确保所有依赖项都已安装。如有必要，可以随时重新运行该脚本以更改设置。

可以按照提示手动运行脚本，也可以采用预先配置的响应自动运行脚本。继续操作前，请查看该脚本的帮助信息：

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

#### 提示配置

运行会提示各种问题的手动配置：

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

#### 自动配置

自动安装时，通过环境变量提供设置脚本所需的选项。如果所需的所有变量都存在，脚本不会提示您提供任何信息。

支持的环境变量包括：

- **CTX_XDL_SUPPORT_DDC_AS_CNAME=Y | N** - Linux VDA 支持使用 DNS CNAME 记录指定 Delivery Controller 名称。默认情况下设置为 N。
- **CTX_XDL_DDC_LIST=’ list-ddc-fqdns’** –Linux VDA 要求提供由空格分隔的 Delivery Controller 完全限定域名 (FQDN) 列表以用于向 Delivery Controller 注册。必须至少指定一个 FQDN 或 CNAME 别名。
- **CTX_XDL_VDA_PORT=port-number** –Linux VDA 通过 TCP/IP 端口（默认为端口 80）与 Delivery Controller 通信。
- **CTX_XDL_REGISTER_SERVICE=Y | N** - 在启动计算机后启动 Linux Virtual Desktop 服务。默认情况下，该值设置为 Y。
- **CTX_XDL_ADD_FIREWALL_RULES=Y | N** - Linux Virtual Desktop 服务要求允许传入网络连接通过系统防火墙。您可以在系统防火墙中自动为 Linux Virtual Desktop 打开所需端口（默认为端口 80 和 1494）。默认情况下设置为 Y。
- **CTX_XDL_AD_INTEGRATION=1 | 2 | 3 | 4 | 5** –Linux VDA 要求使用 Kerberos 配置设置向 Delivery Controller 进行身份验证。Kerberos 配置根据系统上已安装和已配置的 Active Directory 集成工具确定。指定要使用且受支持的 Active Directory 集成方法：
  - 1 –Samba Winbind
  - 2 –Quest Authentication Services
  - 3 –Centrify DirectControl
  - 4 –SSSD
  - 5 –PBIS

- **CTX_XDL_HDX_3D_PRO=Y | N** - Linux VDA 支持 HDX 3D Pro，这是一组 GPU 加速技术，旨在优化富图形应用程序的虚拟化水平。如果选择了 HDX 3D Pro，则要为 VDI 桌面（单会话）模式配置 VDA -（即 CTX_XDL_VDI_MODE=Y）。
- **CTX_XDL_VDI_MODE=Y | N** - 将计算机配置为专用桌面交付模型 (VDI) 还是托管共享桌面交付模型。对于 HDX 3D Pro 环境，将此变量设置为 Y。默认情况下，此变量设置为 N。
- **CTX_XDL_SITE_NAME=dns-name** - Linux VDA 通过 DNS 发现 LDAP 服务器。要将 DNS 搜索结果限制为本地站点，应指定 DNS 站点名称。默认情况下，此变量设置为 **<none>**。
- **CTX_XDL_LDAP_LIST='list-ldap-servers'** –Linux VDA 查询 DNS 来发现 LDAP 服务器。如果 DNS 无法提供 LDAP 服务记录，您可以提供以空格分隔的 LDAP FQDN（带有 LDAP 端口）列表。例如 ad1.mycompany.com:389。默认情况下，此变量设置为 **<none>**。
- **CTX_XDL_SEARCH_BASE=search-base-set** - Linux VDA 通过设置为 Active Directory 域根的搜索库来查询 LDAP（例如，DC=mycompany,DC=com）。为提高搜索性能，可以指定搜索基础（例如 OU=VDI,DC=mycompany,DC=com）。默认情况下，此变量设置为 **<none>**。
- **CTX_XDL_FAS_LIST='list-fas-servers'** –联合身份验证服务 (FAS) 服务器是通过 AD 组策略配置的。Linux VDA 不支持 AD 组策略，但您可以改为提供以分号分隔的 FAS 服务器的列表。顺序必须与在 AD 组策略中配置的顺序相同。如果删除了任何服务器地址，请使用 **<none>** 文本字符串填充其空白，并且不要修改服务器地址的顺序。
- **CTX_XDL_DOTNET_ RUNTIME_PATH=path-to-install-dotnet-runtime** – 安装.NET Core Runtime 3.1 以支持新的 Broker 代理服务 (`ctxvda`) 的路径。默认路径为 /usr/bin。
- **CTX_XDL_START_SERVICE=Y | N** –在完成 Linux VDA 配置后，是否启动 Linux VDA 服务。默认情况下设置为 Y。
- **CTX_XDL_TELEMETRY_SOCKET_PORT** - 用于侦听 Citrix Scout 的套接字端口。默认端口为 7503。
- **CTX_XDL_TELEMETRY_PORT** - 用于与 Citrix Scout 通信的端口。默认端口为 7502。

设置环境变量并运行配置脚本：

```
export CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N

export CTX_XDL_DDC_LIST='list-ddc-fqdns'

export CTX_XDL_VDA_PORT=port-number

export CTX_XDL_REGISTER_SERVICE=Y|N

export CTX_XDL_ADD_FIREWALL_RULES=Y|N

export CTX_XDL_AD_INTEGRATION=1|2|3|4|5

export CTX_XDL_HDX_3D_PRO=Y|N

export CTX_XDL_VDI_MODE=Y|N

export CTX_XDL_SITE_NAME=dns-site-name | '<none>'

export CTX_XDL_LDAP_LIST='list-ldap-servers' | '<none>'

```

```
export CTX_XDL_SEARCH_BASE=search-base-set | '<none>'

export CTX_XDL_FAS_LIST='list-fas-servers' | '<none>'

export CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime

export CTX_XDL_TELEMETRY_SOCKET_PORT=port-number

export CTX_XDL_TELEMETRY_PORT=port-number

export CTX_XDL_START_SERVICE=Y|N

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

运行 sudo 命令时，键入 **-E** 选项以将现有环境变量传递给其创建的新 shell。我们建议您使用前面的命令并加上 **#!/bin/bash** 作为第一行来创建 shell 脚本文件。

另外，您可以使用单个命令指定所有参数：

```
sudo CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N \

CTX_XDL_DDC_LIST='list-ddc-fqdns' \

CTX_XDL_VDA_PORT=port-number \

CTX_XDL_REGISTER_SERVICE=Y|N \

CTX_XDL_ADD_FIREWALL_RULES=Y|N \

CTX_XDL_AD_INTEGRATION=1|2|3|4|5 \

CTX_XDL_HDX_3D_PRO=Y|N \

CTX_XDL_VDI_MODE=Y|N \

CTX_XDL_SITE_NAME=dns-name \

CTX_XDL_LDAP_LIST='list-ldap-servers' \

CTX_XDL_SEARCH_BASE=search-base-set \

CTX_XDL_FAS_LIST='list-fas-servers' \

CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime \

CTX_XDL_TELEMETRY_SOCKET_PORT=port-number \

CTX_XDL_TELEMETRY_PORT=port-number \

CTX_XDL_START_SERVICE=Y|N \

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

删除配置更改

在某些情形下，您可能需要删除 **ctxsetup.sh** 脚本对配置所做的更改，但不卸载 Linux VDA 软件包。

继续操作前，请查看此脚本的帮助信息：

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

删除配置更改：

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

> 重要：
>
> 此脚本会从数据库删除所有配置数据，从而使 Linux VDA 无法使用。

配置日志

**ctxsetup.sh** 和 **ctxcleanup.sh** 脚本会在控制台上显示错误，并将其他信息写入配置日志文件 **/tmp/xdl.configure.log**。

重新启动 Linux VDA 服务，确保更改生效。

### 步骤 7：运行 Linux VDA

使用 **ctxsetup.sh** 脚本配置 Linux VDA 后，可以运行以下命令来控制 Linux VDA。

启动 **Linux VDA**：

启动 Linux VDA 服务：

```
sudo /sbin/service ctxhdx start

sudo /sbin/service ctxvda start
```

停止 **Linux VDA**：

要停止 Linux VDA 服务，请执行以下操作：

```
sudo /sbin/service ctxvda stop

sudo /sbin/service ctxhdx stop
```

> 注意：
>
> 在停止 `ctxvda` 和 `ctxhdx` 服务之前，请运行 `service ctxmonitorservice stop` 命令以停止监视服务守护程序。否则，监视服务守护程序将重新启动您停止的服务。

重新启动 **Linux VDA**：

重新启动 Linux VDA 服务：

```
sudo /sbin/service ctxvda stop

sudo /sbin/service ctxhdx restart

sudo /sbin/service ctxvda start
```

检查 **Linux VDA** 的状态：

要检查 Linux VDA 服务的运行状态，请执行以下操作：

```
sudo /sbin/service ctxvda status

sudo /sbin/service ctxhdx status
```

### 步骤 8：在 Citrix Virtual Apps 或 Citrix Virtual Desktops 中创建计算机目录

创建计算机目录和添加 Linux VDA 计算机的过程与传统的 Windows VDA 方法类似。有关如何完成这些任务的更加详细的说明，请参阅创建计算机目录和管理计算机目录。

创建包含 Linux VDA 计算机的计算机目录时会面临一些限制，使得该过程不同于为 Windows VDA 计算机创建计算机目录：

- 对于操作系统，请选择：
  - 多会话操作系统选项（对于托管共享桌面交付模型）。
  - 单会话操作系统选项（适用于 VDI 专用桌面交付模型）。
- 请勿在同一个计算机目录中混合使用 Linux 和 Windows VDA 计算机。

> 注意：
>
> 早期版本的 Citrix Studio 不支持“Linux 操作系统”的概念。但是，选择 **Windows Server** 操作系统或服务器操作系统选项等同于使用托管共享桌面交付模型。选择 **Windows** 桌面操作系统或桌面操作系统选项等同于使用每计算机一个用户交付模型。
>
> 提示：
>
> 如果删除计算机后将其重新加入 Active Directory 域，则必须删除计算机，然后将其重新添加到计算机目录。

### 步骤 9：在 Citrix Virtual Apps 或 Citrix Virtual Desktops 中创建交付组

创建交付组和添加包含 Linux VDA 计算机的计算机目录的过程与 Windows VDA 计算机几乎相同。有关如何完成这些任务的更加详细的说明，请参阅创建交付组。

创建含有 Linux VDA 计算机目录的交付组时会面临以下限制：

- 确保所选的 AD 用户和组已正确配置，可以登录到 Linux VDA 计算机。
- 请勿允许未经身份验证的（匿名）用户登录。
- 请勿在交付组中混入含有 Windows 计算机的计算机目录。

> 重要：
>
> Linux VDA 1.4 及更高版本支持发布应用程序。但是，Linux VDA 不支持将桌面和应用程序交付给相同的计算机。

有关如何创建计算机目录和交付组的信息，请参阅 Citrix Virtual Apps and Desktops 7 2103。

## 安装 Linux Virtual Delivery Agent for SUSE

June 21, 2022

可以选择按照本文中的步骤进行手动安装，也可以使用轻松安装进行自动安装和配置。轻松安装省时又省力，与手动安装相比，更不易于出错。

> 注意：
>
> 请仅对全新安装使用轻松安装功能。请勿使用轻松安装更新现有安装。

### 步骤 1：准备安装

#### 步骤 1a：启动 YaST 工具

SUSE Linux Enterprise YaST 工具用于对操作系统执行方方面面的配置。

启动基于文本的 YaST 工具：

```
su -

yast
```

或者，启动基于 UI 的 YaST 工具：

```
su -

yast2 &
```

#### 步骤 1b：配置网络连接

以下各部分介绍了如何配置 Linux VDA 使用的各种网络设置和服务。网络配置通过 YaST 工具执行，而不得使用其他方法，例如 Network Manager。以下说明介绍的是使用基于 UI 的 YaST 工具的情形。也可以使用基于文本的 YaST 工具，但导航方法稍有不同，对此本文未作介绍。

配置主机名和 **DNS**

1. 打开 YaST 网络设置。
2. 仅限 SLED 12：在 **Global Options**（全局选项）选项卡上，将 **Network Setup Method**（网络设置方法）更改为 **Wicked Service**（Wicked 服务）。
3. 打开 **Hostname/DNS**（主机名/DNS）选项卡。
4. 取消选中 **Change hostname via DHCP**（通过 DHCP 更改主机名）。
5. 选中 **Assign Hostname to Loopback IP**（向环回 IP 分配主机名）。
6. 编辑以下内容，以反映所作的网络设置：
   - 主机名 - 添加计算机的 DNS 主机名。
   - 域名 - 添加计算机的 DNS 域名。
   - 名称服务器 - 添加 DNS 服务器的 IP 地址。通常是 AD 域控制器的 IP 地址。
   - 域搜索列表 - 添加 DNS 域名。

> 注意：
>
> Linux VDA 当前不支持 NetBIOS 名称截断。因此，主机名不得超过 15 个字符。
>
> 提示：
>
> 只能使用字符 a-z、A-Z、0-9 和连字符 (-)。请避免使用下划线 (_)、空格和其他符号。主机名不得以数字开头和以连字符结尾。此规则也适用于 Delivery Controller 主机名。

禁用多播 **DNS**　只有在 SLED 上，默认设置才会启用多播 DNS (mDNS)，而这可能会导致名称解析结果不一致。默认情况下，SLES 上未启用 mDNS，因此无需任何操作。

要禁用 mDNS，请编辑 **/etc/nsswitch.conf** 并更改包含以下内容的行：

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

更改为：

```
hosts: files dns
```

检查主机名　验证主机名设置是否正确无误：

```
hostname
```

此命令仅返回计算机的主机名，而不返回其完全限定域名 (FQDN)。

验证 FQDN 设置是否正确无误：

```
hostname -f
```

此命令返回计算机的 FQDN。

检查名称解析和服务可访问性　确认可以解析 FQDN 并对域控制器和 Delivery Controller 执行 ping 操作：

```
nslookup domain-controller-fqdn

ping domain-controller-fqdn

nslookup delivery-controller-fqdn

ping delivery-controller-fqdn
```

如果无法解析 FQDN 或 Ping 不通上述任一计算机，请先检查相关步骤，然后再继续。

步骤 **1c**：配置 **NTP** 服务

维护 VDA、Delivery Controller 和域控制器之间的时钟始终精确同步至关重要。将 Linux VDA 托管为虚拟机可能会导致时钟偏差问题。出于此原因，最好使用远程 NTP 服务来保持时间同步。默认 NTP 设置可能需要作一些更改：

1. 打开 YaST 的“NTP Configuration”（NTP 配置），然后选择 **General Settings**（常规设置）选项卡。
2. 在“Start NTP Daemon”（启动 NTP 守护程序）部分，选中 **Now and on Boot**（现在及引导时）。
3. 如果存在 **Undisciplined Local Clock (LOCAL)**（无序本地时钟 (LOCAL)）项，选择该项，然后单击 **Delete**（删除）。
4. 单击 **Add**（添加），添加一个 NTP 服务器条目。
5. 选择 **Server Type**（服务器类型），然后单击 **Next**（下一步）。
6. 在“Address”（地址）字段键入 NTP 服务器的 DNS 名称。此服务通常托管在 Active Directory 域控制器上。
7. 将“Options”（选项）字段保持不变。
8. 单击 **Test**（测试）验证 NTP 服务是否可访问。
9. 在随后的一系列窗口中一直单击 **OK**（确定）保存更改。

> 注意：
>
> 对于 SLES 12 实施，NTP 守护程序可能会因 SUSE 上与 AppArmor 策略有关的一个已知问题而无法启动。请单击解决方案了解其他信息。

步骤 **1d**：安装 **Linux VDA** 依赖软件包

适用于 SUSE Linux Enterprise 的 Linux VDA 软件依赖于以下软件包：

- Postgresql10-server 10.12 或更高版本
- OpenJDK 1.8.0
- OpenMotif Runtime Environment 2.3.1 或更高版本
- Cups 1.6.0 或更高版本
- Foomatic 过滤器 3.0.0 或更高版本
- ImageMagick 6.8 或更高版本

添加存储库　可以从 SUSE Linux Enterprise 软件开发工具包 (SDK) 中获取一些必需的软件包，例如 PostgreSQL 和 ImageMagick。要获取这些软件包，请使用 YaST 添加 SDK 存储库，或者下载 SDK 映像文件，然后使用以下命令在本地进行装载：

```
sudo mkdir -p /mnt/sdk

sudo mount -t iso9660 path-to-iso/SLE-12-SP5-SDK-DVD-x86_64-GM-DVD1.iso
    /mnt/sdk

sudo zypper ar -f /mnt/sdk sdk
```

安装 **Kerberos** 客户端　安装 Kerberos 客户端，在 Linux VDA 与 Delivery Controller 之间实现双向身份验证：

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

Kerberos 客户端配置依赖于所使用的 Active Directory 集成方法。请参阅下面的说明。

安装 **OpenJDK**　Linux VDA 依赖于 OpenJDK 1.8.0。

> 提示：
>
> 为避免在 Delivery Controller 中注册失败，请务必仅安装 OpenJDK 1.8.0。从系统中删除所有其他版本的 Java。

- **SLED**：

1. 在 SLED 上，Java 运行时环境通常随操作系统一起安装。检查是否已安装它：

   ```
   sudo zypper info java-1_8_0-openjdk
   ```

2. 如果状态显示为过时，请更新为最新版本：

   ```
   sudo zypper update java-1_8_0-openjdk
   ```

3. 检查 Java 版本：

   ```
   java -version
   ```

- **SLES**：

1. 在 SLES 上，安装 Java 运行时环境：

   ```
   sudo zypper install java-1_8_0-openjdk
   ```

2. 检查 Java 版本：

   ```
   java -version
   ```

安装 **PostgreSQL**　在 SLED/SLES 12 中，安装以下软件包：

```
sudo zypper install postgresql-init

sudo zypper install postgresql10-server

sudo zypper install postgresql-jdbc
```

此时需要执行安装后步骤，以便初始化数据库服务，并确保 PostgreSQL 在计算机启动时启动：

```
sudo systemctl enable postgresql

sudo systemctl restart postgresql
```

数据库文件位于 /var/lib/pgsql/data。

删除存储库　在安装了相关软件包的情况下，运行以下命令以删除先前设置的 SDK 存储库和装载的媒体：

```
sudo zypper rr sdk

sudo umount /mnt/sdk

sudo rmdir /mnt/sdk
```

### 步骤 2：为虚拟机管理程序准备 Linux VM

在支持的虚拟机管理程序上将 Linux VDA 当作虚拟机运行时，需要作出一些更改。根据使用的虚拟机管理程序平台作出以下更改。如果正在裸机硬件上运行 Linux 计算机，则无需作出任何更改。

#### 修复 Citrix Hypervisor 上的时间同步问题

如果启用了 Citrix Hypervisor 时间同步功能，则在每个半虚拟化 Linux VM 中，您会发现 NTP 和 Citrix Hypervisor 都在尝试管理系统时钟。为避免时钟与其他服务器不同步，每个 Linux 客户机中的系统时钟都必须与 NTP 同步。这种情况要求禁用主机时间同步。无需在 HVM 模式下进行任何更改。

在某些 Linux 发行版中，如果正在运行半虚拟化 Linux 内核，并安装了 Citrix VM Tools，您可以检查 Citrix Hypervisor 时间同步功能是否存在，以及是否已在 Linux VM 中启用：

```
su -

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

此命令返回 0 或 1：

- 0 - 时间同步功能已启用，且必须禁用。
- 1 - 时间同步功能已禁用，无需采取任何操作。

如果 **/proc/sys/xen/independent_wallclock** 文件不存在，则不需要执行以下步骤。

如果已启用，请通过向该文件写入 **1** 以禁用时间同步功能：

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

要使此更改成为永久更改，并在重新启动后仍然有效，请编辑 **/etc/sysctl.conf** 文件并添加以下行：

```
xen.independent_wallclock = 1
```

要验证这些更改，请重新启动系统：

```
reboot
```

重新启动后，验证此设置是否正确：

```
su -

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

此命令返回值 1。

#### 在 **Microsoft Hyper-V** 上修复时间同步问题

安装了 Hyper-V Linux 集成服务的 Linux VM 可应用 Hyper-V 时间同步功能来使用主机操作系统的时间。为确保系统时钟始终精确可靠，请一同启用此功能与 NTP 服务。

从管理操作系统中：

1. 打开 Hyper-V 管理器控制台。
2. 对于 Linux VM 的设置，请选择 **Integration Services**（集成服务）。
3. 确保已选择 **Time synchronization**（时间同步）。

> 注意：
>
> 此方法与 VMware 和 Citrix Hypervisor 不同，这两种产品会禁用主机时间同步功能，以免与 NTP 发生冲突。Hyper-V 时间同步可以与 NTP 时间同步共存并互补。

#### 修复 **ESX** 和 **ESXi** 上的时间同步问题

如果启用了 VMware 时间同步功能，则在每个半虚拟化 Linux VM 中，您会发现 NTP 和虚拟机管理程序都在尝试同步系统时钟。为避免时钟与其他服务器不同步，请将每个 Linux 来宾中的系统时钟与 NTP 同步。这种情况要求禁用主机时间同步。

如果正在运行安装了 VMware Tools 的半虚拟化 Linux 内核：

1. 打开 vSphere Client。
2. 编辑 Linux VM 设置。

3. 在 **Virtual Machine Properties**（虚拟机属性）对话框中，打开 **Options**（选项）选项卡。
4. 选择 **VMware Tools**。
5. 在 **Advanced**（高级）框中，取消选中 **Synchronize guest time with host**（与主机同步客户机时间）。

### 步骤 3：向 Windows 域中添加 Linux 虚拟机 (VM)

Linux VDA 支持多种向 Active Directory (AD) 域添加 Linux 计算机的方法：

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

根据所选的方法，按说明执行操作。

> 注意：
>
> 为 Linux VDA 中的本地帐户和 AD 中的帐户使用相同的用户名时，会话启动可能会失败。

#### Samba Winbind

加入 **Windows** 域　您的域控制器必须可访问，而且您必须具有有权将计算机添加到域的 Active Directory 用户帐户。

1. 打开 YaST 的“Windows Domain Membership”（Windows 域成员身份）。
2. 进行以下更改：
   - 将域或工作组设为 Active Directory 域的名称或域控制器的 IP 地址。确保域名为大写。
   - 选中 **Also Use SMB information for Linux Authentication**（同时为 Linux 身份验证使用 SMB 信息）。
   - 选中 **Create Home Directory on Login**（在登录时创建主目录）。
   - 选中 **Single Sign-on for SSH**（为 SSH 使用单点登录）。
   - 确保未选中 **Offline Authentication**（脱机身份验证）。此选项与 Linux VDA 不兼容。
3. 单击确定。如果提示安装某些软件包，请单击 **Install**（安装）。
4. 如果找到域控制器，则会询问您是否要加入域。单击是。
5. 出现提示时，键入有权将计算机添加到域的域用户的凭据，然后单击 **OK**（确定）。
6. 此时会显示一条消息，说明操作成功。
7. 如果提示安装某些 samba 和 krb5 软件包，请单击 **Install**（安装）。

YaST 可能会说明这些更改需要重新启动计算机或一些服务。我们建议您重新启动计算机：

```
su -

reboot
```

仅限 **SUSE 12**：修补了 **Kerberos** 凭据缓存名称　SUSE 12 已将默认 Kerberos 凭据缓存名称规范从常用的 **FILE:/tmp/krb5cc_%{uid}** 更改为 **DIR:/run/user/%{uid}/krb5cc**。这种全新 DIR 缓存方法与 Linux VDA 不兼容，必须进行手动更改。以 root 用户身份，编辑 **/etc/krb5.conf** 并在 **[libdefaults]** 部分下添加以下设置（如果尚未设置）：

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

验证域成员身份　Delivery Controller 要求所有 VDA 计算机（Windows 和 Linux）都要在 Active Directory 中有一个计算机对象。

运行 Samba 的 **net ads** 命令验证计算机是否已加入域：

```
sudo net ads testjoin
```

运行以下命令验证额外的域和计算机对象信息：

```
sudo net ads info
```

验证 **Kerberos** 配置　为了确保 Kerberos 已正确配置为可与 Linux VDA 配合使用，请验证系统 keytab 文件是否已创建并包含有效密钥：

```
sudo klist -ke
```

此命令显示各种主体名称与密码套件组合可用的密钥列表。运行 Kerberos `kinit` 命令，使用这些密钥向域控制器验证计算机的身份：

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

计算机和领域名称必须指定为大写。美元符号 ($) 必须使用反斜杠 (\) 进行转义，以免发生 shell 替换。在某些环境中，DNS 域名与 Kerberos 领域名称不同。请确保使用领域名称。如果此命令成功运行，则不会显示任何输出。

使用以下命令验证计算机帐户的 TGT 票据已缓存：

```
sudo klist
```

使用以下命令检查计算机帐户详细信息：

```
sudo net ads status
```

验证用户身份验证　使用 `wbinfo` 工具验证是否可向域验证域用户的身份：

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

这里指定的域为 AD 域名，而不是 Kerberos 领域名称。对于 bash shell，必须使用另一个反斜杠对反斜杠 (\) 字符进行转义。此命令返回一条成功或失败消息。

要验证 Winbind PAM 模块是否已正确配置，请使用以前未使用的域用户帐户登录 Linux VDA。

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

验证是否为 **id -u** 命令返回的 uid 创建了对应的 Kerberos 凭据缓存文件：

```
ls /tmp/krb5cc_uid
```

验证用户 Kerberos 凭据缓存中的票据是否有效且未过期：

```
klist
```

退出会话。

```
exit
```

直接登录 Gnome 或 KDE 控制台也可以执行类似的测试。在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### Quest Authentication Service

在域控制器上配置 **Quest**　假定您已在 Active Directory 域控制器上安装并配置了 Quest 软件，而且已获得管理权限，有权在 Active Directory 中创建计算机对象。

允许域用户登录 **Linux VDA** 计算机　为了让域用户能够在 Linux VDA 计算机上建立 HDX 会话：

1. 在 Active Directory 用户和计算机管理控制台中，为该用户帐户打开 Active Directory 用户属性。
2. 选择 **Unix Account**（Unix 帐户）选项卡。
3. 选中 **Unix-enabled**（已启用 Unix）。
4. 将 **Primary GID Number**（首选 GID 编号）设置为实际域用户组的组 ID。

> 注意：
>
> 这些说明相当于设置域用户，以便他们可以使用控制台、RDP、SSH 或任何其他远程协议进行登录。

在 **Linux VDA** 上配置 **Quest**

配置 **VAS** 守护程序　必须启用并断开自动续订 Kerberos 票据功能。必须禁用身份验证（脱机登录）功能：

```
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
```

此命令将续订间隔设为 9 小时（32400 秒），即比默认的 10 小时票据生命周期短 1 小时。请在票据生命周期较短的系统上设置较低的值。

配置 **PAM** 和 **NSS**　要启用通过 HDX 进行的域用户登录以及其他服务（例如 su、ssh 和 RDP），请运行以下命令以手动配置 PAM 和 NSS：

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

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

加入 **Windows** 域　使用 Quest `vastool` 命令将 Linux 计算机加入到 Active Directory 域中：

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

**user** 为有权将计算机加入 Active Directory 域的任何域用户。**domain-name** 为域的 DNS 名称，例如 example.com。

验证域成员身份　Delivery Controller 要求所有 VDA 计算机（Windows 和 Linux）都要在 Active Directory 中有一个计算机对象。验证 Quest 加入的 Linux 计算机是否位于域中：

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

如果计算机已加入域，此命令会返回域名。如果计算机未加入任何域，则会显示以下错误：

```
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
```

验证用户身份验证　要验证 Quest 是否能够通过 PAM 对域用户进行身份验证，请使用以前未使用的域用户帐户登录 Linux VDA。

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

验证是否为 **id -u** 命令返回的 uid 创建了对应的 Kerberos 凭据缓存文件：

```
ls /tmp/krb5cc_uid
```

验证 Kerberos 凭据缓存中的票据是否有效且未过期：

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

退出会话。

```
exit
```

直接登录 Gnome 或 KDE 控制台也可以执行类似的测试。在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### Centrify DirectControl

加入 **Windows** 域　安装 Centrify DirectControl Agent 后，请使用 Centrify **adjoin** 命令将 Linux 计算机加入 Active Directory 域：

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

**user** 为有权将计算机加入 Active Directory 域的任何 Active Directory 域用户。**domain-name** 是将 Linux 计算机加入到的域的名称。

验证域成员身份　Delivery Controller 要求所有 VDA 计算机（Windows 和 Linux）都要在 Active Directory 中有一个计算机对象。验证 Centrify 加入的 Linux 计算机是否位于域中：

```
su -

adinfo
```

验证 **Joined to domain** 值是否有效以及 **CentrifyDC mode** 是否返回了 **connected**。如果模式仍然卡在启动状态，则表明 Centrify 客户端遇到了服务器连接或身份验证问题。

使用以下命令可获得更全面的系统和诊断信息：

```
adinfo --sysinfo all

adinfo - diag
```

测试与各种 Active Directory 和 Kerberos 服务的连接。

```
adinfo --test
```

在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### SSSD

如果您在 SUSE 上使用 SSSD，请按照此部分中的说明进行操作。此部分包含有关如何将 Linux VDA 计算机加入 Windows 域的说明以及如何配置 Kerberos 身份验证的指导。

要在 SUSE 上设置 SSSD，请完成以下步骤：

1. 加入域并创建主机 keytab
2. 为 SSSD 配置 PAM
3. 设置 SSSD
4. 启用 SSSD
5. 验证域成员身份
6. 验证 Kerberos 配置
7. 验证用户身份验证

加入域并创建主机 **keytab**　SSSD 并不提供用于加入域和管理系统 keytab 文件的 Active Directory 客户端功能。可以改为使用 Samba 方法。在配置 SSSD 之前，请完成以下步骤。

1. 停止并禁用 Name Service Cache Daemon (NSCD) 守护进程。

```
sudo systemctl stop nscd

sudo systemctl disable nscd
```

2. 安装或更新所需软件包：

```
sudo zypper install krb5-client

sudo zypper install samba-client
```

3. 以 root 用户身份编辑 /etc/krb5.conf 文件，以允许 kinit 实用程序与目标域进行通信。在 **[libdefaults]**、**[realms]** 和 **[domain_realm]** 部分下添加以下条目：

> 注意：
>
> 根据您的 AD 基础结构配置 Kerberos。以下设置适用于单域、单林模型。

```
```
[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 = REALM
```

**realm** is the Kerberos realm name, such as example.com. **REALM** is the Kerberos realm name in uppercase, such as EXAMPLE.COM. **fqdn-of-domain-controller** is the FQDN of the domain controller.

1. 以 root 用户身份编辑 /etc/samba/smb.conf 以允许 **net** 实用程序与目标域进行通信。将以下条目添加到 **[global]** 部分：

```
[global]
    workgroup = domain

    realm = REALM

    security = ADS

    kerberos method = secrets and keytab

    client signing = yes

    client use spnego = yes
```

   **domain** 是 Active Directory 域的简短 NetBIOS 名称，例如 EXAMPLE。

2. 修改 /etc/nsswitch.conf 文件中的 **passwd** 和 **group** 条目以在解析用户和组时引用 SSSD。

```
passwd: compat sss

group: compat sss
```

3. 加入 Windows 域。请确保域控制器可访问，而且您具有有权将计算机添加到域的 Active Directory 用户帐户。

```
sudo realm join REALM -U user
```

   **user** 为有权将计算机加入域的域用户。

**为 SSSD 配置 PAM** 在为 SSSD 配置 PAM 之前，请安装或更新所需的软件包：

```
sudo zypper install sssd sssd-ad
```

将 PAM 模块配置为通过 SSSD 进行用户身份验证，并为用户登录创建主目录。

```
sudo pam-config --add  --sss
sudo pam-config --add --mkhomedir
```

**设置 SSSD**

1. 以 root 用户身份编辑 /etc/sssd.conf，以允许 SSSD 守护程序与目标域进行通信。sssd.conf 配置示例（可以根据需要添加额外的选项）：

```
[sssd]
    config_file_version = 2
    services = nss,pam
    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** 是 DNS 域名，例如 example.com。

> 注意：
>
> **ldap_id_mapping** 设置为 true，以便 SSSD 本身负责将 Windows SID 映射到 Unix UID。否则，Active Directory 必须能够提供 POSIX 扩展。将 **ad_gpo_access_control** 设置为 **permissive** 以

> 防止 Linux 会话出现无效登录错误。请参阅 sssd.conf 和 sssd-ad 的手册页。

2. 对 sssd.conf 设置文件所有权和权限：

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

启用 **SSSD**　运行以下命令以在系统启动时启用并启动 SSSD 守护程序：

```
sudo systemctl enable sssd
sudo systemctl start sssd
```

验证域成员身份

1. 运行 Samba 的 net ads 命令验证计算机是否已加入域：

```
sudo net ads testjoin
```

2. 运行以下命令验证额外的域和计算机对象信息：

```
sudo net ads info
```

验证 **Kerberos** 配置　要验证 Kerberos 是否已正确配置为可与 Linux VDA 配合使用，请验证系统 keytab 文件是否已创建并包含有效密钥：

```
sudo klist -ke
```

此命令显示各种主体名称与密码套件组合可用的密钥列表。运行 Kerberos kinit 命令，以使用这些密钥向域控制器验证计算机的身份：

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

计算机和领域名称必须指定为大写。美元符号 (**$**) 必须使用反斜杠 (****) 进行转义，以免发生 shell 替换。在某些环境中，DNS 域名与 Kerberos 领域名称不同。请确保使用领域名称。如果此命令成功运行，则不会显示任何输出。

使用以下命令验证计算机帐户的 TGT 票据已缓存：

```
sudo klist
```

验证用户身份验证　SSSD 不直接通过守护程序提供用于测试身份验证的命令行工具，只能通过 PAM 完成。

要验证 SSSD PAM 模块是否已正确配置，请使用以前未使用的域用户帐户登录 Linux VDA。

```
ssh localhost -l domain\username

id -u

```

```
klist

exit
```

验证 klist 命令返回的 Kerberos 票据是否适用于该用户并且尚未过期。

以 root 用户身份，验证是否已为前面的 `id -u` 命令返回的 uid 创建相应的票据缓存文件：

```
ls /tmp/krb5cc_uid
```

直接登录 Gnome 或 KDE 控制台也可以执行类似的测试。在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### 步骤 **4**：安装 **Linux VDA**

#### 步骤 **4a**：卸载旧版本

如果安装了除先前的两个版本和 LTSR 版本之外的早期版本，请在安装新版本之前将其卸载。

1. 停止 Linux VDA 服务：

   ```
   sudo /sbin/service ctxvda stop

   sudo /sbin/service ctxhdx stop
   ```

   > 注意：
   >
   > 在停止 `ctxvda` 和 `ctxhdx` 服务之前，请运行 `service ctxmonitorservice stop` 命令以停止监视服务守护程序。否则，监视服务守护程序将重新启动您停止的服务。

2. 卸载软件包：

   ```
   sudo rpm -e XenDesktopVDA
   ```

> 重要：
>
> 支持从最新的两个版本进行升级。
>
> 注意：
>
> 安装组件位于 **/opt/Citrix/VDA/** 中。
>
> 要运行命令，需要提供完整路径；或者，也可以将 **/opt/Citrix/VDA/sbin** 和 **/opt/Citrix/VDA/bin** 添加到系统路径。

#### 步骤 **4b**：下载 **Linux VDA** 软件包

转至 Citrix Virtual Apps and Desktops 下载页面。展开相应版本的 Citrix Virtual Apps and Desktops，然后单击组件以下载与 Linux 发行版匹配的 Linux VDA 包。

步骤 **4c**：安装 **Linux VDA**

使用 Zypper 安装 Linux VDA 软件：

对于 **SUSE 12**：

```
sudo zypper install XenDesktopVDA-<version>.sle12_x.x86_64.rpm
```

使用 RPM 软件包管理器安装 Linux VDA 软件。在此之前，请解决以下依赖项：

对于 **SUSE 12**：

```
sudo rpm -i XenDesktopVDA-<version>.sle12_x.x86_64.rpm
```

步骤 **4d**：升级 **Linux VDA**（可选）

可以从先前的两个版本和 LTSR 版本升级现有安装。

对于 **SUSE 12**：

```
sudo rpm -U XenDesktopVDA-<version>.sle12_x.x86_64.rpm
```

**SUSE 12** 的 **RPM** 依赖项列表：

```
postgresql-server >= 9.3

postgresql-jdbc >= 9.2

java-1.8.0-openjdk >= 1.8.0

ImageMagick >= 6.8

dbus-1 >= 1.8.8

dbus-1-x11 >= 1.8.8

libXpm4 >= 3.5.11

libXrandr2 >= 1.4.2

libXtst6 >= 1.2.2

motif >= 2.3

pam >= 1.1.8

bash >= 4.2

findutils >= 4.5

gawk >= 4.1
```

```

sed >= 4.2

cups >= 1.6.0

cups-filters-foomatic-rip >= 1.0.0

openldap2 >= 2.4

cyrus-sasl >= 2.1

cyrus-sasl-gssapi >= 2.1

libxml2 >= 2.9

python-requests >= 2.8.1

rpmlib(PayloadFilesHavePrefix) <= 4.0-1

rpmlib(CompressedFileNames) <= 3.0.4-1

rpmlib(PayloadIsLzma) <= 4.4.6-1

libtcmalloc4 >= 2.5

libcap-progs >= 2.22

xorg-x11-server >= 7.6_1.18.3-76.15

ibus >= 1.5

xorg- x11-server = 7.6_1.19.6

xorg-x11 = 7.6_1

postgresql10-server >= 10.12

libgtk-2_0-0 >= 2.24

libgthread-2_0-0 >= 2.48

pulseaudio-utils >= 5.0

lsb-release >= 2.0
```

> 重要：
>
> 在升级后重新启动 Linux VDA 计算机。

### 步骤 5：配置 Linux VDA

安装软件包后，必须运行 ctxsetup.sh 脚本来配置 Linux VDA。执行任何更改之前，该脚本都会验证环境，确保所有依赖项都已安装。如有必要，可以随时重新运行该脚本以更改设置。

可以按照提示手动运行脚本，也可以采用预先配置的响应自动运行脚本。继续操作前，请查看该脚本的帮助信息：

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

#### 提示配置

运行会提示各种问题的手动配置：

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

#### 自动配置

自动安装时，通过环境变量提供设置脚本所需的选项。如果所需的所有变量都存在，脚本不会提示您提供任何信息。

支持的环境变量包括：

- **CTX_XDL_SUPPORT_DDC_AS_CNAME = Y | N** –Linux VDA 支持使用 DNS CNAME 记录指定 Delivery Controller 名称。默认情况下设置为 N。
- **CTX_XDL_DDC_LIST= ‘list-ddc-fqdns’** –Linux VDA 要求提供由空格分隔的 Delivery Controller 完全限定域名 (FQDN) 列表以用于向 Delivery Controller 注册。必须至少指定一个 FQDN 或 CNAME 别名。
- **CTX_XDL_VDA_PORT=port-number** –Linux VDA 通过 TCP/IP 端口（默认为端口 80）与 Delivery Controller 通信。
- **CTX_XDL_REGISTER_SERVICE=Y | N** - 在启动计算机后启动 Linux Virtual Desktop 服务。默认情况下，该值设置为 Y。
- **CTX_XDL_ADD_FIREWALL_RULES=Y | N** –Linux Virtual Desktop 服务要求允许传入网络连接通过系统防火墙。您可以在系统防火墙中自动为 Linux Virtual Desktop 打开所需端口（默认为端口 80 和 1494）。默认情况下设置为 Y。
- **CTX_XDL_AD_INTEGRATION=1 | 2 | 3 | 4** –Linux VDA 要求使用 Kerberos 配置设置向 Delivery Controller 进行身份验证。Kerberos 配置根据系统上已安装和已配置的 Active Directory 集成工具确定。指定要使用且受支持的 Active Directory 集成方法：
  - 1 –Samba Winbind
  - 2 –Quest Authentication Service
  - 3 - Centrify DirectControl
  - 4 - SSSD

- **CTX_XDL_HDX_3D_PRO=Y | N** –Linux VDA 支持 HDX 3D Pro，这是一组 GPU 加速技术，旨在优化富图形应用程序的虚拟化水平。如果选择了 HDX 3D Pro，则要为 VDI 桌面（单会话）模式配置 VDA -（即 CTX_XDL_VDI_MODE=Y）。
- **CTX_XDL_VDI_MODE=Y | N** –将计算机配置为专用桌面交付模型 (VDI) 还是托管共享桌面交付模型。对于 HDX 3D Pro 环境，将此变量设置为 Y。默认情况下，此变量设置为 N。
- **CTX_XDL_SITE_NAME=dns-name** –Linux VDA 通过 DNS 发现 LDAP 服务器。要将 DNS 搜索结果限制为本地站点，应指定 DNS 站点名称。默认情况下，此变量设置为 **<none>**。
- **CTX_XDL_LDAP_LIST= ‘list-ldap-servers’** –Linux VDA 查询 DNS 来发现 LDAP 服务器。如果 DNS 无法提供 LDAP 服务记录，您可以提供以空格分隔的 LDAP FQDN（带有 LDAP 端口）列表。例如 ad1.mycompany.com:389。默认情况下，此变量设置为 **<none>**。
- **CTX_XDL_SEARCH_BASE=search-base-set** –Linux VDA 通过设置为 Active Directory 域根的搜索库来查询 LDAP（例如，DC=mycompany,DC=com）。为提高搜索性能，可以指定搜索基础（例如 OU=VDI,DC=mycompany,DC=com）。默认情况下，此变量设置为 **<none>**。
- **CTX_XDL_FAS_LIST= ‘list-fas-servers’** –联合身份验证服务 (FAS) 服务器是通过 AD 组策略配置的。Linux VDA 不支持 AD 组策略，但您可以改为提供以分号分隔的 FAS 服务器的列表。顺序必须与在 AD 组策略中配置的顺序相同。如果删除了任何服务器地址，请使用 **<none>** 文本字符串填充其空白，并且不要修改服务器地址的顺序。
- **CTX_XDL_DOTNET_ RUNTIME_PATH=path-to-install-dotnet-runtime** – 安装.NET Core Runtime 3.1 以支持新的 Broker 代理服务 (`ctxvda`) 的路径。默认路径为 /usr/bin。
- **CTX_XDL_START_SERVICE=Y | N** –在完成 Linux VDA 配置后，是否启动 Linux VDA 服务。默认情况下设置为 Y。
- **CTX_XDL_TELEMETRY_SOCKET_PORT** - 用于侦听 Citrix Scout 的套接字端口。默认端口为 7503。
- **CTX_XDL_TELEMETRY_PORT** - 用于与 Citrix Scout 通信的端口。默认端口为 7502。

设置环境变量并运行配置脚本：

```
export CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N

export CTX_XDL_DDC_LIST= ‘list-ddc-fqdns’

export CTX_XDL_VDA_PORT=port-number

export CTX_XDL_REGISTER_SERVICE=Y|N

export CTX_XDL_ADD_FIREWALL_RULES=Y|N

export CTX_XDL_AD_INTEGRATION=1|2|3|4

export CTX_XDL_HDX_3D_PRO=Y|N

export CTX_XDL_VDI_MODE=Y|N

export CTX_XDL_SITE_NAME=dns-site-name | '<none>'

export CTX_XDL_LDAP_LIST= ‘list-ldap-servers’  | '<none>'

```

```
export CTX_XDL_SEARCH_BASE=search-base-set | '<none>'

export CTX_XDL_FAS_LIST= ‘list-fas-servers’  | '<none>'

export CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime

export CTX_XDL_TELEMETRY_SOCKET_PORT=port-number

export CTX_XDL_TELEMETRY_PORT=port-number

export CTX_XDL_START_SERVICE=Y|N

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

运行 sudo 命令时，键入 **-E** 选项以将现有环境变量传递给它创建的新 shell。我们建议您使用前面的命令并加上 **#!/bin/bash** 作为第一行来创建 shell 脚本文件。

或者，您可以使用单个命令指定所有参数：

```
sudo CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N \

CTX_XDL_DDC_LIST= ‘list-ddc-fqdns’  \

CTX_XDL_VDA_PORT=port-number \

CTX_XDL_REGISTER_SERVICE=Y|N \

CTX_XDL_ADD_FIREWALL_RULES=Y|N \

CTX_XDL_AD_INTEGRATION=1|2|3|4 \

CTX_XDL_HDX_3D_PRO=Y|N \

CTX_XDL_VDI_MODE=Y|N \

CTX_XDL_SITE_NAME=dns-name \

CTX_XDL_LDAP_LIST= ‘list-ldap-servers’  \

CTX_XDL_SEARCH_BASE=search-base-set \

CTX_XDL_FAS_LIST= ‘list-fas-servers’  \

CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime \

CTX_XDL_TELEMETRY_SOCKET_PORT=port-number \

CTX_XDL_TELEMETRY_PORT=port-number \

CTX_XDL_START_SERVICE=Y|N \

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

删除配置更改

在某些情形下，您可能需要删除 **ctxsetup.sh** 脚本对配置所做的更改，但不卸载 Linux VDA 软件包。

继续操作前，请查看此脚本的帮助信息：

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

删除配置更改：

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

> 重要：
>
> 此脚本会从数据库删除所有配置数据，从而使 Linux VDA 无法使用。

配置日志

**ctxsetup.sh** 和 **ctxcleanup.sh** 脚本会在控制台上显示错误，并将其他信息写入配置日志文件：

`/tmp/xdl.configure.log`

重新启动 Linux VDA 服务，确保更改生效。

### 步骤 6：运行 Linux VDA

使用 **ctxsetup.sh** 脚本配置 Linux VDA 后，可以运行以下命令来控制 Linux VDA。

启动 **Linux VDA**：

启动 Linux VDA 服务：

```
sudo /sbin/service ctxhdx start

sudo /sbin/service ctxvda start
```

停止 **Linux VDA**：

停止 Linux VDA 服务：

```
sudo /sbin/service ctxvda stop

sudo /sbin/service ctxhdx stop
```

> 注意：
>
> 在停止 `ctxvda` 和 `ctxhdx` 服务之前，请运行 `service ctxmonitorservice stop` 命令以停止监视服务守护程序。否则，监视服务守护程序将重新启动您停止的服务。

重新启动 **Linux VDA**：

重新启动 Linux VDA 服务：

```
sudo /sbin/service ctxvda stop

sudo /sbin/service ctxhdx restart

sudo /sbin/service ctxvda start
```

检查 **Linux VDA** 状态：

要检查 Linux VDA 服务的运行状态，请执行以下操作：

```
sudo /sbin/service ctxvda status

sudo /sbin/service ctxhdx status
```

### 步骤 7：在 Citrix Virtual Apps 或 Citrix Virtual Desktops 中创建计算机目录

创建计算机目录和添加 Linux VDA 计算机的过程与传统的 Windows VDA 方法类似。有关如何完成这些任务的更加详细的说明，请参阅创建计算机目录和管理计算机目录。

创建包含 Linux VDA 计算机的计算机目录时会面临一些限制，使得该过程不同于为 Windows VDA 计算机创建计算机目录：

- 对于操作系统，请选择：
  - 多会话操作系统选项（对于托管共享桌面交付模型）。
  - 单会话操作系统选项（对于 VDI 专用桌面交付模型）。
- 请勿在同一个计算机目录中混合使用 Linux 和 Windows VDA 计算机。

> 注意：
>
> 早期版本的 Citrix Studio 不支持“Linux 操作系统”的概念。但是，选择 **Windows Server** 操作系统或服务器操作系统选项等同于使用托管共享桌面交付模型。选择 **Windows** 桌面操作系统或桌面操作系统选项等同于使用每计算机一个用户交付模型。
>
> 提示：
>
> 如果删除计算机后将其重新加入 Active Directory 域，则必须删除计算机，然后将其重新添加到计算机目录。

### 步骤 8：在 Citrix Virtual Apps 或 Citrix Virtual Desktops 中创建交付组

创建交付组和添加包含 Linux VDA 计算机的计算机目录的过程与 Windows VDA 计算机几乎相同。有关如何完成这些任务的更加详细的说明，请参阅创建交付组。

创建含有 Linux VDA 计算机目录的交付组时会面临以下限制：

- 确保所选的 AD 用户和组已正确配置，可以登录到 Linux VDA 计算机。
- 请勿允许未经身份验证的（匿名）用户登录。
- 请勿在交付组中混入含有 Windows 计算机的计算机目录。

> 重要：
>
> Linux VDA 1.4 及更高版本支持发布应用程序。但是，Linux VDA 不支持将桌面和应用程序交付给相同的计算机。

有关如何创建计算机目录和交付组的信息，请参阅 Citrix Virtual Apps and Desktops 7 2103。

## 安装 Linux Virtual Delivery Agent for Ubuntu

June 21, 2022

可以选择按照本文中的步骤进行手动安装，也可以使用轻松安装进行自动安装和配置。轻松安装省时又省力，与手动安装相比，更不易于出错。

> 注意：
>
> 请仅对全新安装使用轻松安装功能。请勿使用轻松安装更新现有安装。

### 步骤 1：为 VDA 安装准备 Ubuntu

#### 步骤 1a：验证网络配置

我们建议您先连接并正确配置网络，然后再继续操作。

如果您使用的是 Ubuntu 18.04 Live Server，请在设置主机名之前在 **/etc/cloud/cloud.cfg** 配置文件中做以下更改：

```
preserve_hostname: true
```

#### 步骤 1b：设置主机名

为确保正确报告计算机的主机名，请更改 **/etc/hostname** 文件，使其仅包含计算机的主机名。

```
hostname
```

#### 步骤 1c：为主机名分配环回地址

确保计算机的 DNS 域名和完全限定域名 (FQDN) 正确地返回报告。方法是更改 **/etc/hosts** 文件的以下行以包含 FQDN 和主机名作为前两个条目：

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

例如：

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

从文件中的其他条目中删除对 **hostname-fqdn** 或 **hostname** 的任何其他引用。

> 注意：
>
> Linux VDA 当前不支持 NetBIOS 名称截断。因此，主机名不得超过 15 个字符。
>
> 提示：
>
> 只能使用字符 a-z、A-Z、0-9 和连字符 (-)。请避免使用下划线 (_)、空格和其他符号。主机名不得以数字开头和以连字符结尾。此规则也适用于 Delivery Controller 主机名。

#### 步骤 **1d**：检查主机名

验证主机名设置是否正确无误：

```
hostname
```

此命令仅返回计算机的主机名，而不返回其 FQDN。

验证 FQDN 设置是否正确无误：

```
hostname -f
```

此命令返回计算机的 FQDN。

#### 步骤 **1e**：禁用多播 **DNS**

默认设置启用了多播 DNS (**mDNS**)，而这会导致名称解析结果不一致。

要禁用 **mDNS**，请编辑 **/etc/nsswitch.conf** 并更改包含以下内容的行：

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

更改为：

```
hosts: files dns
```

#### 步骤 **1f**：检查名称解析和服务可访问性

确认可以解析 FQDN 并对域控制器和 Delivery Controller 执行 ping 操作：

```
nslookup domain-controller-fqdn

ping domain-controller-fqdn

nslookup delivery-controller-fqdn

ping delivery-controller-fqdn
```

如果无法解析 FQDN 或 Ping 不通上述任一计算机，请先检查相关步骤，然后再继续。

### 步骤 **1g**：配置时钟同步 **(chrony)**

确保 VDA、Delivery Controller 和域控制器之间的时钟始终精确同步至关重要。将 Linux VDA 托管为虚拟机可能会导致时钟偏差问题。出于此原因，最好使用远程时间服务来同步时间。

安装 chrony：

```
apt-get install chrony
```

以 root 用户身份，编辑 **/etc/chrony/chrony.conf** 并为每个远程时间服务器添加一个服务器条目：

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

在典型部署中，时间从本地域控制器同步，而不是直接从公共 NTP 池服务器同步。为域中的每个 Active Directory 域控制器添加一个服务器条目。

删除列出的任何其他 **server** 或 **pool** 条目，包括环回 IP 地址、localhost 和公共服务器 ***.pool.ntp.org** 条目。

保存更改并重新启动 Chrony 守护程序：

```
sudo systemctl restart chrony
```

### 步骤 **1h**：安装 **OpenJDK**

Linux VDA 依赖于 OpenJDK。通常，运行时环境作为操作系统安装的一部分进行安装。

在 Ubuntu 16.04 上，使用以下方法安装 OpenJDK：

```
sudo apt-get install -y default-jdk
```

在 Ubuntu 18.04 上，使用以下方法安装 OpenJDK：

```
sudo apt-get install -y openjdk-8-jdk
```

步骤 **1i**：安装 **PostgreSQL**

Linux VDA 要求在 Ubuntu 上安装 PostgreSQL 9.x 版：

```
sudo apt-get install -y postgresql

sudo apt-get install -y libpostgresql-jdbc-java
```

步骤 **1j**：安装 **Motif**

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

步骤 **1k**：安装其他软件包

```
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
```

步骤 **2**：准备虚拟机管理程序

在支持的虚拟机管理程序上将 Linux VDA 当作虚拟机运行时，需要作出一些更改。根据使用的虚拟机管理程序平台作出以下更改。如果正在裸机硬件上运行 Linux 计算机，则无需作出任何更改。

修复 **Citrix Hypervisor** 上的时间同步问题

启用了 Citrix Hypervisor 时间同步功能时，在每个半虚拟化 Linux VM 中，您会发现 NTP 和 Citrix Hypervisor 都尝试管理系统时钟。为避免时钟与其他服务器不同步，请确保每个 Linux 客户机中的系统时钟都与 NTP 同步。这种情况要求禁用主机时间同步。无需在 HVM 模式下进行任何更改。

在某些 Linux 发行版中，如果正在运行半虚拟化 Linux 内核，并安装了 Citrix VM Tools，您可以检查 Citrix Hypervisor 时间同步功能是否存在，以及是否已在 Linux VM 中启用：

```
su -

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

此命令返回 0 或 1：

- 0 - 时间同步功能已启用，且必须禁用。
- 1 - 时间同步功能已禁用，无需采取任何操作。

如果 /proc/sys/xen/independent_wallclock 文件不存在，则不需要执行以下步骤。

如果已启用，请通过向该文件写入 1 以禁用时间同步功能：

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

要使此更改成为永久更改，并在重新启动后仍然有效，请编辑 **/etc/sysctl.conf** 文件并添加以下行：

```
xen.independent_wallclock = 1
```

要验证这些更改，请重新启动系统：

```
su -

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

此命令返回值 1。

#### 在 **Microsoft Hyper-V** 上修复时间同步问题

安装了 Hyper-V Linux 集成服务的 Linux VM 可以使用 Hyper-V 时间同步功能来使用主机操作系统的时间。为确保系统时钟始终精确可靠，请一同启用此功能与 NTP 服务。

从管理操作系统中：

1. 打开 Hyper-V 管理器控制台。
2. 对于 Linux VM 的设置，请选择 **Integration Services**（集成服务）。
3. 确保已选择 **Time synchronization**（时间同步）。

> 注意：
>
> 此方法与 VMware 和 Citrix Hypervisor 不同，这两种产品会禁用主机时间同步功能，以免与 NTP 发生冲突。Hyper-V 时间同步可以与 NTP 时间同步共存并互补。

#### 修复 **ESX** 和 **ESXi** 上的时间同步问题

启用了 VMware 时间同步功能时，在每个半虚拟化 Linux VM 中，您会发现 NTP 和虚拟机管理程序都尝试同步系统时钟。为避免时钟与其他服务器不同步，请确保每个 Linux 客户机中的系统时钟都与 NTP 同步。这种情况要求禁用主机时间同步。

如果正在运行安装了 VMware Tools 的半虚拟化 Linux 内核：

1. 打开 vSphere Client。
2. 编辑 Linux VM 设置。

3. 在 **Virtual Machine Properties**（虚拟机属性）对话框中，打开 **Options**（选项）选项卡。
4. 选择 **VMware Tools**。
5. 在 **Advanced**（高级）框中，取消选中 **Synchronize guest time with host**（与主机同步客户机时间）。

### 步骤 3：向 Windows 域中添加 Linux 虚拟机 (VM)

Linux VDA 支持多种向 Active Directory (AD) 域添加 Linux 计算机的方法：

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

根据所选的方法，按说明执行操作。

> 注意：
>
> 为 Linux VDA 中的本地帐户和 AD 中的帐户使用相同的用户名时，会话启动可能会失败。

#### Samba Winbind

安装或更新所需软件包

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

在计算机启动时启用要启动的 **Winbind** 守护程序　Winbind 守护程序必须配置为在计算机启动时启动：

```
sudo systemctl enable winbind
```

> 注意：
>
> 确保 `winbind` 脚本位于 `/etc/init.d` 下。

配置 **Kerberos**　以 root 用户身份，打开 **/etc/krb5.conf** 并配置以下设置：

> 注意：
>
> 根据您的 AD 基础结构配置 Kerberos。以下设置适用于单域、单林模型。

```
[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
```

此上下文中的 **domain-dns-name** 参数为 DNS 域名，例如 **example.com**。**REALM** 是大写的 Kerberos 领域名称，例如 **EXAMPLE.COM**。

配置 **Winbind** 身份验证　手动配置 Winbind，因为 Ubuntu 没有诸如 RHEL 中的 `authconfig` 和 SUSE 中的 yast2 这类工具。

打开 **/etc/samba/smb.conf** 并配置以下设置：

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

**WORKGROUP** 是 **REALM** 中的第一个字段，**REALM** 是大写的 Kerberos 领域名称。

配置 **nsswitch**　打开 **/etc/nsswitch.conf** 并将 `winbind` 附加到以下行：

```
passwd: compat winbind
group: compat winbind
```

加入 **Windows** 域　您的域控制器必须可访问，而且您必须具有有权将计算机添加到域的 Active Directory 用户帐户。

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

其中，**REALM** 是大写的 Kerberos 领域名称，**user** 是有权将计算机添加到域的域用户。

重新启动 **winbind**

```
sudo systemctl restart winbind
```

为 **Winbind** 配置 **PAM**　运行以下命令，确保选中 **Winbind NT/Active Directory authentication**（Winbind NT/Active Directory 身份验证）和 **Create home directory on login**（在登录时创建主目录）选项：

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

> 提示：
>
> 仅当计算机加入域后，`winbind` 守护程序才会始终保持运行。

验证域成员身份　Delivery Controller 要求所有 VDA 计算机（不论是 Windows 还是 Linux）都要在 Active Directory 中有一个计算机对象。

运行 Samba 的 **net ads** 命令验证计算机是否已加入域：

```
sudo net ads testjoin
```

运行以下命令验证额外的域和计算机对象信息：

```
sudo net ads info
```

验证 **Kerberos** 配置　要验证 Kerberos 是否已正确配置为可与 Linux VDA 配合使用，请验证系统 **keytab** 文件是否已创建并包含有效密钥：

```
sudo klist -ke
```

此命令显示各种主体名称与密码套件组合可用的密钥列表。运行 Kerberos `kinit` 命令，使用这些密钥向域控制器验证计算机的身份：

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

计算机和领域名称必须指定为大写。美元符号 ($) 必须使用反斜杠 (\) 进行转义，以免发生 shell 替换。在某些环境中，DNS 域名与 Kerberos 领域名称不同。请确保使用领域名称。如果此命令成功运行，则不会显示任何输出。

使用以下命令验证计算机帐户的 TGT 票据已缓存：

```
sudo klist
```

使用以下命令检查计算机的帐户详细信息：

```
sudo net ads status
```

验证用户身份验证　使用 **wbinfo** 工具验证是否可向域验证域用户的身份：

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

这里指定的域为 AD 域名，而不是 Kerberos 领域名称。对于 bash shell，必须使用另一个反斜杠对反斜杠 (\) 字符进行转义。此命令返回一条成功或失败消息。

要验证 Winbind PAM 模块是否已正确配置，请使用以前未使用的域用户帐户登录 Linux VDA。

```
ssh localhost -l domain\username

id -u
```

> 注意：
>
> 要成功运行 SSH 命令，请确保 SSH 已启用并正常运行。

验证是否为 **id -u** 命令返回的 uid 创建了对应的 Kerberos 凭据缓存文件：

```
ls /tmp/krb5cc_uid
```

验证用户 Kerberos 凭据缓存中的票据是否有效且未过期：

```
klist
```

退出会话。

```
exit
```

直接登录 Gnome 或 KDE 控制台也可以执行类似的测试。在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

> 提示：
>
> 如果使用域帐户登录时成功执行用户身份验证，但无法显示您的桌面，请重新启动计算机并重试。

### Quest Authentication Service

在域控制器上配置 **Quest**　假定您已在 Active Directory 域控制器上安装并配置了 Quest 软件，而且已获得管理权限，有权在 Active Directory 中创建计算机对象。

允许域用户登录 **Linux VDA** 计算机　为了让域用户能够在 Linux VDA 计算机上建立 HDX 会话：

1. 在 Active Directory 用户和计算机管理控制台中，为该用户帐户打开 Active Directory 用户属性。

2. 选择 **Unix Account**（Unix 帐户）选项卡。
3. 选中 **Unix-enabled**（已启用 Unix）。
4. 将 **Primary GID Number**（首选 GID 编号）设置为实际域用户组的组 ID。

> 注意：
>
> 这些说明相当于设置域用户，以便他们可以使用控制台、RDP、SSH 或任何其他远程协议进行登录。

在 **Linux VDA** 上配置 **Quest**

**SELinux** 策略强制实施解决方法　默认 RHEL 环境会强制实施 SELinux。此强制功能会影响 Quest 使用的 Unix 域套接字 IPC 机制，并阻止域用户登录。

解决此问题的最便捷的方法是禁用 SELinux。以 root 用户身份，编辑 **/etc/selinux/config** 并更改 **SELinux** 设置：

```
SELINUX=disabled
```

此更改要求重新启动计算机：

```
reboot
```

> 重要：
>
> 请谨慎使用此设置。禁用后重新启用 SELinux 策略强制实施会导致完全锁定，即便是对 root 用户和其他本地用户也是如此。

配置 **VAS** 守护程序　必须启用并断开自动续订 Kerberos 票据功能。必须禁用身份验证（脱机登录）功能：

```
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
```

此命令将续订间隔设为 9 小时（32400 秒），即比默认的 10 小时票据生命周期短 1 小时。请在票据生命周期较短的系统上设置较低的值。

配置 **PAM** 和 **NSS**　要启用通过 HDX 进行的域用户登录以及其他服务（例如 su、ssh 和 RDP），请运行以下命令以手动配置 PAM 和 NSS：

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

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

加入 **Windows** 域　使用 Quest `vastool` 命令将 Linux 计算机加入到 Active Directory 域中：

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

user 为有权将计算机加入 Active Directory 域的域用户。domain-name 为域的 DNS 名称，例如 example.com。

验证域成员身份　Delivery Controller 要求所有 VDA 计算机（不论是 Windows 还是 Linux）都要在 Active Directory 中有一个计算机对象。验证 Quest 加入的 Linux 计算机是否位于域中：

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

如果计算机已加入域，此命令会返回域名。如果计算机未加入任何域，则会显示以下错误：

```
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
```

验证用户身份验证　要验证 Quest 是否能够通过 PAM 对域用户进行身份验证，请使用以前未使用的域用户帐户登录 Linux VDA。

```
ssh localhost -l domain\username

id -u
```

验证是否为 **id -u** 命令返回的 UID 创建了对应的 Kerberos 凭据缓存文件：

```
ls /tmp/krb5cc_uid
```

验证 Kerberos 凭据缓存中的票据是否有效且未过期：

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

退出会话。

```
exit
```

在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### Centrify DirectControl

加入 **Windows** 域　安装 Centrify DirectControl Agent 后，请使用 Centrify `adjoin` 命令将 Linux 计算机加入 Active Directory 域：

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

**user** 参数为有权将计算机加入 Active Directory 域的任何 Active Directory 域用户。**domain-name** 参数是将 Linux 计算机加入到的域的名称。

验证域成员身份　Delivery Controller 要求所有 VDA 计算机（不论是 Windows 还是 Linux）都要在 Active Directory 中有一个计算机对象。验证 Centrify 加入的 Linux 计算机是否位于域中：

```
su -

adinfo
```

验证 **Joined to domain** 值是否有效以及 **CentrifyDC mode** 是否返回了 **connected**。如果模式仍然卡在启动状态，则表明 Centrify 客户端遇到了服务器连接或身份验证问题。

使用以下命令可获得更全面的系统和诊断信息：

```
adinfo --sysinfo all

adinfo --diag
```

测试与各种 Active Directory 和 Kerberos 服务的连接。

```
adinfo --test
```

在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### SSSD

配置 **Kerberos**　运行以下命令以安装 Kerberos：

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

要配置 Kerberos，请以 root 用户身份打开 **/etc/krb5.conf** 并配置以下参数：

> 注意：
>
> 根据您的 AD 基础结构配置 Kerberos。以下设置适用于单域、单林模型。

```
[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*

此上下文中的 domain-dns-name 参数为 DNS 域名，例如 example.com。*REALM* 是大写的 Kerberos 领域名称，例如 EXAMPLE.COM。

加入域　必须将 SSSD 配置为使用 Active Directory 作为其身份提供程序并且使用 Kerberos 进行身份验证。但是，SSSD 并不提供用于加入域和管理系统 keytab 文件的 AD 客户端功能。可以改为使用 adcli、realmd 或 Samba。

> 注意：
>
> 本部分内容仅提供 adcli 和 Samba 的信息。

使用 **adcli** 加入域：

安装 **adcli**：

安装所需的软件包：

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

通过 **adcli** 加入域：

使用以下命令删除旧系统 keytab 文件并加入域：

```
su -

rm -rf /etc/krb5.keytab

adcli join domain-dns-name -U user -H hostname-fqdn
```

**user** 是有权将计算机添加到域的域用户。**hostname-fqdn** 是计算机的 FQDN 格式的主机名。

需要 **-H** 选项，adcli 才能生成格式为 host/hostname-fqdn@REALM 的 SPN（Linux VDA 要求使用此格式）。

验证系统 **keytab**：

对于 Ubuntu 20.04 计算机，请运行 adcli testjoin 命令以测试其是否已加入域。

对于 Ubuntu 18.04 或 Ubuntu 16.04 计算机，请运行 sudo klist -ket 命令以确保系统 keytab 文件已创建。

验证每个键的时间戳是否与将计算机加入域的时间相匹配。

使用 **Samba** 加入域：

安装软件包：

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

配置 **Samba**：

打开 **/etc/samba/smb.conf** 并配置以下设置：

```
[global]
workgroup = WORKGROUP
security = ADS
realm = REALM
client signing = yes
client use spnego = yes
kerberos method = secrets and keytab
```

**WORKGROUP** 是 **REALM** 中的第一个字段，**REALM** 是大写的 Kerberos 领域名称。

使用 **Samba** 加入域：

您的域控制器必须可访问，而且您必须具有有权将计算机添加到域的 Windows 帐户。

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

其中，**REALM** 是大写的 Kerberos 领域名称，**user** 是有权将计算机添加到域的域用户。

设置 **SSSD** 安装或更新所需软件包：

如果尚未安装，请安装所需的 SSSD 和配置软件包：

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

如果已安装软件包，则建议进行更新：

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

> 注意：
>
> 默认情况下，Ubuntu 中的安装过程将自动配置 **nsswitch.conf** 和 PAM 登录模块。

配置 **SSSD** 启动 SSSD 守护程序之前，需要更改 SSSD 配置。对于某些版本的 SSSD，默认不安装 **/etc/sssd/sssd.conf** 配置文件，必须手动创建。以 root 用户身份创建或打开 **/etc/sssd/sssd.conf** 并配置以下设置：

```
[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
```

> 注意：
>
> ldap_id_mapping 设置为 **true**，以便 SSSD 本身负责将 Windows SID 映射到 Unix UID。否则，Active Directory 必须能够提供 POSIX 扩展程序。PAM 服务 `ctxhdx` 已添加到 ad_gpo_map_remote_interactive。
>
> 此上下文中的 **domain-dns-name** 参数为 DNS 域名，例如 example.com。**REALM** 是大写的 Kerberos 领域名称，例如 EXAMPLE.COM。不需要配置 NetBIOS 域名。
>
> 有关配置设置的信息，请参阅 sssd.conf 和 `sssd-ad` 的手册页。

SSSD 守护程序要求配置文件必须仅具有所有者读取权限：

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

启动 **SSSD** 守护程序　运行以下命令立即启动 SSSD 守护程序，以及使守护程序在计算机启动时启动：

```
sudo systemctl start sssd

sudo systemctl enable sssd
```

**PAM 配置** 运行以下命令，确保选中 **SSS authentication**（SSS 身份验证）和 **Create home directory on login**（在登录时创建主目录）选项：

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

验证域成员身份 Delivery Controller 要求所有 VDA 计算机（Windows 和 Linux）都要在 Active Directory 中有一个计算机对象。

使用 **adcli** 验证域成员关系：

通过运行以下命令显示域信息：

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

使用 **Samba** 验证域成员关系：

运行 Samba 的 **net ads** 命令验证计算机是否已加入域：

```
sudo net ads testjoin
```

运行以下命令验证额外的域和计算机对象信息：

```
sudo net ads info
```

验证 **Kerberos** 配置 要验证 Kerberos 是否已正确配置为可与 Linux VDA 配合使用，请验证系统 keytab 文件是否已创建并包含有效密钥：

```
sudo klist -ke
```

此命令显示各种主体名称与密码套件组合可用的密钥列表。运行 Kerberos `kinit` 命令，使用这些密钥向域控制器验证计算机的身份：

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

计算机和领域名称必须指定为大写。美元符号 ($) 必须使用反斜杠 (\) 进行转义，以免发生 shell 替换。在某些环境中，DNS 域名与 Kerberos 领域名称不同。请确保使用领域名称。如果此命令成功运行，则不会显示任何输出。

使用以下命令验证计算机帐户的 TGT 已缓存：

```
sudo klist
```

验证用户身份验证　SSSD 不直接通过守护程序提供用于测试身份验证的命令行工具，只能通过 PAM 完成。

要验证 SSSD PAM 模块是否已正确配置，请使用以前未使用的域用户帐户登录 Linux VDA。

```
ssh localhost -l domain\username

id -u

klist

exit
```

验证 **klist** 命令返回的 Kerberos 票据是否适用于该用户并且尚未过期。

以 root 用户身份，验证是否已为前面的 **id -u** 命令返回的 uid 创建相应的票据缓存文件：

```
ls /tmp/krb5cc_uid
```

可以通过登录 KDE 或 Gnome Display Manager 执行类似的测试。在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### PBIS

下载所需的 **PBIS** 软件包　例如：

```
sudo wget https://github.com/BeyondTrust/pbis-open/releases/download
    /8.8.0/pbis-open-8.8.0.506.linux.x86_64.deb.sh
```

使 **PBIS** 安装脚本可执行　例如：

```
sudo chmod +x pbis-open-8.8.0.506.linux.x86_64.deb.sh
```

运行 **PBIS** 安装脚本　例如：

```
sudo sh pbis-open-8.8.0.506.linux.x86_64.deb.sh
```

加入 **Windows** 域　您的域控制器必须可访问，而且您必须具有有权将计算机添加到域的 Active Directory 用户帐户。

```
sudo /opt/pbis/bin/domainjoin-cli join domain-name user
```

**user** 为有权将计算机加入 Active Directory 域的域用户。**domain-name** 为域的 DNS 名称，例如 example.com。

注意：要将 Bash 设置为默认 shell，请运行 **sudo /opt/pbis/bin/config LoginShellTemplate/bin/bash** 命令。

验证域成员身份　Delivery Controller 要求所有 VDA 计算机（Windows 和 Linux）都要在 Active Directory 中有一个计算机对象。验证加入了 PBIS 的 Linux 计算机是否位于域中：

```
/opt/pbis/bin/domainjoin-cli query
```

如果计算机已加入某个域，此命令将返回有关当前加入的 AD 域和 OU 的信息。否则，仅显示主机名。

验证用户身份验证　要验证 PBIS 是否能够通过 PAM 对域用户进行身份验证，请使用以前未使用的域用户帐户登录 Linux VDA。

```
sudo ssh localhost -l domain\user

id -u
```

验证是否为 **id -u** 命令返回的 UID 创建了对应的 Kerberos 凭据缓存文件：

```
ls /tmp/krb5cc_uid
```

退出会话。

```
exit
```

在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### 步骤 **4**：安装 **Linux VDA**

#### 步骤 **4a**：下载 **Linux VDA** 软件包

转至 Citrix Virtual Apps and Desktops 下载页面。展开相应版本的 Citrix Virtual Apps and Desktops，然后单击组件以下载与 Linux 发行版匹配的 Linux VDA 包。

#### 步骤 **4b**：安装 **Linux VDA**

使用 Debian 软件包管理器安装 Linux VDA 软件：

对于 **Ubuntu 20.04**：

```
sudo dpkg -i xendesktopvda_<version>.ubuntu20.04_amd64.deb
```

对于 **Ubuntu 18.04**：

```
sudo dpkg -i xendesktopvda_<version>.ubuntu18.04_amd64.deb
```

对于 **Ubuntu 16.04**：

```
sudo dpkg -i xendesktopvda_<version>.ubuntu16.04_amd64.deb
```

Ubuntu 20.04 的 Debian 依赖项列表：

```
postgresql >= 12

libpostgresql-jdbc-java >= 42.2

openjdk-8-jdk >= 8u252

imagemagick >= 8:6.9.10

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

libgoogle-perftools4 >= 2.7~

libpython2.7 >= 2.7~
```

Ubuntu 18.04 的 Debian 依赖项列表：

```
postgresql >= 9.5

libpostgresql-jdbc-java >= 9.2

openjdk-8-jdk >= 1.8.0

gtk3-nocsd >=3

imagemagick >= 8:6.8.9.9

ufw >= 0.35

ubuntu-desktop >= 1.361

```

```
libxrandr2 >= 2:1.5.0

libxtst6 >= 2:1.2.2

libxm4 >= 2.3.4

util-linux >= 2.27.1

bash >= 4.3

findutils >= 4.6.0

sed >= 4.2.2

cups >= 2.1

libldap-2.4-2 >= 2.4.42

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

python-requests >= 2.9.1

libgoogle-perftools4 >= 2.4~

xserver-xorg-core >= 2:1.18

xserver-xorg-core << 2:1.19

x11vnc>=0.9.13

python-websockify >= 0.6.1
```

Ubuntu 16.04 的 Debian 依赖项列表：

```
postgresql >= 9.5

libpostgresql-jdbc-java >= 9.2

default-jdk >= 2:1.8

imagemagick >= 8:6.8.9.9

ufw >= 0.35

ubuntu-desktop >= 1.361

libxrandr2 >= 2:1.5.0

libxtst6 >= 2:1.2.2

libxm4 >= 2.3.4

util-linux >= 2.27.1
```

```

bash >= 4.3

findutils >= 4.6.0

sed >= 4.2.2

cups >= 2.1

libldap-2.4-2 >= 2.4.42

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

python-requests >= 2.9.1

libgoogle-perftools4 >= 2.4~

xserver-xorg-core >= 2:1.18

xserver-xorg-core << 2:1.19

x11vnc>=0.9.13

python-websockify >= 0.6.1
```

> 注意：
>
> 有关此版本的 Linux VDA 支持的 Linux 发行版和 Xorg 版本的列表，请参阅系统要求。

#### 步骤 **4c**：升级 **Linux VDA**（可选）

可以从先前的两个版本和 LTSR 版本升级现有安装。

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

#### 步骤 **4d**：配置 **Linux VDA**

安装软件包后，必须运行 ctxsetup.sh 脚本来配置 Linux VDA。执行任何更改之前，该脚本都会验证环境，确保所有依赖项都已安装。如有必要，可以随时重新运行该脚本以更改设置。

可以按照提示手动运行脚本，也可以采用预先配置的响应自动运行脚本。继续操作前，请查看该脚本的帮助信息：

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

提示配置　运行会提示各种问题的手动配置：

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

自动配置　自动安装时，设置脚本所需的选项可由环境变量提供。如果所需的所有变量都存在，则脚本不会提示用户提供任何信息，从而允许通过脚本完成安装过程。

支持的环境变量包括：

- **CTX_XDL_SUPPORT_DDC_AS_CNAME = Y | N** –Linux VDA 支持使用 DNS CNAME 记录指定 Delivery Controller 名称。默认情况下设置为 N。
- **CTX_XDL_DDC_LIST=** ‘**list-ddc-fqdns**’ –Linux VDA 要求提供由空格分隔的 Delivery Controller 完全限定域名 (FQDN) 列表以用于向 Delivery Controller 注册。必须至少指定一个 FQDN 或 CNAME 别名。
- **CTX_XDL_VDA_PORT=port-number** –Linux VDA 通过 TCP/IP 端口（默认为端口 80）与 Delivery Controller 通信。
- **CTX_XDL_REGISTER_SERVICE = Y | N** –在启动计算机后启动 Linux Virtual Desktop 服务。默认情况下设置为 Y。
- **CTX_XDL_ADD_FIREWALL_RULES=Y | N** –Linux Virtual Desktop 服务要求允许传入网络连接通过系统防火墙。您可以在系统防火墙中自动为 Linux Virtual Desktop 打开所需端口（默认为端口 80 和 1494）。默认情况下设置为 Y。
- **CTX_XDL_AD_INTEGRATION=1 | 2 | 3 | 4 |5** –Linux VDA 要求使用 Kerberos 配置设置向 Delivery Controller 进行身份验证。Kerberos 配置根据系统上已安装和已配置的 Active Directory 集成工具确定。指定要使用且受支持的 Active Directory 集成方法：
  - 1 –Samba Winbind
  - 2 –Quest Authentication Service
  - 3 - Centrify DirectControl
  - 4 - SSSD
  - 5 –PBIS
- **CTX_XDL_HDX_3D_PRO=Y | N** –Linux VDA 支持 HDX 3D Pro，这是一组 GPU 加速技术，旨在优化富图形应用程序的虚拟化水平。如果选择了 HDX 3D Pro，则要为 VDI 桌面（单会话）模式配置 VDA -（即 CTX_XDL_VDI_MODE=Y）。
- **CTX_XDL_VDI_MODE=Y | N** –将计算机配置为专用桌面交付模型 (VDI) 还是托管共享桌面交付模型。对于 HDX 3D Pro 环境，将此变量设置为 Y。默认情况下，此变量设置为 N。
- **CTX_XDL_SITE_NAME=dns-name** –Linux VDA 通过 DNS 发现 LDAP 服务器。要将 DNS 搜索结果限制为本地站点，应指定 DNS 站点名称。默认情况下，此变量设置为 **<none>**。
- **CTX_XDL_LDAP_LIST=** ‘**list-ldap-servers**’ –Linux VDA 查询 DNS 来发现 LDAP 服务器。如果 DNS 无法提供 LDAP 服务记录，您可以提供以空格分隔的 LDAP FQDN（带有 LDAP 端口）列表。例如 ad1.mycompany.com:389。默认情况下，此变量设置为 **<none>**。
- **CTX_XDL_SEARCH_BASE=search-base-set** –Linux VDA 通过设置为 Active Directory 域根的搜索库来查询 LDAP（例如，DC=mycompany,DC=com）。但是，为提高搜索效能，可以指定搜索基础（例如 OU=VDI,DC=mycompany,DC=com）。默认情况下，此变量设置为 **<none>**。
- **CTX_XDL_FAS_LIST=** ‘**list-fas-servers**’ –联合身份验证服务 (FAS) 服务器是通过 AD 组策略配置的。Linux VDA 不支持 AD 组策略，但您可以改为提供以分号分隔的 FAS 服务器的列表。顺序必须与在 AD 组策略中配置的顺序相同。如果删除了任何服务器地址，请使用 **<none>** 文本字符串填充其空白，并且不要修改服务

器地址的顺序。

- **CTX_XDL_DOTNET_ RUNTIME_PATH=path-to-install-dotnet-runtime** – 安装.NET Core Runtime 3.1 以支持新的 Broker 代理服务 (`ctxvda`) 的路径。默认路径为 /usr/bin。
- **CTX_XDL_START_SERVICE=Y | N** –在完成 Linux VDA 配置后，是否启动 Linux VDA 服务。默认情况下设置为 Y。
- **CTX_XDL_TELEMETRY_SOCKET_PORT** - 用于侦听 Citrix Scout 的套接字端口。默认端口为 7503。
- **CTX_XDL_TELEMETRY_PORT** - 用于与 Citrix Scout 通信的端口。默认端口为 7502。

设置环境变量并运行配置脚本：

```
export CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N

export CTX_XDL_DDC_LIST= ‘list-ddc-fqdns’

export CTX_XDL_VDA_PORT=port-number

export CTX_XDL_REGISTER_SERVICE=Y|N

export CTX_XDL_ADD_FIREWALL_RULES=Y|N

export CTX_XDL_AD_INTEGRATION=1|2|3|4|5

export CTX_XDL_HDX_3D_PRO=Y|N

export CTX_XDL_VDI_MODE=Y|N

export CTX_XDL_SITE_NAME=dns-site-name | '<none>'

export CTX_XDL_LDAP_LIST= ‘list-ldap-servers’  | '<none>'

export CTX_XDL_SEARCH_BASE=search-base-set | '<none>'

export CTX_XDL_FAS_LIST= ‘list-fas-servers’  | '<none>'

export CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime

export CTX_XDL_TELEMETRY_SOCKET_PORT=port-number

export CTX_XDL_TELEMETRY_PORT=port-number

export CTX_XDL_START_SERVICE=Y|N

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

运行 sudo 命令时，键入 **-E** 选项以将现有环境变量传递给它创建的新 shell。我们建议您使用前面的命令并加上 **#!/bin/bash** 作为第一行来创建 shell 脚本文件。

或者，您可以使用单个命令指定所有参数：

```
sudo CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N \

```

```
CTX_XDL_DDC_LIST= ‘list-ddc-fqdns’  \

CTX_XDL_VDA_PORT=port-number \

CTX_XDL_REGISTER_SERVICE=Y|N \

CTX_XDL_ADD_FIREWALL_RULES=Y|N \

CTX_XDL_AD_INTEGRATION=1|2|3|4|5 \

CTX_XDL_HDX_3D_PRO=Y|N \

CTX_XDL_VDI_MODE=Y|N \

CTX_XDL_SITE_NAME=dns-name \

CTX_XDL_LDAP_LIST= ‘list-ldap-servers’  \

CTX_XDL_SEARCH_BASE=search-base-set \

CTX_XDL_FAS_LIST= ‘list-fas-servers’  \

CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime \

CTX_XDL_TELEMETRY_SOCKET_PORT=port-number \

CTX_XDL_TELEMETRY_PORT=port-number \

CTX_XDL_START_SERVICE=Y|N \

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

删除配置更改　在某些情形下，您可能需要删除 **ctxsetup.sh** 脚本对配置所做的更改，但不卸载 Linux VDA 软件包。

继续操作前，请查看此脚本的帮助信息：

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

删除配置更改：

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

> 重要：
>
> 此脚本会从数据库删除所有配置数据，从而使 Linux VDA 无法使用。

配置日志　**ctxsetup.sh** 和 **ctxcleanup.sh** 脚本会在控制台上显示错误，并将其他信息写入配置日志文件 **/tmp/xdl.configure.log**：

重新启动 Linux VDA 服务，确保更改生效。

卸载 **Linux VDA** 软件　要检查 Linux VDA 是否已安装并查看已安装软件包的版本，请运行以下命令：

```
dpkg -l xendesktopvda
```

查看更多详细信息：

```
apt-cache show xendesktopvda
```

要卸载 Linux VDA 软件，请执行以下操作：

```
dpkg -r xendesktopvda
```

> 注意：
>
> 卸载 Linux VDA 软件会删除关联的 PostgreSQL 和其他配置数据。但是，不会删除在安装 Linux VDA 之前设置的 PostgreSQL 软件包和其他依赖软件包。
>
> 提示：
>
> 本节中的信息未介绍包括 PostgreSQL 在内的依赖软件包的删除操作。

### 步骤 5：运行 Linux VDA

使用 **ctxsetup.sh** 脚本配置 Linux VDA 后，请使用以下命令控制 Linux VDA。

启动 **Linux VDA**：

启动 Linux VDA 服务：

```
sudo systemctl start ctxhdx

sudo systemctl start ctxvda
```

停止 **Linux VDA**：

停止 Linux VDA 服务：

```
sudo systemctl stop ctxvda

sudo systemctl stop ctxhdx
```

> 注意：
>
> 在停止 `ctxvda` 和 `ctxhdx` 服务之前，请运行 `service ctxmonitorservice stop` 命令以停止监视服务守护程序。否则，监视服务守护程序将重新启动您停止的服务。

重新启动 **Linux VDA**：

重新启动 Linux VDA 服务：

```
sudo systemctl stop ctxvda

sudo systemctl restart ctxhdx

sudo systemctl restart ctxvda
```

检查 **Linux VDA** 状态：

要检查 Linux VDA 服务的运行状态，请执行以下操作：

```
sudo systemctl status ctxvda

sudo systemctl status ctxhdx
```

### 步骤 6：在 Citrix Virtual Apps 或 Citrix Virtual Desktops 中创建计算机目录

创建计算机目录和添加 Linux VDA 计算机的过程与传统的 Windows VDA 方法类似。有关如何完成这些任务的更加详细的说明，请参阅创建计算机目录和管理计算机目录。

创建包含 Linux VDA 计算机的计算机目录时会面临一些限制，使得该过程不同于为 Windows VDA 计算机创建计算机目录：

- 对于操作系统，请选择：
  - 多会话操作系统选项（对于托管共享桌面交付模型）。
  - 单会话操作系统选项（对于 VDI 专用桌面交付模型）。
- 请勿在同一个计算机目录中混合使用 Linux 和 Windows VDA 计算机。

> 注意：
>
> 早期版本的 Citrix Studio 不支持“Linux 操作系统”的概念。但是，选择 **Windows Server** 操作系统或服务器操作系统选项等同于使用托管共享桌面交付模型。选择 **Windows** 桌面操作系统或桌面操作系统选项等同于使用每计算机一个用户交付模型。
>
> 提示：
>
> 如果删除计算机后将其重新加入 Active Directory 域，则必须删除计算机，然后将其重新添加到计算机目录。

### 步骤 7：在 Citrix Virtual Apps 或 Citrix Virtual Desktops 中创建交付组

创建交付组和添加包含 Linux VDA 计算机的计算机目录的过程与 Windows VDA 计算机几乎相同。有关如何完成这些任务的更加详细的说明，请参阅创建交付组。

创建含有 Linux VDA 计算机目录的交付组时会面临以下限制：

- 确保所选的 AD 用户和组已正确配置，可以登录到 Linux VDA 计算机。

- 请勿允许未经身份验证的（匿名）用户登录。
- 请勿在交付组中混入含有 Windows 计算机的计算机目录。

有关如何创建计算机目录和交付组的信息，请参阅 Citrix Virtual Apps and Desktops 7 2103。

## 安装 Linux Virtual Delivery Agent for Debian

June 21, 2022

可以选择按照本文中的步骤进行手动安装，也可以使用轻松安装进行自动安装和配置。轻松安装省时又省力，与手动安装相比，更不易于出错。

> 注意：
>
> 请仅对全新安装使用轻松安装功能。请勿使用轻松安装更新现有安装。

### 步骤 1：准备 Debian 以便进行 VDA 安装

#### 步骤 1a：验证网络配置

我们建议您先连接并正确配置网络，然后再继续操作。

#### 步骤 1b：设置主机名

为确保正确报告计算机的主机名，请更改 **/etc/hostname** 文件，使其仅包含计算机的主机名。

```
hostname
```

#### 步骤 1c：为主机名分配环回地址

确保计算机的 DNS 域名和完全限定域名 (FQDN) 正确地返回报告。方法是更改 **/etc/hosts** 文件的以下行以包含 FQDN 和主机名作为前两个条目：

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

例如：

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

从文件中的其他条目中删除对 **hostname-fqdn** 或 **hostname** 的任何其他引用。

> 注意：
>
> Linux VDA 当前不支持 NetBIOS 名称截断。因此，主机名不得超过 15 个字符。
>
> 提示：
>
> 只能使用字符 a-z、A-Z、0-9 和连字符 (-)。请避免使用下划线 (_)、空格和其他符号。主机名不得以数字开头和以连字符结尾。此规则也适用于 Delivery Controller 主机名。

#### 步骤 **1d**：检查主机名

验证主机名设置是否正确无误：

```
hostname
```

此命令仅返回计算机的主机名，而不返回其 FQDN。

验证 FQDN 设置是否正确无误：

```
hostname -f
```

此命令返回计算机的 FQDN。

#### 步骤 **1e**：禁用多播 **DNS**

默认设置启用了多播 DNS (**mDNS**)，而这会导致名称解析结果不一致。

要禁用 **mDNS**，请编辑 **/etc/nsswitch.conf** 并更改包含以下内容的行：

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

更改为：

```
hosts: files dns
```

#### 步骤 **1f**：检查名称解析和服务可访问性

确认可以解析 FQDN 并对域控制器和 Delivery Controller 执行 ping 操作：

```
nslookup domain-controller-fqdn

ping domain-controller-fqdn

nslookup delivery-controller-fqdn

ping delivery-controller-fqdn
```

如果无法解析 FQDN 或 Ping 不通上述任一计算机，请先检查相关步骤，然后再继续。

#### 步骤 **1g**：配置时钟同步 **(chrony)**

确保 VDA、Delivery Controller 和域控制器之间的时钟始终精确同步至关重要。将 Linux VDA 托管为虚拟机可能会导致时钟偏差问题。出于此原因，最好使用远程时间服务来同步时间。

安装 chrony：

```
apt-get install chrony
```

以 root 用户身份，编辑 **/etc/chrony/chrony.conf** 并为每个远程时间服务器添加一个服务器条目：

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

在典型部署中，时间从本地域控制器同步，而不是直接从公共 NTP 池服务器同步。为域中的每个 Active Directory 域控制器添加一个服务器条目。

删除列出的任何其他 **server** 或 **pool** 条目，包括环回 IP 地址、localhost 和公共服务器 ***.pool.ntp.org** 条目。

保存更改并重新启动 Chrony 守护程序：

```
sudo systemctl restart chrony
```

#### 步骤 **1h**：安装软件包

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

sudo apt-get install -y libgtk2.0-0
```

#### 步骤 **1i**：添加旧版稳定的存储库

要为 Debian 发行版安装必要的依赖项，请将 `deb http://deb.debian.org/debian/ oldstable main` 行添加到 `/etc/apt/sources.list` 文件。

#### 步骤 **2**：准备虚拟机管理程序

在支持的虚拟机管理程序上将 Linux VDA 当作虚拟机运行时，需要作出一些更改。根据使用的虚拟机管理程序平台作出以下更改。如果正在裸机硬件上运行 Linux 计算机，则无需作出任何更改。

#### 修复 **Citrix Hypervisor** 上的时间同步问题

启用了 Citrix Hypervisor 时间同步功能时，在每个半虚拟化 Linux VM 中，您会发现 NTP 和 Citrix Hypervisor 都尝试管理系统时钟。为避免时钟与其他服务器不同步，请确保每个 Linux 客户机中的系统时钟都与 NTP 同步。这种情况要求禁用主机时间同步。无需在 HVM 模式下进行任何更改。

在某些 Linux 发行版中，如果正在运行半虚拟化 Linux 内核，并安装了 Citrix VM Tools，您可以检查 Citrix Hypervisor 时间同步功能是否存在，以及是否已在 Linux VM 中启用：

```
su -

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

此命令返回 0 或 1：

- 0 - 时间同步功能已启用，且必须禁用。
- 1 - 时间同步功能已禁用，无需采取任何操作。

如果 /proc/sys/xen/independent_wallclock 文件不存在，则不需要执行以下步骤。

如果已启用，请通过向该文件写入 1 以禁用时间同步功能：

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

要使此更改成为永久更改，并在重新启动后仍然有效，请编辑 **/etc/sysctl.conf** 文件并添加以下行：

```
xen.independent_wallclock = 1
```

要验证这些更改，请重新启动系统：

```
su -

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

此命令返回值 1。

#### 在 **Microsoft Hyper-V** 上修复时间同步问题

安装了 Hyper-V Linux 集成服务的 Linux VM 可以使用 Hyper-V 时间同步功能来使用主机操作系统的时间。为确保系统时钟始终精确可靠，请一同启用此功能与 NTP 服务。

从管理操作系统中：

1. 打开 Hyper-V 管理器控制台。
2. 对于 Linux VM 的设置，请选择 **Integration Services**（集成服务）。
3. 确保已选择 **Time synchronization**（时间同步）。

> 注意：
>
> 此方法与 VMware 和 Citrix Hypervisor 不同，这两种产品会禁用主机时间同步功能，以免与 NTP 发生冲突。Hyper-V 时间同步可以与 NTP 时间同步共存并互补。

修复 **ESX** 和 **ESXi** 上的时间同步问题

启用了 VMware 时间同步功能时，在每个半虚拟化 Linux VM 中，您会发现 NTP 和虚拟机管理程序都尝试同步系统时钟。为避免时钟与其他服务器不同步，请确保每个 Linux 客户机中的系统时钟都与 NTP 同步。这种情况要求禁用主机时间同步。

如果正在运行安装了 VMware Tools 的半虚拟化 Linux 内核：

1. 打开 vSphere Client。
2. 编辑 Linux VM 设置。
3. 在 **Virtual Machine Properties**（虚拟机属性）对话框中，打开 **Options**（选项）选项卡。
4. 选择 **VMware Tools**。
5. 在 **Advanced**（高级）框中，取消选中 **Synchronize guest time with host**（与主机同步客户机时间）。

## 步骤 3：向 Windows 域中添加 Linux 虚拟机 (VM)

Linux VDA 支持多种向 Active Directory (AD) 域添加 Linux 计算机的方法：

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

根据所选的方法，按说明执行操作。

> 注意：
>
> 为 Linux VDA 中的本地帐户和 AD 中的帐户使用相同的用户名时，会话启动可能会失败。

### Samba Winbind

安装或更新所需软件包

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

在计算机启动时启用要启动的 **Winbind** 守护程序　Winbind 守护程序必须配置为在计算机启动时启动：

```
sudo systemctl enable winbind
```

> 注意：
>
> 确保 winbind 脚本位于 `/etc/init.d` 下。

配置 **Kerberos**　以 root 用户身份，打开 **/etc/krb5.conf** 并配置以下设置：

> 注意：
>
> 根据您的 AD 基础结构配置 Kerberos。以下设置适用于单域、单林模型。

```
[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
```

此上下文中的 **domain-dns-name** 参数为 DNS 域名，例如 **example.com**。**REALM** 是大写的 Kerberos 领域名称，例如 **EXAMPLE.COM**。

配置 **Winbind** 身份验证　打开 **/etc/samba/smb.conf** 并配置以下设置：

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

**WORKGROUP** 是 **REALM** 中的第一个字段，**REALM** 是大写的 Kerberos 领域名称。

**配置 nsswitch** 打开 **/etc/nsswitch.conf** 并将 `winbind` 附加到以下行：

```
passwd: systemd winbind
group: systemd winbind
```

**加入 Windows 域** 您的域控制器必须可访问，而且您必须具有有权将计算机添加到域的 Active Directory 用户帐户。

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

其中，**REALM** 是大写的 Kerberos 领域名称，**user** 是有权将计算机添加到域的域用户。

**重新启动 Winbind**

```
sudo systemctl restart winbind
```

**为 Winbind 配置 PAM** 运行以下命令，确保选中 **Winbind NT/Active Directory authentication**（Winbind NT/Active Directory 身份验证）和 **Create home directory on login**（在登录时创建主目录）选项：

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

> 提示：
>
> 仅当计算机加入域后，`winbind` 守护程序才会始终保持运行。

**验证域成员身份** Delivery Controller 要求所有 VDA 计算机（不论是 Windows 还是 Linux）都要在 Active Directory 中有一个计算机对象。

运行 Samba 的 **net ads** 命令验证计算机是否已加入域：

```
sudo net ads testjoin
```

运行以下命令验证额外的域和计算机对象信息：

```
sudo net ads info
```

**验证 Kerberos 配置** 要验证 Kerberos 是否已正确配置为可与 Linux VDA 配合使用，请验证系统 **keytab** 文件是否已创建并包含有效密钥：

```
sudo klist -ke
```

此命令显示各种主体名称与密码套件组合可用的密钥列表。运行 Kerberos `kinit` 命令，使用这些密钥向域控制器验证计算机的身份：

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

计算机和领域名称必须指定为大写。美元符号 ($) 必须使用反斜杠 (\) 进行转义，以免发生 shell 替换。在某些环境中，DNS 域名与 Kerberos 领域名称不同。请确保使用领域名称。如果此命令成功运行，则不会显示任何输出。

使用以下命令验证计算机帐户的 TGT 票据已缓存：

```
sudo klist
```

使用以下命令检查计算机的帐户详细信息：

```
sudo net ads status
```

验证用户身份验证　使用 **wbinfo** 工具验证是否可向域验证域用户的身份：

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

这里指定的域为 AD 域名，而不是 Kerberos 领域名称。对于 bash shell，必须使用另一个反斜杠对反斜杠 (\) 字符进行转义。此命令返回一条成功或失败消息。

要验证 Winbind PAM 模块是否已正确配置，请使用以前未使用的域用户帐户登录 Linux VDA。

```
ssh localhost -l domain\username

id -u
```

> 注意：
>
> 要成功运行 SSH 命令，请确保 SSH 已启用并正常运行。

验证是否为 **id -u** 命令返回的 uid 创建了对应的 Kerberos 凭据缓存文件：

```
ls /tmp/krb5cc_uid
```

验证用户 Kerberos 凭据缓存中的票据是否有效且未过期：

```
klist
```

退出会话。

```
exit
```

直接登录 Gnome 或 KDE 控制台也可以执行类似的测试。在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

> 提示：
>
> 如果使用域帐户登录时成功执行用户身份验证，但无法显示您的桌面，请重新启动计算机并重试。

### Quest Authentication Service

在域控制器上配置 **Quest**　假定您已在 Active Directory 域控制器上安装并配置了 Quest 软件，而且已获得管理权限，有权在 Active Directory 中创建计算机对象。

允许域用户登录 **Linux VDA** 计算机　为了让域用户能够在 Linux VDA 计算机上建立 HDX 会话：

1. 在 Active Directory 用户和计算机管理控制台中，为该用户帐户打开 Active Directory 用户属性。
2. 选择 **Unix Account**（Unix 帐户）选项卡。
3. 选中 **Unix-enabled**（已启用 Unix）。
4. 将 **Primary GID Number**（首选 GID 编号）设置为实际域用户组的组 ID。

> 注意：
>
> 这些说明相当于设置域用户，以便他们可以使用控制台、RDP、SSH 或任何其他远程协议进行登录。

在 **Linux VDA** 上配置 **Quest**

**SELinux** 策略强制实施解决方法　默认 RHEL 环境会强制实施 SELinux。此强制功能会影响 Quest 使用的 Unix 域套接字 IPC 机制，并阻止域用户登录。

解决此问题的最便捷的方法是禁用 SELinux。以 root 用户身份，编辑 **/etc/selinux/config** 并更改 **SELinux** 设置：

```
SELINUX=disabled
```

此更改要求重新启动计算机：

```
reboot
```

> 重要：
>
> 请谨慎使用此设置。禁用后重新启用 SELinux 策略强制实施会导致完全锁定，即便是对 root 用户和其他本地用户也是如此。

配置 **VAS** 守护程序　必须启用并断开自动续订 Kerberos 票据功能。必须禁用身份验证（脱机登录）功能：

```
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
```

此命令将续订间隔设为 9 小时（32400 秒），即比默认的 10 小时票据生命周期短 1 小时。请在票据生命周期较短的系统上设置较低的值。

配置 **PAM** 和 **NSS**　要启用通过 HDX 进行的域用户登录以及其他服务（例如 su、ssh 和 RDP），请运行以下命令以手动配置 PAM 和 NSS：

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

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

加入 **Windows** 域　使用 Quest `vastool` 命令将 Linux 计算机加入到 Active Directory 域中：

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

user 为有权将计算机加入 Active Directory 域的域用户。domain-name 为域的 DNS 名称，例如 example.com。

验证域成员身份　Delivery Controller 要求所有 VDA 计算机（不论是 Windows 还是 Linux）都要在 Active Directory 中有一个计算机对象。验证 Quest 加入的 Linux 计算机是否位于域中：

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

如果计算机已加入域，此命令会返回域名。如果计算机未加入任何域，则会显示以下错误：

```
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
```

验证用户身份验证　要验证 Quest 是否能够通过 PAM 对域用户进行身份验证，请使用以前未使用的域用户帐户登录 Linux VDA。

```
ssh localhost -l domain\username

id -u
```

验证是否为 **id -u** 命令返回的 UID 创建了对应的 Kerberos 凭据缓存文件：

```
ls /tmp/krb5cc_uid
```

验证 Kerberos 凭据缓存中的票据是否有效且未过期：

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

退出会话。

```
exit
```

在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### Centrify DirectControl

**加入 Windows 域** 安装 Centrify DirectControl Agent 后，请使用 Centrify `adjoin` 命令将 Linux 计算机加入 Active Directory 域：

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

**user** 参数为有权将计算机加入 Active Directory 域的任何 Active Directory 域用户。**domain-name** 参数是将 Linux 计算机加入到的域的名称。

**验证域成员身份** Delivery Controller 要求所有 VDA 计算机（不论是 Windows 还是 Linux）都要在 Active Directory 中有一个计算机对象。验证 Centrify 加入的 Linux 计算机是否位于域中：

```
su -

adinfo
```

验证 **Joined to domain** 值是否有效以及 **CentrifyDC mode** 是否返回了 **connected**。如果模式仍然卡在启动状态，则表明 Centrify 客户端遇到了服务器连接或身份验证问题。

使用以下命令可获得更全面的系统和诊断信息：

```
adinfo --sysinfo all

adinfo --diag
```

测试与各种 Active Directory 和 Kerberos 服务的连接。

```
adinfo --test
```

在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### SSSD

**配置 Kerberos** 运行以下命令以安装 Kerberos：

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

要配置 Kerberos，请以 root 用户身份打开 **/etc/krb5.conf** 并配置以下参数：

> 注意：
>
> 根据您的 AD 基础结构配置 Kerberos。以下设置适用于单域、单林模型。

```
[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
```

此上下文中的 domain-dns-name 参数为 DNS 域名，例如 example.com。*REALM* 是大写的 Kerberos 领域名称，例如 EXAMPLE.COM。

加入域　必须将 SSSD 配置为使用 Active Directory 作为其身份提供程序并且使用 Kerberos 进行身份验证。但是，SSSD 并不提供用于加入域和管理系统 keytab 文件的 AD 客户端功能。可以改为使用 adcli、realmd 或 Samba。

> 注意：
>
> 本部分内容仅提供 adcli 和 Samba 的信息。

使用 **adcli** 加入域：

安装 **adcli**：

安装所需的软件包：

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

通过 **adcli** 加入域：

使用以下命令删除旧系统 keytab 文件并加入域：

```
su -

rm -rf /etc/krb5.keytab

adcli join domain-dns-name -U user -H hostname-fqdn
```

**user** 是有权将计算机添加到域的域用户。**hostname-fqdn** 是计算机的 FQDN 格式的主机名。

需要 **-H** 选项，adcli 才能生成格式为 host/hostname-fqdn@REALM 的 SPN（Linux VDA 要求使用此格式）。

验证系统 **keytab**：

运行 `sudo klist -ket` 命令以确保系统 keytab 文件已创建。

验证每个键的时间戳是否与将计算机加入域的时间相匹配。

使用 **Samba** 加入域：

安装软件包：

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

配置 **Samba**：

打开 **/etc/samba/smb.conf** 并配置以下设置：

```
[global]
workgroup = WORKGROUP
security = ADS
realm = REALM
client signing = yes
client use spnego = yes
kerberos method = secrets and keytab
```

**WORKGROUP** 是 **REALM** 中的第一个字段，**REALM** 是大写的 Kerberos 领域名称。

使用 **Samba** 加入域：

您的域控制器必须可访问，而且您必须具有有权将计算机添加到域的 Windows 帐户。

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

其中，**REALM** 是大写的 Kerberos 领域名称，**user** 是有权将计算机添加到域的域用户。

设置 **SSSD**　安装或更新所需软件包：

如果尚未安装，请安装所需的 SSSD 和配置软件包：

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

如果已安装软件包，则建议进行更新：

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

> 注意：
>
> 默认情况下，Ubuntu 中的安装过程将自动配置 **nsswitch.conf** 和 PAM 登录模块。

**配置 SSSD** 启动 SSSD 守护程序之前，需要更改 SSSD 配置。对于某些版本的 SSSD，默认不安装 **/etc/sssd/sssd.conf** 配置文件，必须手动创建。以 root 用户身份创建或打开 **/etc/sssd/sssd.conf** 并配置以下设置：

```
[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
```

> 注意：
>
> ldap_id_mapping 设置为 **true**，以便 SSSD 本身负责将 Windows SID 映射到 Unix UID。否则，Active Directory 必须能够提供 POSIX 扩展程序。PAM 服务 `ctxhdx` 已添加到 ad_gpo_map_remote_interactive。
>
> 此上下文中的 **domain-dns-name** 参数为 DNS 域名，例如 example.com。**REALM** 是大写的 Kerberos 领域名称，例如 EXAMPLE.COM。不需要配置 NetBIOS 域名。
>
> 有关配置设置的信息，请参阅 sssd.conf 和 `sssd-ad` 的手册页。

SSSD 守护程序要求配置文件必须仅具有所有者读取权限：

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

启动 **SSSD** 守护程序　运行以下命令立即启动 SSSD 守护程序，以及使守护程序在计算机启动时启动：

```
sudo systemctl start sssd

sudo systemctl enable sssd
```

**PAM** 配置　运行以下命令，确保选中 **SSS authentication**（SSS 身份验证）和 **Create home directory on login**（在登录时创建主目录）选项：

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

验证域成员身份　Delivery Controller 要求所有 VDA 计算机（Windows 和 Linux）都要在 Active Directory 中有一个计算机对象。

使用 **adcli** 验证域成员关系：

通过运行以下命令显示域信息：

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

使用 **Samba** 验证域成员关系：

运行 Samba 的 **net ads** 命令验证计算机是否已加入域：

```
sudo net ads testjoin
```

运行以下命令验证额外的域和计算机对象信息：

```
sudo net ads info
```

验证 **Kerberos** 配置　要验证 Kerberos 是否已正确配置为可与 Linux VDA 配合使用，请验证系统 keytab 文件是否已创建并包含有效密钥：

```
sudo klist -ke
```

此命令显示各种主体名称与密码套件组合可用的密钥列表。运行 Kerberos `kinit` 命令，使用这些密钥向域控制器验证计算机的身份：

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

计算机和领域名称必须指定为大写。美元符号 ($) 必须使用反斜杠 (\) 进行转义，以免发生 shell 替换。在某些环境中，DNS 域名与 Kerberos 领域名称不同。请确保使用领域名称。如果此命令成功运行，则不会显示任何输出。

使用以下命令验证计算机帐户的 TGT 已缓存：

```
sudo klist
```

验证用户身份验证　SSSD 不直接通过守护程序提供用于测试身份验证的命令行工具，只能通过 PAM 完成。

要验证 SSSD PAM 模块是否已正确配置，请使用以前未使用的域用户帐户登录 Linux VDA。

```
ssh localhost -l domain\username

id -u

klist

exit
```

验证 **klist** 命令返回的 Kerberos 票据是否适用于该用户并且尚未过期。

以 root 用户身份，验证是否已为前面的 **id -u** 命令返回的 uid 创建相应的票据缓存文件：

```
ls /tmp/krb5cc_uid
```

可以通过登录 KDE 或 Gnome Display Manager 执行类似的测试。在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### PBIS

下载所需的 **PBIS** 软件包　例如：

```
sudo wget https://github.com/BeyondTrust/pbis-open/releases/download
    /8.8.0/pbis-open-8.8.0.506.linux.x86_64.deb.sh
```

使 **PBIS** 安装脚本可执行　例如：

```
sudo chmod +x pbis-open-8.8.0.506.linux.x86_64.deb.sh
```

运行 **PBIS** 安装脚本　例如：

```
sudo sh pbis-open-8.8.0.506.linux.x86_64.deb.sh
```

加入 **Windows** 域　您的域控制器必须可访问，而且您必须具有有权将计算机添加到域的 Active Directory 用户帐户。

```
sudo /opt/pbis/bin/domainjoin-cli join domain-name user
```

**user** 为有权将计算机加入 Active Directory 域的域用户。**domain-name** 为域的 DNS 名称，例如 example.com。

注意：要将 Bash 设置为默认 shell，请运行 **sudo /opt/pbis/bin/config LoginShellTemplate/bin/bash** 命令。

验证域成员身份　Delivery Controller 要求所有 VDA 计算机（Windows 和 Linux）都要在 Active Directory 中有一个计算机对象。验证加入了 PBIS 的 Linux 计算机是否位于域中：

```
/opt/pbis/bin/domainjoin-cli query
```

如果计算机已加入某个域，此命令将返回有关当前加入的 AD 域和 OU 的信息。否则，仅显示主机名。

验证用户身份验证　要验证 PBIS 是否能够通过 PAM 对域用户进行身份验证，请使用以前未使用的域用户帐户登录 Linux VDA。

```
sudo ssh localhost -l domain\user

id -u
```

验证是否为 **id -u** 命令返回的 UID 创建了对应的 Kerberos 凭据缓存文件：

```
ls /tmp/krb5cc_uid
```

退出会话。

```
exit
```

在进行域加入验证后继续执行步骤 4：安装 Linux VDA。

### 步骤 4：安装 Linux VDA

#### 步骤 4a：下载 Linux VDA 软件包

转至 Citrix Virtual Apps and Desktops 下载页面。展开相应版本的 Citrix Virtual Apps and Desktops，然后单击组件以下载与 Linux 发行版匹配的 Linux VDA 包。

#### 步骤 4b：安装 Linux VDA

使用 Debian 软件包管理器安装 Linux VDA 软件：

```
sudo dpkg -i xendesktopvda_<version>.debian10_amd64.deb
```

Debian 10.7 的 Debian 依赖项列表：

```
postgresql                >= 11
libpostgresql-jdbc-java >= 42.2
openjdk-8-jdk             >= 8u252
imagemagick               >= 8:6.9.10
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               >= 9.27~
libmspack0                >= 0.10
libgoogle-perftools4      >= 2.7~
libpython2.7              >= 2.7~
libsasl2-modules-gssapi-mit >= 2.1.~
```

> 注意：
>
> 有关此版本的 Linux VDA 支持的 Linux 发行版和 Xorg 版本的列表，请参阅系统要求。

步骤 **4c**：升级 **Linux VDA**（可选）

可以从先前的两个版本和 LTSR 版本升级现有安装。

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

步骤 **4d**：配置 **Linux VDA**

安装软件包后，必须运行 ctxsetup.sh 脚本来配置 Linux VDA。执行任何更改之前，该脚本都会验证环境，确保所有依赖项都已安装。如有必要，可以随时重新运行该脚本以更改设置。

可以按照提示手动运行脚本，也可以采用预先配置的响应自动运行脚本。继续操作前，请查看该脚本的帮助信息：

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

提示配置　运行会提示各种问题的手动配置：

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

自动配置　自动安装时，设置脚本所需的选项可由环境变量提供。如果所需的所有变量都存在，则脚本不会提示用户提供任何信息，从而允许通过脚本完成安装过程。

支持的环境变量包括：

- **CTX_XDL_SUPPORT_DDC_AS_CNAME = Y | N** –Linux VDA 支持使用 DNS CNAME 记录指定 Delivery Controller 名称。默认情况下设置为 N。
- **CTX_XDL_DDC_LIST=** ‘**list-ddc-fqdns**’ –Linux VDA 要求提供由空格分隔的 Delivery Controller 完全限定域名 (FQDN) 列表以用于向 Delivery Controller 注册。必须至少指定一个 FQDN 或 CNAME 别名。
- **CTX_XDL_VDA_PORT=port-number** –Linux VDA 通过 TCP/IP 端口（默认为端口 80）与 Delivery Controller 通信。
- **CTX_XDL_REGISTER_SERVICE = Y | N** –在启动计算机后启动 Linux Virtual Desktop 服务。默认情况下设置为 Y。
- **CTX_XDL_ADD_FIREWALL_RULES=Y | N** –Linux Virtual Desktop 服务要求允许传入网络连接通过系统防火墙。您可以在系统防火墙中自动为 Linux Virtual Desktop 打开所需端口（默认为端口 80 和 1494）。默认情况下设置为 Y。
- **CTX_XDL_AD_INTEGRATION=1 | 2 | 3 | 4 |5** –Linux VDA 要求使用 Kerberos 配置设置向 Delivery Controller 进行身份验证。Kerberos 配置根据系统上已安装和已配置的 Active Directory 集成工具确定。指定要使用且受支持的 Active Directory 集成方法：
  - 1 –Samba Winbind
  - 2 –Quest Authentication Service
  - 3 - Centrify DirectControl
  - 4 - SSSD
  - 5 –PBIS
- **CTX_XDL_HDX_3D_PRO=Y | N** –Linux VDA 支持 HDX 3D Pro，这是一组 GPU 加速技术，旨在优化富图形应用程序的虚拟化水平。如果选择了 HDX 3D Pro，则要为 VDI 桌面（单会话）模式配置 VDA -（即 CTX_XDL_VDI_MODE=Y）。
- **CTX_XDL_VDI_MODE=Y | N** –将计算机配置为专用桌面交付模型 (VDI) 还是托管共享桌面交付模型。对于 HDX 3D Pro 环境，将此变量设置为 Y。默认情况下，此变量设置为 N。
- **CTX_XDL_SITE_NAME=dns-name** –Linux VDA 通过 DNS 发现 LDAP 服务器。要将 DNS 搜索结果限制为本地站点，应指定 DNS 站点名称。默认情况下，此变量设置为 **<none>**。
- **CTX_XDL_LDAP_LIST=** ‘**list-ldap-servers**’ –Linux VDA 查询 DNS 来发现 LDAP 服务器。如果 DNS 无法提供 LDAP 服务记录，您可以提供以空格分隔的 LDAP FQDN（带有 LDAP 端口）列表。例如 ad1.mycompany.com:389。默认情况下，此变量设置为 **<none>**。
- **CTX_XDL_SEARCH_BASE=search-base-set** –Linux VDA 通过设置为 Active Directory 域根的搜索库来查询 LDAP（例如，DC=mycompany,DC=com）。但是，为提高搜索效能，可以指定搜索基础（例如 OU=VDI,DC=mycompany,DC=com）。默认情况下，此变量设置为 **<none>**。
- **CTX_XDL_FAS_LIST=** ‘**list-fas-servers**’ –联合身份验证服务 (FAS) 服务器是通过 AD 组策略配置的。Linux VDA 不支持 AD 组策略，但您可以改为提供以分号分隔的 FAS 服务器的列表。顺序必须与在 AD 组策略中配置的顺序相同。如果删除了任何服务器地址，请使用 **<none>** 文本字符串填充其空白，并且不要修改服务

器地址的顺序。

- **CTX_XDL_DOTNET_ RUNTIME_PATH=path-to-install-dotnet-runtime** – 安装.NET Core Runtime 3.1 以支持新的 Broker 代理服务 (`ctxvda`) 的路径。默认路径为 /usr/bin。
- **CTX_XDL_START_SERVICE=Y | N** –在完成 Linux VDA 配置后，是否启动 Linux VDA 服务。默认情况下设置为 Y。
- **CTX_XDL_TELEMETRY_SOCKET_PORT** - 用于侦听 Citrix Scout 的套接字端口。默认端口为 7503。
- **CTX_XDL_TELEMETRY_PORT** - 用于与 Citrix Scout 通信的端口。默认端口为 7502。

设置环境变量并运行配置脚本：

```
export CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N

export CTX_XDL_DDC_LIST= ‘list-ddc-fqdns’

export CTX_XDL_VDA_PORT=port-number

export CTX_XDL_REGISTER_SERVICE=Y|N

export CTX_XDL_ADD_FIREWALL_RULES=Y|N

export CTX_XDL_AD_INTEGRATION=1|2|3|4|5

export CTX_XDL_HDX_3D_PRO=Y|N

export CTX_XDL_VDI_MODE=Y|N

export CTX_XDL_SITE_NAME=dns-site-name | '<none>'

export CTX_XDL_LDAP_LIST= ‘list-ldap-servers’  | '<none>'

export CTX_XDL_SEARCH_BASE=search-base-set | '<none>'

export CTX_XDL_FAS_LIST= ‘list-fas-servers’  | '<none>'

export CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime

export CTX_XDL_TELEMETRY_SOCKET_PORT=port-number

export CTX_XDL_TELEMETRY_PORT=port-number

export CTX_XDL_START_SERVICE=Y|N

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

运行 sudo 命令时，键入 **-E** 选项以将现有环境变量传递给它创建的新 shell。我们建议您使用前面的命令并加上 **#!/bin/bash** 作为第一行来创建 shell 脚本文件。

或者，您可以使用单个命令指定所有参数：

```
sudo CTX_XDL_SUPPORT_DDC_AS_CNAME=Y|N \

```

```
CTX_XDL_DDC_LIST= ‘list-ddc-fqdns’  \

CTX_XDL_VDA_PORT=port-number \

CTX_XDL_REGISTER_SERVICE=Y|N \

CTX_XDL_ADD_FIREWALL_RULES=Y|N \

CTX_XDL_AD_INTEGRATION=1|2|3|4|5 \

CTX_XDL_HDX_3D_PRO=Y|N \

CTX_XDL_VDI_MODE=Y|N \

CTX_XDL_SITE_NAME=dns-name \

CTX_XDL_LDAP_LIST= ‘list-ldap-servers’  \

CTX_XDL_SEARCH_BASE=search-base-set \

CTX_XDL_FAS_LIST= ‘list-fas-servers’  \

CTX_XDL_DOTNET_RUNTIME_PATH=path-to-install-dotnet-runtime \

CTX_XDL_TELEMETRY_SOCKET_PORT=port-number \

CTX_XDL_TELEMETRY_PORT=port-number \

CTX_XDL_START_SERVICE=Y|N \

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

删除配置更改　在某些情形下，您可能需要删除 **ctxsetup.sh** 脚本对配置所做的更改，但不卸载 Linux VDA 软件包。

继续操作前，请查看此脚本的帮助信息：

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

删除配置更改：

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

> 重要：
>
> 此脚本会从数据库删除所有配置数据，从而使 Linux VDA 无法使用。

配置日志　**ctxsetup.sh** 和 **ctxcleanup.sh** 脚本会在控制台上显示错误，并将其他信息写入配置日志文件 **/tmp/xdl.configure.log**：

重新启动 Linux VDA 服务，确保更改生效。

卸载 **Linux VDA** 软件　要检查 Linux VDA 是否已安装并查看已安装软件包的版本，请运行以下命令：

```
dpkg -l xendesktopvda
```

查看更多详细信息：

```
apt-cache show xendesktopvda
```

要卸载 Linux VDA 软件，请执行以下操作：

```
dpkg -r xendesktopvda
```

> 注意：
>
> 卸载 Linux VDA 软件会删除关联的 PostgreSQL 和其他配置数据。但是，不会删除在安装 Linux VDA 之前设置的 PostgreSQL 软件包和其他依赖软件包。
>
> 提示：
>
> 本节中的信息未介绍包括 PostgreSQL 在内的依赖软件包的删除操作。

### 步骤 5：运行 Linux VDA

使用 **ctxsetup.sh** 脚本配置 Linux VDA 后，请使用以下命令控制 Linux VDA。

启动 **Linux VDA**：

启动 Linux VDA 服务：

```
sudo systemctl start ctxhdx

sudo systemctl start ctxvda
```

停止 **Linux VDA**：

停止 Linux VDA 服务：

```
sudo systemctl stop ctxvda

sudo systemctl stop ctxhdx
```

> 注意：
>
> 在停止 `ctxvda` 和 `ctxhdx` 服务之前，请运行 `service ctxmonitorservice stop` 命令以停止监视服务守护程序。否则，监视服务守护程序将重新启动您停止的服务。

重新启动 **Linux VDA**：

重新启动 Linux VDA 服务：

```
sudo systemctl stop ctxvda

sudo systemctl restart ctxhdx

sudo systemctl restart ctxvda
```

检查 **Linux VDA** 状态：

要检查 Linux VDA 服务的运行状态，请执行以下操作：

```
sudo systemctl status ctxvda

sudo systemctl status ctxhdx
```

### 步骤 6：在 Citrix Virtual Apps 或 Citrix Virtual Desktops 中创建计算机目录

创建计算机目录和添加 Linux VDA 计算机的过程与传统的 Windows VDA 方法类似。有关如何完成这些任务的更加详细的说明，请参阅创建计算机目录和管理计算机目录。

创建包含 Linux VDA 计算机的计算机目录时会面临一些限制，使得该过程不同于为 Windows VDA 计算机创建计算机目录：

- 对于操作系统，请选择：
  - 多会话操作系统选项（对于托管共享桌面交付模型）。
  - 单会话操作系统选项（对于 VDI 专用桌面交付模型）。
- 请勿在同一个计算机目录中混合使用 Linux 和 Windows VDA 计算机。

> 注意：
>
> 早期版本的 Citrix Studio 不支持“Linux 操作系统”的概念。但是，选择 **Windows Server** 操作系统或服务器操作系统选项等同于使用托管共享桌面交付模型。选择 **Windows** 桌面操作系统或桌面操作系统选项等同于使用每计算机一个用户交付模型。
>
> 提示：
>
> 如果删除计算机后将其重新加入 Active Directory 域，则必须删除计算机，然后将其重新添加到计算机目录。

### 步骤 7：在 Citrix Virtual Apps 或 Citrix Virtual Desktops 中创建交付组

创建交付组和添加包含 Linux VDA 计算机的计算机目录的过程与 Windows VDA 计算机几乎相同。有关如何完成这些任务的更加详细的说明，请参阅创建交付组。

创建含有 Linux VDA 计算机目录的交付组时会面临以下限制：

- 确保所选的 AD 用户和组已正确配置，可以登录到 Linux VDA 计算机。

- 请勿允许未经身份验证的（匿名）用户登录。
- 请勿在交付组中混入含有 Windows 计算机的计算机目录。

有关如何创建计算机目录和交付组的信息，请参阅 Citrix Virtual Apps and Desktops 7 2103。

## 配置 Linux VDA

April 16, 2021

本部风内容详细介绍了 Linux VDA 的功能，包括功能说明、配置和故障排除。

> 提示：
>
> 用于收集日志的 `xdlcollect` Bash 脚本将集成到 Linux VDA 软件中并位于 /opt/Citrix/VDA/bin 下。安装 Linux VDA 后，可以运行 `bash /opt/Citrix/VDA/bin/xdlcollect.sh` 命令来收集日志。
>
> 日志收集完成后，压缩的日志文件将在与脚本相同的文件夹中生成。`xdlcollect` 可以询问您是否将压缩的日志文件上载到 Citrix Insight Services (CIS)。如果您同意，`xdlcollect` 将在上载完成后返回 upload_ID。上载不会从您的本地计算机中删除压缩的日志文件。其他用户可以使用 upload_ID 访问 CIS 中的日志文件。

## 将 NIS 与 Active Directory 集成

November 8, 2021

本文介绍如何在 Linux VDA 中使用 SSSD 将 NIS 与 Windows Active Directory (AD) 集成。Linux VDA 被视为 Citrix Virtual Apps and Desktops 的一个组件。因此，它与 Windows AD 环境紧密结合。

使用 NIS（而非 AD）作为 UID 和 GID 提供程序要求帐户信息（用户名和密码组合）在 AD 和 NIS 中相同。

> 注意：
>
> 仍由 AD 服务器执行身份验证。不支持 NIS+。如果使用 NIS 作为 UID 和 GID 提供程序，则不再使用来自 Windows 服务器的 POSIX 属性。
>
> 提示：
>
> 此方法代表已弃用的 Linux VDA 部署方法，这种方法仅用于特殊用例。对于 RHEL/CentOS 发行版，请按照安装 Linux Virtual Delivery Agent for RHEL/CentOS 中的说明进行操作。对于 Ubuntu 发行版，请按照安装 Linux Virtual Delivery Agent for Ubuntu 中的说明进行操作。

SSSD 是什么？

SSSD 是系统守护程序。其主要功能是为了实现通过可以为系统提供缓存和脱机支持的通用框架来识别远程资源并对其进行身份验证。它提供 PAM 和 NSS 两种模块，将来可以为扩展用户信息支持基于 D-BUS 的接口。此外它还提供更好的数据库来存储本地用户帐户和扩展用户数据。

### 将 **NIS** 与 **AD** 相集成

要将 NIS 与 AD 集成，请执行以下操作：

1. 将 Linux VDA 添加为 NIS 客户端
2. 使用 Samba 加入域并创建主机 keytab
3. 设置 SSSD
4. 配置 NSS/PAM
5. 验证 Kerberos 配置
6. 验证用户身份验证

#### 将 **Linux VDA** 添加为 **NIS** 客户端

配置 NIS 客户端：

```
yum -y install ypbind rpcbind oddjob-mkhomedir
```

设置 NIS 域：

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

在 **/etc/hosts** 中添加 NIS 服务器和客户端的 IP 地址：

```
{ NIS server IP address }    server.nis.domain nis.domain
```

通过 `authconfig` 配置 NIS：

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

**nis.domain** 表示 NIS 服务器的域名。**server.nis.domain** 是 NIS 服务器的主机名，也可以是 NIS 服务器的 IP 地址。

配置 NIS 服务：

```
sudo systemctl start rpcbind ypbind

sudo systemctl enable rpcbind ypbind
```

确保 NIS 配置正确：

```
ypwhich
```

验证可从 NIS 服务器获得帐户信息：

```
getent passwd nisaccount
```

> 注意：
>
> **nisaccount** 表示 NIS 服务器上的实际 NIS 帐户。确保 UID、GID、主目录和登录 shell 都已正确配置。

#### 使用 **Samba** 加入域并创建主机 **keytab**

SSSD 并不提供用于加入域和管理系统 keytab 文件的 AD 客户端功能。实现这些功能可以采用几种方法，包括：

- `adcli`
- `realmd`
- `Winbind`
- `Samba`

本节信息只介绍 Samba 方法。对于 `realmd`，请参阅 RHEL 或 CentOS 供应商的文档。必须在配置 SSSD 之前执行这些步骤。

使用 **Samba** 加入域并创建主机 **keytab**：

在正确配置了以下文件的 Linux 客户端上：

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

为计算机配置 Samba 和 Kerberos 身份验证：

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

其中，**REALM** 是大写的 Kerberos 领域名称，而 **domain** 是域的 NetBIOS 名称。

如果需要通过 DNS 查找 KDC 服务器和领域名称，请将以下两个选项添加至前面的命令：

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

打开 **/etc/samba/smb.conf** 并将以下条目添加到 **[Global]** 部分下方，但要放在 **authconfig** 工具生成的部分后面：

```
kerberos method = secrets and keytab
winbind offline logon = no
```

加入 Windows 域要求您的域控制器可访问，而且您具有有权将计算机添加到域的 AD 用户帐户。

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

`REALM` 是大写的 Kerberos 领域名称，`user` 是有权将计算机添加到域的域用户。

### 设置 SSSD

设置 SSSD 的步骤如下：

- 在 Linux 客户端计算机上安装 **sssd-ad** 和 **sssd-proxy** 软件包。
- 对各种文件（例如 **sssd.conf**）进行配置更改。
- 启动 **sssd** 服务。

**/etc/sssd/sssd.conf** **sssd.conf** 配置示例（可以根据需要添加更多选项）：

```
[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
```

将 **ad.domain.com**、**server.ad.example.com** 替换为相应的值。有关详细信息，请参阅 sssd-ad(5) - Linux 手册页。

对 **sssd.conf** 设置文件所有权和权限：

```
chown root:root /etc/sssd/sssd.conf
```

```
chmod 0600 /etc/sssd/sssd.conf
restorecon /etc/sssd/sssd.conf
```

#### 配置 NSS/PAM

**RHEL/CentOS**:

使用 **authconfig** 启用 SSSD。安装 **oddjob mkhomedir** 以确保主目录创建与 SELinux 兼容:

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

sudo systemctl start sssd

sudo systemctl enable sssd
```

> 提示:
>
> 配置 Linux VDA 设置时，要为 SSSD 考虑上述操作，而对 Linux VDA 客户端无需特殊设置。对于 **ctxsetup.sh** 脚本中的额外解决方案，请使用默认值。

#### 验证 Kerberos 配置

为了确保 Kerberos 已正确配置为可与 Linux VDA 配合使用，请检查系统 **keytab** 文件是否已创建并包含有效密钥:

```
sudo klist -ke
```

此命令显示各种主体名称与密码套件组合可用的密钥列表。运行 Kerberos **kinit** 命令，以使用这些密钥向域控制器验证计算机的身份:

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

计算机和领域名称必须指定为大写。美元符号 ($) 必须使用反斜杠 (\) 进行转义，以免发生 shell 替换。在某些环境中，DNS 域名与 Kerberos 领域名称不同。请确保使用领域名称。如果此命令成功运行，则不会显示任何输出。

使用以下命令验证计算机帐户的 TGT 票据已缓存:

```
sudo klist -ke
```

#### 验证用户身份验证

使用 **getent** 命令确认支持的登录格式以及 NSS 是否工作:

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

**DOMAIN** 参数指示简短形式的域名。如果需要使用另一种登录格式，请先使用 **getent** 命令进行验证。

支持的登录格式如下：

- 低级别登录名称：`DOMAIN\username`
- UPN：`username@domain.com`
- NetBIOS 前缀格式：`username@DOMAIN`

要验证 SSSD PAM 模块是否已正确配置，请使用域用户帐户登录 Linux VDA。该域用户帐户以前未曾使用过。

```
sudo ssh localhost  - l DOMAIN\username

id -u
```

检查是否为以下命令返回的 **uid** 创建了对应的 Kerberos 凭据缓存文件：

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

检查用户 Kerberos 凭据缓存中的票据是否有效且未过期：

```
klist
```

## 发布应用程序

July 7, 2022

借助 Linux VDA 7.13 版，Citrix 向所有受支持的 Linux 平台中添加了无缝应用程序功能。不需要执行任何特定安装过程即可使用此功能。

> 提示：
>
> 在 Linux VDA 1.4 中，Citrix 增加了对已发布的非无缝应用程序和会话共享功能的支持。

### 使用 **Citrix Studio** 发布应用程序

您可以在创建交付组或将应用程序添加到现有交付组时发布 Linux VDA 上安装的应用程序。该过程与发布 Windows VDA 上安装的应用程序类似。有关详细信息，请参阅 Citrix Virtual Apps and Desktops 文档 （根据使用的 Citrix Virtual Apps and Desktops 版本）。

> 提示：
>
> 配置交付组时，请确保交付类型设置为桌面和应用程序或应用程序。
>
> 重要：

> Linux VDA 1.4 及更高版本支持发布应用程序。但是，Linux VDA 不支持将桌面和应用程序交付给相同的计算机。要解决此问题，我们建议您为应用程序和桌面交付创建单独的交付组。
>
> 注意：
>
> 要使用无缝应用程序，请勿在 StoreFront 上禁用无缝模式。无缝模式默认处于启用状态。如果您已通过设置 TWIMode=Off 禁用该模式，请删除此设置，而非将其更改为 TWIMode=On。否则可能无法启动已发布的桌面。

限制

Linux VDA 不支持单个用户启动同一应用程序的多个并发实例。

已知问题

在发布应用程序期间发现了以下已知问题：

- 不支持非长方形窗口。窗口的边角可能会显示服务器端背景。
- 不支持从已发布的应用程序预览窗口的内容。
- 目前，无缝模式支持以下窗口管理器：Mutter、Metacity 和 Compiz (Ubuntu 16.04)。Kwin 和其他窗口管理器不受支持。请确保您的窗口管理器属于受支持的窗口管理器。
- 运行多个 LibreOffice 应用程序时，只有第一个启动的应用程序显示在 Citrix Studio 上，因为这些应用程序共享进程。
- “Dolphin” 之类的基于 Qt5 的已发布应用程序可能不显示图标。要解决此问题，请参阅网址为 https://wiki.archlinux.org/title/Qt 的文章。
- 同一 ICA 会话中运行的已发布应用程序的所有任务栏按钮都组合在同一个组中。要解决此问题，请将任务栏属性设置为不组合任务栏按钮。

## Linux 流技术推送

June 16, 2023

本文提供了有关 Citrix Provisioning Linux 流技术推送功能的信息。使用此功能，您可以直接在 Citrix Virtual Apps and Desktops 环境中预配 Linux 虚拟桌面。

支持以下 Linux 发行版：

- Ubuntu 16.04
- Ubuntu 18.04.5（预览版）
- RHEL 8.3（预览版）

> 重要：
>
> - 要将此功能用于 Ubuntu 18.04.5 和 RHEL 8.3，请分别使用 **PVS Linux Streaming Agent (Ubuntu 18.04)-Experimental** 软件包和 **PVS Linux Streaming Agent (RHEL8.3)-Experimental** 软件包。Linux VDA 下载页面上提供了安装包。
> - 要将此功能用于 Ubuntu 16.04，请下载最新的 Citrix Provisioning ISO 并找到 Ubuntu 16.04 的目标软件。有关详细信息，请参阅 Citrix Provisioning 文档中的配置 Linux 流技术推送功能。

预配 Linux 目标设备时，请注意以下事项：

- 有时，无法将客户端驱动器映射到已预配的 Linux VM 会话。要解决此问题，请在安装 Citrix Provisioning 目标设备之前使用 `service ctxcdm stop` 停止 CDM 服务，然后再运行 `pvs-imager` 命令以对其进行转换。
- Linux 流技术推送功能仅支持将 Winbind 用作加入 Windows 域的工具。
- 如果为 Linux 设备启用了 RAM 缓存，请将缓存大小设置为 8 MB（最小值）。Linux 对写入缓存使用尽可能多的 RAM，包括所有可用内存。在控制台中指定的数量为预先保留的数量。Citrix 建议您尽可能少地预留，这样将有效地允许 Linux 管理内存使用量。
- Citrix Provisioning Imager UI 中的目标设备名称通常默认为 `im\\_localhost`。创建多个虚拟磁盘时，必须更改此值。使用相同的目标设备名称可能会导致 imager 命令失败。
- 安装（及后续更新）必须在超级用户模式下完成。以超级用户身份安装有两种方式：
  - 使用 `su` 命令在终端中进入用户模式。
  - 在该命令前面输入 `sudo`。例如，`sudo yum install tdb-tools`；为每个命令输入 `sudo`。
- 必须使用 Active Directory 控制器同步 Linux 客户端的系统时钟。
- 不支持 UEFI。
- 不支持 VMM。
- 写入缓存驱动器必须带有标签 `PVS_Cache`，才能将其用作写入缓存。将使用整个分区。
- 英语本地化信息在非英语安装中显示。
- 不支持 SE Linux。
- 在 XenServer 上运行的目标必须在 HVM 模式下运行。
- 启动 Linux 目标设备后，可能会显示一条指示 SE Linux 警报浏览器的警告消息。
- ESXi 上托管的两个流 Ubuntu 18.04 VM 通过 DHCP 获得相同的 IP 地址。要解决此问题，请将 VM 配置为使用 MAC 地址作为通过 DHCP 检索 IP 地址的唯一 ID。
- 对于 Ubuntu 18.04.5 和 RHEL 8.3，计算机帐户密码不会在 Active Directory 中自动更新。当密码过期且流 VM 无法加入域时，请尝试通过 Citrix Provisioning 控制台重置密码。
- 对于 Ubuntu 16.04，使用 Citrix Provisioning 预配 Linux 目标设备时，仅支持 Samba 4.4 及更早版本提供的 Winbind。

安装选项

必须以管理员身份登录，才能安装 Linux 流技术推送组件。安装时请注意，以下命令必须在 root shell 中发出，或者使用 `sudo` 权限发出。

> 注意：
>
> 如果通过流技术推送 Citrix Provisioning Linux 目标设备，则必须创建自签名证书。SOAP 服务器使用的 SSL 连接要求您在 SOAP 服务器上配置 X.509 证书。
>
> 证书的 CA 还必须存在于 Provisioning 服务器和 Linux 目标设备上。有关创建自签名证书的信息，请参阅为 Linux 流技术推送创建自签名证书。

对于 Ubuntu 16.04 发行版：

```
sudo dpkg -i pvs-<version>.deb

sudo apt-get -yf install
```

对于 Ubuntu 18.04 发行版：

```
sudo apt-get -y install dracut dracut-network tdb-tools python3 python3
    -distutils
sudo dpkg -i pvs_<version>_ubuntu18.04_amd64.deb
```

对于 RHEL 8.3 发行版：

```
yum - nogpgcheck localinstall pvs_<version>_rhel8.3_x86_64.deb
```

### 使用 **GUI** 创建 **Linux** 黄金映像

要调用 GUI 以安装此功能，请执行以下操作：

1. 以管理员身份登录。
2. 运行 `pvs-imager` 命令：

   > 提示：
   >
   > `pvs-imager` 命令由于主机名问题失败时，请验证您的网络配置是否正确。请勿将系统的主机名设置为 `localhost`。在 RHEL8.3 上，请使用 X11 显示服务器而非 Wayland 登录以使用 GUI。

运行命令后，用户界面页面将显示以下内容：


### 使用命令行接口安装 **Linux** 流技术推送功能

要调用命令行以安装此功能，请执行以下操作：

1. 以管理员身份登录。
2. 运行以下命令：

```
pvs-imager -C
```

命令行安装包括两个选项：

- \-C 允许您创建虚拟磁盘
- \-U 允许您更新现有虚拟磁盘

以下信息说明了 Linux 流技术推送功能的非 GUI 相关的安装选项：

```
Usage: ./pvs-imager \[-hCU] \[-a|--address=<IPaddr>] \[-u|--username=<
    username>] \[-p|--password=<password>] \[-P|--port=<port>] \[-d|--
    domain=<domain] \[-S|--store=<store>] \[-v|--vdisk=<vdisk name>] \[-
    s|--size=<vdisk size] \[-D|--device=<sourceDevice>] \[-c|--
    collection=<collection>] \[-n|--name=<name>]
 Non-GUI Modes:
  -C      -  Create a new vDisk
    ---OR----
  -U      -  Update an existing vDisk

 General Options:
  -a <server IP> -  Address or hostname of PVS server
  -u <username>  -  Username for API login
  -p <password>  -  Password for API login
  -d <domain>   -  AD domain for API login
  -P <port>    -  Base port for API login (default: 54321)
  -S <store>   -  Store containing vDisk
  -c <collection> -  Collection to store imaging device in
  -n <name>    -  Device name for imaging device
  -v <name>    -  vDisk name
  -s <size>    -  vDisk size (Create Mode only, default: sourceDevice
     size)
  -D <sourceDev> -  devnode to clone
  -V       -  increment debug verbosity (up to 5 times)
  -g <grubMode>  -  Supported Grub settings ( 'debug' )
```

适用于映像操作的受支持的文件系统包括 `ext4`、`xfs` 或 `btrfs`。

> 提示：
>
> 使用 `-VVVVV` 开关创建的 `pvs-imager` 的调试日志将在执行 `pvs-imager` 工具的文件夹中创建。日志文件的名称为 `pvs-imager.log`。

### 关于磁盘缓存

对于不使用 Citrix Virtual Apps and Desktops 设置向导情况下的硬盘缓存或硬盘溢出缓存，请使用格式化分区对目标设备磁盘进行格式化。包括标签 `PVS_Cache`。可以在目标设备上通过 `mkfs -L PVS_Cache` 命令创建此对象。可以对缓存使用任何区分大小写的文件系统，但建议使用 XFS。

> 提示：
>
> 管理员可以通过编写在启动时运行的 Bash 脚本为其环境创建任何缓存磁盘选择逻辑。该脚本将根据最适合环境的机制查找缓存设备候选，在其上运行 `mkfs`，然后重新启动。

配置磁盘缓存时：

- Citrix 建议使用 Citrix Virtual Apps and Desktops 设置向导创建 Linux 目标设备。
- 手动创建标签需要区分大小写以避免配置冲突。
- 或者，请考虑使用手动方法创建写入缓存。

手动创建目标设备的写入缓存

默认情况下，Citrix Virtual Apps and Desktops 设置向导将忽略附加到当前模板的驱动器。向导会根据您提供的参数创建写入缓存。有时，写入缓存驱动器在使用向导自动创建过程中会遇到问题。或者，当目标设备由于创建的驱动器出现问题而持续回退到服务器端缓存时。要解决这些问题，请在目标设备上使用 `mkfs -L PVS_Cache` 命令手动创建对象。

在使用 `UseTemplatecache` 参数时，Citrix Virtual Apps and Desktops 设置向导默认识别为目标设备的手动创建的写入缓存更改。在运行 Citrix Virtual Apps and Desktops 设置向导的 Provisioning 服务器上或远程预配控制台所指向的 Provisioning 服务器上，更改注册表设置：

在预配控制台计算机上创建以下注册表项以禁用模板缓存：

`HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\ProvisioningServices`

`Name: UseTemplateCache`

`Type: DWORD`

`Value: 0`

运行 Citrix Virtual Apps and Desktops 设置向导. 在虚拟机页面上，将本地写入缓存磁盘大小更改为 0 GB（默认为 6 GB）。

## Remote PC Access

July 15, 2024

概述

Remote PC Access 是 Citrix Virtual Apps and Desktops 的扩展程序。它使组织能够轻松地允许员工以安全的方式远程访问其物理办公室 PC。如果用户可以访问其办公室 PC，他们可以访问完成工作所需的所有应用程序、数据和资源。

Remote PC Access 使用交付虚拟桌面和应用程序的相同 Citrix Virtual Apps and Desktops 组件。部署和配置 Remote PC Access 的要求和流程与部署 Citrix Virtual Apps and Desktops 以交付虚拟资源所需的要求和流程相同。这种统一性提供了一致且统一的管理体验。用户通过使用 Citrix HDX 交付其远程办公室 PC 会话，获得最佳用户体验。

有关详细信息，请参阅 Citrix Virtual Apps and Desktops 文档中的 Remote PC Access。

### 注意事项

这些注意事项是 Linux VDA 特有的：

- 在物理机上，请仅在非 3D 模式下使用 Linux VDA。由于 NVIDIA 驱动程序的限制，当启用了 HDX 3D 模式时，PC 的本地屏幕无法停止，并显示会话的活动。显示此屏幕存在潜在的安全风险。
- 请对物理 Linux 计算机使用单会话操作系统类型的计算机目录。
- 自动用户分配不适用于 Linux 计算机。通过自动用户分配，用户在本地登录 PC 时会自动将其分配给各自的计算机。在没有管理员干预的情况下执行此登录。通过在客户端设备上运行的 Citrix Workspace 应用程序，用户可以在 Remote PC Access 桌面会话中访问办公室 PC 上的应用程序和数据。
- 如果用户已在本地登录到其 PC，尝试从 StoreFront 启动 PC 将失败。
- 节能选项不适用于 Linux 计算机。

### 配置

要交付 Linux PC 会话，请在目标 PC 上安装 Linux VDA，创建 **Remote PC Access** 类型的计算机目录，然后创建交付组，以使计算机目录中的 PC 可供请求访问的用户使用。以下部分详细介绍了该过程：

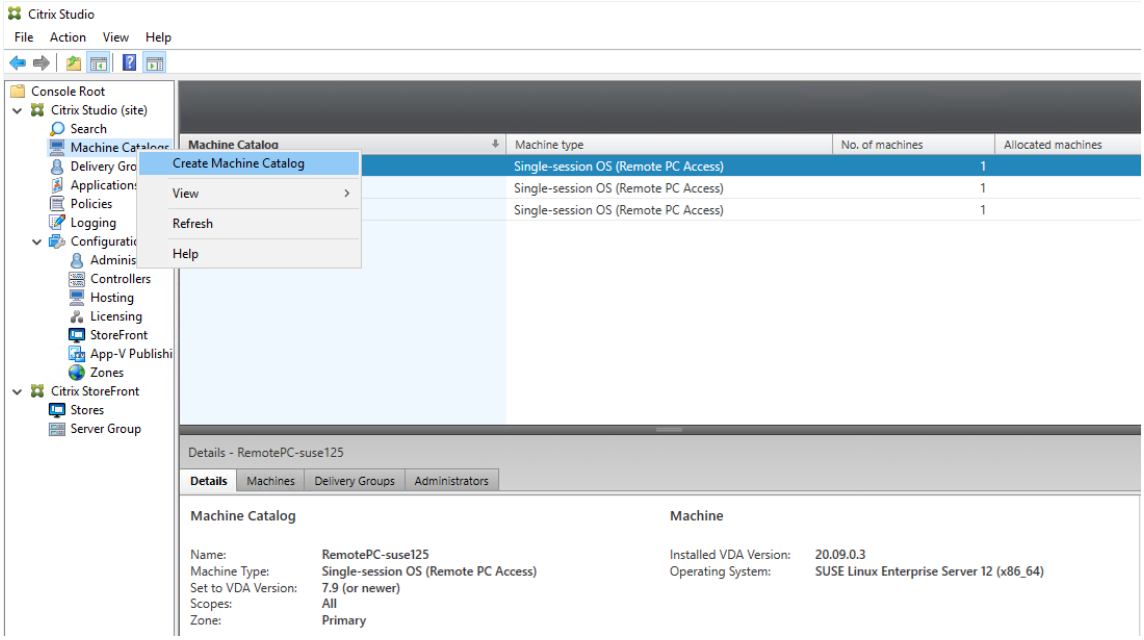
#### 步骤 **1 -** 在目标 **PC** 上安装 **Linux VDA**

建议您使用轻松安装来安装 Linux VDA。在安装过程中，将 `CTX_XDL_VDI_MODE` 变量的值设置为 `Y`。

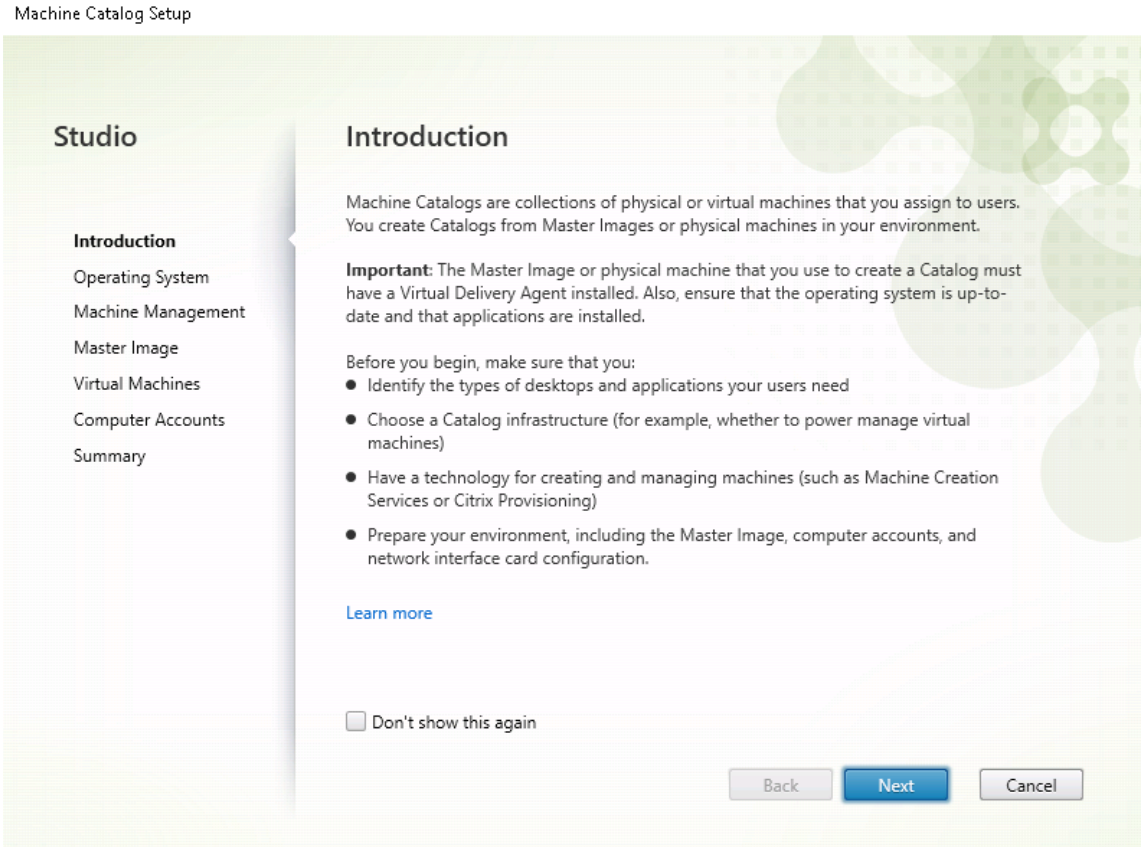
#### 步骤 **2 -** 创建 **Remote PC Access** 类型的计算机目录

1. 在 Citrix Studio 中，右键单击计算机目录，然后从快捷菜单中选择创建计算机目录。

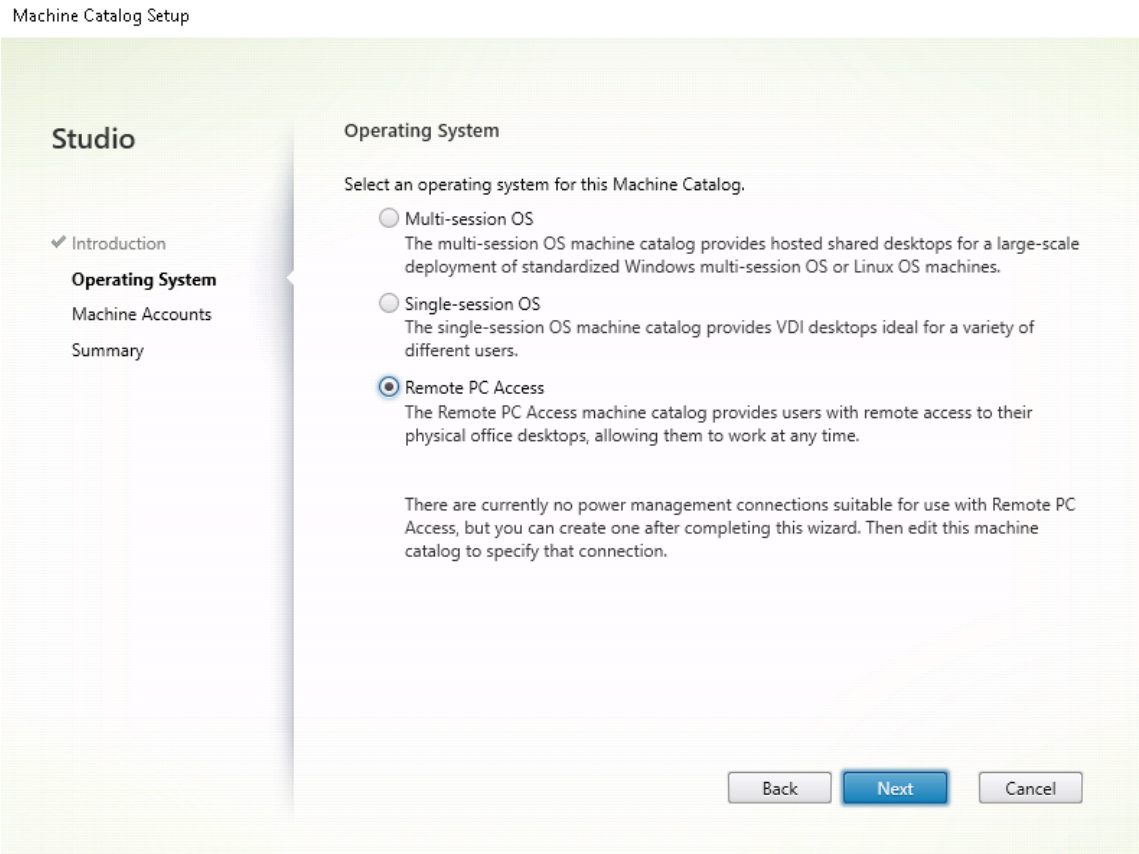



2. 单击简介页面上的下一步。



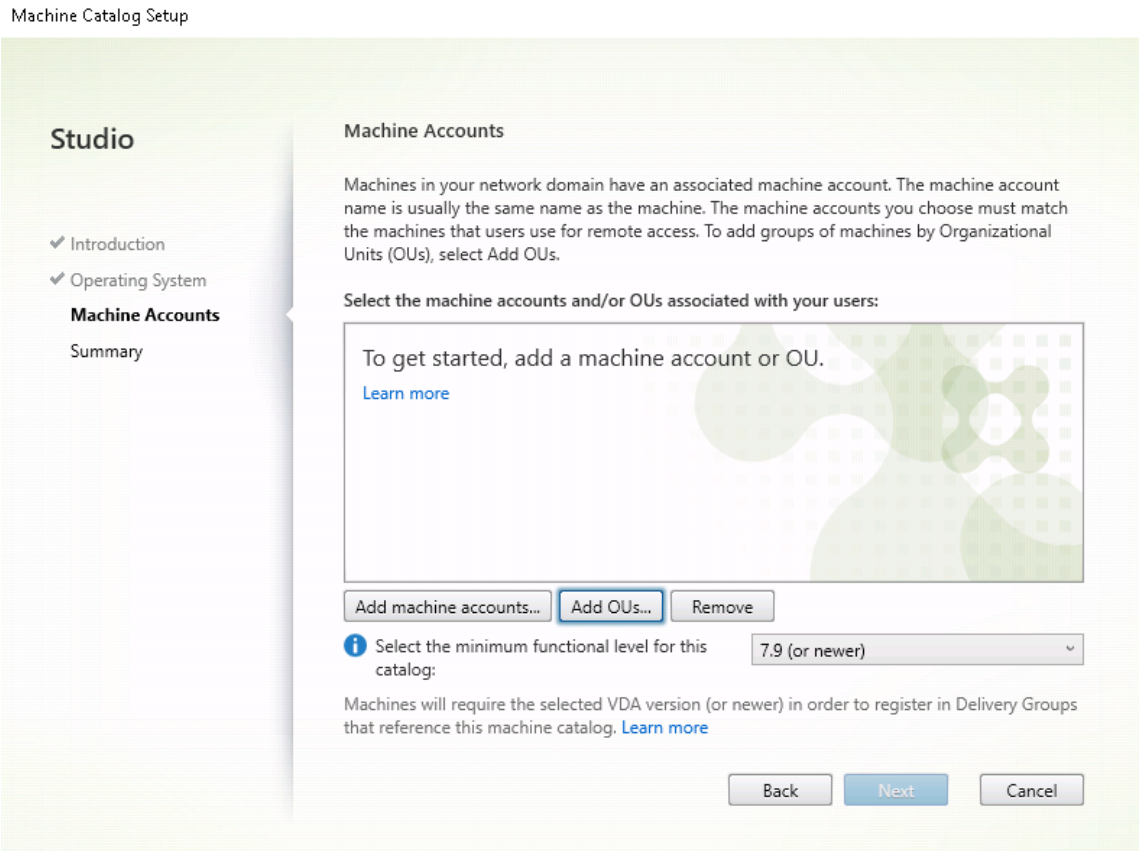

3. 在操作系统页面上选择 **Remote PC Access**。



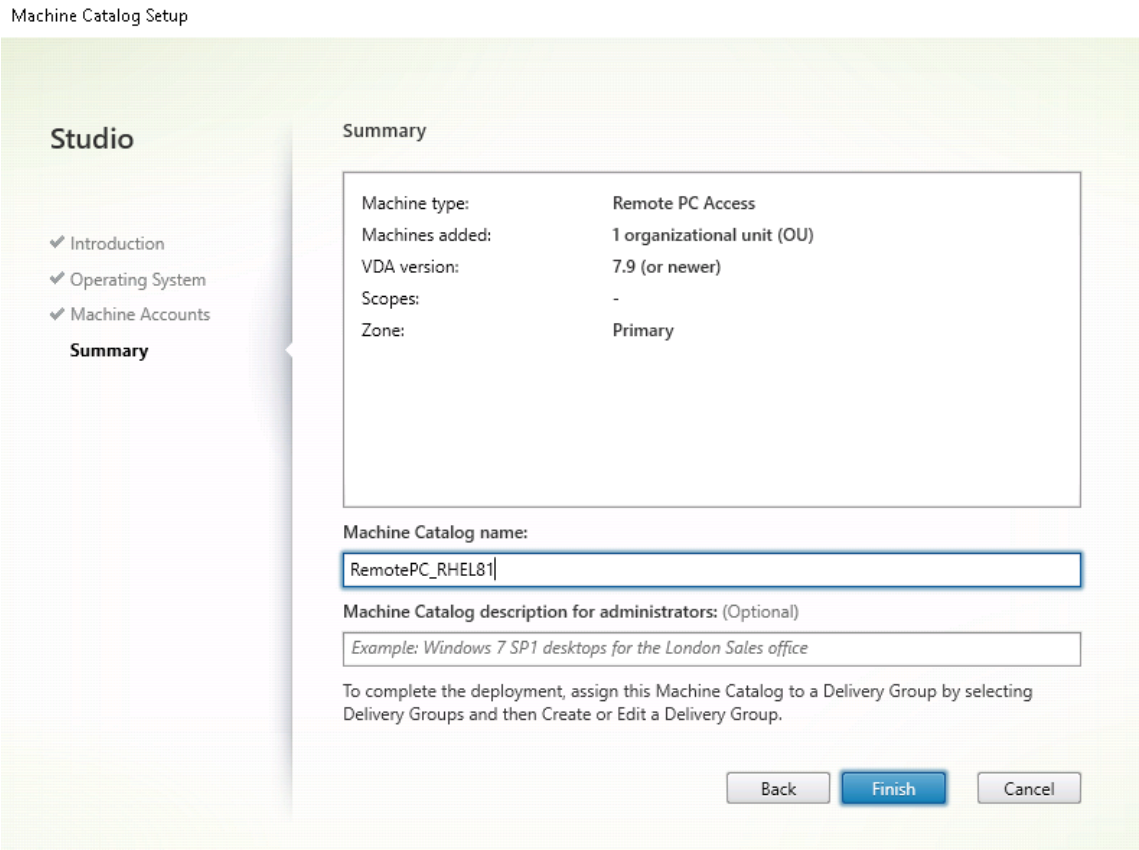

4. 单击添加 **OU** 以选择包含目标 PC 的 OU，或者单击添加计算机帐户将单个计算机添加到计算机目录中。

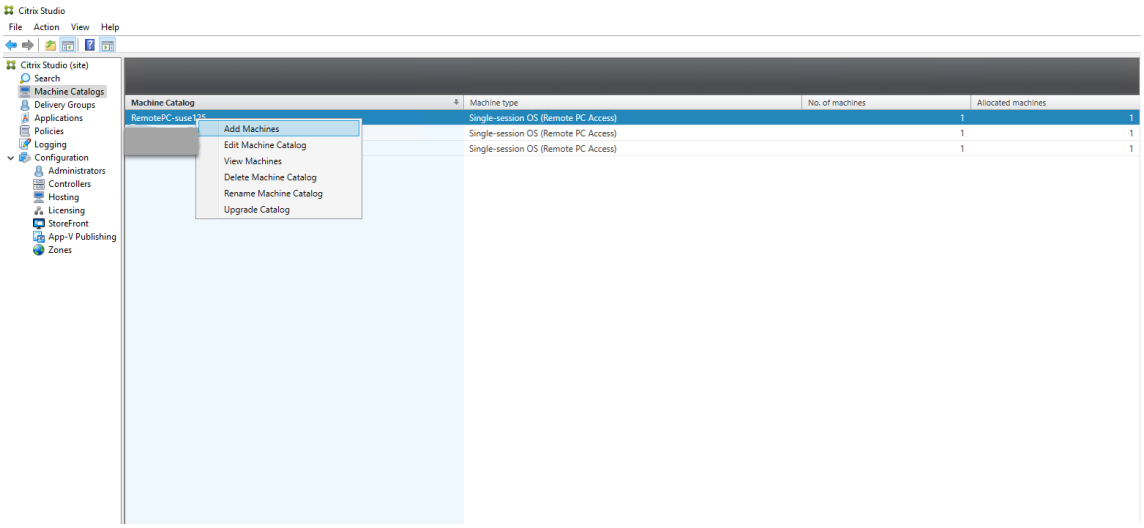



5. 命名计算机目录。



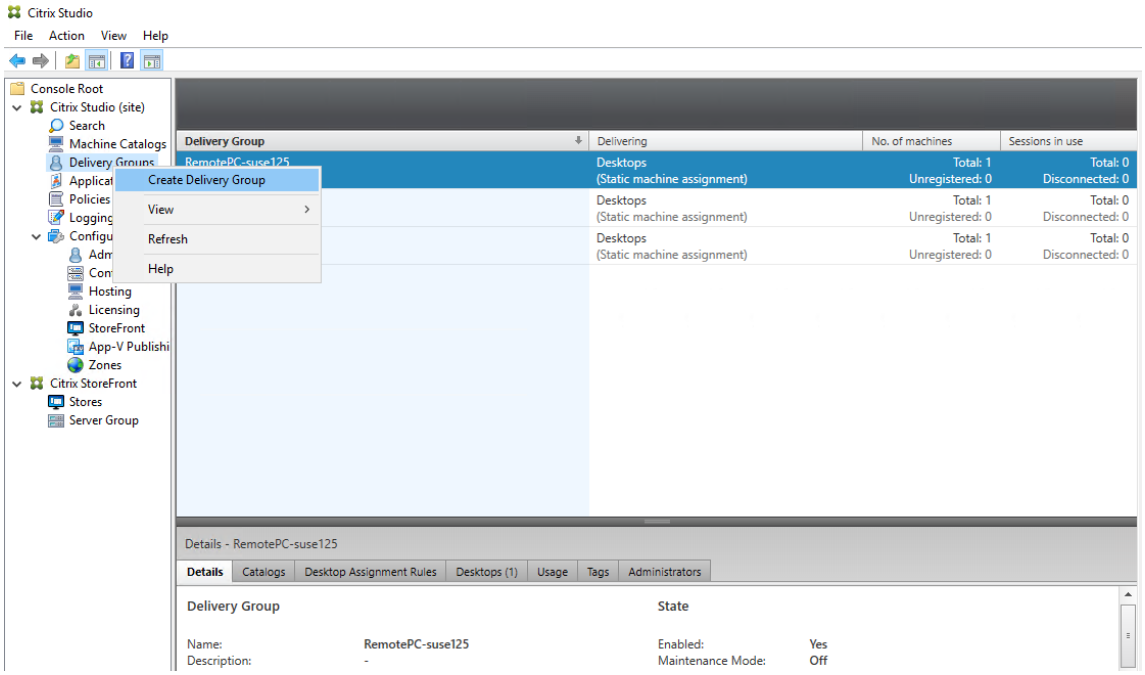

6.（可选）右键单击计算机目录以执行相关操作。



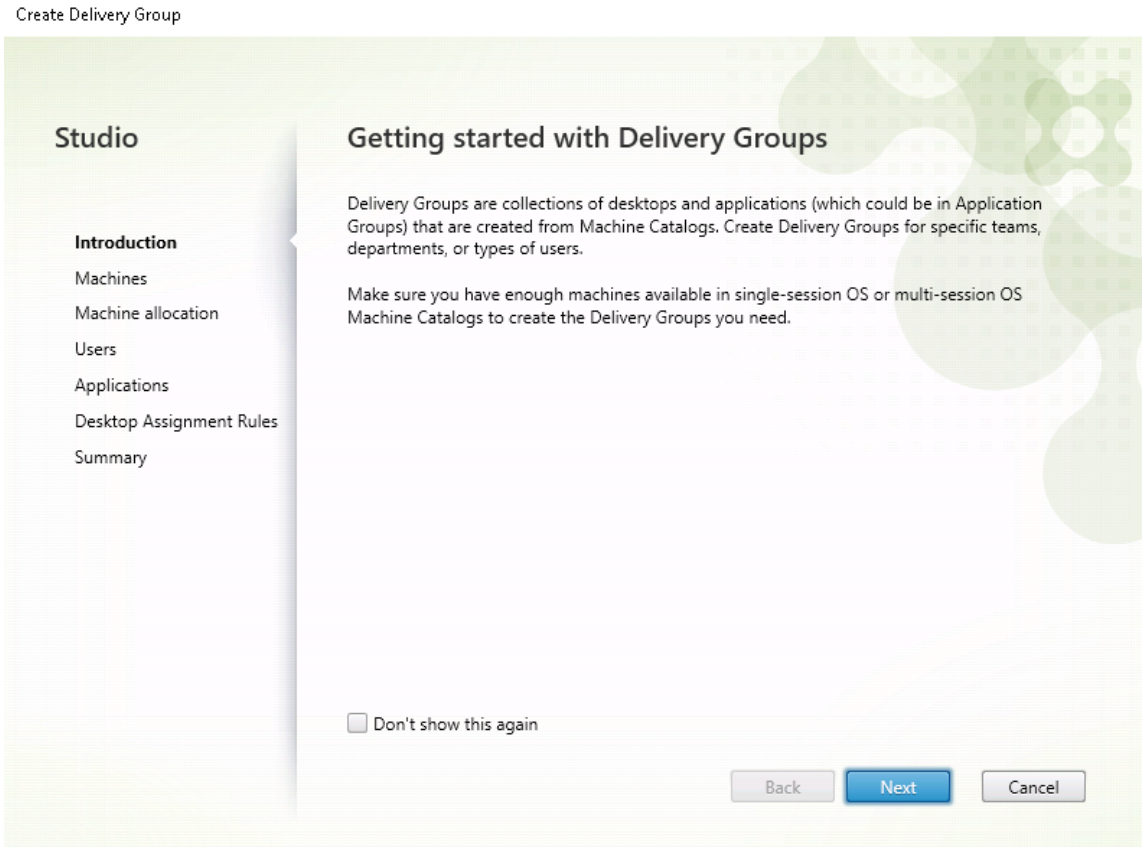

步骤 **3 -** 创建交付组，以使计算机目录中的 **PC** 可供请求访问权限的用户使用

1. 在 Citrix Studio 中，右键单击交付组，然后从快捷菜单中选择创建交付组。

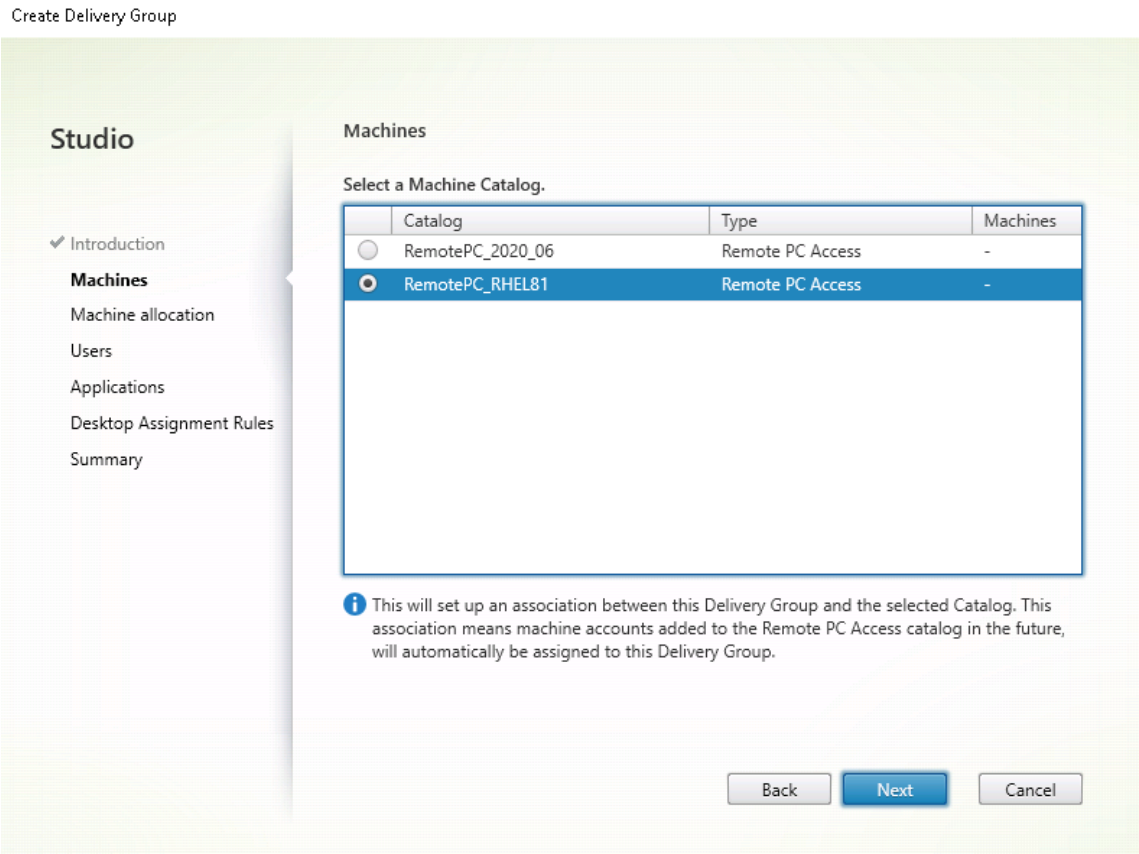



2. 单击交付组入门页面上的下一步。



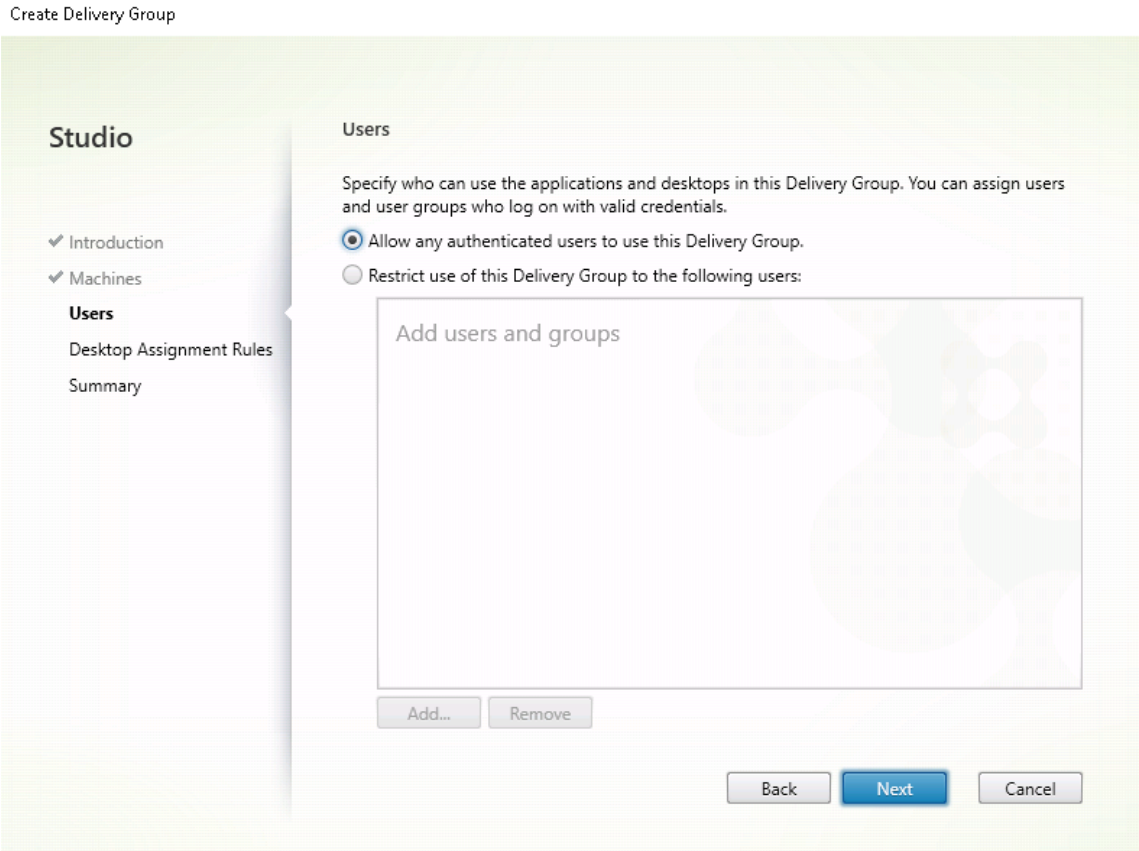

3. 选择在步骤 2 中创建的计算机目录，以将其与交付组关联。




4. 添加可以访问计算机目录中 PC 的用户。添加的用户可以使用客户端设备上的 Citrix Workspace 应用程序远程访问 PC。




局域网唤醒

Remote PC Access 支持局域网唤醒功能，用户可以使用此功能远程开启物理 PC。借助此功能，用户可以在办公室 PC 不使用时将其关闭，以节约能源成本。用户还可以在计算机意外关闭时进行远程访问。

借助局域网唤醒功能，幻数据包会在 Delivery Controller 指示时直接从 PC 上运行的 VDA 发送到 PC 所在的子网。这允许此功能在不依赖额外的基础结构组件或第三方解决方案的情况下运行，以便传送幻数据包。

局域网唤醒功能不同于传统的基于 SCCM 的局域网唤醒功能。有关基于 SCCM 的局域网唤醒的信息，请参阅 局域网唤醒–SCCM 集成。

系统要求

下面是使用局域网唤醒功能的系统要求：

- 控制平面：
  - Citrix Virtual Apps and Desktops 服务
  - Citrix Virtual Apps and Desktops 2012 或更高版本
- 物理 PC：

- VDA 版本 2012 或更高版本
- 在 BIOS 中和 NIC 上启用了局域网唤醒

### 配置局域网唤醒

目前，仅在使用 PowerShell 时支持集成的局域网唤醒的配置。

要配置局域网唤醒，请执行以下操作：

1. 如果您还没有 Remote PC Access 计算机目录，请创建一个目录。
2. 如果您还没有局域网唤醒主机连接，请创建一个连接。

   > 注意：
   >
   > 要使用局域网唤醒功能，如果您具有“Microsoft 配置管理器局域网唤醒”类型的主机连接，请创建一个主机连接。

3. 检索局域网唤醒主机连接的唯一标识符。
4. 将局域网唤醒主机连接与计算机目录相关联。

   要创建局域网唤醒主机连接，请执行以下操作：

```
# 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
while (-not $bhc.IsReady)
{

    Start-Sleep -s 5
    $bhc = Get-BrokerHypervisorConnection -HypHypervisorConnectionUid $hypHc.HypervisorConnectionUid
```

```
}
```

主机连接准备就绪后，运行以下命令以检索主机连接的唯一标识符：

```
$bhc = Get-BrokerHypervisorConnection -Name "<WoL Connection Name>"
$hypUid = $bhc.Uid
```

检索连接的唯一标识符后，运行以下命令以将连接与 Remote PC Access 计算机目录相关联：

```
Get-BrokerCatalog -Name "<Catalog Name>" | Set-BrokerCatalog -RemotePCHypervisorConnectionUid $hypUid
```

5. 在计算机目录中的每台 VM 上的 BIOS 中和 NIC 上启用局域网唤醒。

   注意：启用局域网唤醒的方法因计算机配置的不同而异。

   - 要在 BIOS 中启用局域网唤醒，请执行以下操作：
     a) 进入 BIOS 并启用局域网唤醒功能。
        访问 BIOS 的方法取决于主板的制造商和制造商选择的 BIOS 供应商。
     b) 保存您的设置并重新启动计算机。
   - 要在 NIC 上启用局域网唤醒，请执行以下操作：
     a) 运行 `sudo ethtool <NIC>` 命令以检查您的 NIC 是否支持幻数据包。
        `<NIC>` 是您的 NIC 的设备名称，例如 `eth0`。`sudo ethtool <NIC>` 命令提供有关 NIC 功能的输出：
        - 如果输出中包含一个类似于 `Supports Wake-on: <letters>` 的行（其中 `<letters>` 包含字母 `g`），您的 NIC 将支持局域网唤醒幻数据包方法。
        - 如果输出中包含一个类似于 `Wake-on: <letters>` 的行（其中 `<letters>` 包含字母 `g`，但不包含字母 `d`），则局域网唤醒幻数据报方法已启用。但是，如果 `<letters>` 包含字母 `d`，则表示局域网唤醒功能已禁用。在这种情况下，请通过运行 `sudo ethtool -s <NIC> wol g` 命令启用局域网唤醒。
     b) 在大多数发行版中，每次启动后都需要运行 `sudo ethtool -s <NIC> wol g` 命令。要永久设置此选项，请根据您的发行版完成以下步骤：
        **Ubuntu**：
        将 `up ethtool -s <NIC> wol g` 行添加到接口配置文件 `/etc/network/interfaces` 中。例如：

```
# 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**:

将以下 ETHTOOL_OPTS 参数添加到接口配置文件 /etc/sysconfig/network-scripts/ifcfg-<NIC> 中:

```
ETHTOOL_OPTS="-s ${
 DEVICE }
  wol g"
```

设计注意事项

当您计划在 Remote PC Access 中使用局域网唤醒时，请注意以下事项：

- 多个计算机目录可以使用相同的局域网唤醒主机连接。
- 要使一台 PC 唤醒另一台 PC，两台 PC 必须位于同一子网中，并使用相同的局域网唤醒主机连接。这些 PC 是在相同还是不同的计算机目录中并不重要。
- 将主机连接分配给特定区域。如果您的部署中包含多个区域，则需要在每个区域中使用局域网唤醒主机连接。这同样适用于计算机目录。
- 幻数据包使用全局广播地址 255.255.255.255 进行广播。确保该地址未被阻止。
- 子网中必须至少打开一台 PC（对于每个局域网唤醒连接），才能唤醒该子网中的计算机。

操作注意事项

下面是使用局域网唤醒功能的注意事项：

- VDA 必须至少注册一次，才能使用集成的局域网唤醒功能唤醒 PC。
- 局域网唤醒功能只能用于唤醒 PC。该功能不支持其他电源操作，例如重新启动或关闭。
- 创建局域网唤醒连接后，该功能在 Studio 中可见。但是，不支持在 Studio 中编辑其属性。
- 幻数据包通过以下两种方式之一发送：
  - 当用户尝试启动到其 PC 的会话并且 VDA 未注册时
  - 当管理员从 Studio 或 PowerShell 手动发送打开电源命令时
- 由于 Delivery Controller 不知道 PC 的电源状态，因此，Studio 在电源状态下显示不支持。Delivery Controller 使用 VDA 注册状态来确定 PC 是打开还是关闭。

更多资源

下面是 Remote PC Access 的其他资源：

- 解决方案设计指南：Remote PC Access 设计决策。
- Remote PC Access 体系结构的示例：Citrix Remote PC Access 解决方案的参考体系结构。

# 打印

September 1, 2022

本文提供有关打印最佳实践的信息。

## 安装

Linux VDA 要求同时启用 **cups** 和 **foomatic** 过滤器。在安装 VDA 时安装过滤器。还可以根据分发情况手动安装过滤器。例如：

在 **RHEL 7** 上：

```
sudo yum -y install cups

sudo yum -y install foomatic-filters
```

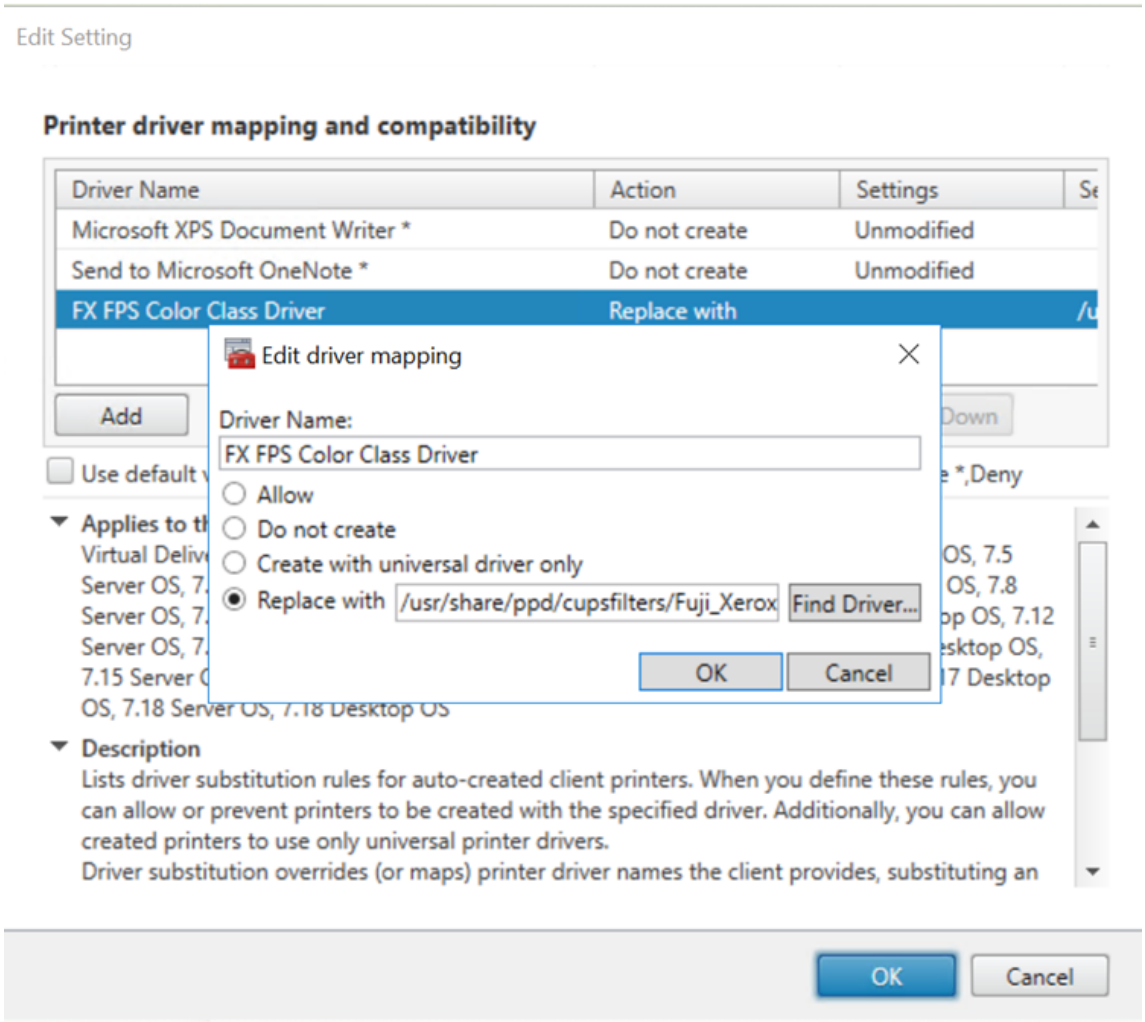
## 配置

Citrix 提供了三种类型的通用打印机驱动程序（postscript、pcl5 和 pcl6）。但是，通用打印机驱动程序可能与您的客户端打印机不兼容。在这种情况下，早期版本中的唯一选择为编辑 ~/.CtxlpProfile$CLIENT_NAME 配置文件。自版本 1906 起，可以选择改为在 Citrix Studio 中配置打印机驱动程序映射和兼容性策略。

要在 Citrix Studio 中配置打印机驱动程序映射和兼容性策略，请执行以下操作：

1. 选择打印机驱动程序映射和兼容性策略。
2. 单击添加。
3. 使用客户端打印机的驱动程序名称填写驱动程序名称。如果使用适用于 Linux 的 Citrix Workspace 应用程序，请改为填写打印机名称。
4. 选择替换为并键入 VDA 上的驱动程序文件的绝对路径。




> 注意：
>
> - 仅支持 PPD 驱动程序文件。
> - 不支持打印机驱动程序映射和兼容性策略的其他选项。只有替换为生效。

### 使用情况

可以从已发布的桌面和已发布的应用程序打印。仅客户端默认打印机会映射到 Linux VDA 会话。对于桌面和应用程序，打印机名称不同。

- 对于发布的桌面：
  `CitrixUniversalPrinter:$CLIENT_NAME:dsk$SESSION_ID`
- 对于发布的应用程序：
  `CitrixUniversalPrinter:$CLIENT_NAME:app$SESSION_ID`

> 注意：
>
> 如果同一用户同时打开了已发布的桌面和已发布的应用程序，会话可以访问两种打印机。无法在已发布的应用程序会话中的桌面打印机上打印，也无法在已发布的桌面会话中的应用程序打印机上打印。

### 故障排除

#### 无法打印

打印无法正常工作时，请检查打印守护程序 **ctxlpmngt** 和 CUPS 框架。

打印守护程序 **ctxlpmngt** 是一个按会话进程，必须在会话期间内运行。运行以下命令以确认打印守护程序是否正在运行。如果 **ctxlpmngt** 未运行，请从命令行中手动启动 **ctxlpmngt**。

```
ps -ef | grep ctxlpmngt
```

如果仍无法打印，请检查 CUPS 框架。**ctxcups** 服务用于打印机管理，并与 Linux CUPS 框架通信。此进程在每个计算机上有一个，可通过运行以下命令进行检查：

```
service ctxcups status
```

#### 收集 **CUPS** 日志的额外步骤

要收集 CUPS 日志，请运行以下命令以配置 CUPS 服务文件。否则，CUPS 日志无法记录在 **hdx.log** 中：

```
sudo service cups stop

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

PrivateTmp=false

sudo service cups start

sudo systemctl daemon-reload
```

> 注意：
>
> 此配置仅在出现问题时收集完整的打印日志时设置。在正常情况下，不建议做此配置，因为这会损害 CUPS 安全性。

#### 打印输出为乱码

打印机驱动程序不兼容可能会导致输出乱码。系统中为每个用户提供了驱动程序配置，该配置可通过编辑 **~/.CtxlpProfile$CLIENT_NAME** 配置文件进行配置。

```
[DEFAULT_PRINTER]

printername=

model=

ppdpath=

drivertype=
```

> 重要：
>
> **printername** 字段包含的是当前客户端默认打印机的名称。它是一个只读值。请勿编辑。
>
> 不能同时设置字段 **ppdpath**、**model** 和 **drivertype**，因为映射的打印机只能使用其中一个字段。

- 如果通用打印机驱动程序与客户端打印机不兼容，请使用 **model**= 选项配置本机打印机驱动程序的型号。可以使用 **lpinfo** 命令查找打印机的当前型号名称：

  ```
  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
  ```

  然后可以设置型号以与打印机匹配：

  ```
  model=xerox/ph3115.ppd.gz
  ```

- 如果通用打印机驱动程序与客户端打印机不兼容，请配置本机打印机驱动程序的 PPD 文件路径。**ppdpath** 值是本机打印机驱动程序文件的绝对路径。

  例如，/home/tester/NATIVE_PRINTER_DRIVER.ppd 下存在一个 **ppd** 驱动程序：

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

- Citrix 提供了三种类型的通用打印机驱动程序（postscript、pcl5 和 pcl6）。可以根据打印机属性配置驱动程序类型。

  例如，如果客户端默认打印机驱动程序类型为 PCL5，请将 **drivertype** 设置为：

  ```
  drivertype=pcl5
  ```

输出大小为零

尝试使用其他类型的打印机。并尝试使用 CutePDF 和 PDFCreator 之类的虚拟打印机以确定此问题是否与打印机驱动程序有关。

打印作业取决于客户端默认打印机的打印机驱动程序。务必确定当前活动驱动程序类型。如果客户端打印机使用的是 PCL5 驱动程序，而 Linux VDA 选择的是 Postscript 驱动程序，则会出现问题。

如果打印机驱动程序类型正确，可以执行以下步骤来确定问题：

1. 登录到已发布的桌面会话。
2. 运行 **vi ~/.CtxlpProfile$CLIENT_NAME** 命令。
3. 添加以下字段以在 Linux VDA 上保存后台打印文件：

   ```
   deletespoolfile=no
   ```

4. 注销并重新登录以加载配置更改。
5. 打印文档以重现问题。打印后，将有一个 spool 文件保存在 **/var/spool/cups-ctx/$logon_user/$spool_file** 下。
6. 检查后台打印是否为空。如果 spool 文件大小为零，表示有问题。请联系 Citrix 支持（并提供打印日志）以获取更多指导。
7. 如果 spool 大小不为零，则将该文件复制到客户端。spool 文件内容取决于客户端默认打印机的打印机驱动程序类型。如果映射的打印机（本机）驱动程序是 postscript，可以直接在 Linux 操作系统上打开 spool 文件。检查内容是否正确。

   如果后台打印文件是 PCL，或者客户端操作系统是 Windows，则将后台打印文件复制到客户端，并使用不同的打印机驱动程序在客户端打印机上打印该文件。
8. 将映射的打印机更改为使用不同的打印机驱动程序。下例以 PostScript 客户端打印机为例：

   a) 登录活动会话，在客户端桌面上打开浏览器。

   b) 打开打印管理门户：

      ```
      localhost:631
      ```

   c) 选择映射的打印机 **CitrixUniversalPrinter:$ClientName:app/dsk$SESSION_ID** 和 **Modify Printer**（修改打印机）。此操作要求使用管理员权限。

   d) 保持 cups-ctx 连接，然后单击“Continue”（继续）以更改打印机驱动程序。

   e) 在 **Make**（制造商）和 **Model**（型号）字段中，从 Citrix UPD 驱动程序中选择不同的打印机驱动程序。例如，如果安装了 CUPS-PDF 虚拟打印机，可以选择“Generic CUPS-PDF Printer”（通用 CUPS-PDF 打印机）驱动程序。保存更改。

   f) 如果此过程成功，则会在 **.CtxlpProfile$CLIENT_NAME** 中配置驱动程序的 PPD 文件路径，以允许映射的打印机使用新选择的驱动程序。

### 已知问题

下面是已确定的在 Linux VDA 上打印时存在的问题：

**CTXPS** 驱动程序与部分 **PLC** 打印机不兼容

如果发生打印输出损坏，请将打印机驱动程序设置为制造商提供的本机打印机驱动程序。

打印大文档时打印速度较慢

在本地客户端打印机上打印大文档时，文档会通过服务器连接进行传输。如果连接的速度很慢，传输可能需要很长时间。

在其他会话中看到打印机和打印作业通知

Linux 的会话概念与 Windows 操作系统不同。因此，所有用户都会获得系统范围的通知。您可以禁用这些通知，方法是更改 CUPS 配置文件：**/etc/cups/cupsd.conf**。

找到文件中配置的当前策略名称：

```
DefaultPolicy default
```

如果策略名称为 *default*，则将以下行添加到默认策略 XML 块中：

```
<Policy default>

     # Job/subscription privacy...

     JobPrivateAccess default

     JobPrivateValues default

     SubscriptionPrivateAccess default

     SubscriptionPrivateValues default

     … …

     <Limit Create-Printer-Subscription>

          Require user @OWNER

          Order deny,allow

     </Limit>

     <Limit All>

          Order deny,allow

     </Limit>

</Policy>
```

## 文件复制和粘贴

May 18, 2021

使用 Citrix Virtual Apps and Desktops 2006 和适用于 Windows 的 Citrix Workspace 应用程序 1903 或更高版本，用户现在可以在会话与本地客户端之间复制和粘贴文件。复制和粘贴通过右键菜单或键盘快捷方式发挥作用。

要成功复制和粘贴文件，请确保：

- 文件的最大数量不超过 20。
- 最大文件大小不超过 200 MB。

### 支持的平台

文件复制和粘贴功能适用于：

- RHEL 7.8
- SLES 12.5
- Ubuntu 16.04
- Ubuntu 18.04
- Debian 10

### 相关策略

以下剪贴板策略与配置功能相关。有关剪贴板策略的详细信息，请参阅策略支持列表。

- 客户端剪贴板重定向
- 剪贴板选择更新模式
- 主选定内容更新模式

### 限制

- 不支持剪切。剪切文件请求被视为复制。
- 不支持拖放。
- 不支持复制目录。

## 文件传输

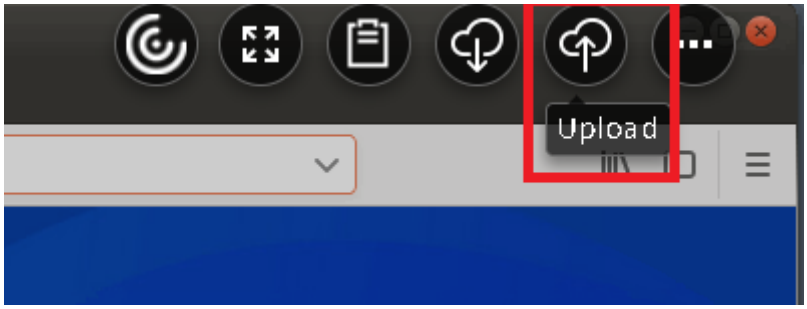
November 8, 2021

支持在 Linux VDA 和客户端设备之间进行文件传输。当客户端设备运行支持 HTML5 沙盒属性的 Web 浏览器时，可以使用此功能。HTML5 沙盒属性允许用户使用适用于 HTML5 和 Chrome 的 Citrix Workspace 应用程序访问虚拟桌面和应用程序。

> 注意：
>
> 适用于 HTML5 和 Chrome 的 Citrix Workspace 应用程序的文件传输可用。

在已发布的应用程序和桌面会话中，文件传输允许在 Linux VDA 与客户端设备之间上载和下载文件。要将文件从客户端设备上载到 Linux VDA，请单击 Citrix Workspace 应用程序工具栏上的上载图标，然后从文件对话框中选择所需的文件。要将文件从 Linux VDA 下载到客户端设备，请单击下载图标。可以在上载或下载过程中添加文件。一次最多可以传输 100 个文件。




> 注意：
>
> 要在 Linux VDA 与客户端设备之间上载和下载文件，请确保启用 Citrix Workspace 应用程序的工具栏。
> 可以使用允许拖放文件的 Citrix Workspace 应用程序版本。

自动下载是文件传输的增强功能。下载或移动到 VDA 上的保存到我的设备目录的文件将自动传输到客户端设备。

> 注意：
>
> 自动下载要求将允许在桌面与客户端之间传输文件和从桌面下载文件策略设置为允许。

下面是自动下载的一些用例：

- 下载文件以保存到我的设备

  在已发布的桌面和 Web 浏览器应用程序会话中，从 Web 站点下载的文件可以保存到 VDA 上的保存到我的设备目录中，以便自动传输到客户端设备。要实现自动下载，请将会话中 Web 浏览器的默认下载目录设置为保存到我的设备，并在运行适用于 HTML5 或 Chrome 的 Citrix Workspace 应用程序的 Web 浏览器中设置本地下载目录。

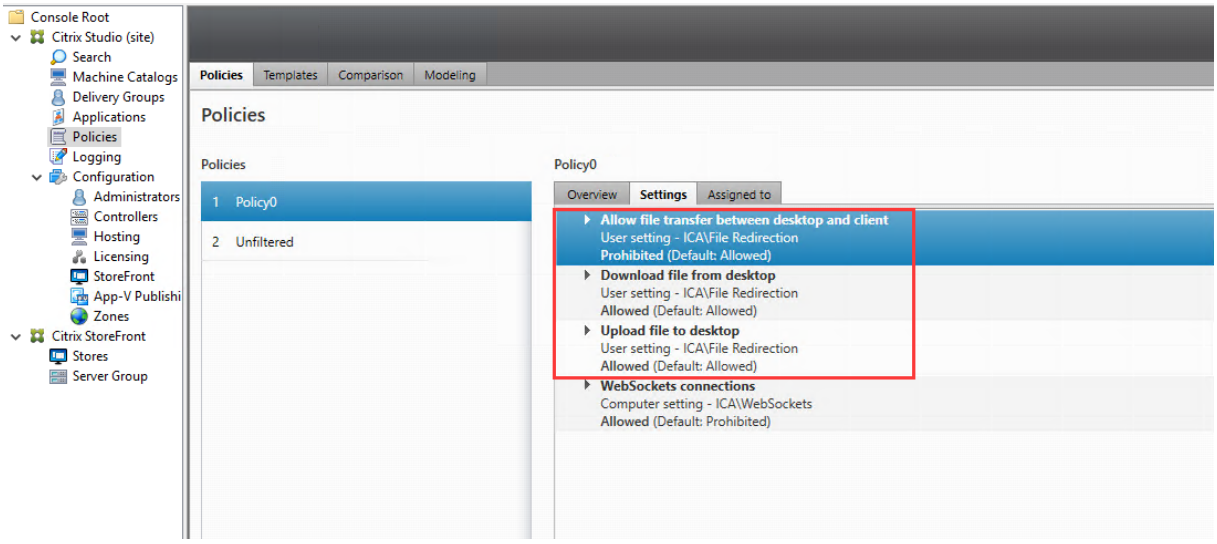
- 将文件移动或复制到保存到我的设备

  在已发布的桌面会话中，选择目标文件，然后将其移动或复制到保存到我的设备目录中，以便在客户端设备上可用。

### 文件传输策略

可以使用 Citrix Studio 设置文件传输策略。默认启用文件传输。




策略说明：

- 允许在桌面与客户端之间传输文件。允许或阻止用户在 Citrix Virtual Apps and Desktops 会话与其设备之间传输文件。
- 从桌面下载文件。允许或阻止用户将文件从 Citrix Virtual Apps and Desktops 会话下载到其设备。
- 将文件上载到桌面。允许或阻止用户将文件从其设备上载到 Citrix Virtual Apps and Desktops 会话。

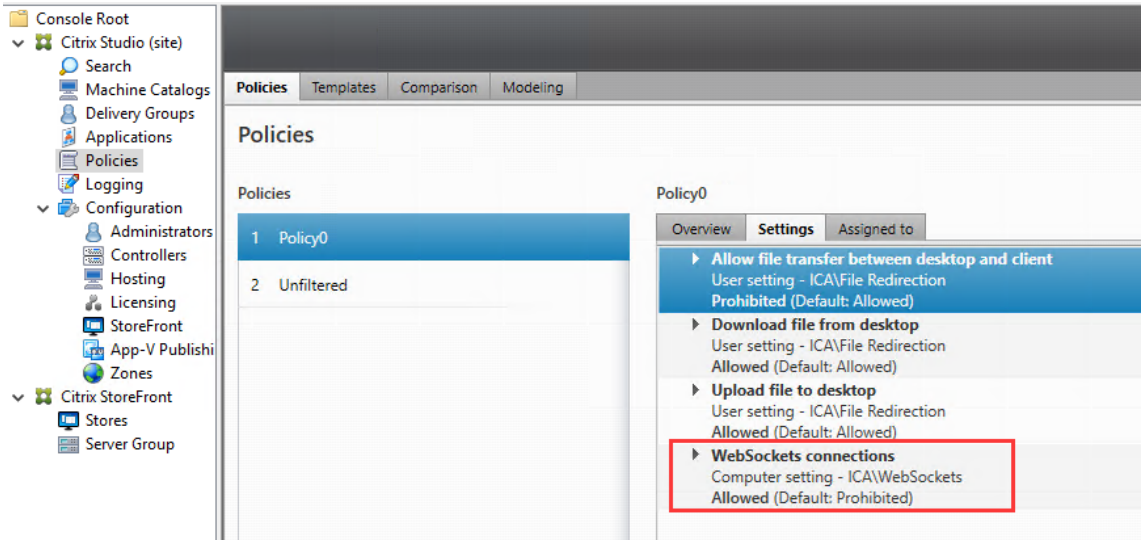
> 注意：
>
> 为确保从桌面下载文件和将文件上载到桌面策略生效，请将允许在桌面与客户端之间传输文件策略设置为允许。

### 使用情况

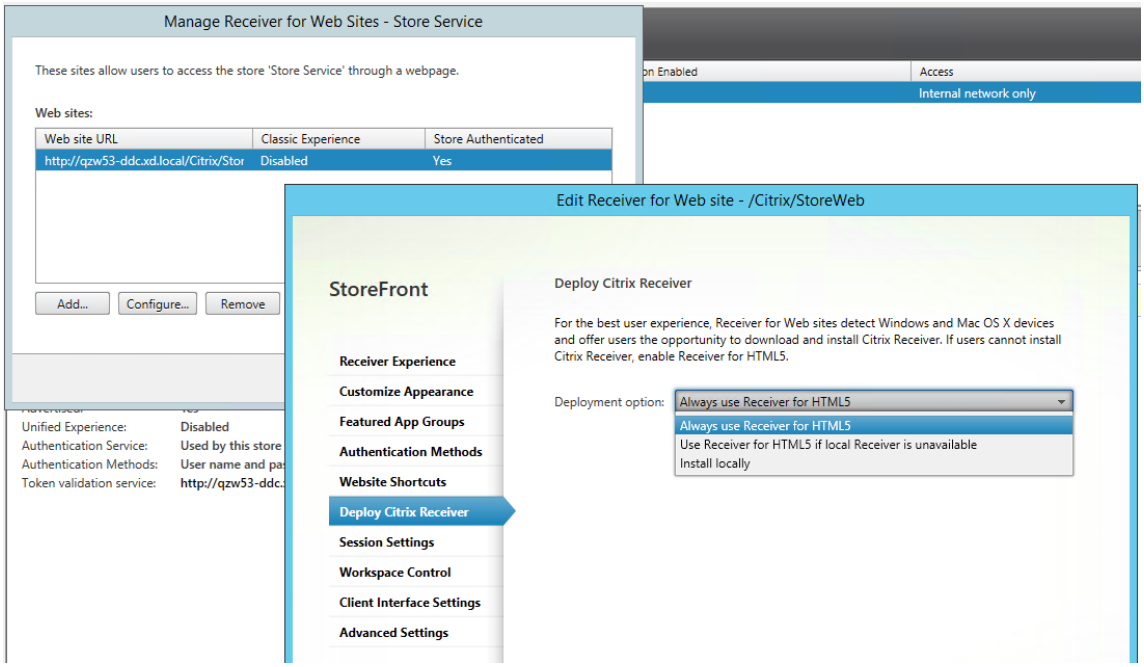
要通过适用于 **HTML5** 的 **Citrix Workspace** 应用程序使用文件传输功能，请执行以下操作：

1. 在 Citrix Studio 中，将 **WebSocket** 连接策略设置为允许。




2. 在 Citrix Studio 中，通过上述文件传输策略启用文件传输功能。
3. 在 Citrix StoreFront 管理控制台中，单击应用商店，选择管理 **Receiver for Web** 站点节点，然后通过选择始终使用 **Receiver for HTML5** 选项来启用 Citrix Receiver for HTML5。




4. 启动虚拟桌面或 Web 浏览器应用程序会话。在 Linux VDA 与客户端设备之间执行一个或多个文件传输。

要通过适用于 **Chrome** 的 **Citrix Workspace** 应用程序使用文件传输功能，请执行以下操作：

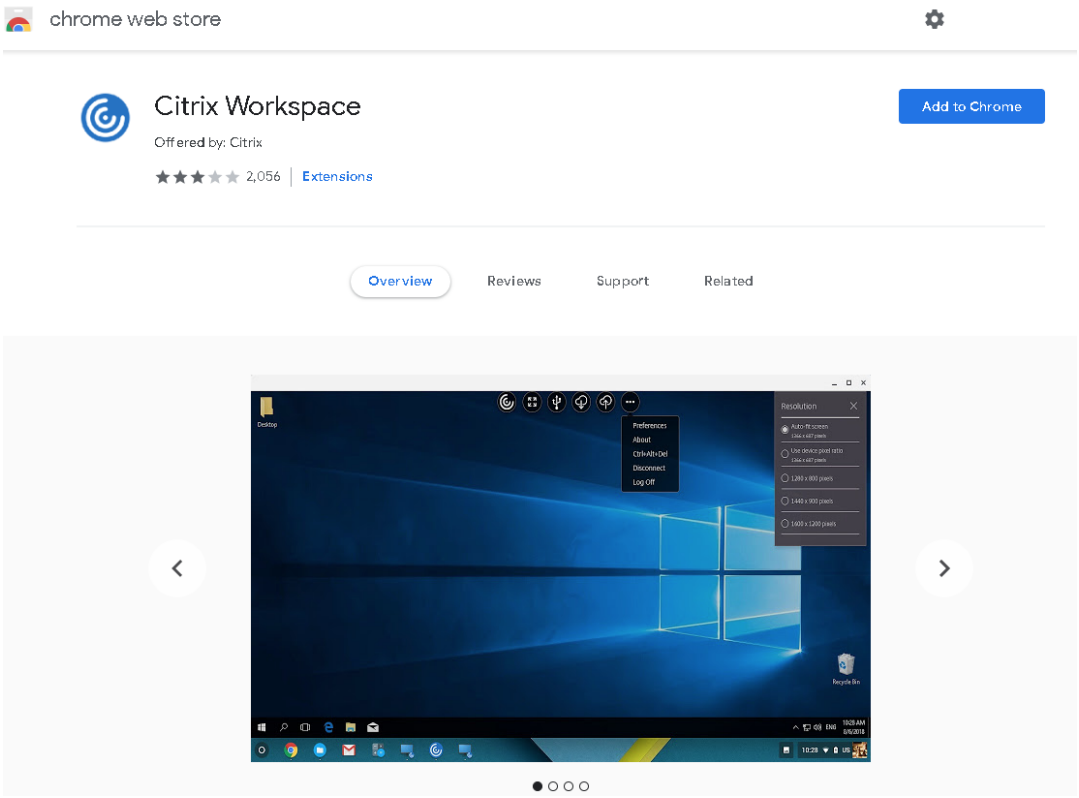
1. 通过上述文件传输策略启用文件传输功能。
2. 从 Chrome 网上应用店中获取 Citrix Workspace 应用程序。

   如果您已将适用于 Chrome 的 Citrix Workspace 应用程序添加到“Chrome 应用程序”页面，请跳过此步骤。

a) 在 Google Chrome 的搜索框中键入适用于 **Chrome** 的 **Citrix Workspace**。单击搜索图标。

b) 在搜索结果中，单击指向 Chrome 网上应用店（提供 Citrix Workspace 应用程序）的 URL。




c) 单击添加到 **Chrome** 以将 Citrix Workspace 应用程序添加到 Google Chrome 中。

3. 在“Chrome 应用程序”页面上单击“适用于 Chrome 的 Citrix Workspace 应用程序”。
4. 键入要连接的 StoreFront 应用商店的 URL。

   如果您之前键入了该 URL，请跳过此步骤。
5. 启动虚拟桌面或应用程序会话。在 Linux VDA 与客户端设备之间执行一个或多个文件传输。

## PDF 打印

April 21, 2021

使用支持 PDF 打印的 Citrix Workspace 应用程序版本，可以打印从 Linux VDA 会话内部转换的 PDF。会话打印作业将被发送到安装了 Citrix Workspace 应用程序的本地计算机。在本地计算机上，可以使用所选 PDF 查看器打开 PDF，并在所选打印机上进行打印。

Linux VDA 支持在以下版本的 Citrix Workspace 应用程序上进行 PDF 打印：

- Citrix Receiver for HTML5 版本 2.4 到 2.6.9、适用于 HTML5 的 Citrix Workspace 应用程序 1808 及更高版本
- Citrix Receiver for Chrome 版本 2.4 到 2.6.9、适用于 Chrome 的 Citrix Workspace 应用程序 1808 及更高版本
- 适用于 Windows 的 Citrix Workspace 应用程序 1905 及更高版本

### 配置

除了使用支持 PDF 打印的 Citrix Workspace 应用程序版本外，还要在 Citrix Studio 中启用以下策略：

- 客户端打印机重定向（默认启用）
- 自动创建 **PDF** 通用打印机（默认禁用）

启用这些策略后，当您在已启动的会话中单击打印时，打印预览将显示在本地计算机上，以便您选择打印机。有关设置默认打印机的信息，请参阅 Citrix Workspace 应用程序文档。

## 配置图形

April 18, 2024

本文指导如何完成 Linux VDA 显卡配置和微调。

有关详细信息，请参阅系统要求和安装概述部分。

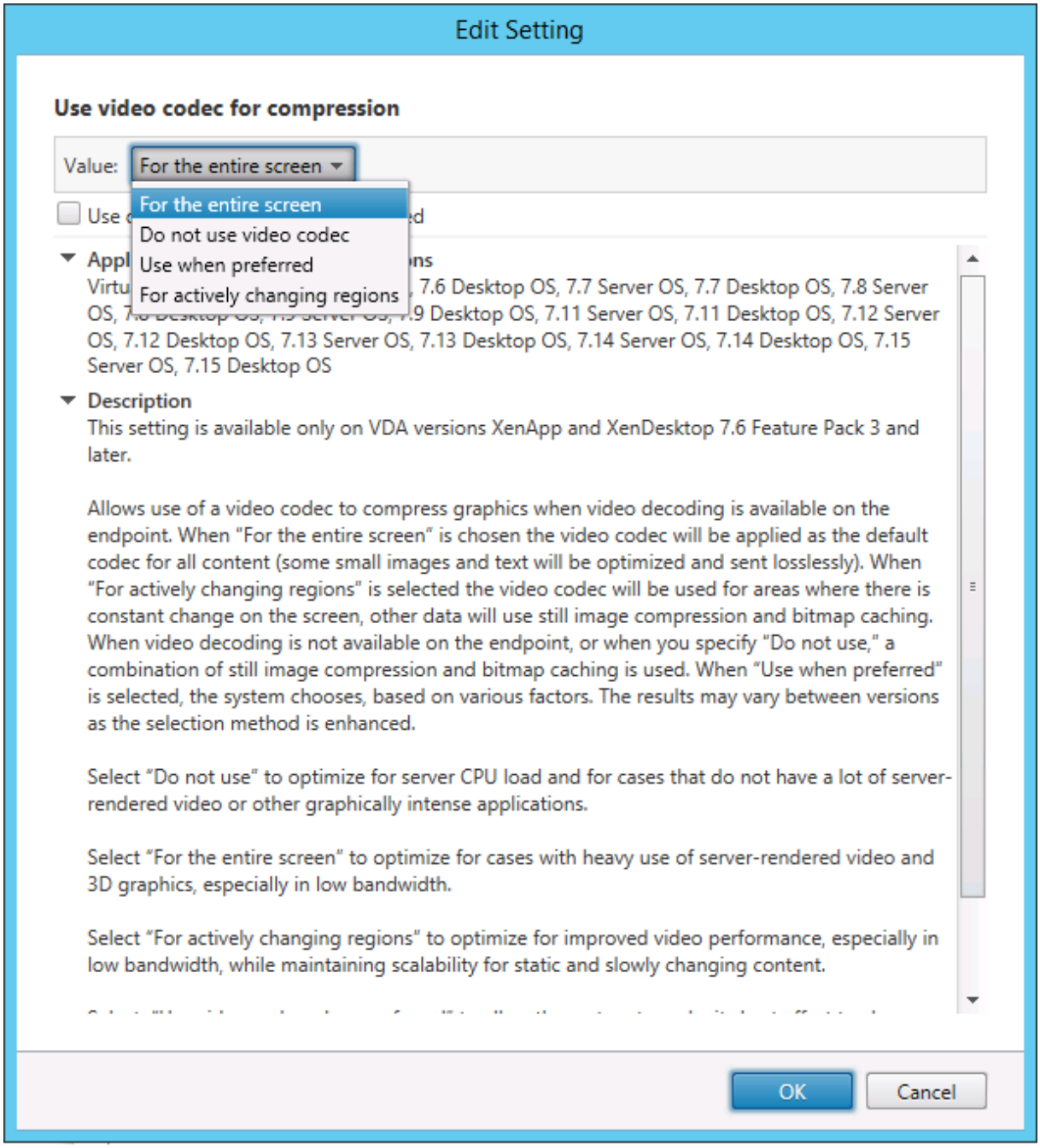
### 配置

Thinwire 是 Linux VDA 中使用的显示内容远程处理技术。该技术允许一台计算机上生成的图形传输（通常跨网络）到另一台计算机上进行显示。

使用视频编解码器进行压缩图形策略设置默认图形模式，并针对不同的用例提供以下选项：

- 偏好时使用。这是默认设置。无需执行其他配置。保持此设置可确保为所有 Citrix 连接选择 Thinwire，且 Thinwire 已针对典型桌面工作负载在可扩展性、带宽和卓越图像质量方面经过优化。
- 针对整个屏幕。为 Thinwire 提供全屏 H.264 或 H.265，以针对改进用户体验和带宽使用情况进行优化，尤其是在大量使用 3D 图形的情况下。
- 针对主动变化的区域。Thinwire 中的自适应显示技术识别移动图像（视频、动态 3D），并只在图像移动的屏幕部分使用 H.264。选择性使用 ***H.264*** 视频编解码器支持 HDX Thinwire 检测屏幕的频繁更新部分（例如视频内容），并使用 H.264 视频编解码器对其进行编码。对于屏幕的其余部分（包括文本和摄影图片），仍继续使用图像压缩（JPEG、RLE）和位图缓存。用户可获得以下益处：使用较低带宽、提高视频内容质量以及在其他位置获得无损文本或高质量图像。要启用此功能，请将策略设置使用视频编解码器进行压缩更改为偏好时使用（默认设置）或针对主动变化的区域。有关详细信息，请参阅图形策略设置。




一些其他策略设置（包括以下视频显示策略设置）可以用于对显示远程处理技术的性能进行完善：

- 简单图形的首选颜色深度
- 目标帧速率
- 视觉质量

### 在 **Thinwire** 中将 **H.264** 用于“无损构建”

默认情况下，现在对于移动图像，视觉质量策略设置的无损构建首选项为 H.264 而不是 JPEG。

H.264 编码可提供卓越的图像质量。使用视频编解码器进行压缩策略控制该首选项，默认设置为偏好时使用。要强制无损构建使用 JPEG，请将使用视频编解码器进行压缩策略设置为不使用视频编解码器。如果客户端不支持选择性 H.264，无论策略设置是什么，无损构建都将回退到 JPEG。Citrix Receiver for Windows 4.9 到 4.12、Citrix Receiver for Linux 13.5 到 13.10、适用于 Windows 的 Citrix Workspace 应用程序 1808 及更高版本，以及适用于 Linux 的 Citrix Workspace 应用程序 1808 及更高版本支持选择性 H.264。有关视觉质量和使用视频编解码器进行压缩策略设置的详细信息，请参阅视觉显示策略设置和图形策略设置。

### 支持 **H.265** 视频编解码器

从 7.18 版起，Linux VDA 支持使用 H.265 视频编解码器对远程图形和视频进行硬件加速。您可以在 Citrix Receiver for Windows 4.10 到 4.12 和适用于 Windows 的 Citrix Workspace 应用程序 1808 及更高版本上使用此功能。要从此功能受益，请在 Linux VDA 和客户端上启用此功能。如果客户端的 GPU 不支持使用 DXVA 接口进行 H.265 解码，“图形的 H.265 解码”策略设置将被忽略，会话将回退到使用 H.264 视频编解码器。有关详细信息，请参阅 H.265 视频编码。

要在 VDA 上启用 H.265 硬件编码，请执行以下操作：

1. 启用使用视频编解码器的硬件编码策略。
2. 启用针对 **3D** 图形工作负载优化策略。
3. 确保使用视频编解码器进行压缩策略采用默设置，或设置为针对整个屏幕。
4. 确保视觉质量策略未设置为无损构建或始终无损。

要在客户端上启用 H.265 硬件编码，请参阅 H.265 视频编码。

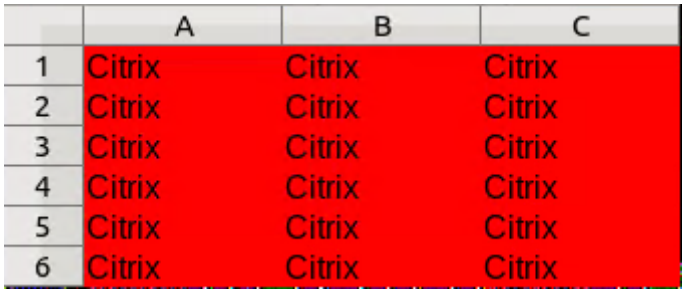
### 支持 **YUV444** 软件编码

Linux VDA 支持 YUV444 软件编码。YUV 编码方案会向每个像素分配亮度和颜色值。在 YUV 中，Y 表示亮度或 luma 值，UV 表示颜色或 chroma 值。您可以在 Citrix Receiver for Windows 4.10 到 4.12 和适用于 Windows 的 Citrix Workspace 应用程序 1808 及更高版本上使用 Linux VDA 的这一功能。

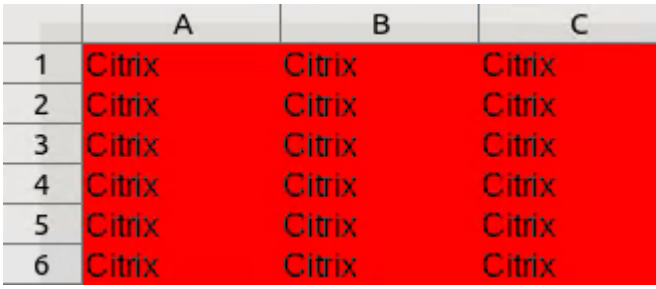
每个唯一的 Y、U 和 V 值均包含 8 位数或一个字节。YUV444 数据格式以 24 位/像素的速度传输。YUV422 数据格式可在两个像素之间共享 U 和 V 值，从而导致平均传输速率为 16 位/像素。下表显示了 YUV444 和 YUV420 之间的直观比较。

YUV444




YUV420




要在 VDA 上启用 YUV444 软件编码，请执行以下操作：

1. 确保将使用视频编解码器进行压缩策略设置为针对整个屏幕。
2. 确保将视觉质量策略设置为始终无损或无损构建。

根据带宽估算值调整平均位速率

通过根据带宽估算值调整平均位速率，Citrix 增强了 HDX 3D Pro 硬件编码。

使用 HDX 3D Pro 硬件编码时，VDA 可以间歇性地估计网络带宽并根据带宽估算值调整已编码的帧的位速率。这一新增功能提供了一种平衡清晰度和流畅度的机制。

默认情况下启用此功能。要禁用此功能，请运行以下命令：

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\SYSTEM\
    CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "
    DisableReconfigureEncoder" -d "0x00000001" --force
```

除了使用此功能之外，您还可以运行以下命令来调整清晰度和流畅度。**AverageBitRatePercent** 和 **MaxBitRate-Percent** 参数将设置带宽使用情况的百分比。设置的值越高，图形就越清晰，但流畅度越低。建议的设置范围为 50 到 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
```

在调整平均位速率时，如果您的屏幕画面保持不变，则最近的帧将处于低质量状态，因为未发送任何新的帧。通过重新配置并即时发送最高质量的最新帧，锐化支持可以解决此问题。

有关 Linux VDA Thinwire 支持的策略的完整列表，请参阅策略支持列表。

有关 Linux VDA 上的多监视器支持配置的信息，请参阅 CTX220128。

故障排除

检查正在使用哪种图形模式

运行以下命令来检查正在使用哪种图形模式（**0** 表示 TW+；**1** 表示全屏视频编解码器）：

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

结果类似于：

```
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"
-v "GraphicsMode"-d "0x00000000"--force
```

检查是否正在使用 **H.264**

运行以下命令来检查是否正在使用 H.264（**0** 表示未在使用；**1** 表示正在使用）：

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

结果类似于：

```
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"
-v "H264"-d "0x00000000"--force
```

检查是否正在使用 **H.265**

运行以下命令来检查是否正在使用全屏 H.265（**0** 表示未使用；**1** 表示正在使用）：

```
sudo /opt/Citrix/VDA/bin/ctxreg dump | grep H265
```

结果类似于：

```
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"
-v "H265"-d "0x00000000"--force
```

检查正在使用哪种 **YUV** 编码方案

运行以下命令来检查正在使用哪种 YUV 编码方案（**0** 表示 YUV420；**1** 表示 YUV422；**2** 表示 YUV444）：

注意：仅当正在使用视频编解码器时，YUVFormat 的值才有意义。

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

结果类似于：

```
create -k "HKLM\Software\Citrix\Ica\Session\4\Graphics"-t "REG_DWORD"
-v "YUVFormat"-d "0x00000000"--force
```

检查是否正在使用 **YUV444** 软件编码

运行以下命令来检查是否正在使用 YUV444 软件编码：

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

当 YUV444 处于运行状态时，结果类似于：

```
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
```

检查是否在为 **3D Pro** 使用硬件编码

运行以下命令（**0** 表示未使用；**1** 表示正在使用）：

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

结果类似于：

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

另一个方法是使用 **nvidia-smi** 命令。如果正在使用硬件编码，输出将类似于以下内容：

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

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

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


```

```
+-----------------------------------------------------------------------------+
| Processes:                                                       GPU Memory |
|  GPU       PID  Type  Process name                             Usage      |
|=============================================================================|
|    0      2164  C+G   /usr/local/bin/ctxgfx                    106MiB |
|    0      2187    G   Xorg                                       85MiB |
+-----------------------------------------------------------------------------+
```

确认 **NVIDIA GRID** 图形驱动程序是否已正确安装

要确认 NVIDIA GRID 图形驱动程序是否已正确安装，请运行 **nvidia-smi**。结果类似于：

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

为显卡设置正确的配置：

`etc/X11/ctx-nvidia.sh`

#### **HDX 3D Pro** 多监视器重绘问题

如果在非主监视器屏幕上发生重绘问题，请检查 NVIDIA GRID 许可证是否可用。

#### 检查 **Xorg** 错误日志

Xorg 的日志文件命名类似于 **Xorg.{DISPLAY}.log**，位于 **/var/log/** 文件夹中。

#### 已知问题及限制

#### 对于 **vGPU**，**Citrix Hypervisor** 本地控制台显示 **ICA** 桌面会话屏幕

解决方法：通过运行以下命令禁用 VM 的本地 VGA 控制台：

```
xe vm-param-set uuid=<vm-uuid> platform:vgpu_extra_args="disable_vnc=1"
```

#### **NVIDIA K2** 显卡不支持在直通模式下进行 **YUV444** 硬件编码

通过策略设置启用了无损构建时，在使用 NVIDIA K2 显卡的情况下，用户启动应用程序/桌面会话时会出现黑屏或灰屏。出现该问题是因为 NVIDIA K2 显卡不支持在直通模式下进行 YUV444 硬件编码。有关详细信息，请参阅视频编码和解码 GPU 支持列表。

#### 登录时 **Gnome 3** 桌面显示很慢

这是 Gnome 3 桌面会话启动的限制。

#### 在调整 **Citrix Workspace** 应用程序窗口大小时，有些 **OpenGL/WebGL** 应用程序无法很好地呈现

调整 Citrix Workspace 应用程序窗口大小会改变屏幕分辨率。NVIDIA 专用驱动程序会更改某些内部状态，并可能要求应用程序做出相应的响应。例如，WebGL 库元素 **lightgl.js** 可能会生成错误，指示 `Rendering to this texture is not supported (incomplete frame buffer)`。

## **Thinwire** 渐进式显示

November 8, 2021

会话交互性在低带宽或高延迟连接中会降级。例如，对于带宽小于 2 Mbps 或延迟超过 200 毫秒的连接，在 Web 页面上滚动可能会变得缓慢、无响应或不稳定。键盘和鼠标操作可能会滞后于图形更新。

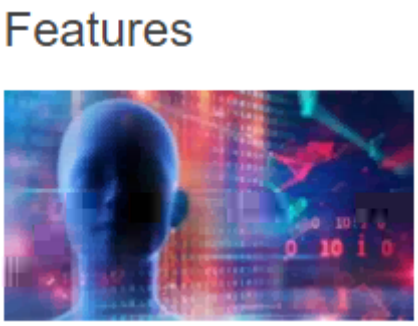
在 7.17 版中，能够使用策略设置来降低带宽占用量，方法是将会话配置为低视觉质量，或者设置较低的颜色深度（16 位或 8 位图形）。但是，您必须知道用户是在使用弱连接。HDX Thinwire 不会根据网络状况动态调整静态图像质量。

从 7.18 版起，当可用带宽低于 2 Mbps 或网络延迟超过 200 毫秒时，HDX Thinwire 会默认切换到渐进式更新模式。在此模式下：

- 所有静态图像都将深度压缩。
- 文本质量降低。

例如，在下图中，渐进式更新模式处于活动状态，字母 **F** 和 **e** 具有蓝色赝像，图像已深度压缩。此方法可显著降低带宽占用量，从而提高图像和文本的接收速度，并改进会话交互性。


用户停止与会话交互后，降级的图像和文本将逐渐锐化到无损效果。例如，在下图中，字母不再包含蓝色赝像，图像以源质量显示。




对于图像，锐化使用随机块状方法。对于文本，锐化各个字母或单词的各个部分。锐化过程是在多个帧上进行。此方法可避免在处理单个大型锐化帧时出现延迟。

瞬变影像（视频）仍通过自适应显示或选择性 H.264 进行管理。

### 如何使用渐进式模式

默认情况下，渐进式模式对视觉质量策略设置高、中（默认设置）和低而言处于随时准备使用状态。

在以下情况下强制关闭（不使用）渐进式模式：

- 视觉质量 = 始终无损或无损构建

- 简单图形的首选颜色深度 = 8 位
- 使用视频编解码器进行压缩 = 针对整个屏幕（需要全屏 H.264 时）

渐进式模式处于随时准备使用状态时，如果出现以下情况之一，则默认启用此模式：

- 可用带宽低于 2 Mbps
- 网络延迟超过 200 毫秒

发生模式切换后，在该模式中最短将花费 10 秒钟时间，即使网络条件暂时不利亦如此。

### 更改渐进式模式行为

可以通过运行以下命令更改渐进式模式的行为：

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\SOFTWARE\
    CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "
    ProgressiveDisplay" -d "<value>" --force
```

其中 <value>：

0 = 始终关闭（在任何情况下都不使用）

1 = 自动（根据网络状况切换，默认值）

2 = 始终开启

处于自动模式 (1) 时，可以运行以下命令之一更改切换渐进式模式时的阈值：

```
sudo /opt/Citrix/VDA/bin/ctxreg  create -k "HKEY_LOCAL_MACHINE\SYSTEM\
    CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "
    ProgressiveDisplayBandwidthThreshold" -d "<value>" --force
```

其中 <value> 为 < 阈值，以 Kbps 为单位 >（默认值 = 2048）

示例：4096 = 在带宽低于 4 Mbps 时开启渐进式模式

```
sudo /opt/Citrix/VDA/bin/ctxreg  create -k "HKEY_LOCAL_MACHINE\SOFTWARE
    \CurrentControlSet\Control\Citrix\Thinwire" -t "REG_DWORD" -v "
    ProgressiveDisplayLatencyThreshold" -d "<value>" --force
```

其中 <value> 为 < 阈值，以毫秒为单位 >（默认值 = 200）

示例：100 = 在网络延迟低于 100 毫秒时开启渐进式模式。

## 非 GRID 3D 图形

March 11, 2024

### 概述

通过此功能增强，Linux VDA 不仅支持 NVIDIA GRID 3D 卡，而且还支持非 GRID 3D 卡。

### 安装

要使用非 GRID 3D 图形功能，您必须：

- 安装 XDamage 作为必备条件。通常情况下，XDamage 是作为 XServer 的扩展程序。
- 在安装 Linux VDA 时将 `CTX_XDL_HDX_3D_PRO` 设置为 `Y`。有关环境变量的信息，请参阅步骤 3：设置运行时环境以完成安装。

### 配置

#### **Xorg** 配置文件

如果您的 3D 卡驱动程序是 NVIDIA，则会自动安装和设置配置文件。

#### 其他类型的 **3D** 卡

如果您的 3D 卡驱动程序不是 NVIDIA，则必须修改安装在 `/etc/X11/` 下面的四个模板配置文件：

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

以 **ctx-driver_name-1.conf** 为例，执行以下操作来修改模板配置文件：

1. 将 ***driver_name*** 替换为实际的驱动程序名称。

   例如，如果您的驱动程序名称为 `intel`，可以将配置文件名称更改为 `ctx-intel-1.conf`。

2. 添加视频驱动程序信息。

   每个模板配置文件都一个包含名为“Device”的部分，这部分被注释掉。本节介绍视频驱动程序信息。请在添加您的视频驱动程序信息之前先完成本节内容。要启用本部分内容，请执行以下操作：

   a) 请参阅制造商提供的 3D 卡指南了解配置信息。可以生成本机配置文件。确认在未使用 Linux VDA ICA 会话时，您的 3D 卡在使用本机配置文件的本地环境中是否能正常使用。

   b) 将本机配置文件的“Device”部分复制到 **ctx-driver_name-1.conf**。

3. 运行以下命令来设置注册表项，以使 Linux VDA 能够识别在步骤 1 中设置的配置文件名称。

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

启用非 **GRID 3D** 图形功能

默认情况下，非 GRID 3D 图形功能处于禁用状态。可以运行以下命令将 XDamageEnabled 设置为 1 来启用它。

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKEY_LOCAL_MACHINE\System\
    CurrentControlSet\Control\Citrix\XDamage" -t "REG_DWORD" -v "
    XDamageEnabled" -d "0x00000001" --force
```

故障排除

无图形输出或图形输出为乱码

如果您可以在本地运行 3D 应用程序，且所有配置均正确，则丢失图形输出或图形输出为乱码是因为缺陷。请使用 /opt/Citrix/VDA/bin/setlog 并将 GFX_X11 设置为“verbose”来收集跟踪信息以进行调试。

不能进行硬件编码

此功能仅支持软件编码。

## 配置策略

November 8, 2021

安装

按照安装文章来准备 Linux VDA。

依赖项

请务必在安装 Linux VDA 软件包之前安装这些依赖项。

**RHEL/CentOS**:

```
sudo yum -y install openldap

sudo yum -y install libxml2

sudo yum -y install cyrus-sasl

sudo yum -y install cyrus-sasl-gssapi
```

**SLES/SELD**:

```
sudo zypper install openldap2

sudo zypper install libxml2

sudo zypper install cyrus-sasl

sudo zypper install cyrus-sasl-gssapi
```

**Ubuntu**:

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

sudo apt-get install -y libsasl2-2

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

## 配置

### Citrix Studio 中的策略设置

要在 Citrix Studio 中设置策略，请执行以下操作：

1. 打开 **Citrix Studio**。
2. 选择策略面板。
3. 单击创建策略。
4. 根据策略支持列表设置策略。

### VDA 上的 LDAP 服务器设置

对于单域环境，VDA 上的 LDAP 服务器设置是可选的，但对于多域和多级林环境，该设置是必需的。策略服务需要此设置才能在这些环境中执行 LDAP 搜索。

安装 Linux VDA 软件包后，运行以下命令：

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

按建议的格式键入所有 LDAP 服务器：由空格分隔的 LDAP 完全限定域名 (FQDN)（带有 LDAP 端口）列表（例如 ad1.mycompany.com:389 ad2.mycomany.com:389）。

```
Checking CTX_XDL_LDAP_LIST... value not set.
The Virtual Delivery Agent by default queries DNS to discover LDAP servers, however if DNS is unable to provide
LDAP service records, you may provide a space-separated list of LDAP Fully Qualified Domain Names (FQDNs) with
LDAP port (e.g. ad1.mycompany.com:389).
If required, please provide the FQDN:port of at least one LDAP server. [<none>]:
```

还可以通过运行 **ctxreg** 命令将此设置直接写入注册表：

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

## 策略支持列表

November 8, 2021

### **Linux VDA** 策略支持列表

| Studio 策略 | 注册表项名称 | 类型 | 模块 | 默认值 |
|---|---|---|---|---|
| 使用客户端本地时间 | UseLocalTimeOfClient | 用户 | ICA\时区控制 | 使用服务器时区 |
| ICA 往返行程计算 | IcaRoundTripCheckEnabled | 计算机 | ICA\最终用户监视 | 已启用 (1) |
| ICA 往返行程计算间隔 | IcaRoundTripCheckPeriod | 计算机 | ICA\最终用户监视 | 15 |
| 空闲连接的 ICA 往返行程计算 | IcaRoundTripCheckWhenIdle | 计算机 | ICA\最终用户监视 | 已禁用 (0) |
| 总会话带宽限制 | LimitOverallBw | 用户 | ICA\带宽 | 0 |
| 音频重定向带宽限制 | LimitAudioBw | 用户 | ICA\带宽 | 0 |
| 音频重定向带宽限制百分比 | LimitAudioBwPercent | 用户 | ICA\带宽 | 0 |
| 客户端 USB 设备重定向带宽限制 | LimitUSBBw | 用户 | ICA\带宽 | 0 |
| 客户端 USB 设备重定向带宽百分比 | LimitUSBBwPercent | 用户 | ICA\带宽 | 0 |
| 剪贴板重定向带宽限制 | LimitClipbdBW | 用户 | ICA\带宽 | 0 |

| Studio 策略 | 注册表项名称 | 类型 | 模块 | 默认值 |
|---|---|---|---|---|
| 剪贴板重定向带宽限制百分比 | LimitClipbdBWPercent | 用户 | ICA\带宽 | 0 |
| 文件重定向带宽限制 | LimitCdmBw | 用户 | ICA\带宽 | 0 |
| 文件重定向带宽限制百分比 | LimitCdmBwPercent | 用户 | ICA\带宽 | 0 |
| 打印机重定向带宽限制 | LimitPrinterBw | 用户 | ICA\带宽 | 0 |
| 打印机重定向带宽限制百分比 | LimitPrinterBwPercent | 用户 | ICA\带宽 | 0 |
| WebSocket 连接 | AcceptWebSocketsConnections | 计算机 | ICA\WebSockets | 禁止 |
| WebSocket 端口号 | WebSocketsPort | 计算机 | ICA\WebSockets | 8008 |
| WebSocket 可信源服务器列表 | WSTrustedOriginServerList | 计算机 | ICA\WebSockets | * |
| ICA 保持活动状态 | SendICAKeepAlives | 计算机 | ICA 保持活动状态 | 不发送 ICA 保持活动状态消息 (0) |
| ICA 保持活动状态超时 | ICAKeepAliveTimeout | 计算机 | ICA 保持活动状态 | 60 秒 |
| ICA 侦听器端口号 | IcaListenerPortNumber | 计算机 | ICA | 1494 |
| HDX 自适应传输 | HDXoverUDP | 计算机 | ICA | Preferred(2) |
| 会话可靠性连接 | AcceptSessionReliabilityConnections | 计算机 | ICA\会话可靠性 | 允许 (1) |
| 重新连接 UI 透明度级别 | ReconnectionUiTransparencyLevel | 计算机 | ICA\客户端自动重新连接 | 80% |
| 会话可靠性端口号 | SessionReliabilityPort | 计算机 | ICA\会话可靠性 | 2598 |
| 会话可靠性超时 | SessionReliabilityTimeout | 计算机 | ICA\会话可靠性 | 180 秒 |
| 客户端自动重新连接 | AllowAutoClientReconnect | 用户 | ICA\客户端自动重新连接 | 允许 (1) |
| 客户端音频重定向 | AllowAudioRedirection | 用户 | 音频 | 允许 (1) |
| 客户端打印机重定向 | AllowPrinterRedir | 用户 | 打印 | 允许 (1) |
| 自动创建 PDF 通用打印机 | AutoCreatePDFPrinter | 用户 | 打印 | 已禁用 (0) |

| Studio 策略 | 注册表项名称 | 类型 | 模块 | 默认值 |
|---|---|---|---|---|
| 打印机驱动程序映射和兼容性 | DriverMappingList | 用户 | 打印 | `"Microsoft XPS Document Writer *, Deny;Send to Microsoft OneNote *, Deny"` |
| 客户端剪贴板重定向 | AllowClipboardRedir | 用户 | 剪贴板 | 允许 (1) |
| 客户端 USB 设备重定向 | AllowUSBRedir | 用户 | USB | 禁止 (0) |
| 客户端 USB 设备重定向规则 | USBDeviceRules | 用户 | USB | “\0” |
| 移动图像压缩 | MovingImageCompressionConfiguration | 用户 | Thinwire | 已启用 (1) |
| 额外颜色压缩 | ExtraColorCompression | 用户 | Thinwire | 已禁用 (0) |
| 目标最低帧速率 | TargetedMinimumFramesPerSecond | 用户 | Thinwire | 10 fps |
| 目标帧速率 | FramesPerSecond | 用户 | Thinwire | 30 fps |
| 视觉质量 | VisualQuality | 用户 | Thinwire | 中 (3) |
| 使用视频编解码器进行压缩 | VideoCodec | 用户 | Thinwire | 首选时使用 (3) |
| 使用视频编解码器的硬件编码 | UseHardwareEncodingForVideoCodec | 用户 | Thinwire | 已启用 (1) |
| 允许视觉无损压缩 | AllowVisuallyLosslessCompression | 用户 | Thinwire | 已禁用 (0) |
| 针对 3D 图形工作负载优化 | OptimizeFor3dWorkload | 用户 | Thinwire | 已禁用 (0) |
| 简单图形的首选颜色深度 | PreferredColorDepth | 用户 | Thinwire | 24 位/像素 (1) |
| 音频质量 | SoundQuality | 用户 | 音频 | 高 - 高清晰度音频 (2) |
| 客户端麦克风重定向 | AllowMicrophoneRedir | 用户 | 音频 | 允许 (1) |
| 最大会话数 | MaximumNumberOfSessions | 计算机 | 负载管理 | 250 |
| 并发登录数容差 | ConcurrentLogonsTolerance | 计算机 | 负载管理 | 2 |
| 启用控制器自动更新 | EnableAutoUpdateOfControllers | 计算机 | Virtual Delivery Agent 设置 | 允许 (1) |
| 剪贴板选择更新模式 | ClipboardSelectionUpdateMode | 用户 | 剪贴板 | 3 |

| Studio 策略 | 注册表项名称 | 类型 | 模块 | 默认值 |
|---|---|---|---|---|
| 主选定内容更新模式 | PrimarySelectionUpdateMode | 用户 | 剪贴板 | 3 |
| 最高 speex 质量 | MaxSpeexQuality | 用户 | 音频 | 5 |
| 自动连接客户端驱动器 | AutoConnectDrives | 用户 | 文件重定向/CDM | 已启用 (1) |
| 客户端光盘驱动器 | AllowCdromDrives | 用户 | 文件重定向/CDM | 允许 (1) |
| 客户端固定驱动器 | AllowFixedDrives | 用户 | 文件重定向/CDM | 允许 (1) |
| 客户端软盘驱动器 | AllowFloppyDrives | 用户 | 文件重定向/CDM | 允许 (1) |
| 客户端网络驱动器 | AllowNetworkDrives | 用户 | 文件重定向/CDM | 允许 (1) |
| 客户端驱动器重定向 | AllowDriveRedir | 用户 | 文件重定向/CDM | 允许 (1) |
| 只读客户端驱动器访问 | ReadOnlyMappedDrive | 用户 | 文件重定向/CDM | 已禁用 (0) |
| 自动显示键盘 | AllowAutoKeyboardPopUp | 用户 | MRVC | 已禁用 (0) |
| 允许在桌面与客户端之间传输文件 | AllowFileTransfer | 用户 | 文件传输 | 允许 |
| 从桌面下载文件 | AllowFileDownload | 用户 | 文件传输 | 允许 |
| 将文件上载到桌面 | AllowFileUpload | 用户 | 文件传输 | 允许 |
| 会话空闲计时器 | EnableSessionIdleTimer | 用户 | 会话计时器 | 已启用 (1) |
| 会话空闲计时器间隔 | SessionIdleTimerInterval | 用户 | 会话计时器 | 1440 分钟 |
| 断开会话计时器 | EnableSessionDisconnectTimer | 用户 | 会话计时器 | 已禁用 (0) |
| 断开会话计时器间隔 | SessionDisconnectTimerPeriod | 用户 | 会话计时器 | 1440 分钟 |

> 注意：
>
> 只有 Windows Virtual Delivery Agent (VDA) 支持通过用户数据报协议 (UDP) 传输音频。Linux VDA 不支持。有关详细信息，请参阅通过用户数据报协议 (UDP) 实时传输音频。

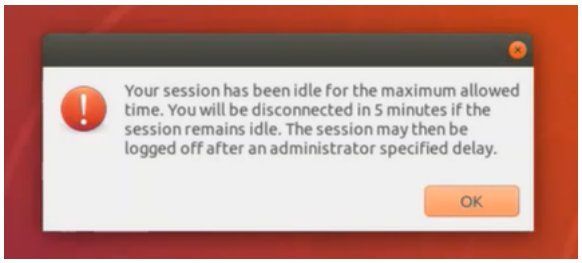
可以使用以下 Citrix 策略设置在 Citrix Studio 中配置会话连接计时器：

- 会话空闲计时器：确定是否对空闲会话强制实施时间限制。
- 会话空闲计时器时隔：设置空闲会话的时间限制。如果会话空闲计时器已启用，并且活动会话在设定的时间内未收到用户输入，会话将断开连接。
- 断开连接的会话计时器：确定是否对断开连接的会话强制实施时间限制。
- 断开连接的会话计时器间隔：设置断开连接的会话注销之前的时间间隔。

更新任何策略设置时，请确保这些设置在部署中保持一致。

空闲会话的时间限制到期时，将显示一条警告消息。有关示例，请参见以下屏幕截图：按确定关闭警告消息，但无法使会话保持活动状态。要使会话保持活动状态，请提供用户输入以重置空闲计时器。




可以在 Citrix Studio 7.12 版及更高版本中配置以下策略。

- MaxSpeexQuality

  值（整数）：[0–10]

  默认值：5

  详细信息：

  音频质量为中低时，音频重定向采用 Speex 编解码器对音频数据进行编码（请参阅音频质量策略）。Speex 是一种有损编解码器，这意味着它会牺牲输入语音信号保真度来进行压缩。与其他一些语音编解码器不同，这可以控制在质量和比特率之间所做的权衡。大多数时候 Speex 编码过程通过范围在 0 到 10 之间的质量参数控制。质量越高，比特率越高。

  最高 Speex 质量根据音频质量和带宽限制（请参阅音频重定向带宽限制策略）选择最佳 Speex 质量对音频数据编码。如果音频质量为中，编码器将处于宽带模式，这意味着采样率较高。如果音频质量为低，编码器将处于窄带模式，这意味着采样率较低。相同的 Speex 质量在不同的模式下有不同的比特率。最佳 Speex 质量出现在最大值满足以下条件时：

  - 不高于最高 Speex 质量。
  - 其比特率等于或小于带宽限制。

  相关设置：音频质量、音频重定向带宽限制

- PrimarySelectionUpdateMode

  值（枚举）：[0, 1, 2, 3]

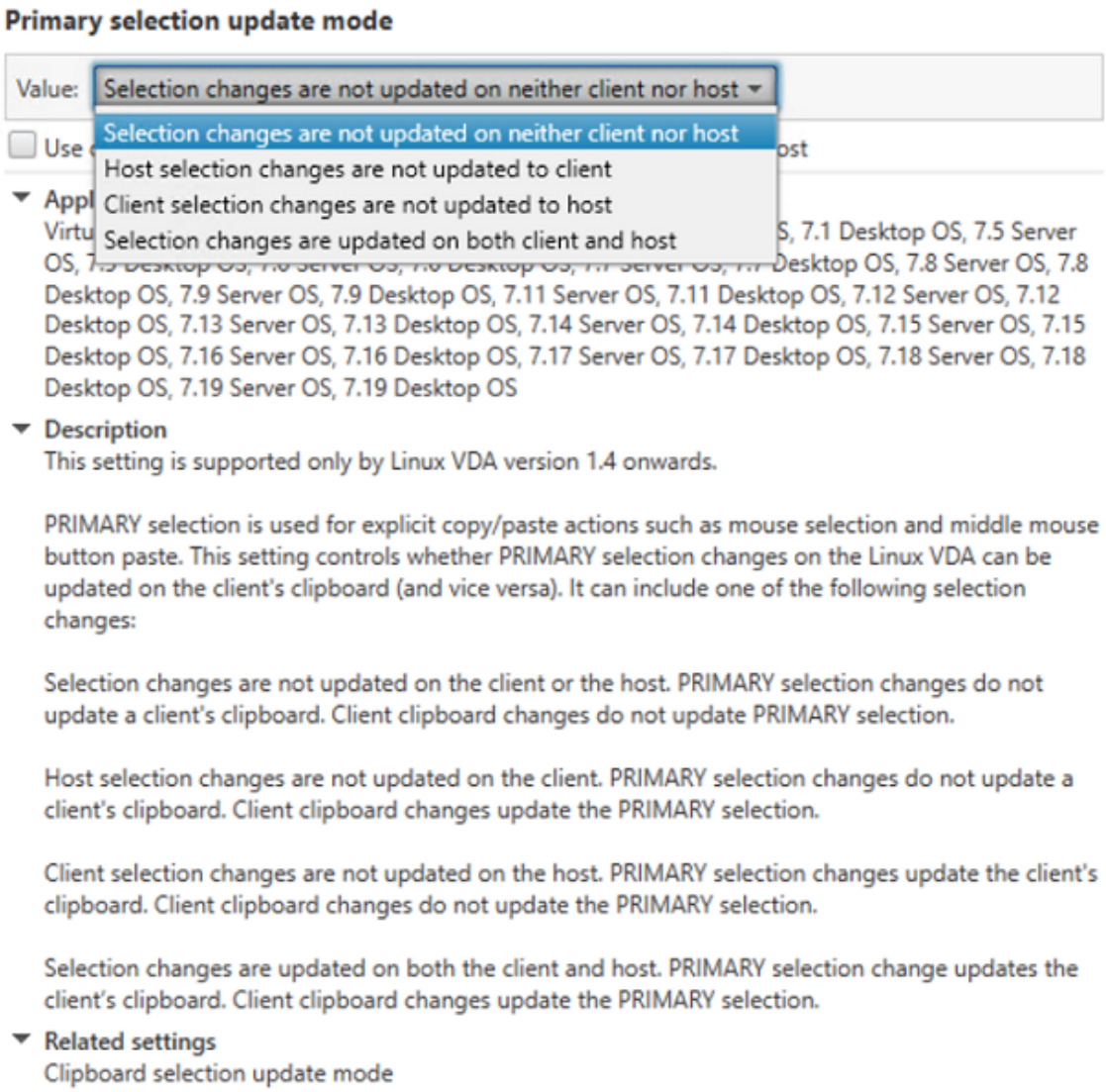
  默认值：3

  详细信息：

  选择数据并通过按下鼠标中键粘贴数据时，将使用主选定内容。

  此设置控制 Linux VDA 和客户端上的主选定内容变更是否可以在对方的剪贴板上更新。有四个值选项：




- 选定内容变更既不在客户端上也不在主机上更新

  Linux VDA 上的主选定内容变更不更新客户端上的剪贴板。客户端上的主选定内容变更不更新 Linux VDA 上的剪贴板。

- 主机选定内容变更不更新到客户端

  Linux VDA 上的主选定内容变更不更新客户端上的剪贴板。客户端上的主选定内容变更更新 Linux VDA 上的剪贴板。

- 客户端选定内容变更不更新到主机

  Linux VDA 上的主选定内容变更更新客户端上的剪贴板。客户端上的主选定内容变更不更新 Linux VDA 上的剪贴板。

- 选定内容变更同时在客户端和主机上更新

  Linux VDA 上的主选定内容变更更新客户端上的剪贴板。客户端上的主选定内容变更更新 Linux VDA 上的剪贴板。此选项为默认值。

相关设置：剪贴板选定内容更新模式

- ClipboardSelectionUpdateMode

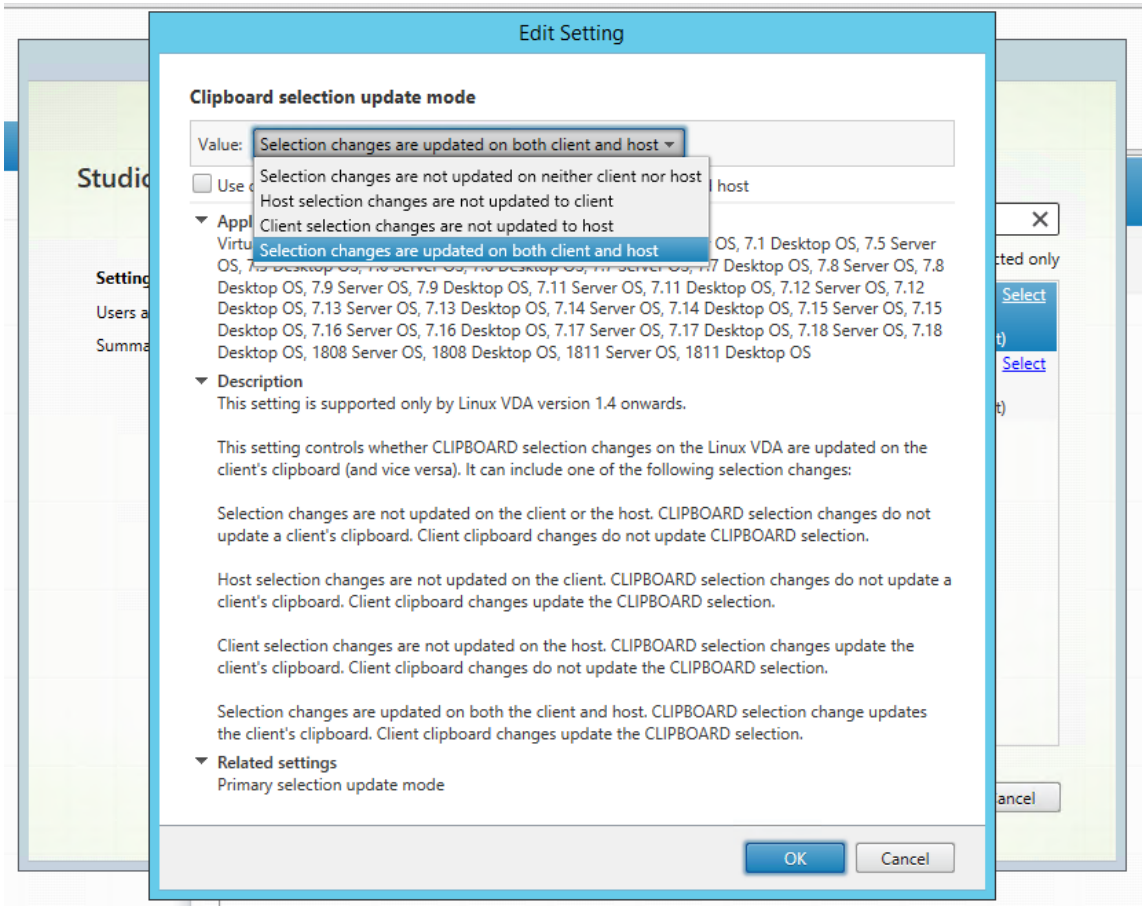
值（枚举）：[0, 1, 2, 3]

默认值：3

详细信息：

当您选择某些数据并明确请求将其“复制”到剪贴板时使用剪贴板选定内容，例如通过从快捷菜单中选择“复制”。剪贴板选定内容主要用于 Microsoft Windows 剪贴板操作，而主选定内容对 Linux 而言是唯一的。

此策略控制 Linux VDA 和客户端上的剪贴板选定内容变更是否可以在对方的剪贴板上更新。有四个值选项：




  - 选定内容变更既不在客户端上也不在主机上更新

    Linux VDA 上的剪贴板选定内容变更不更新客户端上的剪贴板。客户端上的剪贴板选定内容变更不更新 Linux VDA 上的剪贴板。

  - 主机选定内容变更不更新到客户端

    Linux VDA 上的剪贴板选定内容变更不更新客户端上的剪贴板。客户端上的剪贴板选定内容变更更新 Linux VDA 上的剪贴板。

  - 客户端选定内容变更不更新到主机

Linux VDA 上的剪贴板选定内容变更更新客户端上的剪贴板。客户端上的剪贴板选定内容变更不更新 Linux VDA 上的剪贴板。

– 选定内容变更同时在客户端和主机上更新
  Linux VDA 上的剪贴板选定内容变更更新客户端上的剪贴板。客户端上的剪贴板选定内容变更更新 Linux VDA 上的剪贴板。此选项为默认值。

相关设置：主选定内容更新模式

> 注意：
>
> Linux VDA 支持剪贴板选定内容和主选定内容。要控制 Linux VDA 与客户端之间的复制和粘贴行为，我们建议您同时将剪贴板选定内容更新模式和主选定内容更新模式设置为相同的值。

## 配置 IPv6

November 8, 2021

Linux VDA 支持 IPv6 以便与 Citrix Virtual Apps and Desktops 保持一致。使用此功能时，请注意以下事项：

- 对于双堆栈环境，除非显式启用 IPv6，否则使用 IPv4。
- 如果在 IPv4 环境中启用了 IPv6，Linux VDA 将无法运行。

> 重要
> :
>
> - 整个网络环境必须为 IPv6，而非仅针对 Linux VDA。
> - Centrify 不支持纯 IPv6。

安装 Linux VDA 时，不需要对 IPv6 执行任何特殊的设置任务。

### 为 Linux VDA 配置 IPv6

更改 Linux VDA 的配置之前，请确保您的 Linux 虚拟机以前在 IPv6 网络中运行。有两个与 IPv6 配置有关的注册表项：

```
1  “HKLM\Software\Policies\Citrix\VirtualDesktopAgent” -t “REG_DWORD” -v “OnlyUseIPv6ControllerRegistration”
2  “HKLM\Software\Policies\Citrix\VirtualDesktopAgent” -t “REG_DWORD” -v “ControllerRegistrationIPv6Netmask”
```

必须将 **OnlyUseIPv6ControllerRegistration** 设置为 1 才能在 Linux VDA 上启用 IPv6：

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

如果 Linux VDA 有多个网络接口，则可以使用 **ControllerRegistrationIPv6Netmask** 指定用于 Linux VDA 注册的网络接口：

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

请将 **{IPv6 netmask}** 替换为真实的网络掩码（例如 2000::/64）。

有关 Citrix Virtual Apps and Desktops 中的 IPv6 部署的详细信息，请参阅 IPv4/IPv6 支持。

### 故障排除

检查基础 IPv6 网络环境并使用 ping6 检查 AD 和 Delivery Controller 是否可访问。

## 配置 Citrix 客户体验改善计划 (CEIP)

April 16, 2021

参与 CEIP 时，系统会向 Citrix 发送匿名统计数据和使用情况信息以提高 Citrix 产品的质量和性能。此外，匿名数据的副本将被发送到 Google Analytics (GA) 以进行快速、高效地分析。

### 注册表设置

默认情况下，您在安装 Linux VDA 时会自动参与 CEIP。大约在您安装 Linux VDA 七天后第一次上载数据。可以在注册表中更改此默认设置。

- **CEIPSwitch**

启用或禁用 CEIP 的注册表设置（默认值 = 0）：

位置：HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\CEIP

名称：CEIPSwitch

值：1 = 禁用，0 = 启用

未指定时，CEIP 处于启用状态。

可以在客户端上运行以下命令来禁用 CEIP：

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SOFTWARE\
    Citrix\CEIP" -v "CEIPSwitch" -d "1"
```

- **GASwitch**

启用或禁用 GA 的注册表设置（默认值 = 0）：

位置：HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\CEIP

名称：GASwitch

值：1 = 禁用，0 = 启用

未指定时，GA 处于启用状态。

可以在客户端上运行以下命令来禁用 GA：

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SOFTWARE\
    Citrix\CEIP" -v "GASwitch" -d "1"
```

- **DataPersistPath**

控制数据保留路径的注册表设置（默认值 = /var/xdl/ceip）：

位置：HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\CEIP

名称：DataPersistPath

值：字符串

可以运行以下命令来设置此路径：

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

如果配置的路径不存在或无法访问，数据将保存在默认路径中。

### 从 **Linux VDA** 收集的 **CEIP** 数据

下表提供了收集的匿名信息的类型示例。数据中不包含任何识别出您是客户的详细信息。

| 数据点 | 注册表项名称 | 说明 |
| --- | --- | --- |
| 计算机 GUID | machine_guid | 标识从其获取数据的计算机 |
| AD 解决方案 | ad_solution | 表示计算机的域加入方法的文本字符串 |
| Linux 内核版本 | kernel_version | 表示计算机的内核版本的文本字符串 |

| 数据点 | 注册表项名称 | 说明 |
| --- | --- | --- |
| LVDA 版本 | vda_version | 表示安装的 Linux VDA 版本的文本字符串。 |
| LVDA 更新或全新安装 | update_or_fresh_install | 表示正在全新安装或更新当前 Linux VDA 软件包的文本字符串 |
| LVDA 安装方法 | install_method | 表示通过使用 MCS、PVS、轻松安装或手动安装安装了当前 Linux VDA 软件包的文本字符串。 |
| 是否启用 HDX 3D pro | hdx_3d_pro | 表示是否在计算机上启用 HDX 3D Pro 的文本字符串 |
| 是否启用 VDI 模式 | vdi_mode | 表示是否启用 VDI 模式的文本字符串 |
| 系统区域设置 | system_locale | 表示此计算机的区域设置的文本字符串 |
| LVDA 主要服务上次重新启动时间 | ctxhdx ctxvda | `ctxhdx` 和 `ctxvda` 服务的上次重新启动时间，格式为 dd-hh:mm:ss，例如 10-17:22:19 |
| GPU 类型 | gpu_type | 表示计算机的 GPU 类型 |
| CPU 内核 | cpu_cores | 表示计算机的 CPU 内核数的整数 |
| CPU 频率 | cpu_frequency | 表示 CPU 频率的浮点数（以 MHz 为单位） |
| 物理内存大小 | memory_size | 表示物理内存大小的整数（以 KB 为单位） |
| 启动的会话数 | session_launch | 表示我们收集此数据点时计算机上启动（登录或重新连接）的会话数的整数 |
| Linux 操作系统名称和版本 | os_name_version | 表示计算机的 Linux 操作系统名称和版本的文本字符串 |
| 会话密钥 | session_key | 标识从其获取数据的会话 |
| 资源类型 | resource_type | 表示启动的会话的资源类型的文本字符串：桌面或 `<appname>` |
| 活动会话时间 | active_session_time | 用于保存会话的活动时间。一个会话可以有多个活动时间，因为会话可以断开连接/重新连接 |
| 会话持续时间 | session_duration_time | 用于保存从登录到注销的会话持续时间 |
| Receiver 客户端类型 | receiver_type | 表示用于启动会话的 Citrix Workspace 应用程序的类型的整数 |

| 数据点 | 注册表项名称 | 说明 |
| --- | --- | --- |
| Receiver 客户端版本 | receiver_version | 表示用于启动会话的 Citrix Workspace 应用程序的版本的文本字符串 |
| 打印计数 | printing_count | 表示会话使用打印功能的次数的整数 |
| USB 重定向计数 | usb_redirecting_count | 表示会话使用 USB 设备的次数的整数 |
| Gfx 提供程序类型 | gfx_provider_type | 表示会话的图形提供程序类型的文本字符串 |
| 重影计数 | shadow_count | 表示会话被重影的次数的整数 |
| 用户选择的语言 | ctxism_select | 包含用户选择的所有语言的组合长字符串 |
| 智能卡重定向计数 | scard_redirecting_count | 表示智能卡重定向用于会话中应用程序的会话登录和用户身份验证的次数的整数 |

## 配置 **USB** 重定向

September 23, 2024

USB 设备在 Citrix Workspace 应用程序与 Linux VDA 桌面之间共享。将 USB 设备重定向到桌面后，用户就可以像使用本地连接的 USB 设备那样使用它。

> 提示：
>
> 当网络延迟低于 100 毫秒时，我们建议使用 USB 重定向。当网络延迟超过 200 毫秒时，请勿使用 USB 重定向。

USB 重定向包含三个主要方面的功能：

- 开源项目实施 (VHCI)
- VHCI 服务
- USB 服务

开源 **VHCI**：

这部分 USB 重定向功能发展了通过 IP 网络的通用 USB 设备共享系统。它由 Linux 内核驱动程序和一些用户模式库组成，这些库使您可以与内核驱动程序通信以获取所有 USB 数据。在 Linux VDA 实现中，Citrix 重用 VHCI 的内核驱动程序。但是，Linux VDA 与 Citrix Workspace 应用程序之间的所有 USB 数据传输都封装在 Citrix ICA 协议软件包中。

**VHCI** 服务：

VHCI 服务是 Citrix 提供用来与 VHCI 内核模块通信的开源服务。此服务充当 VHCI 与 Citrix USB 服务之间的网关。

**USB** 服务：

USB 服务相当于管理 USB 设备上的所有虚拟化和数据传输的 Citrix 模块。

### **USB** 重定向的工作方式

通常情况下，如果 USB 设备成功重定向至 Linux VDA，会在系统 /dev 路径中创建一个或多个设备节点。但是，重定向的设备有时不可用于活动的 Linux VDA 会话。USB 设备依赖于驱动程序才能正常使用，且有些设备需要特殊的驱动程序。如果并未提供驱动程序，活动 Linux VDA 会话无法使用重定向的 USB 设备。为确保 USB 设备的连接性，请安装驱动程序并正确配置系统。

Linux VDA 支持一组成功重定向至客户端和从客户端重定向的 USB 设备。此外，正确装载设备（尤其是 USB 磁盘）后，用户就可以访问磁盘，而无需进行任何其他配置。

### 支持的 **USB** 设备

下列设备已确认支持此 Linux VDA 版本。可以随意使用其他设备，但会有意外结果：

> 注意：
>
> Linux VDA 仅支持 USB 2.0 协议。

| USB 大容量存储设备 | VID:PID | 文件系统 |
|---|---|---|
| Netac Technology Co., Ltd | 0dd8:173c | FAT32 |
| Kingston Datatraveler 101 II | 0951:1625 | FAT32 |
| Kingston Datatraveler GT101 G2 | 1567:8902 | FAT32 |
| SanDisk SDCZ80 闪存驱动器 | 0781:5580 | FAT32 |
| WD HDD | 1058:10B8 | FAT32 |

| USB 3D 鼠标 | VID:PID |
|---|---|
| 3DConnexion SpaceMouse Pro | 046d: c62b |

| USB 扫描仪 | VID:PID |
| --- | --- |
| Epson Perfection V330 照片 | 04B8: 0142 |

### 配置 **USB** 重定向

有一个 Citrix 策略控制是否启用或禁用 USB 设备重定向。此外，还可以使用 Delivery Controller 策略指定设备类型。为 Linux VDA 配置 USB 重定向时，请配置以下策略和规则：

- 客户端 USB 设备重定向策略
- 客户端 USB 设备重定向规则

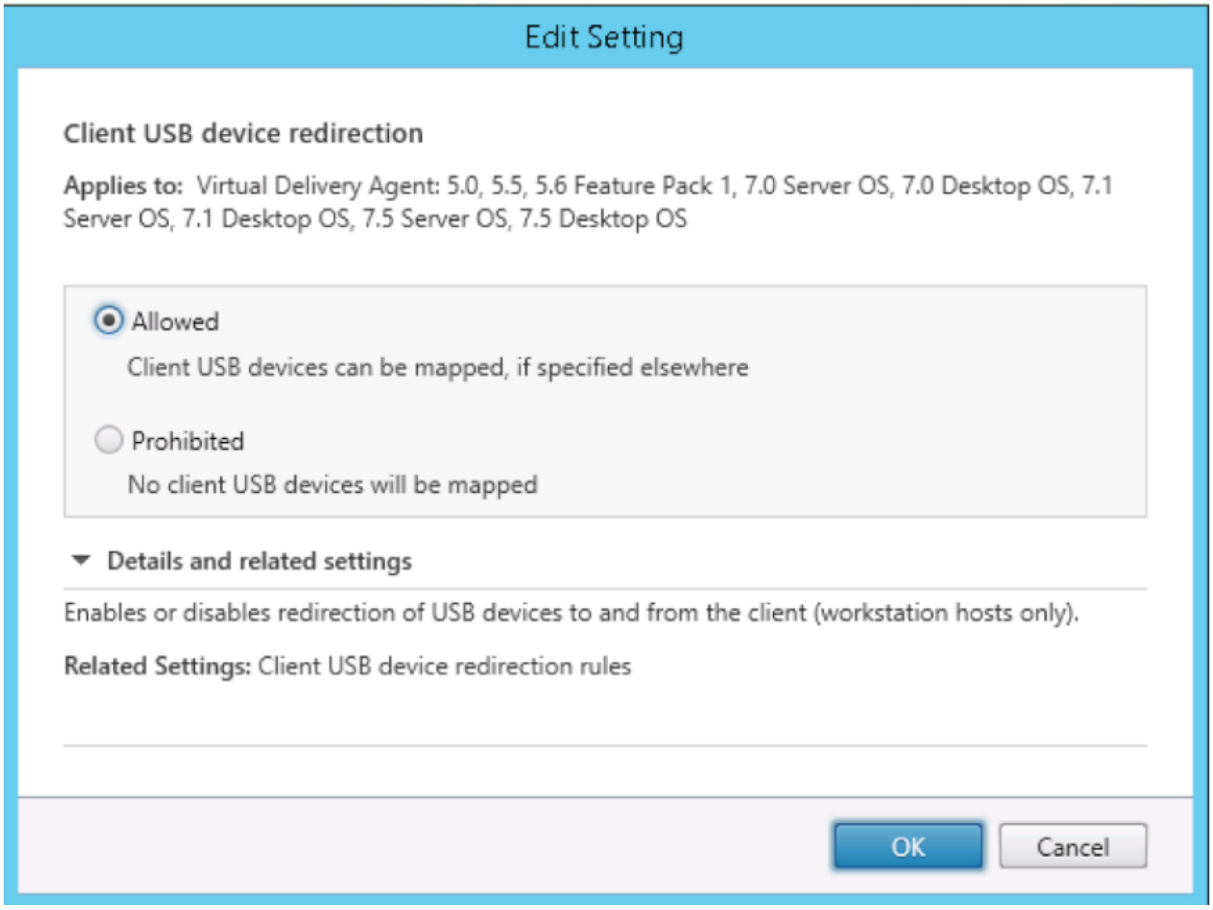
#### 启用 **USB** 重定向策略

在 Citrix Studio 中，启用（或禁用）与客户端之间的 USB 设备重定向（仅限工作站主机）。

在编辑设置对话框中：

1. 选择允许。
2. 单击确定。



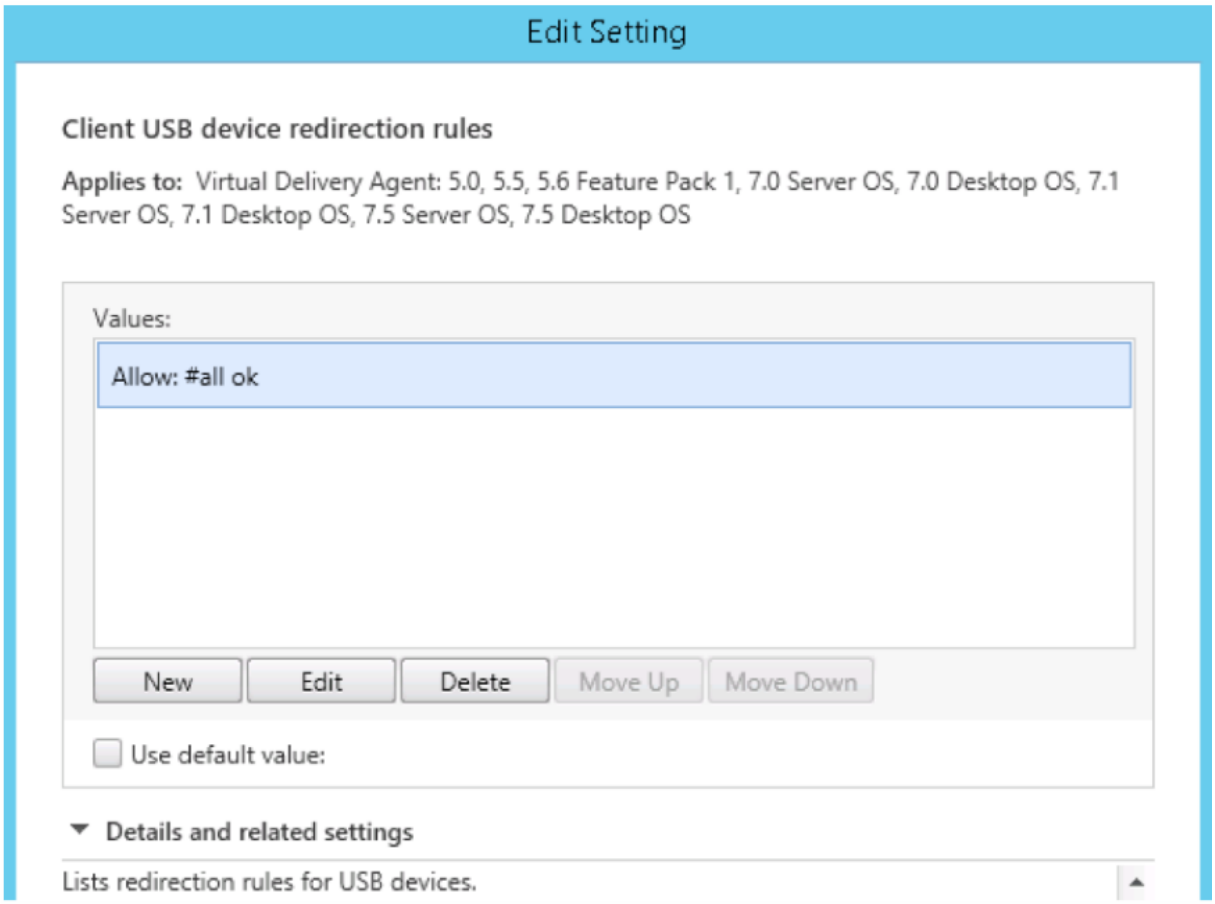

设置 **USB** 重定向规则

启用 USB 重定向策略后，使用 Citrix Studio 设置重定向规则，方法是指定允许（或拒绝）在 Linux VDA 上使用哪些设备。

在客户端 USB 设备重定向规则对话框中：

1. 单击新建添加重定向规则，或单击编辑检查现有规则。
2. 创建（或编辑）规则后，单击确定。




有关如何配置通用 USB 重定向的详细信息，请参阅 Citrix Generic USB Redirection Configuration Guide（《Citrix 通用 USB 重定向配置指南》）。

### 构建 **VHCI** 内核模块

USB 重定向依赖于 VHCI 内核模块（`usb-vhci-hcd.ko` 和 `usb-vhci-iocif.ko`）。这些模块包含在 Linux VDA 发行版中（作为 RPM 软件包的一部分）。它们根据正式的 Linux 发行版内核进行编译，请见下表：

| 支持的 Linux 发行版 | 内核版本 |
| --- | --- |
| RHEL 8.3 | 4.18.0-240.10.1 |
| RHEL 8.2 | 4.18.0-240.10.1 |
| RHEL 8.1 | 4.18.0-240.10.1 |
| RHEL 7.9 | 3.10.0-1160.11.1 |
| RHEL 7.8 | 3.10.0-1160.11.1 |
| SUSE 12.5 | 4.12.14-122.57.1 |

| 支持的 Linux 发行版 | 内核版本 |
| --- | --- |
| Ubuntu 20.04 | 5.4.0-58 |
| Ubuntu 18.04 | 4.15.0-128 |
| Ubuntu 16.04 | 4.4.0-197 |
| Debian 10 | 4.19.0-13 |

> 重要：
>
> 如果您的计算机的内核与为 Linux VDA 构建的驱动程序不兼容，USB 服务可能无法启动。在这种情况下，仅当您构建自己的 VHCI 内核模块时，才能使用 USB 重定向功能。

确认您的内核与 **Citrix** 构建的模块是否一致

在命令行上，运行以下命令来确认内核是否一致：

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

如果命令运行成功，则内核模块已成功加载，且版本与 Citrix 安装的模块一致。

如果命令运行后显示错误，则内核与 Citrix 模块不一致，必须重新构建。

重新构建 **VHCI** 内核模块

如果您的内核模块与 Citrix 的版本不一致，请执行以下操作：

1. 从 Citrix 下载站点下载 LVDA 源代码。选择 **Linux Virtual Delivery Agent (sources)** 部分中包含的文件。
2. 解压缩 **citrix-linux-vda-sources.zip** 文件。导航到 **linux-vda-souces/vhci-hcd-1.15.tar.bz2**，并使用 **tar xvf vhci-hcd-1.15.tar.bz2** 解压 VHCI 源文件。
3. 基于头文件和 **Module.symvers** 文件构建内核模块。按照以下步骤安装内核头文件并根据相应的 Linux 发行版创建 **Module.symvers**：

   **RHEL/CentOS**：

   ```
   yum install kernel-devel
   ```

   **SUSE 12**：

   ```
   zypper install kernel-devel

   zypper install kernel-source
   ```

   **Ubuntu**：

```
apt-get install linux-headers
```

> 提示：
>
> 如果安装成功，将会有类似如下文件夹的内核文件夹：
>
> /usr/src/kernels/3.10.0-327.10.1.el7.x86_64

4. 在 /usr/src/kernels/3.10.0-327.10.1.el7.x86_64 文件夹中，确认是否存在 **Module.symvers** 文件。如果该文件夹中没有此文件，请（通过按顺序运行以下命令：`make oldconfig`、`make prepare`、`make modules`、`make`）构建内核以获取此文件，或者从 **/usr/src/kernels/3.10.0-327.10.1.el7.x86_64-obj/x86_64/defaults/module*** 复制此文件
5. 运行以下命令以安装开发工具。

   **RHEL 8**、**CentOS 8**：

   ```
   yum groupinstall 'Development Tools'

   yum install elfutils-libelf-devel
   ```

   **RHEL 7**、**CentOS 7**：

   ```
   yum groupinstall 'Development Tools'
   ```

   **Ubuntu 20.04**、**Ubuntu 18.04**、**Debian 10**：

   ```
   apt install build-essential flex bison libelf-dev
   ```

   **Ubuntu 16.04**：

   ```
   apt install build-essential flex bison
   ```

6. 在 `vhci-hcd-1.15/Makefile` 文件中，更改 VCHI 的 Makefile，并将 KDIR 设置为内核目录：

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

   KDIR = /usr/src/kernels/3.10.0-327.10.1.el7.x86_64
   ```

7. 在文件夹 `vhci-hcd-1.15/` 中，运行 `make` 以构建 VHCI 内核。

   > 注意：
   >
   > 如果构建成功，则会在文件夹 `vhci-hcd-1.15/` 中创建 `usb-vhci-hcd.ko` 和 `usb-vhci-iocifc.ko`。

8. 将内核模块替换为新构建的模块：**cp -f usb-vhci-*.ko /opt/Citrix/VDA/lib64/**
9. 重新启动 USB 服务：

```
service ctxusbsd restart
```

10. 注销并重新登录会话。检查 USB 重定向是否正常。

解决内核构建问题

使用特定内核构建 VHCI 模块时可能会出现以下错误：

- 可能会出现 `implicit declaration of function 'copy\\_to\\_user'` 错误，请参见下面的屏幕截图：

  ```
  usb-vhci-iocifc.c:216:5: error: implicit declaration of function 'copy_to_user'
  ```

  该错误是由于内核中的头文件发生变化所致。解决方法：将 **#include <linux/uaccess.h>** 行添加到 `vhci-hcd-1.15/usb-vhci-iocifc.c` 文件中。

  ```
  #include <linux/fs.h>
  #include <linux/uaccess.h>

  #include "usb-vhci-hcd.h"
  ```

- 可能会出现 `'driver\\_attr\\_debug_output'undeclared` 错误，请参见下面的屏幕截图：

  ```
  error: 'driver_attr_debug_output' undeclared (first use in this function)
  ```

  当内核缺少符号时，将出现错误。解决方法：禁用 `vhci-hcd-1.15/usb-vhci-iocifc.c` 和 `vhci-hcd-1.15/usb-vhci-hcd.c` 文件中 DEBUG 的宏定义。

  ```
  //#define DEBUG

  #include <linux/module.h>
  ```

- 可能会出现 `'make[3]: *** No rule to make target 'arch/x86/tools/relocs_32.c', needed by 'arch/x86/tools/relocs_32.o'. Stop.` 错误，请参见下面的屏幕截图：

  ```
  scripts/kconfig/conf  --syncconfig Kconfig
  make[3]: *** No rule to make target 'arch/x86/tools/relocs_32.c', needed by 'arch/x86/tools/relocs_32.o'.  Stop.
  arch/x86/Makefile:232: recipe for target 'archscripts' failed
  make[2]: *** [archscripts] Error 2
  make[2]: Leaving directory '/usr/src/linux-headers-5.4.0-1031-azure'
  Makefile:102: recipe for target 'testcc' failed
  make[1]: *** [testcc] Error 2
  make[1]: Leaving directory '/home/administrator1/linuxvda-vhci'
  Makefile:97: recipe for target 'conf/usb-vhci.config.h' failed
  make: *** [conf/usb-vhci.config.h] Error 2
  ```

  解决方法为，使用 `vhci-hcd-1.15/` 路径下的以下命令将 `SUBDIRS=$(PWD)` 替换为 `M=$(shell pwd)`：

```
sed -i 's/SUBDIRS=$(PWD)/M=$(shell pwd)/g' Makefile

sed -i 's/SUBDIRS=$(PWD)/M=$(shell pwd)/g' test/Makefile
```

- 可能会出现 `./include/uapi/linux/stat.h:30:17: error: expected ')' before numeric constant`
  `#define S_IRUSR 00400` 错误，请参见下面的屏幕截图：

```
In file included from ./include/linux/stat.h:7:0,
                 from ./include/linux/module.h:10,
                 from /home/administrator1/vhci-hcd-1.15/usb-vhci-hcd.c:24:
./include/uapi/linux/stat.h:30:17: error: expected ')' before numeric constant
 #define S_IRUSR 00400
                 ^
/home/administrator1/vhci-hcd-1.15/usb-vhci-hcd.c:1312:34: note: in expansion of macro 'S_IRUSR'
 static DRIVER_ATTR(debug_output, S_IRUSR | S_IWUSR, show_debug_output, store_debug_output);
                                  ^~~~~~~
```

Run the following commands to work around the issue:

```
sed -i 's/show_debug_output/debug_output_show/g' usb-vhci-iocifc.c usb-vhci-hcd.c

sed -i 's/store_debug_output/debug_output_store/g' usb-vhci-iocifc.c usb-vhci-hcd.c

sed -i 's/static DRIVER_ATTR(debug_output, S_IRUSR | S_IWUSR, debug_output_show, debug_output_store);/static DRIVER_ATTR_RW(debug_output);/g' usb-vhci-iocifc.c usb-vhci-hcd.c
```

- 可能会出现 `./arch/x86/include/asm/uaccess.h:433:29: error: invalid initializer`
  `__typeof__(ptr)__pu_ptr = (ptr); \` 错误，请参见下面的屏幕截图：

```
/home/administrator1/vhci-hcd-1.15/usb-vhci-iocifc.c: In function 'ioc_register':
./arch/x86/include/asm/uaccess.h:433:29: error: invalid initializer
  __typeof__(ptr) __pu_ptr = (ptr);   \
                             ^
./arch/x86/include/asm/uaccess.h:553:2: note: in expansion of macro '__put_user_nocheck'
  __put_user_nocheck((__typeof__(*(ptr)))(x), (ptr), sizeof(*(ptr)))
  ^~~~~~~~~~~~~~~~~~
/home/administrator1/vhci-hcd-1.15/usb-vhci-iocifc.c:219:3: note: in expansion of macro '__put_user'
   __put_user('\0', arg->bus_id);
   ^~~~~~~~~~
```

As a workaround, change the 219 line of the `usb-vhci-iocifc.c` file from `__put_user('\0', arg->bus_id);` to `__put_user('\0', arg->bus_id + 0);`.

- 可能会出现 `error: 'access_ok'undeclared (first use in this function)`
  `if(unlikely((_IOC_DIR(cmd)& _IOC_READ)&& !access_ok(VERIFY_WRITE, arg, _IOC_SIZE(cmd))))` 错误，请参见下面的屏幕截图：

```
/root/linuxvda-vhci/usb-vhci-iocifc.c:963:46: error: 'access_ok' undeclared (first use in this function)
  if(unlikely((_IOC_DIR(cmd) & _IOC_READ) && !access_ok(VERIFY_WRITE, arg, _IOC_SIZE(cmd))))
                                              ^~~~~~~~
./include/linux/compiler.h:77:42: note: in definition of macro 'unlikely'
 # define unlikely(x) __builtin_expect(!!(x), 0)
                                         ^
```

Run the following commands to work around the issue:

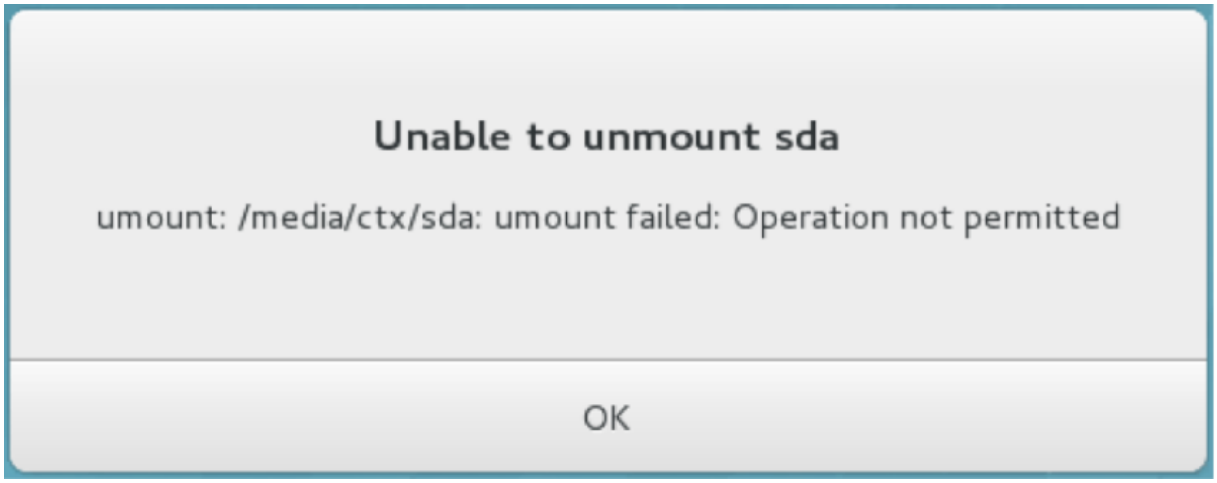
```
sed -i 's/VERIFY_READ, //g' usb-vhci-iocifc.c
sed -i 's/VERIFY_WRITE, //g' usb-vhci-iocifc.c
```

### 解决 **USB** 重定向问题

请根据本节中的信息解决您在使用 Linux VDA 时可能遇到的各种问题。

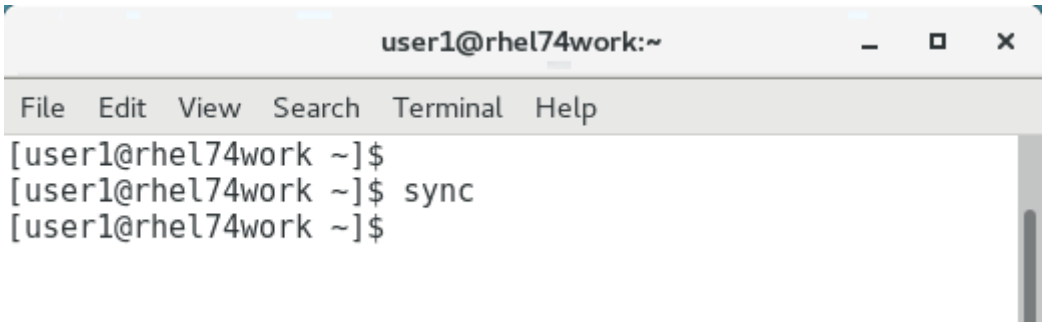
#### 无法卸载重定向的 **USB** 磁盘

为了对从 Citrix Workspace 应用程序重定向的所有 USB 磁盘进行访问控制，Linux VDA 采用管理权限管理所有这些设备，确保只有所有者才能访问重定向的设备。因此，没有管理权限的用户不能卸载设备。




#### 停止重定向 **USB** 磁盘时文件丢失

如果将 USB 磁盘重定向到会话并尝试修改该磁盘（例如，在磁盘上创建一些文件），然后立即使用 Citrix Workspace 应用程序的工具栏停止重定向，则您修改或创建的文件可能会丢失。出现此问题是因为您将数据写入文件系统时，系统在文件系统中装载内存缓存。数据并未写入磁盘本身。如果使用 Citrix Workspace 应用程序的工具栏停止重定向，则没有时间将数据刷新至磁盘，从而导致数据丢失。为了解决此问题，请先在终端使用 sync 命令将数据刷新至磁盘，然后再停止 USB 重定向。

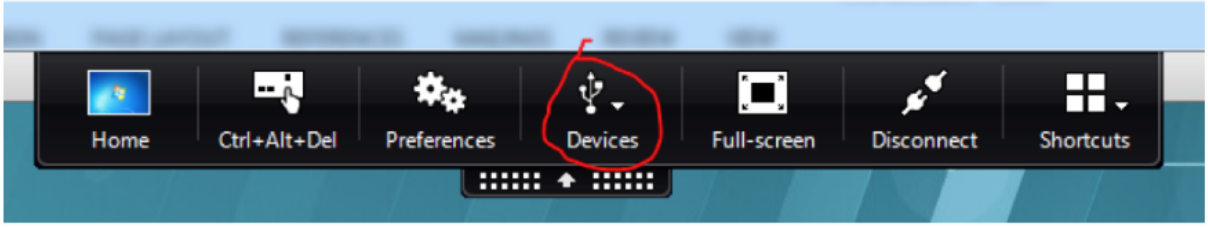


**Citrix Workspace** 应用程序的工具栏中无设备

有时，您可能无法看到 Citrix Workspace 应用程序的工具栏中没有列出设备，这表示没有进行 USB 重定向。如果遇到该问题，请确认以下事项：

- 策略已配置为允许 USB 重定向
- 内核模块与您的内核兼容




> 注意：
>
> 设备选项卡在适用于 Linux 的 Citrix Workspace 应用程序中不可用。

当 **USB** 设备在 **Citrix Workspace** 应用程序的工具栏中显示，但这些设备都标有受策略限制时，重定向失败

出现此问题时，请执行以下操作：

- 配置 Linux VDA 策略以启用重定向。
- 检查是否在 Citrix Workspace 应用程序的注册表中配置了任何其他策略限制。请检查注册表路径中的 **DeviceRules**，以确保此设置未拒绝访问该设备：

  HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Citrix\ICA Client\GenericUSB

**USB** 设备已成功重定向，但无法在会话中使用

通常情况下，只能重定向受支持的 USB 设备。其他设备可能也会重定向到活动 Linux VDA 会话。对于每个重定向的设备，都会在系统 **/dev** 路径中创建用户拥有的节点。但是，用户是否可以成功使用设备由驱动程序和配置决定。如果您发现拥有（已插入）的某个设备无法访问，请将该设备添加到不受限制策略。

> 注意：
>
> 如果是 USB 驱动器，Linux VDA 会配置和装载磁盘。用户（且仅限安装它的所有者）无需执行任何其他配置即可访问该磁盘。未包含在受支持设备列表中的设备可能不是这种情况。

## 配置会话可靠性

May 18, 2021

Citrix 将会话可靠性功能引入所有支持的 Linux 平台。会话可靠性在默认情况下处于启用状态。

会话可靠性将在网络中断时无缝重新连接 ICA 会话。有关会话可靠性的详细信息，请参阅客户端自动重新连接和会话可靠性。

注意：默认情况下，通过会话可靠性连接传输的数据是纯文本数据。出于安全考虑，我们建议您启用 TLS 加密。有关 TLS 加密的详细信息，请参阅使用 TLS 保护用户会话安全。

## 配置

### Citrix Studio 中的策略设置

可以在 Citrix Studio 中为会话可靠性设置以下策略：

- 会话可靠性连接
- 会话可靠性超时
- 会话可靠性端口号
- 重新连接 UI 透明度级别

有关详细信息，请参阅会话可靠性策略设置和客户端自动重新连接策略设置。

注意：设置会话可靠性连接或会话可靠性端口号策略后，请按此顺序重新启动 VDA 服务和 HDX 服务，以使设置生效。

### Linux VDA 上的设置

- 启用/禁用会话可靠性 **TCP** 侦听器

默认情况下，在端口 2598 上启用和侦听会话可靠性 TCP 侦听器。要禁用该侦听器，请运行以下命令。

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

注意：请重新启动 HDX 服务以使设置生效。禁用 TCP 侦听器不会禁用会话可靠性。仍可通过其他侦听器（例如 SSL）获得会话可靠性，前提是通过会话可靠性连接策略启用了该功能。

- 会话可靠性端口号

还可以使用以下命令设置会话可靠性端口号（以端口号 2599 为例）。

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

注意：请重新启动 HDX 服务以使设置生效。如果在 Citrix Studio 中通过策略设置设置了端口号，则将忽略 Linux VDA 上的设置。确保 VDA 上的防火墙配置为不禁止通过设置端口传输网络流量。

- 服务器到客户端保持活动状态时间间隔

会话中没有活动（例如，没有鼠标移动、没有屏幕更新）时，在 Linux VDA 和 ICA 客户端之间发送会话可靠性保持活动状态消息。保持活动状态消息用于检测客户端是否仍可响应。如果客户端没有响应，则挂起会话，直到客户端重新连接。此设置指定相邻保持活动状态消息之间间隔的秒数。默认情况下未配置此设置。要对其进行配置，请运行以下命令（以 10 秒为例）。

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

- 客户端到服务器保持活动状态时间间隔

此设置指定从 ICA 客户端发送到 Linux VDA 的相邻保持活动状态消息之间间隔的秒数。默认情况下未配置此设置。要对其进行配置，请运行以下命令（以 10 秒为例）。

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

### 故障排除

通过策略设置启用会话可靠性后，无法启动会话。

要解决此问题，请执行以下操作：

1. 在 Citrix Studio 中通过策略设置启用会话可靠性后，请务必按此顺序重新启动 VDA 服务和 HDX 服务。
2. 在 VDA 上，运行以下命令以验证会话可靠性 TCP 侦听器是否正在运行（以端口 2598 为例）。

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

   如果会话可靠性端口上没有 TCP 侦听器，请运行以下命令启用该侦听器。

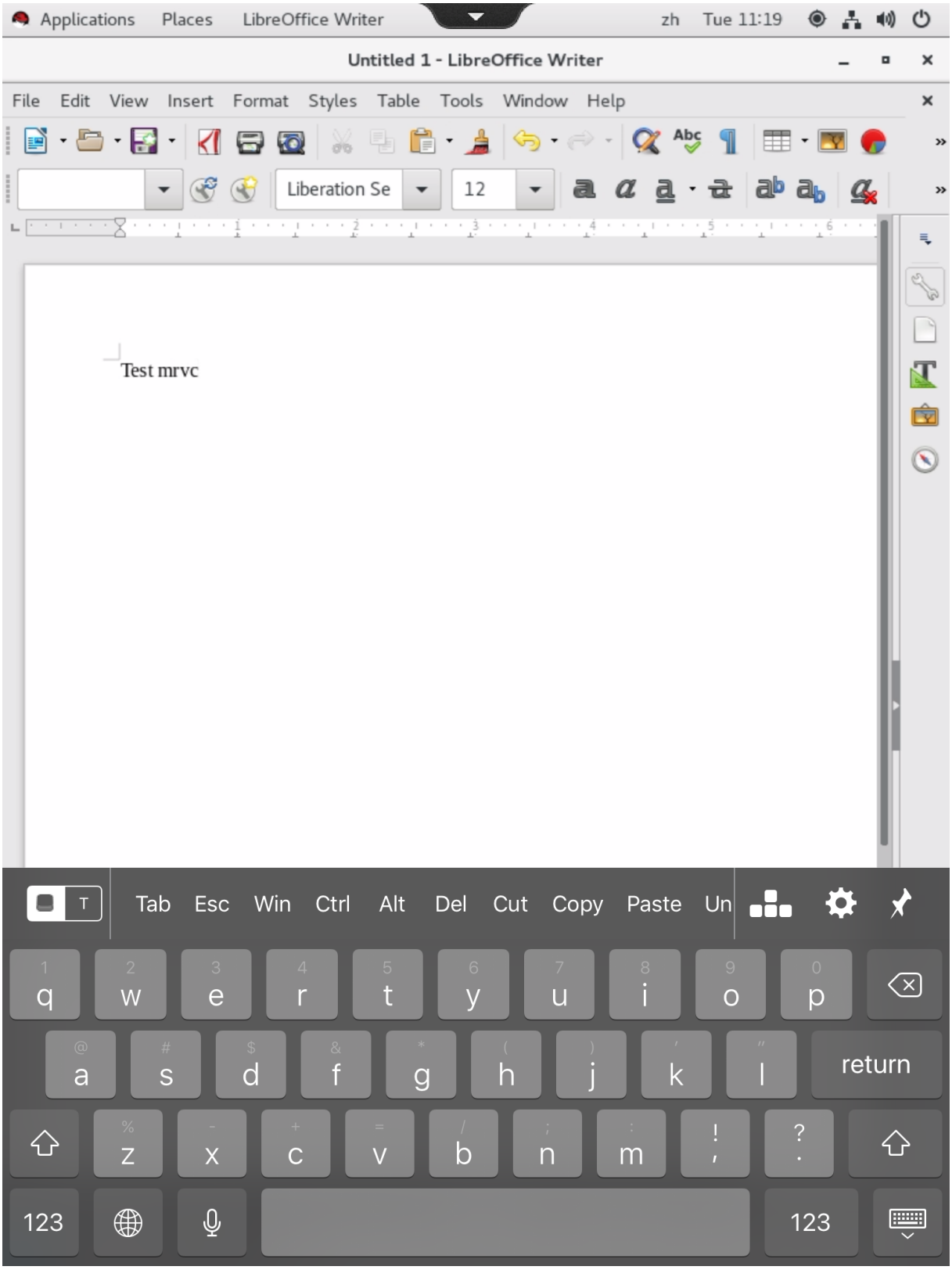
   ```
   /opt/Citrix/VDA/bin/ctxreg update -k "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Citrix\WinStations\cgp" -v "fEnableWinStation" -d "0x00000001"
   ```

## 软键盘

### September 1, 2022

可以在 Linux 虚拟桌面或应用程序会话中使用软键盘功能。软键盘会在您输入或离开输入字段时显示或隐藏。



> 注意：
>
> 该功能适用于 RHEL 7.8、RHEL 7.9、RHEL 8.1、RHEL 8.2、RHEL 8.3、SUSE 12.5、Ubuntu 16.04、Ubuntu 18.04 和 Ubuntu 20.04。适用于 iOS 和 Android 的 Citrix Workspace 应用程序支持该功能。

### 启用和禁用该功能

默认情况下，该功能处于禁用状态。可以使用 **ctxreg** 实用程序启用或禁用该功能。给定 Linux VDA 上的功能配置应用于该 VDA 上发布的所有会话。

要启用此功能，请执行以下操作：

1. 运行以下命令：

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\
    CurrentControlSet\Control\Citrix\VirtualChannels\MrVc"  -v "
    Enabled" -d "0x00000001"
```

2. 在 Citrix Studio 中，将自动显示键盘策略设置为允许。
3. （可选）对于 RHEL 7 和 CentOS 7，运行以下命令将智能输入总线 (`IBus`) 配置为默认 IM 服务：

```
echo "GTK_IM_MODULE=ibus" >>/etc/bashrc
```

要禁用该功能，请运行以下命令：

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\CurrentControlSet\
    Control\Citrix\VirtualChannels\MrVc"  -v "Enabled" -d "0x00000000"
```

> 注意：
>
> 之前的设置将在您登录到新会话或注销后返回到当前会话时生效。

### 限制

- 该功能在 Google Chrome、LibreOffice 和其他应用程序中可能无法正常运行。
- 要在手动隐藏软键盘后再次显示它，请单击非输入字段，然后再次单击当前输入字段。
- 在 Web 浏览器中从一个输入字段切换到单击另一个输入字段时，软键盘可能不显示。要解决此问题，请单击非输入字段，然后单击目标输入字段。
- 此功能不支持 Unicode 字符和双字节字符（例如，中文、日语和韩语字符）。
- 软键盘不适用于密码输入字段。
- 软键盘可能会与当前输入字段重叠。在这种情况下，请移动应用程序窗口或向上滚动屏幕以将输入字段移到可访问的位置。

- 由于 Citrix Workspace 应用程序和 Huawei 平板电脑之间存在兼容性问题，因此即使连接了物理键盘，软键盘也会显示在 Huawei 平板电脑上。

## 客户端输入法编辑器 (IME)

April 16, 2021

### 概述

必须通过 IME 输入双字节字符，例如中文、日语和朝鲜语字符。通过与客户端上的 Citrix Workspace 应用程序兼容的任何 IME（例如 Windows 本机 CJK IME）键入此类字符。

### 安装

在安装 Linux VDA 时，会自动安装此功能。

### 使用情况

照常打开 Citrix Virtual Apps 或 Citrix Virtual Desktops 会话。

根据需要在客户端上更改您的输入法以开始使用客户端 IME 功能。

### 已知问题

- 必须先双击 Google 电子表格中的单元格，才能使用客户端 IME 功能在单元格中键入字符。
- 不会在“密码”字段中自动禁用客户端 IME 功能。
- IME 用户界面不在输入区域中跟随光标。

## 支持多语言输入

November 8, 2021

自 Linux VDA 版本 1.4 起，Citrix 添加了对已发布的应用程序的支持。用户可以在没有 Linux 桌面环境的情况下访问所需的 Linux 应用程序。

但是，由于语言栏与 Linux 桌面环境高度集成，因此，Linux VDA 上的本地语言栏对于已发布的应用程序不可用。因此，用户无法以需要 IME 的语言（例如，中文、日语或韩语）输入文本。在应用程序会话期间，用户也不能在键盘布局之间切换。

为了解决这些问题，此功能为接受文本输入的已发布应用程序提供语言栏。通过语言栏，用户可以在应用程序会话期间选择服务器端 IME 以及在键盘布局之间切换。

### 配置

您可以使用 **ctxreg** 实用程序启用或禁用此功能（默认情况下禁用）。给定 Linux VDA 服务器上的功能配置应用于在该 VDA 上发布的所有应用程序。

配置注册表项为 “HKEY_LOCAL_MACHINE \SYSTEM\CurrentControlSet\Control\Citrix\LanguageBar”，类型为 DWORD。

要启用此功能，请运行以下命令：

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

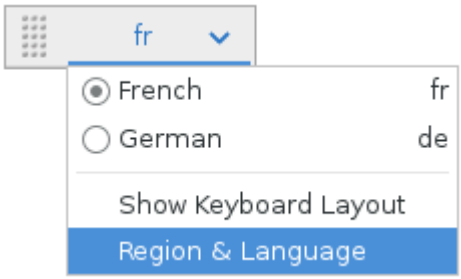
要禁用此功能，请运行以下命令：

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

### 使用情况

用法很简单。

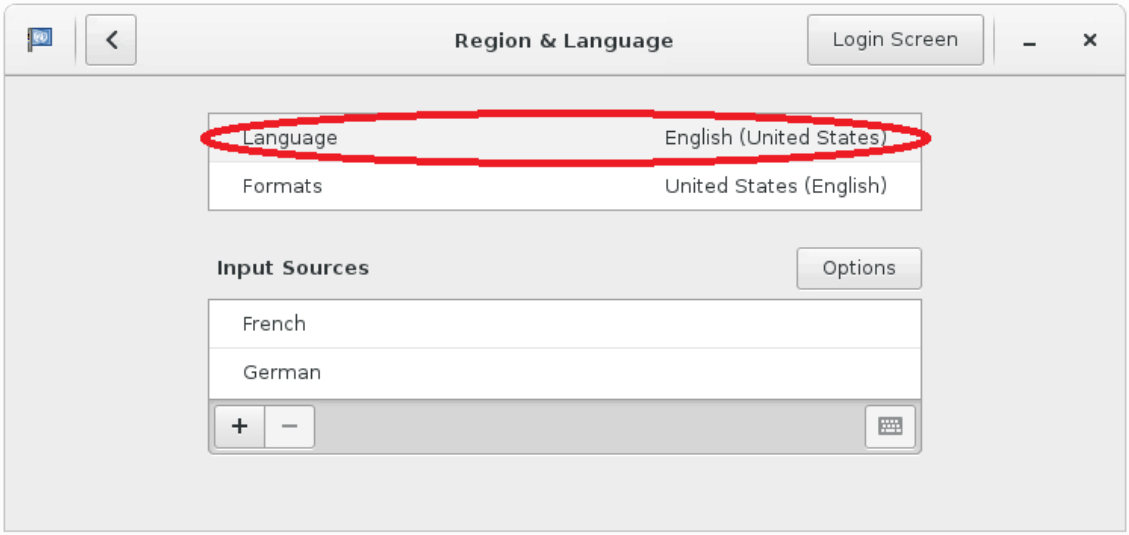
1. 启用功能。
2. 访问可以接受文本输入的已发布的应用程序。在会话中应用程序旁边会显示语言栏。
3. 从下拉菜单中选择区域和语言以添加所需的语言（输入源）。




4. 从下拉菜单中选择 IME 或键盘布局。
5. 使用所选的 IME 或键盘布局键入语言。

> 注意：
>
> - 在 VDA 端语言栏上更改键盘布局时，请确保在运行 Citrix Workspace 应用程序的客户端上使用相同的键盘布局。
> - 必须将 **accountsservice** 软件包升级到版本 0.6.37 或更高版本，才能在区域和语言对话框中执行设置。




## 动态键盘布局同步

November 8, 2021

以前，Linux VDA 和客户端设备上的键盘布局必须相同。例如，如果客户端设备上的键盘布局从“英语”更改为“法语”，但在 VDA 上未更改，会发生键映射问题并持续存在，直至 VDA 也更改为“法语”。

Citrix 通过自动将 VDA 的键盘布局与客户端设备的键盘布局同步来解决此问题。无论何时客户端设备上的键盘布局发生变化，VDA 随后都会做出恰当的调整。

> 注意：
>
> 适用于 HTML5 的 Citrix Workspace 应用程序不支持动态键盘布局同步功能。

### 配置

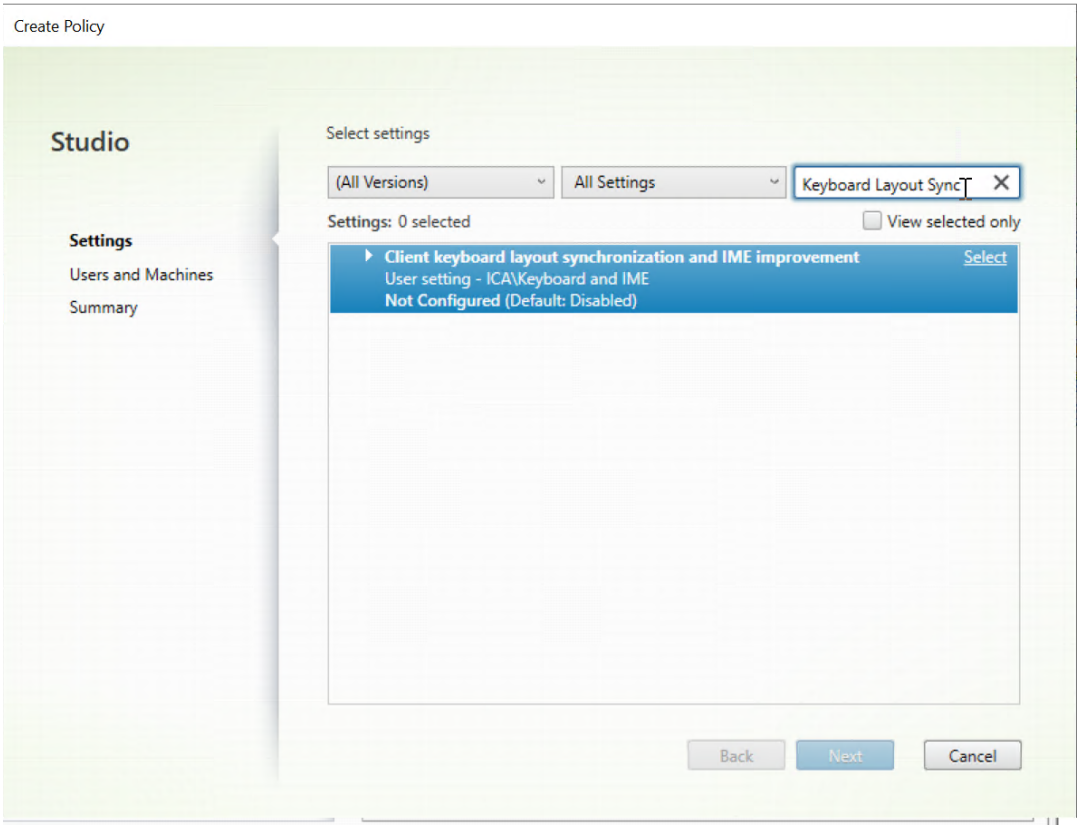
默认情况下，动态键盘布局同步功能处于禁用状态。要启用或禁用此功能，请设置客户端键盘布局同步和 **IME** 改进功能策略，或通过 `ctxreg` 实用程序编辑注册表。

> 注意：
>
> 客户端键盘布局同步和 **IME** 改进功能策略的优先级高于注册表设置，可以应用到您指定的用户和计算机对象或站点中的所有对象。给定 Linux VDA 上的注册表设置应用到该 VDA 上的所有会话。

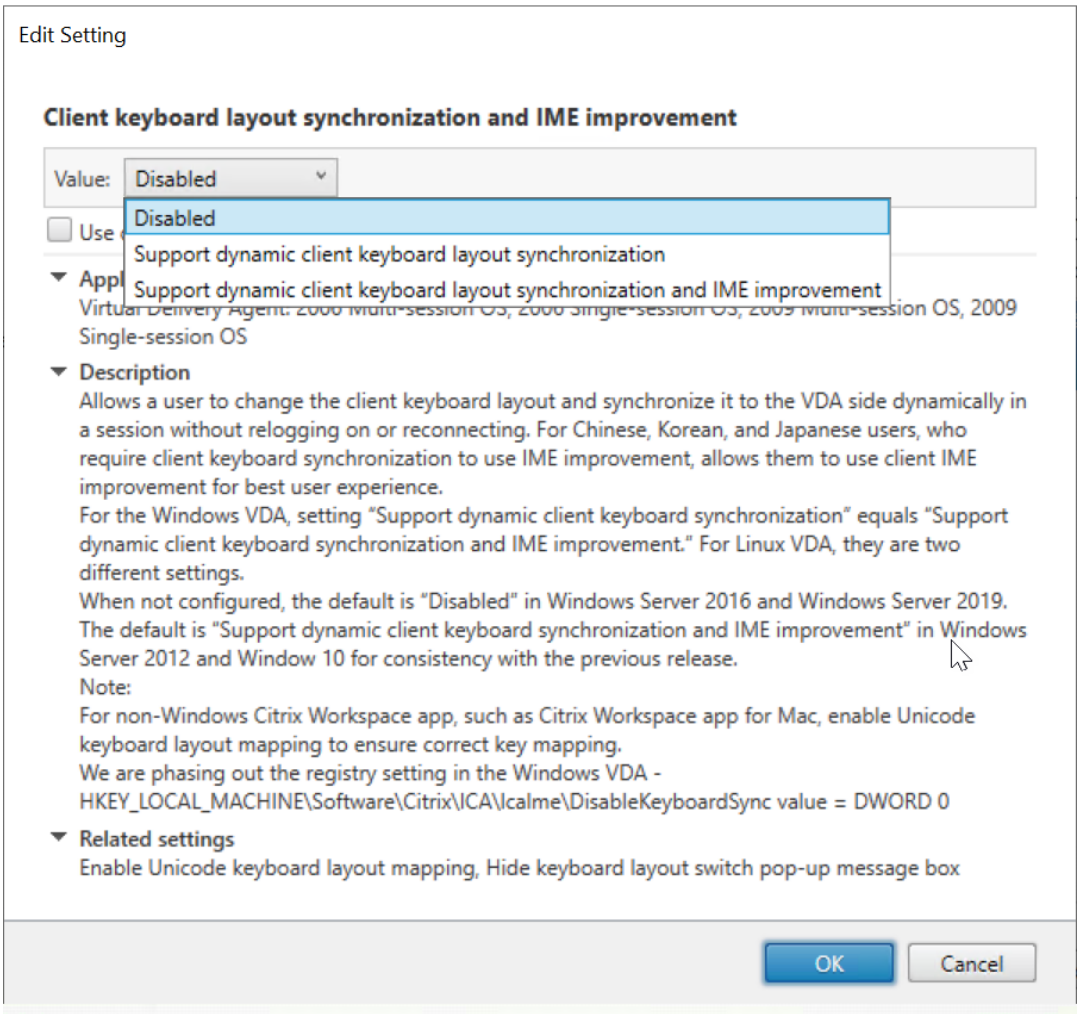
- 设置客户端键盘布局同步和 **IME** 改进功能策略以启用或禁用动态键盘布局同步功能：

  1. 在 Studio 中，右键单击策略，然后选择创建策略。
  2. 搜索客户端键盘布局同步和 **IME** 改进功能策略。




  3. 单击策略名称旁边的选择。
  4. 设置策略。




共有三个可用选项：

- 已禁用：禁用动态键盘布局同步和客户端 IME 用户界面同步。
- 支持动态客户端键盘布局同步：启用动态键盘布局同步，而不考虑位于 `HKEY_LOCAL_MACHINE\SYSTEM \CurrentControlSet\Control\Citrix\LanguageBar` 的 **SyncKeyboardLayout** 注册表项的 DWORD 值。
- 支持动态客户端键盘布局同步和 **IME** 改进功能：启用动态键盘布局同步和客户端 IME 用户界面同步，而不考虑位于 `HKEY_LOCAL_MACHINE\SYSTEM \CurrentControlSet\Control\Citrix\LanguageBar` 的 **SyncKeyboardLayout** 和 **SyncClientIME** 注册表项的 DWORD 值。

- 通过 `ctxreg` 实用程序编辑注册表以启用或禁用动态键盘布局同步功能：

  要启用此功能，请运行以下命令：

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

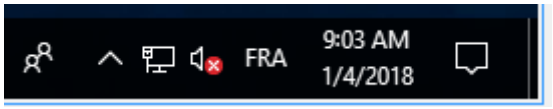
要禁用此功能，请运行以下命令：

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

### 使用情况

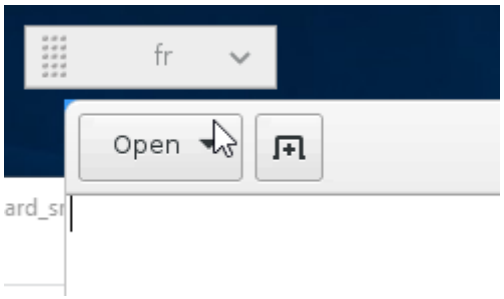
启用此功能后，如果会话过程中客户端设备上的键盘布局发生变化，会话的键盘布局也将相应地发生变化。

例如，如果将客户端设备上的键盘布局更改为“法语 (FR)”：

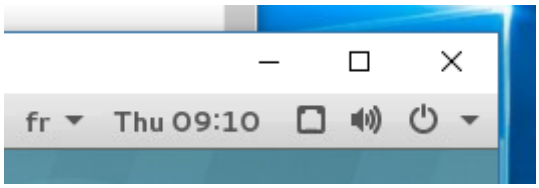



Linux VDA 会话的键盘布局随后也将更改为“fr”。

在应用程序会话中，如果启用了语言栏，则可以看到这一自动变化情形：




在桌面会话中，可以在任务栏中看到这一自动发生的变化：



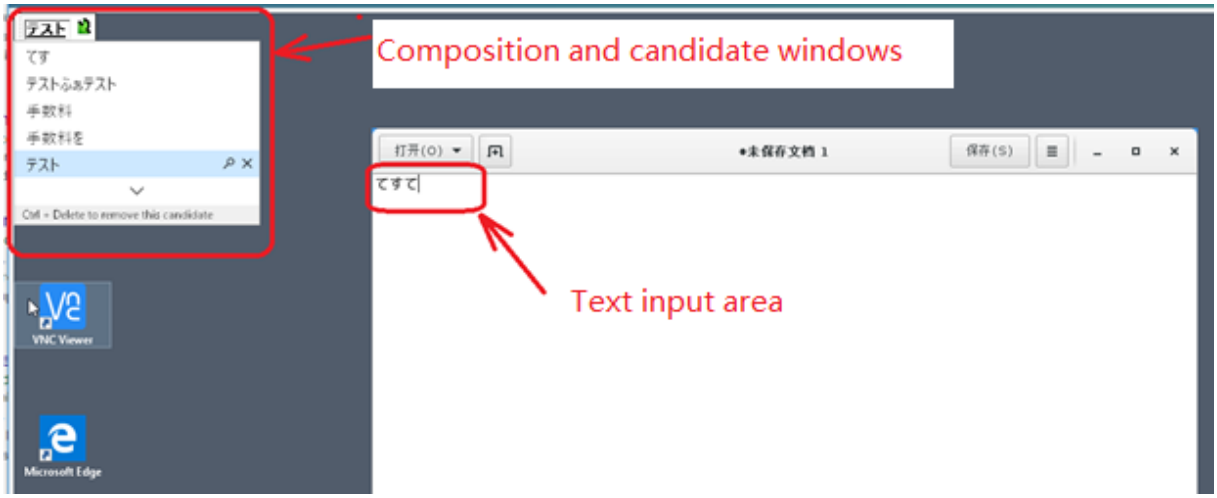

## 客户端 IME 用户界面同步

November 8, 2021

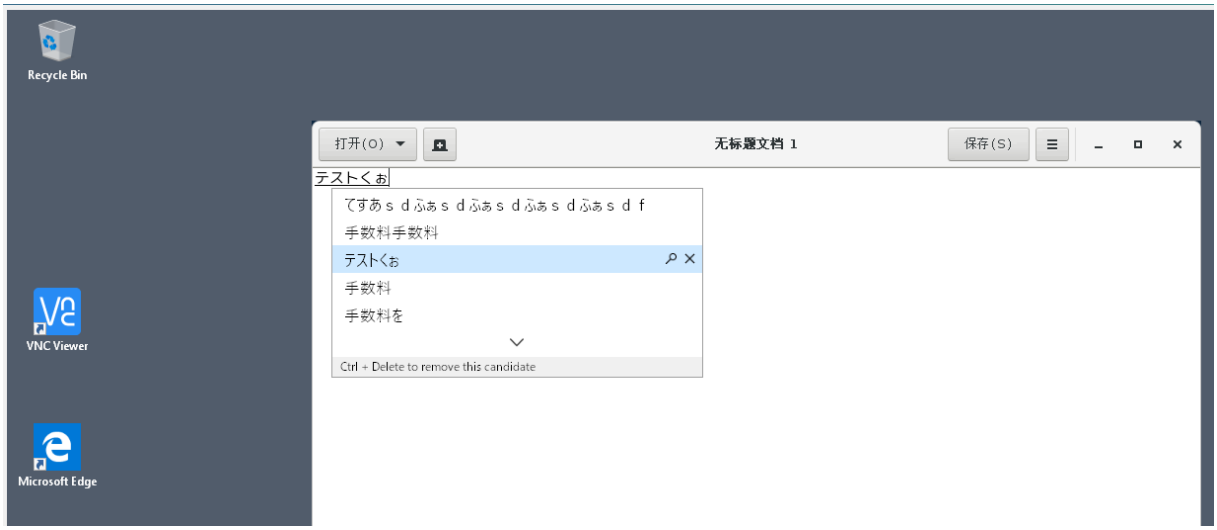
### 概述

迄今为止，客户端 IME 用户界面（包括撰写窗口和候选窗口）置于屏幕左上角。以前不跟随光标，有时距离文本输入区域中的光标较远：




Citrix 增强了可用性，并进一步提高了客户端 IME 的用户体验，如下所示：




> 注意：
>
> 该功能不适用于 RHEL 7.x、CentOS 7.x、Ubuntu 16.04、Ubuntu 18.04 和 SUSE 12.x。它在适用于 Windows 和 Mac 的 Citrix Workspace 应用程序上受支持。
>
> 要在 RHEL 7.x 桌面会话中使用该功能，必须启用 `IBus`。例如，将用户界面语言设置为需要 IME 输入的语言，或者将 **GTK_IM_MODULE=ibus** 添加到 **${HOME}/.config/imsettings/xinputrc** 文件中。
>
> 该功能是自动安装的，但您必须启用该功能，才能使用它。

### 启用和禁用该功能

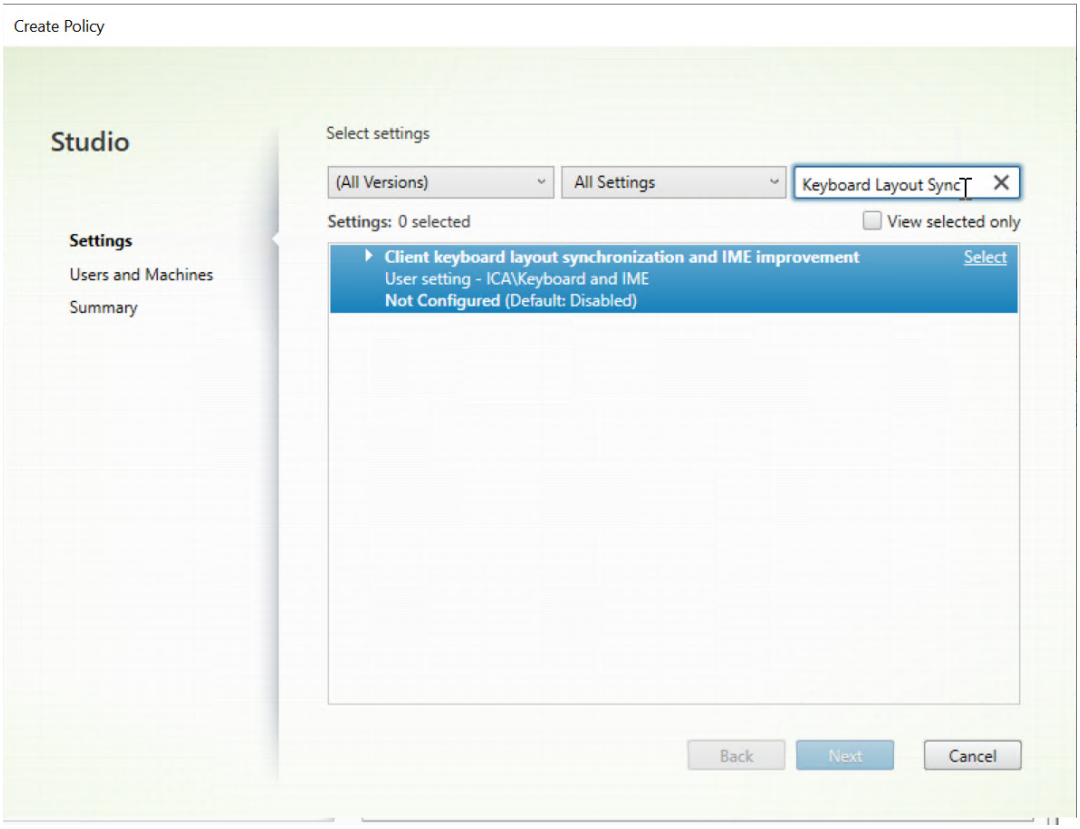
默认情况下禁用客户端 IME 用户界面同步功能。要启用或禁用此功能，请设置客户端键盘布局同步和 **IME** 改进功能策略，或通过 `ctxreg` 实用程序编辑注册表。

> 注意：
>
> 客户端键盘布局同步和 **IME** 改进功能策略的优先级高于注册表设置，可以应用到您指定的用户和计算机对象或站点中的所有对象。给定 Linux VDA 上的注册表设置应用到该 VDA 上的所有会话。

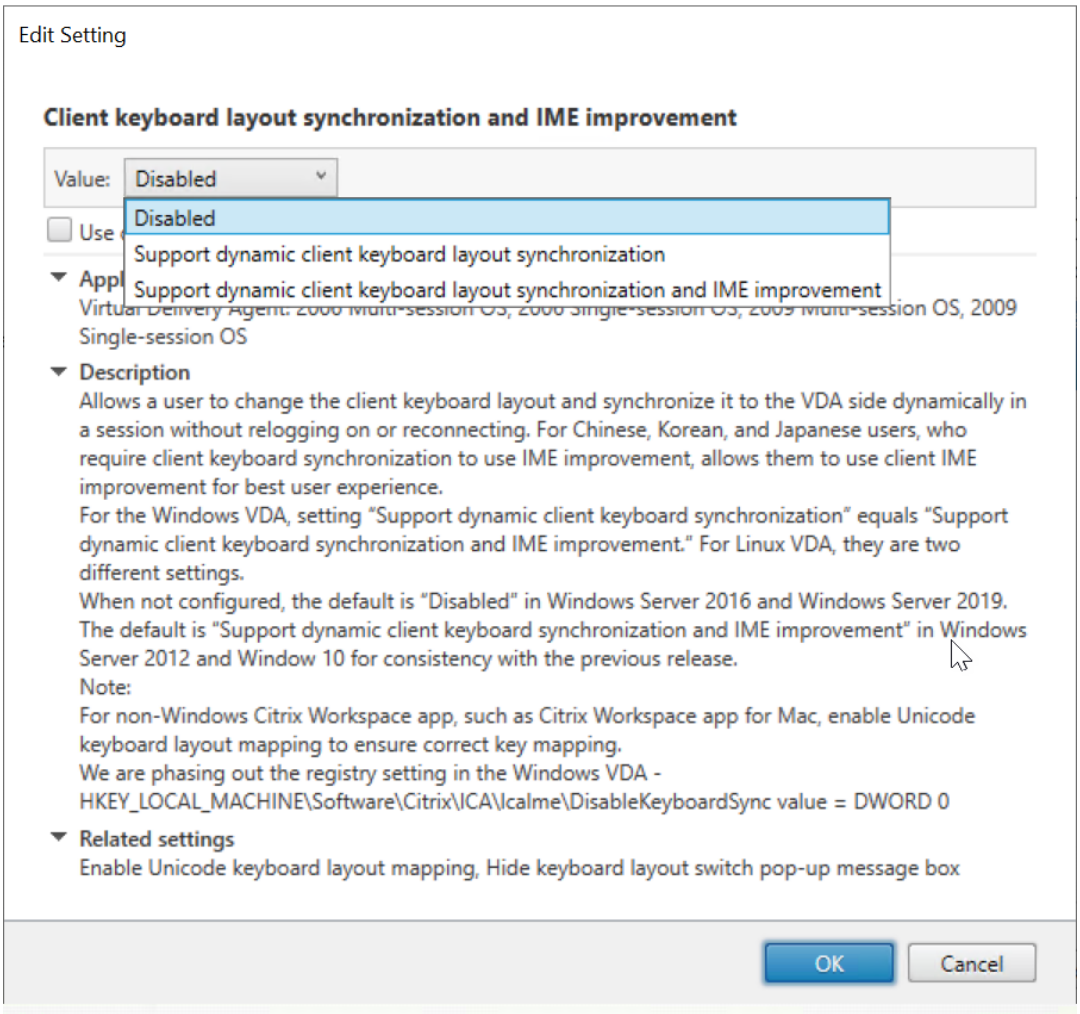
- 设置客户端键盘布局同步和 **IME** 改进功能策略以启用或禁用客户端 IME 用户界面同步功能：

  1. 在 Studio 中，右键单击策略，然后选择创建策略。
  2. 搜索客户端键盘布局同步和 **IME** 改进功能策略。




  3. 单击策略名称旁边的选择。
  4. 设置策略。




共有三个可用选项：

- 已禁用：禁用动态键盘布局同步和客户端 IME 用户界面同步。
- 支持动态客户端键盘布局同步：启用动态键盘布局同步，而不考虑位于 `HKEY_LOCAL_MACHINE\SYSTEM \CurrentControlSet\Control\Citrix\LanguageBar` 的 **SyncKeyboardLayout** 注册表项的 DWORD 值。
- 支持动态客户端键盘布局同步和 **IME** 改进功能：启用动态键盘布局同步和客户端 IME 用户界面同步，而不考虑位于 `HKEY_LOCAL_MACHINE\SYSTEM \CurrentControlSet\Control\Citrix\LanguageBar` 的 **SyncKeyboardLayout** 和 **SyncClientIME** 注册表项的 DWORD 值。

- 通过 `ctxreg` 实用程序编辑注册表以启用或禁用客户端 IME 用户界面同步功能：

  要启用该功能，请运行以下命令：

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

要禁用该功能，请运行以下命令：

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

## HDX Insight

August 20, 2024

### 概述

Linux VDA 部分支持 HDX Insight 功能。HDX Insight 是 Citrix Application Delivery Management (ADM) 的一部分，基于行业通用标准 AppFlow。通过 HDX Insight，IT 可以提供对通过 Citrix ADC 或 Citrix SD-WAN 应用程序网络结构传送的 Citrix ICA 通信前所未有的端到端可见性，从而实现卓越的用户体验。有关详细信息，请参阅 HDX Insight。

### 安装

没有依赖软件包需要安装。

### 使用情况

HDX Insight 将分析通过 Citrix Workspace 应用程序和 Linux VDA 之间的 Citrix ADC 传递的 ICA 消息。所有 HDX Insight 数据均来自 NSAP 虚拟通道并以未压缩方式发送。NSAP 虚拟通道默认处于启用状态。

以下命令分别禁用和启用 NSAP 虚拟通道：

```
sudo /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\
    VirtualDesktopAgent" -t "REG_DWORD" -v "EnableNSAP" -d "0x00000000"
    --force
```

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

### 故障排除

#### 不显示任何数据点

可能有两个原因：

- HDX Insight 未正确配置。

  例如，未在 Citrix ADC 上启用 AppFlow，或者在 Citrix ADM 上配置了不正确的 Citrix ADC 实例。

- 在 Linux VDA 上未启动 ICA 控制虚拟通道。

  ```
  ps aux | grep -i ctxctl
  ```

  如果 `ctxctl` 未运行，请与管理员联系以向 Citrix 报告缺陷。

不显示任何应用程序数据点

请确认是否启用了无缝虚拟通道且无缝应用程序启动了一段时间。

## Rendezvous 协议

October 9, 2022

在使用 Citrix Gateway 服务的环境中，Rendezvous 协议允许 HDX 会话绕过 Citrix Cloud Connector 直接安全地连接到 Citrix Gateway 服务。

要求：

- 使用 Citrix Workspace 和 Citrix Gateway 服务访问环境。
- 控制平面：Citrix Virtual Apps and Desktops 服务 (Citrix Cloud)。
- Linux VDA 版本 2012 或更高版本。
- 在 Citrix 策略中启用 Rendezvous 协议。有关详细信息，请参阅 Rendezvous 协议策略设置。
- VDA 必须对 `https://*.nssvc.net` 具有访问权限，包括所有子域。如果您无法通过该方式将所有子域列入白名单，请改为使用 `https://*.c.nssvc.net` 和 `https://*.g.nssvc.net`。有关详细信息，请参阅 Citrix Cloud 文档（在 Virtual Apps and Desktops 服务下方）的 Internet 连接要求部分和知识中心文章 CTX270584。
- 在代理会话时，Cloud Connector 必须获取 VDA 的 FQDN。要实现此目标，请为站点启用 DNS 解析：使用 Citrix Virtual Apps and Desktops 远程 PowerShell SDK，运行命令 `Set-BrokerSite -DnsResolutionEnabled $true`。有关 Citrix Virtual Apps and Desktops 远程 PowerShell SDK 的详细信息，请参阅 SDK 和 API。

> 重要
> :
> Rendezvous 协议不支持透明代理或显式代理。要使用的代理，请继续使用 Cloud Connector 传输 ICA 流量。

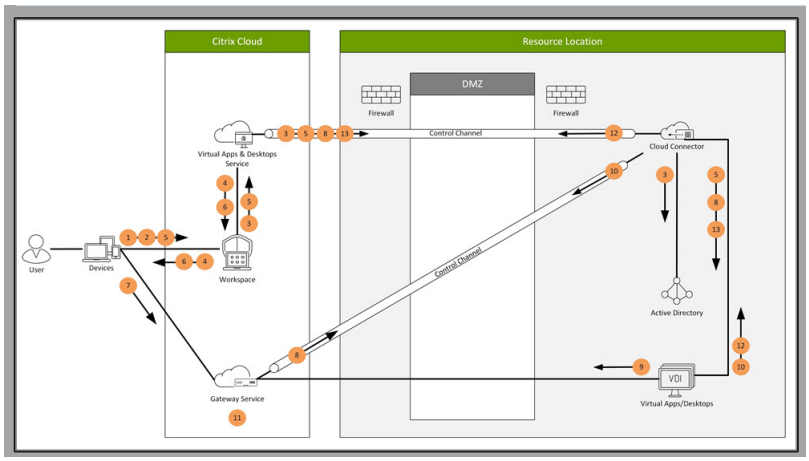
如果启用了 Rendezvous，并且 VDA 无法直接访问 Citrix Gateway 服务，则 VDA 将回退到通过 Cloud Connector 代理 HDX 会话。

如果您满足所有要求，请按照下列步骤验证是否正在使用 Rendezvous：

1. 在 VDA 上启动端点。
2. 运行 `su root -c "/opt/Citrix/VDA/bin/ctxquery -f iuStdP"`。
3. 传输协议指明连接类型：
   - TCP Rendezvous：TCP - SSL - CGP - ICA
   - EDT Rendezvous：UDP - DTLS - CGP - ICA
   - 通过 Cloud Connector 代理：TCP - CGP - ICA

下图概述了 Rendezvous 连接流程。请按照步骤了解流程。




1. 导航到 Citrix Workspace。
2. 在 Citrix Workspace 中输入凭据。
3. 如果使用本地 Active Directory，Citrix Virtual Apps and Desktops 服务将使用 Cloud Connector 通道通过 Active Directory 对凭据进行验证。
4. Citrix Workspace 显示 Citrix Virtual Apps and Desktops 服务中的枚举资源。
5. 从 Citrix Workspace 中选择资源。Citrix Virtual Apps and Desktops 服务向 VDA 发送一条消息，以便为传入会话做好准备。
6. Citrix Workspace 将 ICA 文件发送到包含 Citrix Cloud 生成的 STA 票证的端点。
7. 端点连接到 Citrix Gateway 服务，提供连接到 VDA 的票证，Citrix Cloud 将验证该票证。
8. Citrix Gateway 服务将连接信息发送到 Cloud Connector。Cloud Connector 确定连接是否应为 Rendezvous 连接，并将信息发送到 VDA。
9. VDA 建立与 Citrix Gateway 服务的直接连接。
10. 如果无法在 VDA 与 Citrix Gateway 服务之间建立直接连接，VDA 将通过 Cloud Connector 代理其连接。
11. Citrix Gateway 服务在端点与 VDA 之间建立连接。
12. VDA 通过 Cloud Connector 借助 Citrix Virtual Apps and Desktops 服务验证其许可证。
13. Citrix Virtual Apps and Desktops 服务通过 Cloud Connector 向 VDA 发送会话策略。这些政策已应用。

# 自适应传输

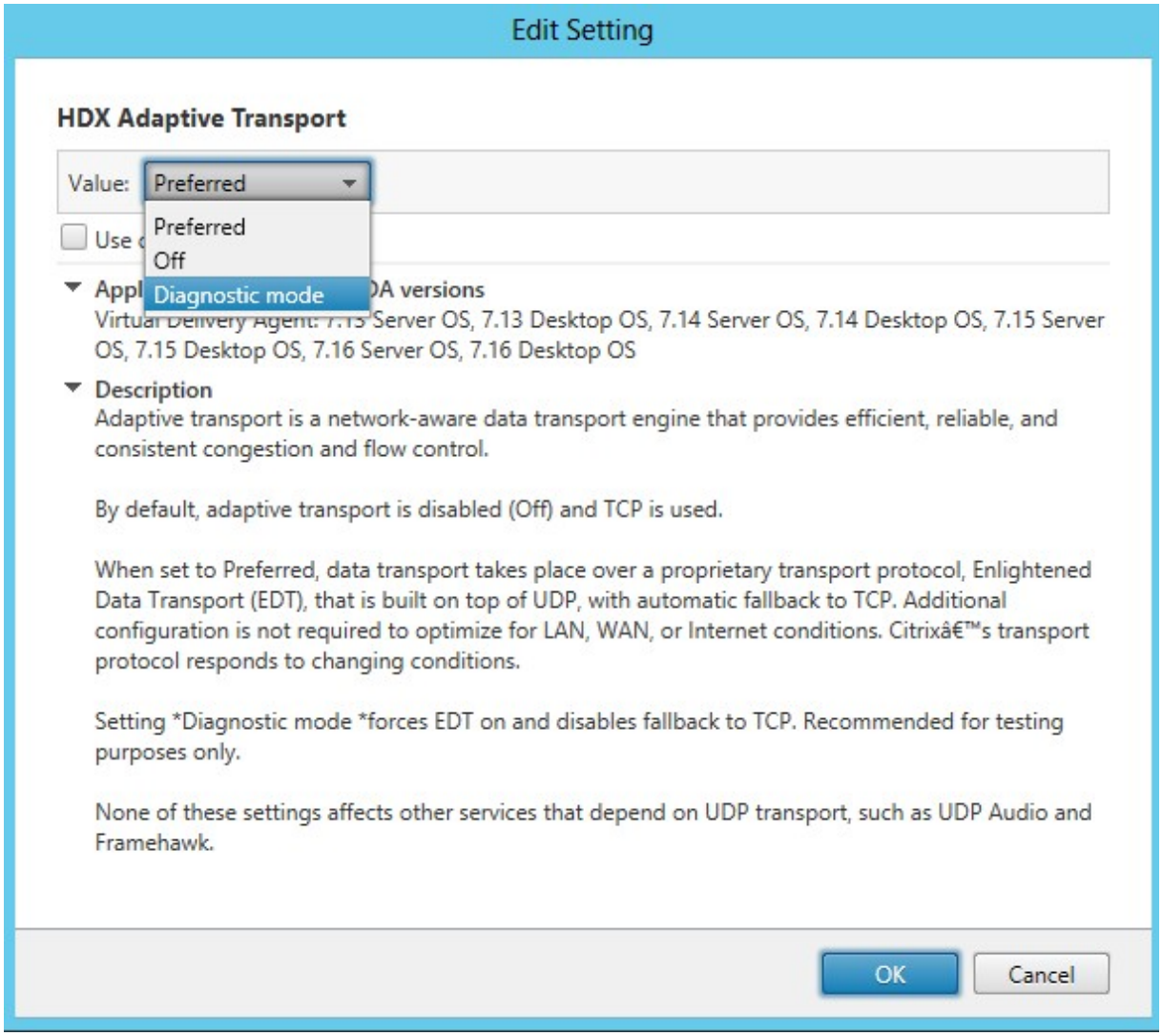
November 8, 2021

自适应传输是 Citrix Virtual Apps and Desktops 的数据传输机制。此传输速度更快，更具可扩展性，改进了应用程序的交互性，并且在具有挑战性的远距离 WAN 和 Internet 连接中互动性更强。有关自适应传输的详细信息，请参阅自适应传输。

## 启用自适应传输

在 Citrix Studio 中，验证 **HDX** 自适应传输策略设置为首选还是诊断模式。首选默认处于选中状态。

- 首选：尽可能使用基于 Enlightened Data Transport (EDT) 的自适应传输，并回退到 TCP。
- 诊断 模式：强制启用 EDT，并禁用回退到 TCP。



#### 禁用自适应传输

要禁用自适应传输，请在 Citrix Studio 中将 **HDX** 自适应传输策略设置为关。

#### 检查是否启用了自适应传输

要检查 UDP 侦听器是否正在运行，请运行以下命令。

```
netstat -an | grep "1494|2598"
```

在正常情况下，输出类似于如下所示。

```
udp        0      0 0.0.0.0:2598            0.0.0.0:*

udp       0      0 :::1494                 :::*
```

### EDT MTU 发现

EDT 在建立会话时自动确定最大传输单位 (MTU)。这样做可以防止出现可能会导致性能下降或无法建立会话的 EDT 数据包碎片。

最低要求：

- Linux VDA 2012
- 适用于 Windows 的 Citrix Workspace 应用程序 1911
- Citrix ADC：
  - 13.0.52.24
  - 12.1.56.22
- 必须启用会话可靠性

如果使用不支持此功能的客户端平台或版本，请参阅知识中心文章 CTX231821，以详细了解有关如何配置适合您环境的自定义 EDT MTU。

> 警告：
>
> 注册表编辑不当会导致严重问题，可能导致需要重新安装操作系统。Citrix 无法保证因注册表编辑器使用不当导致出现的问题能够得以解决。使用注册表编辑器需自担风险。在编辑注册表之前，请务必进行备份。

#### 在 **VDA** 上启用或禁用 **EDT MTU** 发现

默认情况下，EDT MTU 发现处于禁用状态。

- 要启用 EDT MTU 发现，请使用以下命令设置 `MtuDiscovery` 注册表项，重新启动 VDA，然后等待 VDA 注册：

  ```
  /opt/Citrix/VDA/bin/ctxreg create -k "HKLM\System\CurrentControlSet\Control\Terminal Server\Wds\icawd"-t "REG_DWORD"-v "MtuDiscovery"-d "0x00000001"--force
  ```

- 要禁用 EDT MTU 发现，请删除 `MtuDiscovery` 注册表值。

此设置在计算机范围内适用，影响从受支持的客户端连接的所有会话。

#### 控制客户端上的 **EDT MTU** 发现

可以通过在 ICA 文件中添加 `MtuDiscovery` 参数，在客户端上有选择地控制 EDT MTU 发现。要禁用此功能，请在 `Application` 部分下设置以下策略：

```
MtuDiscovery=Off
```

要重新启用此功能，请从 ICA 文件中删除 `MtuDiscovery` 参数。

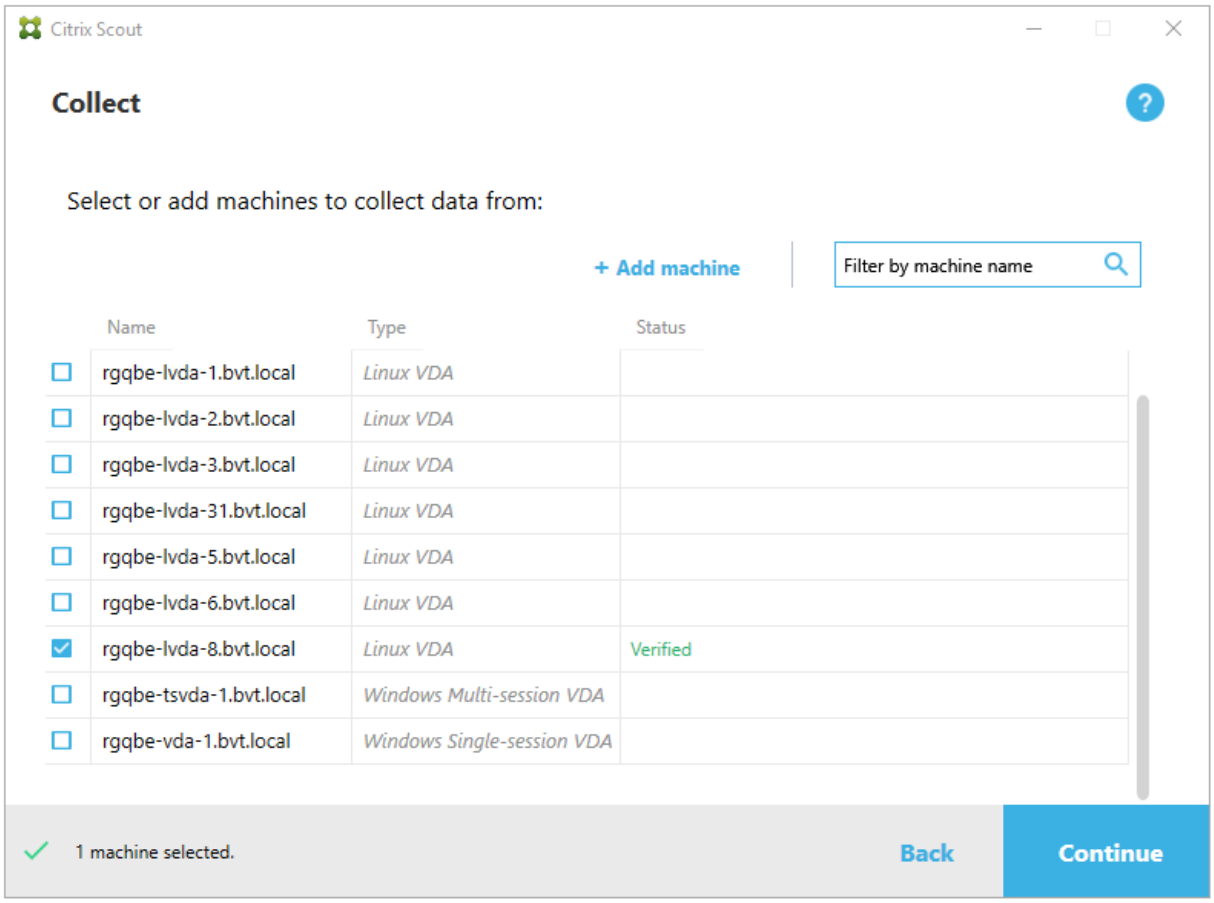
> 重要：
>
> 要使此 ICA 文件参数起作用，请在 VDA 上启用 EDT MTU 发现。如果未在 VDA 上启用 EDT MTU 发现，则 ICA 文件参数无效。

## 与 **Citrix Telemetry Service** 集成

November 8, 2021

通过与 Linux VDA 软件集成的 Citrix Telemetry Service (ctxtelemetry)，您可以运行 Citrix Scout，然后使用 /opt/Citrix/VDA/bin/xdlcollect.sh 脚本来收集有关 Linux VDA 的日志。




> 注意：
>
> 从 Linux VDA 1912 及更早版本升级后，必须重新运行 /opt/Citrix/VDA/sbin/ctxsetup.sh 以配置 Citrix Telemetry Service (ctxtelemetry) 的变量。有关这些变量的详细信息，请参阅轻松安装。

### 启用和禁用 **Citrix Telemetry Service**

- 要启用该服务，请运行 **sudo systemctl enable ctxtelemetry.socket** 命令。
- 要禁用该服务，请运行 **sudo systemctl disable ctxtelemetry.socket**。

### 端口

默认情况下，Citrix Telemetry Service (ctxtelemetry) 使用 TCP/IP 端口 7503 侦听 Citrix Scout。它使用 Delivery Controller 上的 TCP/IP 端口 7502 与 Citrix Scout 进行通信。

在安装 Linux VDA 时，可以使用默认端口或者通过以下变量更改端口。

- **CTX_XDL_TELEMETRY_SOCKET_PORT** - 用于侦听 Citrix Scout 的套接字端口。默认端口为 7503。
- **CTX_XDL_TELEMETRY_PORT** - 用于与 Citrix Scout 通信的端口。默认端口为 7502。

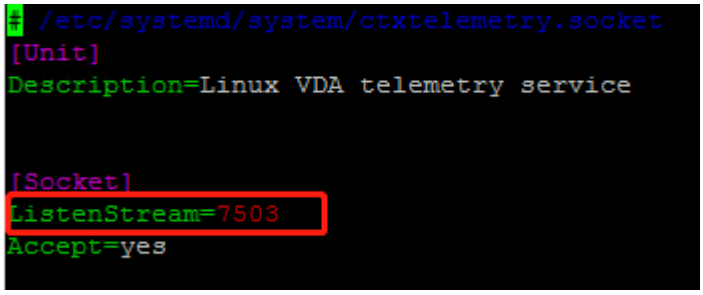
要在安装 VDA 后更改端口，请执行以下操作：

1. 要更改用于与 Scout 通信的端口，请运行以下命令。

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\Software\Citrix\
    VirtualDesktopAgent" -v "TelemetryServicePort" -d <port number>
     -t REG_DWORD
```

2. 要更改侦听 Scout 的套接字端口，请运行以下命令以打开并编辑 ctxtelemetry.socket 文件。

```
sudo vi /etc/systemd/system/ctxtelemetry.socket
```




3. 运行以下命令以重新启动套接字端口。

```
sudo systemctl daemon-reload
sudo systemctl stop ctxtelemetry.socket
sudo systemctl start ctxtelemetry.socket
```

4. 在防火墙配置中启用新端口。

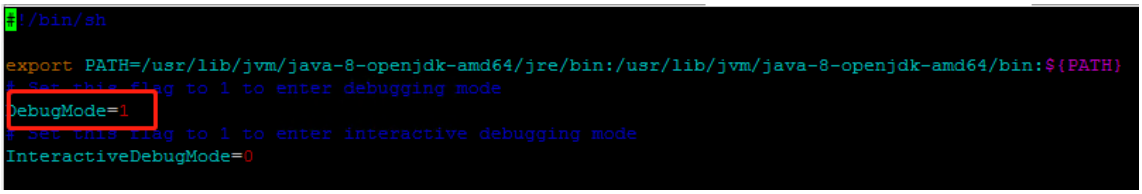
   例如，如果您使用的是 Ubuntu 发行版，请运行 **sudo ufw allow 7503** 命令以启用端口 7503。

## 调试模式

如果 Citrix Telemetry Service 无法按预期方式运行，则可以启用调试模式来确定原因。

1. 要启用调试模式，请运行以下命令以打开 ctxtelemetry 文件，然后将 DebugMode 值更改为 1。

```
sudo vi /opt/Citrix/VDA/sbin/ctxtelemetry
```




2. 手动停止 Citrix Telemetry Service，或者等待 15 分钟以使服务自动停止。

```
administrator@RGQBE-LVDA-3:~$ sudo netstat -ntlp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:139             0.0.0.0:*               LISTEN      1447/smbd
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      971/systemd-resolve
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      1309/sshd
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      25158/cupsd
tcp        0      0 127.0.0.1:5432          0.0.0.0:*               LISTEN      998/postgres
tcp        0      0 0.0.0.0:445             0.0.0.0:*               LISTEN      1447/smbd
tcp6       0      0 :::2598                 :::*                    LISTEN      28100/ctxhdx
tcp6       0      0 :::139                  :::*                    LISTEN      1447/smbd
tcp6       0      0 :::7502                 :::*                    LISTEN      1958/java
tcp6       0      0 :::7503                 :::*                    LISTEN      1/init
tcp6       0      0 :::80                   :::*                    LISTEN      1610/java
tcp6       0      0 :::1494                 :::*                    LISTEN      28100/ctxhdx
tcp6       0      0 :::22                   :::*                    LISTEN      1309/sshd
tcp6       0      0 ::1:631                 :::*                    LISTEN      25158/cupsd
tcp6       0      0 :::445                  :::*                    LISTEN      1447/smbd
administrator@RGQBE-LVDA-3:~$
```

在此示例中，可以运行以下命令来停止 Citrix Telemetry Service。

```
sudo netstat -ntlp
Kill -9 1958
```

3. 要重新启动 Citrix Telemetry Service，请在 Scout 上选择您的 Linux VDA 并在 /var/log/xdl/ 中查找 telemetry-debug.log。

### 服务等待时间

打开套接字端口的 `systemd` 守护程序默认启动并使用少量资源。Citrix Telemetry Service 默认处于停止状态，并仅在从 Delivery Controller 发出日志收集请求时启动。日志收集完成后，服务会等待新的收集请求，持续时间为 15 分钟，如果没有任何收集请求，则会再次停止。可以通过以下命令配置等待时间。最小值为 10 分钟。如果设置的值小于 10 分钟，则最小值 10 分钟将生效。设置等待时间后，停止并重新启动服务。

```
/opt/Citrix/VDA/bin/ctxreg create -k "HKLM\Software\Citrix\
    VirtualDesktopAgent" -v "TelemetryServiceIdleTimeoutInMinutes" -d <
    number> -t REG_DWORD
```

### 验证测试

在开始收集信息之前，验证测试将针对选定的每台计算机自动运行。这些测试将确保满足这些要求。如果某台计算机的测试失败，Scout 将显示一条消息，提供建议的更正措施。有关验证测试的详细信息，请参阅 Citrix Scout 文档中的验证测试部分。

## 启用跟踪

August 10, 2021

概述

收集日志并重现问题减慢了诊断速度，并且降低了用户体验。“启用跟踪”功能可以简化此类工作。跟踪功能默认对 Linux VDA 启用。

配置

`ctxlogd` 守护程序和 `setlog` 实用程序现在包括在 Linux VDA 版本软件包中。默认情况下，`ctxlogd` 守护程序在您安装并配置 Linux VDA 后启动。

**ctxlogd** 守护程序

跟踪的所有其他服务都基于 `ctxlogd` 守护程序。如果不希望跟踪 Linux VDA，可以停止 `ctxlogd` 守护程序。

**setlog** 实用程序

“启用跟踪”功能是使用 `setlog` 实用程序配置的，该程序位于 **/opt/Citrix/VDA/bin/** 路径下。只有 root 用户有权运行该程序。可以使用 GUI 或运行命令来查看和更改配置。请运行以下命令以获取使用 `setlog` 实用程序的帮助信息：

```
setlog help
```

值　默认情况下，**Log Output Path**（日志输出路径）设置为 **/var/log/xdl/hdx.log**，**Max Log Size**（最大日志大小）设置为 200 MB，您最多可以在 **Log Output Path**（日志输出路径）下保存两个旧日志文件。

查看当前的 `setlog` 值：

```
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
```

查看或设置单个 `setlog` 值：

```
setlog value <name> [<value>]
```

例如：

```
setlog value log_size 100
```

级别　默认情况下，日志级别设置为警告。

查看为不同组件设置的日志级别：

```
setlog levels
```

可以通过以下命令设置所有日志级别（包括禁用、已继承、详细、信息、警告、错误和致命错误）：

```
setlog level <class> [<level>]
```

**<class>** 变量指定 Linux VDA 的一个组件。要涵盖所有组件，请将其设置为 all：

```
setlog level all error

Setting log class ALL to ERROR.
```

标志　默认情况下，标志设置如下：

```
setlog flags

DATE = true

TIME = true

NAME = true

PID = true

TID = false

SID = true

UID = false

GID = false

CLASS = false

LEVEL = false

FUNC = true

FILE = false
```

查看当前标志：

```
setlog flags
```

查看或设置单个日志标志：

```
setlog flag <flag> [<state>]
```

还原默认值　将所有级别、标志和值还原到默认设置：

```
setlog default
```

> 重要
> :
> ctxlogd 服务是使用 **/var/xdl/.ctxlog** 文件配置的，只有 root 用户能够创建该文件。其他用户对该文件没有写入权限。我们建议 root 用户不要向其他用户授予写入权限。否则会导致对 ctxlogd 进行任意配置或恶意配置，这样会影响服务器性能，进而影响用户体验。

故障排除

**/var/xdl/.ctxlog** 文件丢失（例如，意外删除）时，ctxlogd 守护程序失败，您将无法重新启动 ctxlogd 服务。

/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.
```

要解决此问题，请以 root 用户身份运行 setlog 以重新创建 **/var/xdl/.ctxlog** 文件。然后重新启动其他服务所基于的 ctxlogd 服务。

## 重影会话

November 8, 2021

通过会话重影功能，域管理员可以在 Intranet 中查看用户的 ICA 会话。该功能使用 noVNC 连接到 ICA 会话，仅 RHEL 7.x 和 Ubuntu 16.04 支持该功能。

> 注意：
>
> Citrix Director 的版本必须为 7.16 或更高版本，才能使用会话重影功能。

## 安装和配置

### 依赖项

会话重影需要两个新的依赖关系 `python-websockify` 和 `x11vnc`。在 Ubuntu 16.04 上安装 Linux VDA 时，会自动安装 `python-websockify` 和 `x11vnc` 依赖项。在 RHEL 7.x 上，必须在安装 Linux VDA 之后手动安装 `python-websockify` 和 `x11vnc`。

在 RHEL 7.x 上运行以下命令以安装 `python-websockify` 和 `x11vnc`（`x11vnc` 版本 0.9.13 或更高版本）。

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

要解决 `python-websockify` 和 `x11vnc`，请在 RHEL 7.x 上启用以下存储库：

- 适用于 Enterprise Linux (EPEL) 的额外软件包

  `python-websockify` 和 `x11vnc` 都需要 EPEL 存储库。运行以下命令以启用 EPEL 存储库：

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

- 可选 RPM

  要启用可选的 RPM 存储库以安装 `x11vnc` 的一些依赖项软件包，请运行以下任一命令：

  对于工作站：

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

  对于服务器：

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

### 端口

会话重影功能会自动选择 6001-6099 内的可用端口来建立从 Linux VDA 到 Citrix Director 的连接。因此，可以并行重影的 ICA 会话数不应超过 99 个。请确保有足够的端口可用来满足您的要求，尤其是多会话重影。

### 注册表

下表列出了相关注册表：

| 注册表 | 说明 | 默认值 |
| --- | --- | --- |
| EnableSessionShadowing | 启用或禁用会话重影功能 | 1（启用） |
| ShadowingUseSSL | 确定是否对 Linux VDA 和 Citrix Director 之间的连接加密 | 0（禁用） |

可在 Linux VDA 上运行 `ctxreg` 命令来更改注册表值。例如，要禁用会话影子处理，请运行以下命令：

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\Software\Citrix\
    VirtualDesktopAgent" -v "EnableSessionShadowing" -d 0x00000000
```

### SSL

Linux VDA 和 Citrix Director 之间的 noVNC 连接使用 WebSocket 协议。对于会话重影，由前面提及的“ShadowingUseSSL”注册表确定选择 `ws://` 或 `wss://`。默认情况下，选择 `ws://`。但是，出于安全原因，我们建议您使用 `wss://`，并在每个 Citrix Director 客户端和每台 Linux VDA 服务器上安装证书。Citrix 拒绝承担使用 `ws://` 进行 Linux VDA 会话重影的任何安全责任。

获取服务器证书和根 **SSL** 证书　证书必须由可信证书颁发机构 (CA) 签发。

对于要在其中配置 SSL 的每台 Linux VDA 服务器来说，都需要一份单独的服务器证书（包括密钥）。服务器证书标识特定的计算机，因此您必须知道每台服务器的完全限定的域名 (FQDN)。为了方便起见，您可以对整个域使用通配符证书。在这种情况下，您必须至少知道域名。

除了在每台服务器上安装一份服务器证书之外，还必须在与 Linux VDA 服务器通信的每个 Citrix Director 客户端上安装从同一 CA 获取的根证书。根证书可从颁发服务器证书的同一 CA 获得。您可以安装来自以下 CA 的服务器证书和客户端证书：与您的操作系统捆绑的 CA，企业 CA（组织授予您访问权限的 CA），或没有与您的操作系统捆绑的 CA。请咨询您组织的安全团队，了解他们需要使用哪种方法来获取证书。

重要：

- 服务器证书的公用名必须是 Linux VDA 服务器的确切 FQDN 或至少是正确的通配符加域字符。例如 vda1.basedomain.com 或 *.basedomain.com。
- 哈希算法（包括 SHA1 和 MD5）对于数字证书中的签名而言太弱，某些浏览器不支持。因此，SHA-256 指定为最低标准。

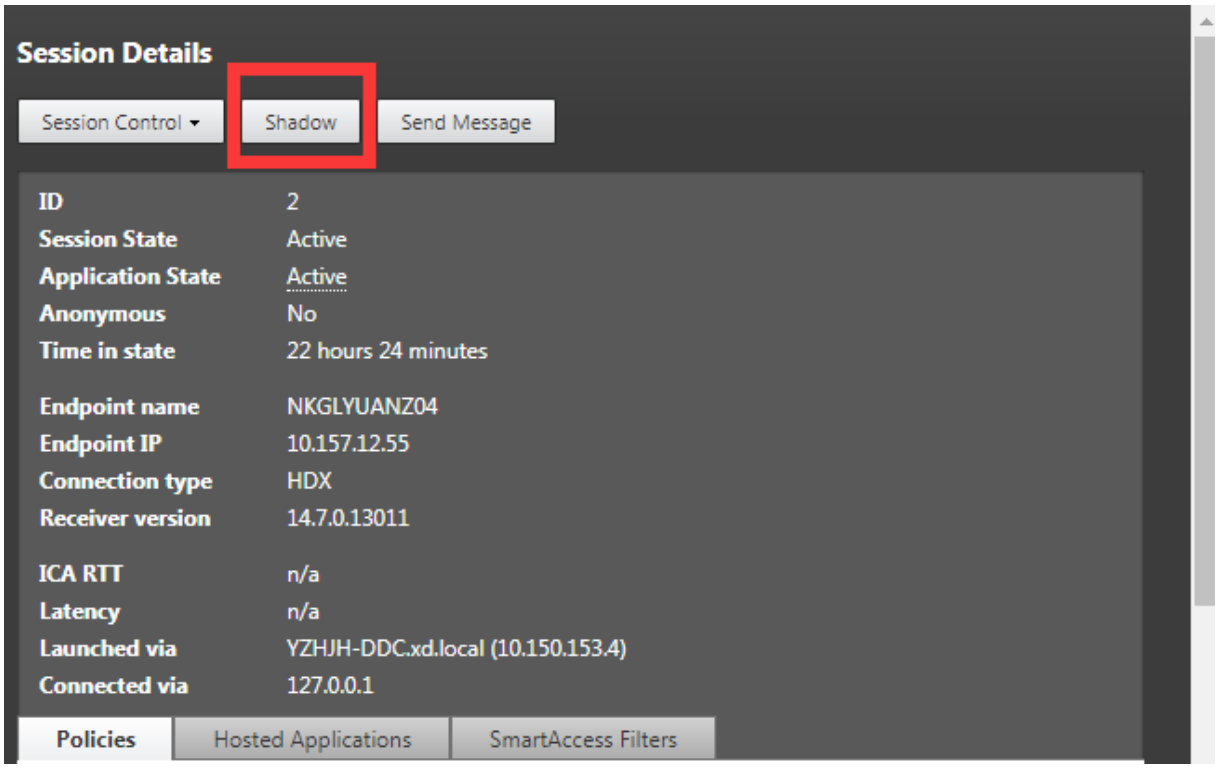
在每个 **Citrix Director** 客户端上安装根证书　会话重影与 IIS 使用相同的基于注册表的证书存储，因此您可以使用 IIS 或 Microsoft 管理控制台 (MMC) 证书管理单元安装根证书。收到 CA 的证书后，可在 IIS 中重新启动 Web 服务器证书向导，然后此向导将安装该证书。另外，您可以在使用 MMC 并将证书作为独立的管理单元添加的计算机上，查看并导入证书。默认情况下，Internet Explorer 和 Google Chrome 会导入安装在操作系统上的证书。对于 Mozilla Firefox，必须在证书管理器的 **Authorities**（颁发机构）选项卡上导入您的根 SSL 证书。

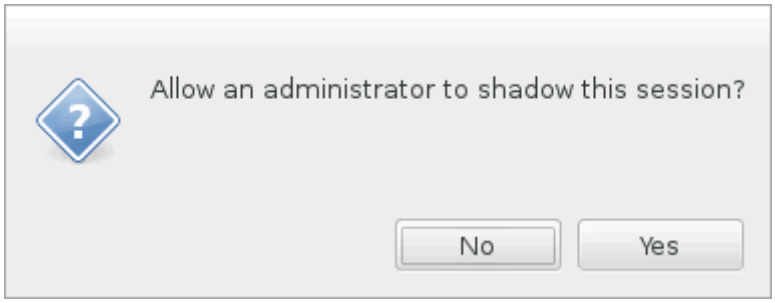
在每台 **Linux VDA** 服务器上安装服务器证书及其密钥　将服务器证书命名为“shadowingcert.*”，并将密钥文件命名为“shadowingkey.*”（* 可以指示格式，例如 shadowingcert.csr 和 shadowingkey.key）。将服务器证书和密钥文件放在路径 **/etc/xdl/shadowingssl** 下，并使用受限权限适当地对其进行保护。如果名称或路径不正确，Linux VDA 将无法找到特定的证书或密钥文件，从而导致与 Citrix Director 的连接失败。

使用情况

在 Citrix Director 中，找到目标会话，然后单击会话详细信息视图中的重影以向 Linux VDA 发送重影请求。




连接初始化后，ICA 会话客户端（不是 Citrix Director 客户端）上会显示一条确认消息，要求用户允许对会话进行重影。




如果用户单击是，则 Citrix Director 端将显示一个窗口，指示正在重影 ICA 会话。

有关用法的详细信息，请参阅 Citrix Director 文档。

限制

- 会话重影仅适用于 Intranet。它不适用于外部网络，即使通过 Citrix Gateway 连接也是如此。Citrix 拒绝承担在外部网络中进行 Linux VDA 会话重影的任何责任。
- 启用了会话重影后，域管理员只能查看 ICA 会话，但无权对其写入或控制。
- 管理员在 Citrix Director 中单击重影后，系统会显示一条确认消息，要求用户允许对会话进行重影。只有在会话用户允许后才能对会话进行重影。
- 前面提及的确认消息有超时限制，即 20 秒。超时后，重影请求将失败。
- 在一个 Citrix Director 窗口中，一个 ICA 会话只能由一位管理员重影。如果某个 ICA 会话已由管理员 A 重影，此时，管理员 B 发送重影请求，则会在用户设备上重新出现要求用户允许的确认消息。如果用户同意，则管理员 A 的重影连接将停止，并为管理员 B 创建一个新的重影连接。如果同一位管理员发送另一个针对同一 ICA 会话的重影请求，也是同样情况。
- 要使用会话重影，请安装 Citrix Director 7.16 或更高版本。
- Citrix Director 客户端使用 FQDN 而不是 IP 地址来连接到目标 Linux VDA 服务器。因此，Citrix Director 客户端必须能够解析 Linux VDA 服务器的 FQDN。

故障排除

如果会话重影失败，请在 Citrix Director 客户端和 Linux VDA 上执行调试。

在 **Citrix Director** 客户端上

通过浏览器的开发人员工具，在控制台选项卡上检查输出日志。或者在网络选项卡上检查 ShadowLinuxSession API 的响应。如果出现了要求用户允许的确认消息，但无法建立连接，请手动 ping Linux VDA 的 FQDN 以验证 Citrix Director 是否可以解析 FQDN。如果 `wss://` 连接出现问题，请检查您的证书。

在 **Linux VDA** 上

确认在出现重影请求时，是否显示要求用户允许的确认消息。如果没有，请检查 vda.log 和 hdx.log 文件以获取相关线索。要获取 vda.log 文件，请执行以下操作：

1. 找到 /etc/xdl/ctx-vda.conf 文件。取消注释以下行以启用 vda.log 配置：

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

2. 打开 /etc/xdl/log4j.xml，找到 com.citrix.dmc 部分，并将“info”更改为“trace”，如下所示：

```
<!-- Broker Agent Plugin - Director VDA plugin Logger -->

 <logger name="com.citrix.dmc">

   <level value="trace"/>

```

```
7    </logger>
```

3. 运行 `service ctxvda restart` 命令以重新启动 `ctxvda` 服务。

如果在建立连接期间出现错误：

1. 检查是否有任何防火墙限制阻止会话重影打开端口。
2. 在使用 SSL 的情况下，确认证书和密钥文件是否正确命名，并放在正确的路径下。
3. 确认 6001-6099 之间是否有足够的端口用于新的重影请求。

## 浏览器内容重定向

January 26, 2022

### 概述

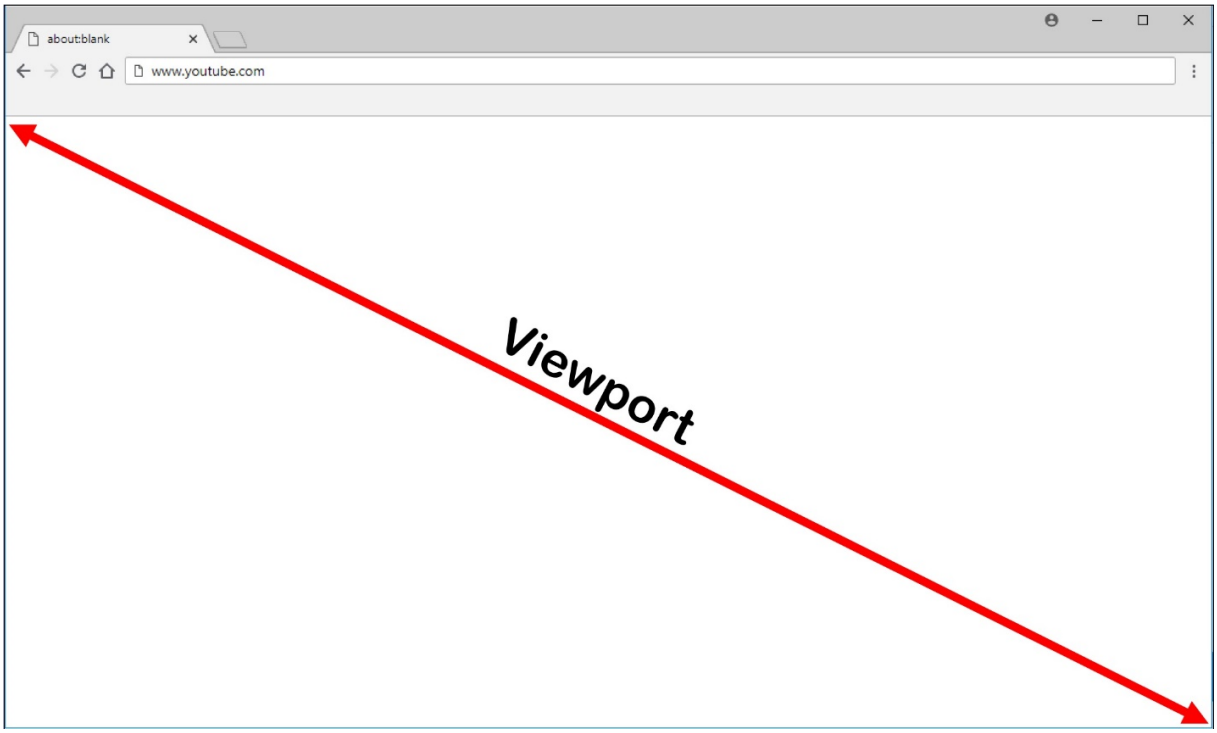
Linux VDA 在 Google Chrome 中支持浏览器内容重定向。浏览器内容重定向提供了在客户端呈现允许列表中的 Web 页面的功能。此功能会使用 Citrix Workspace 应用程序在客户端实例化相应的呈现引擎，该引擎会从 URL 提取 HTTP 和 HTTPS 内容。

> 注意：
>
> 可以使用允许列表指定哪些 Web 页面被重定向到客户端。反过来，可以使用阻止列表指定哪些 Wen 页面不重定向到客户端。

此叠加 Web 布局引擎在客户端上运行，而非在 VDA 上运行，并且使用客户端 CPU、GPU、RAM 和网络。

只有浏览器视口会进行重定向。视口是浏览器中显示内容的矩形区域。视口不包括地址栏、收藏夹栏和状态栏等项目。这些项目仍在 VDA 上的浏览器中运行。




系统要求

**Windows** 客户端：

- 适用于 Windows 的 Citrix Workspace 应用程序 1809 或更高版本

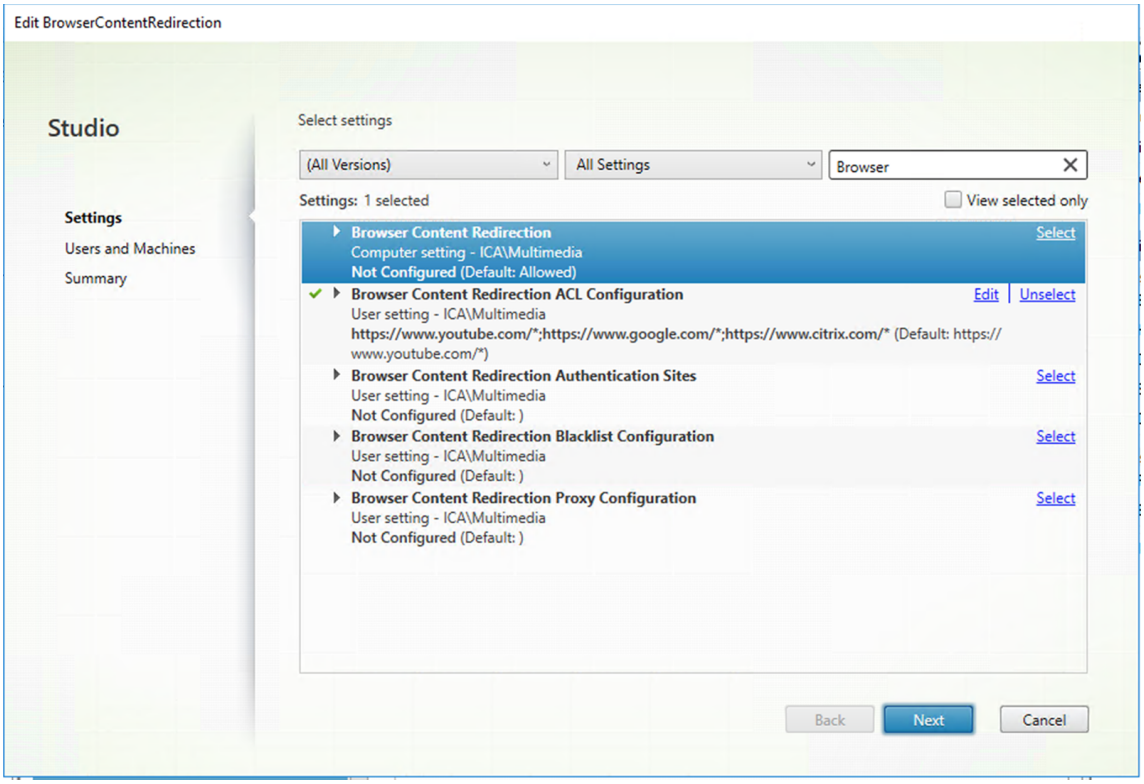
**Linux VDA**：

- VDA 操作系统：Ubuntu 16.04、Ubuntu 18.04、RHEL 7.8、RHEL 8.2、RHEL 8.1、SLES 12.5
- VDA 上的浏览器：添加了 Citrix 浏览器内容重定向扩展程序的 Google Chrome v66 或更高版本

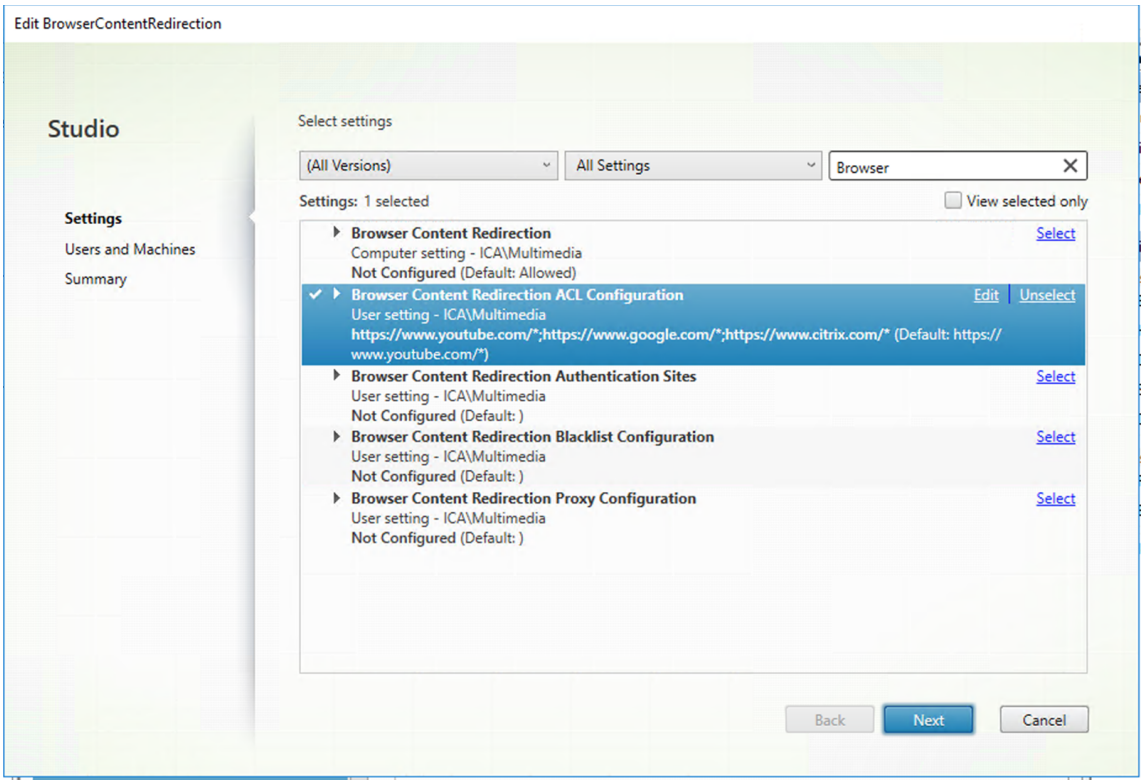
配置浏览器内容重定向

1. 在 Citrix Studio 中，配置一个策略，该策略指定可以使用浏览器内容重定向的 URL 的允许列表以及不能使用浏览器内容重定向的 URL 的阻止列表。默认情况下，浏览器内容重定向设置为允许。

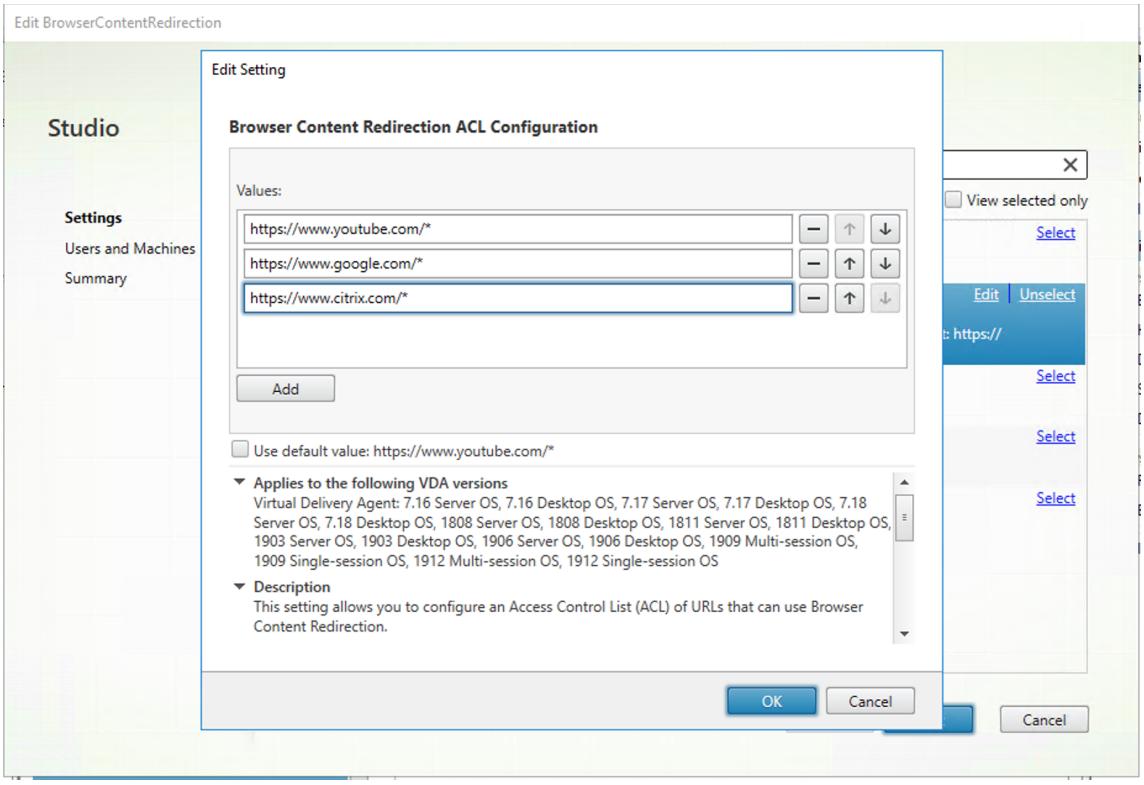
 



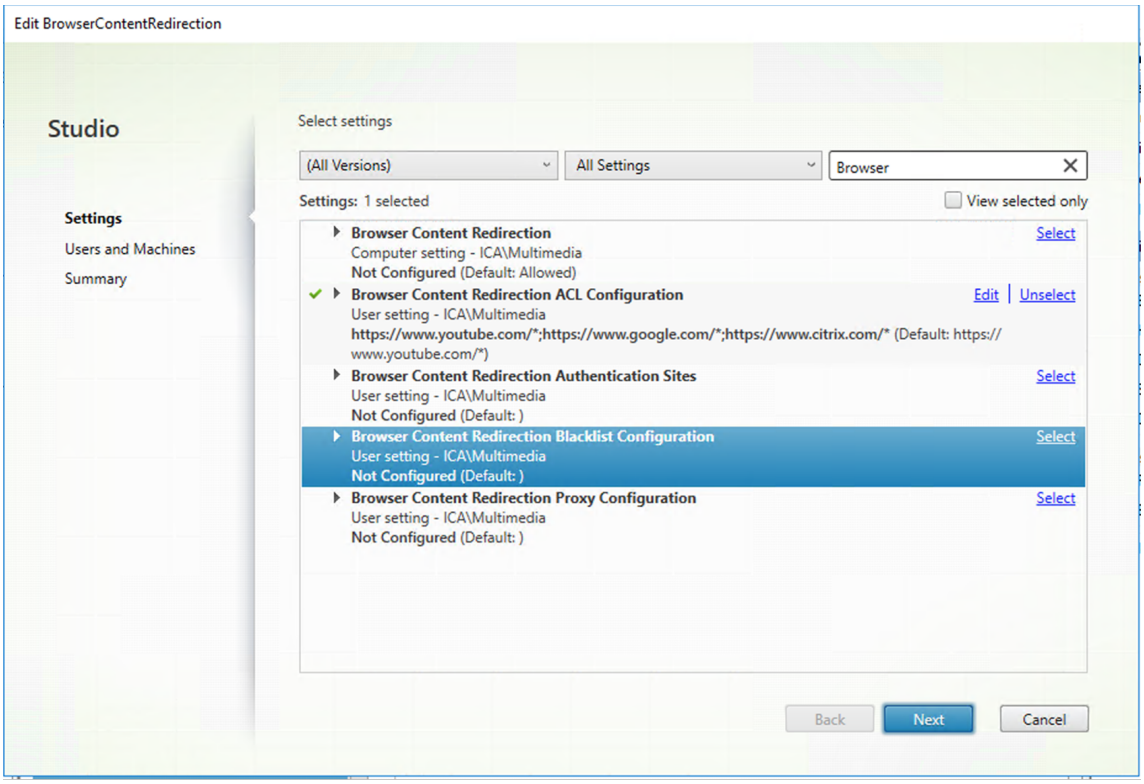

浏览器内容重定向 **ACL** 配置设置指定可以使用浏览器内容重定向的 URL 的允许列表。






浏览器内容重定向黑名单配置设置指定不能使用浏览器内容重定向的 URL 的阻止列表。

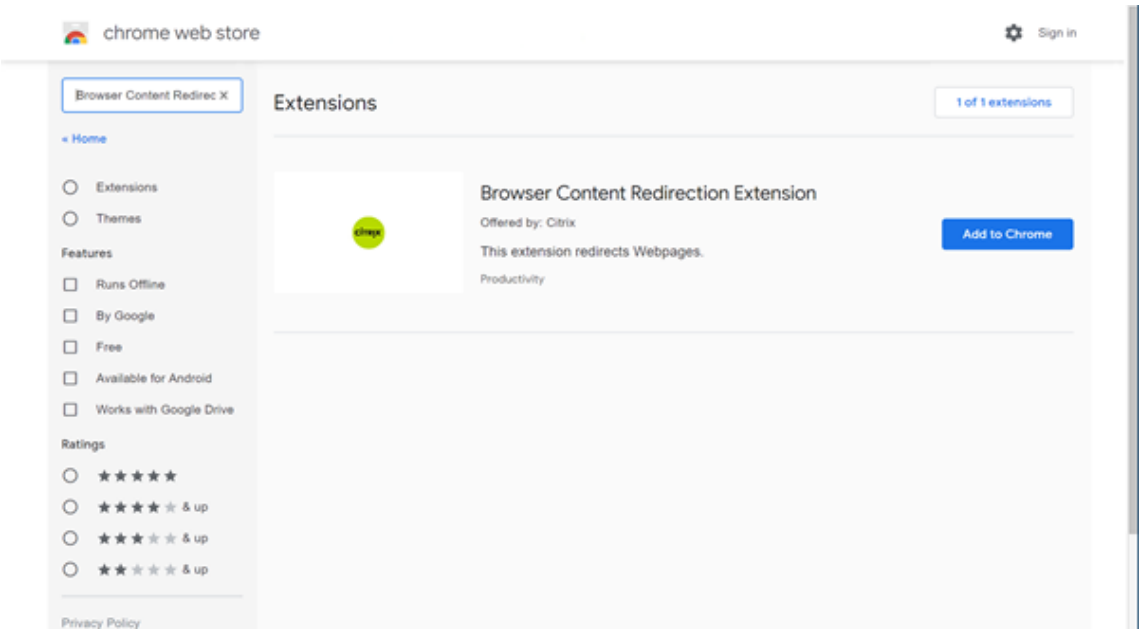


> 注意：
>
> Linux VDA 当前不支持浏览器内容重定向代理配置设置。

2. 要使 VDA 上的浏览器检测（导航到的）URL 是否与允许列表或阻止列表匹配，请从 Chrome 网上应用店添加 Citrix 浏览器内容重定向扩展程序。单击 VDA 上的添加到 **Chrome**。

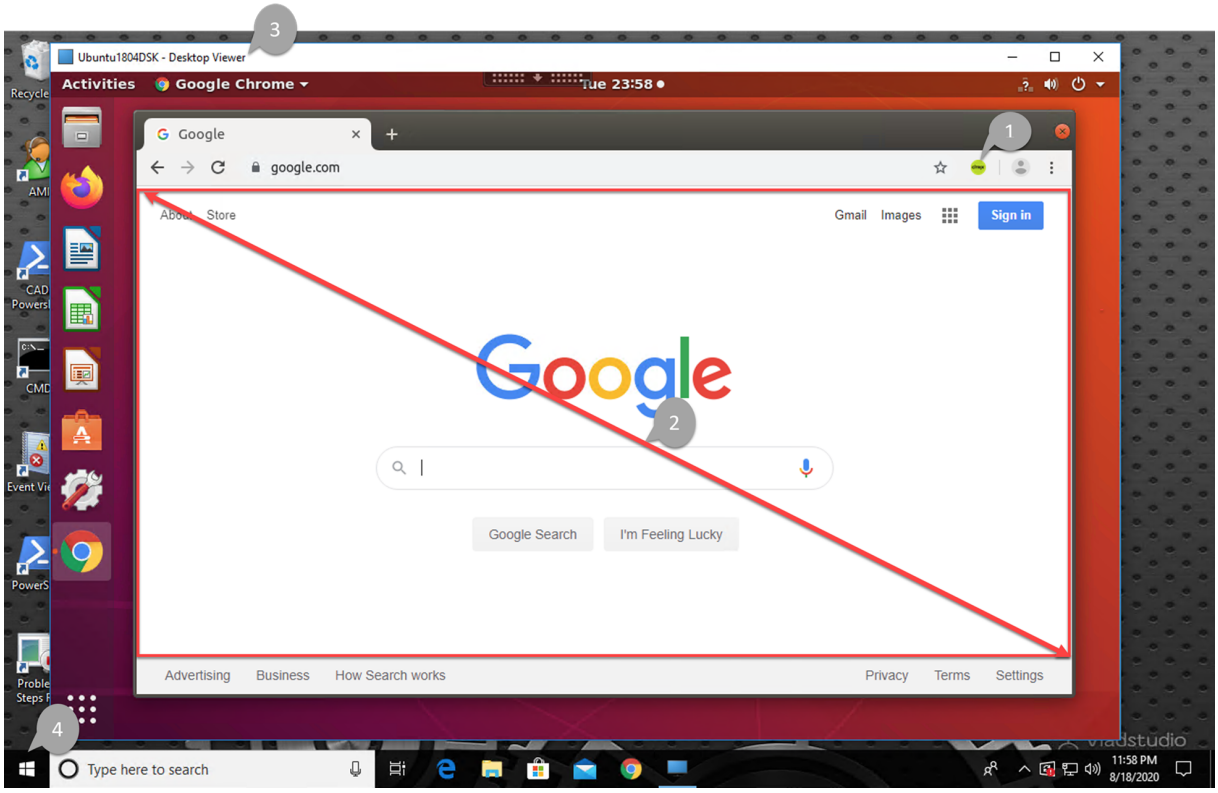
> 重要：
>
> 客户端上不需要此扩展程序。仅在 VDA 上添加。
>
> Chrome 扩展程序基于每个用户安装。不需要更新黄金映像即可添加或删除扩展程序。




如果在允许列表（例如 `https://www.mycompany.com/`）中找到与 URL 匹配的 URL，但在任何阻止列表中都未找到，虚拟通道 (CTXCSB) 会指示 Citrix Workspace 应用程序需要重定向并中继 URL。然后，Citrix Workspace 应用程序会实例化一个本地呈现引擎并显示此 Web 站点。

之后，Citrix Workspace 应用程序会将此 Web 站点无缝融入虚拟桌面浏览器内容区域中。




1. Citrix 浏览器内容重定向扩展程序的图标

   扩展程序图标的颜色指定 Chrome 扩展程序的状态。其颜色为以下三种颜色之一：

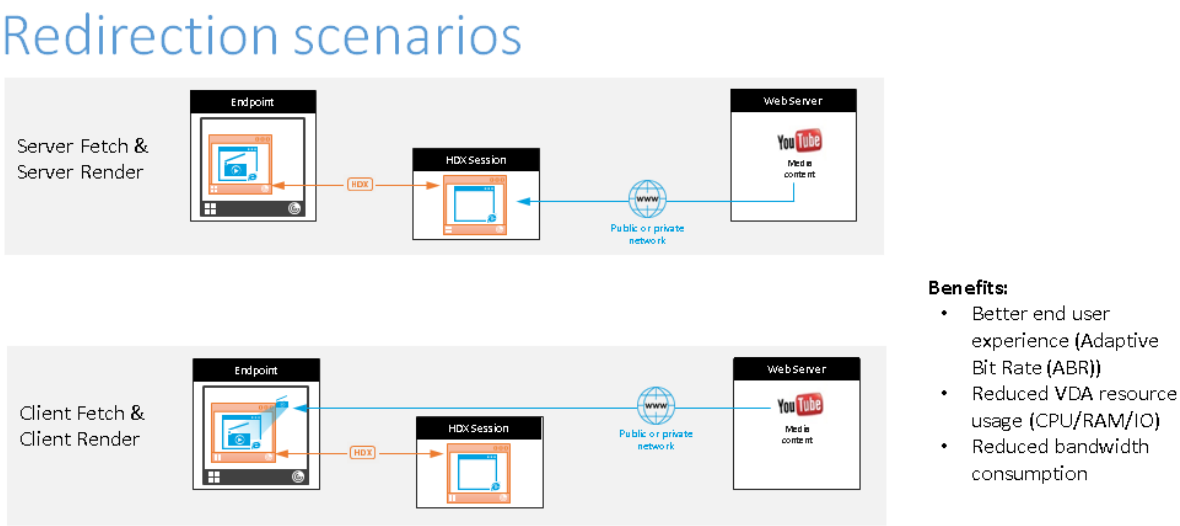
   - 绿色：活动并连接
   - 灰色：在当前选项卡上不活动/空闲
   - 红色：已损坏/不运行

2. 视口在客户端上呈现或混合回虚拟桌面
3. Linux VDA
4. Windows 客户端

## 重定向场景

下面是 Citrix Workspace 应用程序提取内容的方式的几种情况：




- 服务器提取和服务器呈现：由于没有将站点添加到允许列表或重定向失败，因此没有重定向。我们将回退到在 VDA 上呈现 Web 页面，并使用 Thinwire 来远程显示图形。使用策略来控制回退行为。这种情况会导致 VDA 上的 CPU、RAM 和带宽消耗较高。
- 客户端提取和客户端呈现：由于 Citrix Workspace 应用程序直接连接 Web 服务器，因此需要访问 Internet。在这种情况下，会从 Citrix Virtual Apps and Desktops 站点卸载所有网络、CPU 和 RAM 使用量。

### 回退机制

客户端重定向有时可能会失败。例如，如果客户端计算机无法直接访问 Internet，则可能会向 VDA 返回一条错误响应。在这种情况下，VDA 上的浏览器可以在服务器上重新加载并呈现页面。

## 支持适用于 **HTML5** 的 **Citrix Workspace** 应用程序

### November 8, 2021

从此版本开始，您可以使用适用于 HTML5 的 Citrix Workspace 应用程序来直接访问 Linux 虚拟应用程序和桌面，而无需将客户端连接到 Citrix Gateway。有关适用于 HTML5 的 Citrix Workspace 应用程序的信息，请参阅 Citrix 文档。

### 启用此功能

默认情况下，此功能处于禁用状态。要启用该功能，请执行以下操作：

1. 在 Citrix StoreFront 中，启用适用于 HTML5 的 Citrix Workspace 应用程序。

   有关详细过程，请参阅知识中心文章 CTX208163 的“步骤 1”。

2. 启用 WebSocket 连接。

   a) 在 Citrix Studio 中，将 **WebSocket** 连接策略设置为允许。

      您还可以设置其他 WebSocket 策略。有关 WebSocket 策略的完整列表，请参阅 WebSockets 策略设置。

   b) 在 VDA 上，请按此顺序重新启动 `ctxvda` 服务和 `ctxhdx` 服务，以使设置生效。

   c) 在 VDA 上，运行以下命令来检查 WebSocket 侦听器是否正在运行。

      ```
      netstat -an | grep 8008
      ```

      WebSocket 侦听器处于运行状态时，命令输出将如下所示：

      ```
      tcp 0 0 :::8008 :::* LISTEN
      ```

   注意：您还可以启用 TLS 加密以保护 WebSocket 连接的安全。有关启用 TLS 加密的信息，请参阅使用 TLS 保护用户会话安全。

## 监视 **Citrix Director** 中的 **Linux VM** 和 **Linux** 会话

June 16, 2023

本文列出了 Citrix Director 中可用于 Linux VM 和 Linux 会话的一些指标。

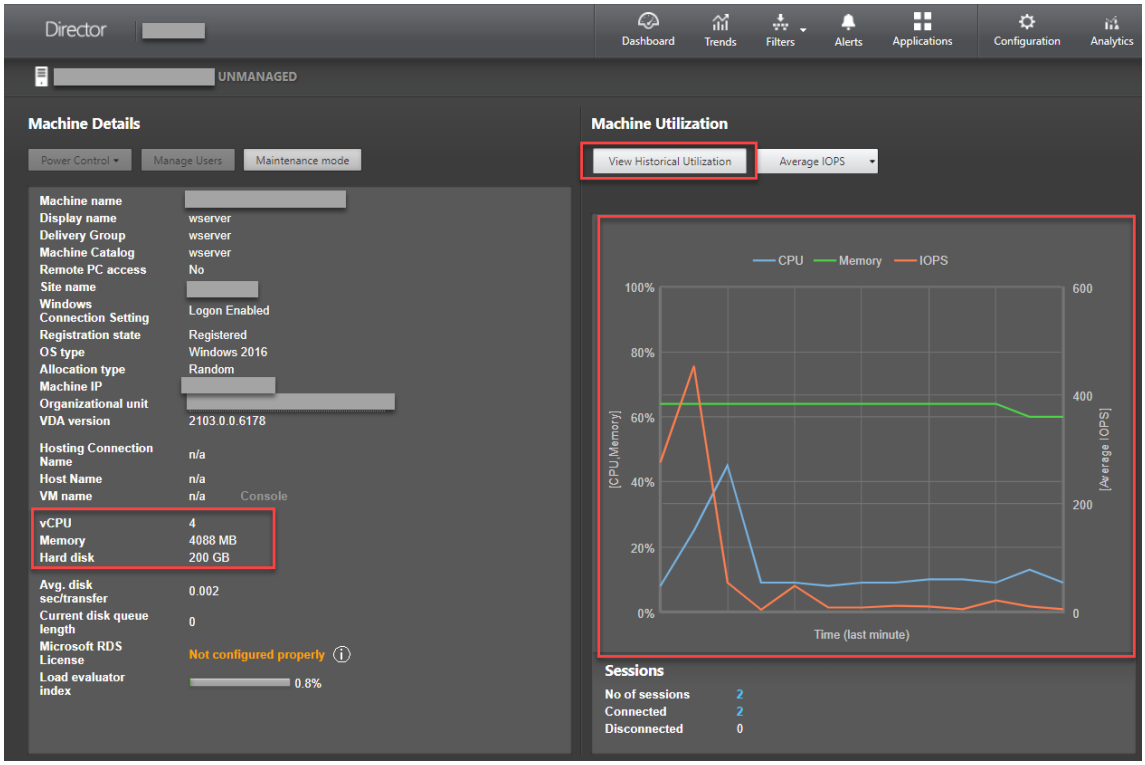
### **Linux VM** 的指标

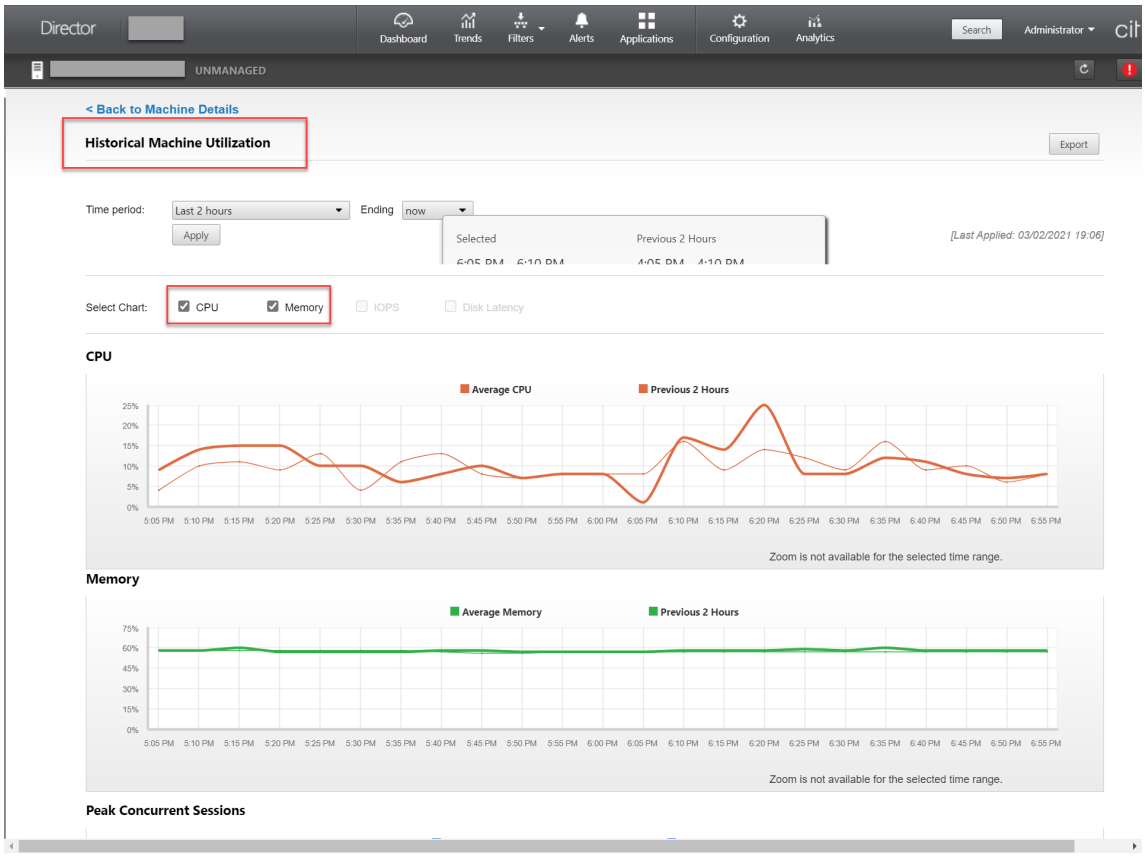
要访问 Linux VM 的指标，请在 Citrix Director 中找到该 VM，然后检查计算机详细信息面板。

自 Linux VDA 版本 2103 起，Citrix Director 中提供了以下 Linux VM 的指标。

- CPU 核心的数量
- 内存大小
- 硬盘容量
- 当前和历史 CPU 和内存利用率

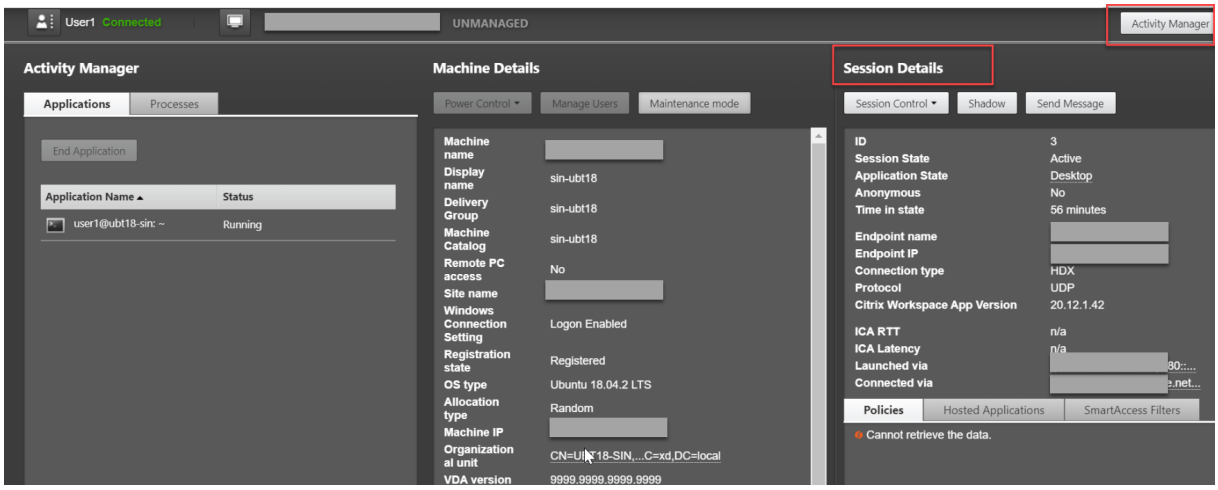


## Linux 会话的指标

要查看 Linux 会话的指标，请通过选择过滤器 > 会话 > 所有会话打开所有会话页面，或者访问会话详细信息面板。要访问会话详细信息面板，请打开所有会话页面，然后单击目标会话以访问其活动管理器视图。例如：




- ICA RTT

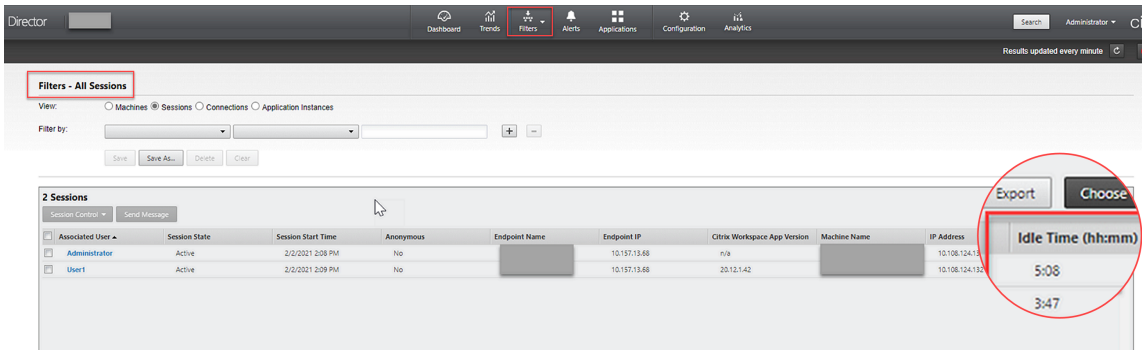
  自 Linux VDA 1903 版起，可以查看 ICA RTT 衡量指标。要查看 ICA RTT 衡量指标，请使用 Citrix Director 1903 或更高版本并在 Citrix Studio 中创建 **ICA** 往返行程计算和 **ICA** 往返行程计算时间间隔策略。有关创建策略的信息，请参阅使用 Studio 创建策略。

- 协议

  自 Linux VDA 1909 版起，可以查看协议信息。Linux 会话的传输协议将在会话详细信息控制板中显示为 **UDP** 或 **TCP**。

- 空闲时间

  自 Linux VDA 版本 2103 起，空闲时间指标可用于 Linux 会话。要访问此指标，请通过选择过滤器 > 会话 > 所有会话打开所有会话页面。



# 监视服务守护程序

November 8, 2021

监视服务守护程序通过执行定期扫描来监视关键服务。检测到异常时，该守护程序会重新启动或停止服务进程，并清除进程残留以释放资源。检测到的异常记录在 **/var/log/xdl/ms.log** 文件中。

## 配置

当您启动 VDA 时，监视服务守护程序将自动启动。

可以使用管理员权限通过 **scanningpolicy.conf**、**rulesets.conf** 和 **whitelist.conf** 文件配置该功能。配置文件位于 **/opt/Citrix/VDA/sbin** 下。

要使 **scanningpolicy.conf**、**rulesets.conf** 和 **whitelist.conf** 文件中的更改生效，请运行以下命令以重新启动监视服务守护程序。

```
service ctxmonitorservice restart
```

- **scanningpolicy.conf**

  此配置文件启用或禁用监视服务守护程序。它设置服务检测时间间隔，并指定是否修复检测到的异常。

  - MonitorEnable：true/false（默认值为 true）
  - DetectTime：20（单位：秒，默认值：20，最小值：5）
  - AutoRepair：true/false（默认值为 true）
  - MultBalance：false
  - ReportAlarm：false

- **rulesets.conf**

  此配置文件指定要监视的目标服务。默认情况下有四种受监视的服务，如下面的屏幕截图所示。

```
MonitorUser: all
MonitorType: 3
ProcessName: ctxhdx
Operation: 4
DBRecord: false

MonitorUser: all
MonitorType: 3
ProcessName: ctxvda
Operation: 4
DBRecord: false

MonitorUser: all
MonitorType: 3
ProcessName: ctxpolicyd
Operation: 4
DBRecord: false

MonitorUser: all
MonitorType: 3
ProcessName: Xorg
Operation: 8
DBRecord: false
```

要配置要监视的每个服务，请设置以下字段。

- MonitorUser：all
- MonitorType：3
- ProcessName: <>（进程名称不能留空，必须完全匹配。）
- Operation：1/2/4/8（1 = 检测到异常时停止服务。2 = 检测到异常时终止服务。4 = 重新启动服务。8 = 清理 Xorg 进程残留。）
- DBRecord：false

- **whitelist.conf**

  在 **rulesets.conf** 文件中指定的目标服务也必须在 **whitelist.conf** 文件中配置。白名单配置是安全性的辅助筛选器。

  要配置白名单，请在 **whitelist.conf** 文件中仅包含进程名称（必须完全匹配）。有关示例，请参阅下面的屏幕截图。

```
ctxcdmd
ctxcdmmount
ctxcdmstat
ctxceip
ctxclipboard
ctxconnect
ctxcredentialctl
ctxctl
ctxcupsd
ctxdisconnect
ctxeuem
ctxfiletransfer
ctxgfx
ctxhdx
ctxism
ctxlogd
ctxlogin
ctxmonitorservice
ctxmrvc
ctxpolicyd
ctxscardsd
ctxvhcid
ctxvda
Xorg
```

> 注意：
>
> 在停止 `ctxvda`、`ctxhdx` 和 `ctxpolicyd` 服务之前，请运行 `service ctxmonitorservice stop` 命令以停止监视服务守护程序。否则，监视服务守护程序将重新启动您停止的服务。

## 使用 **TLS** 保护用户会话安全

March 1, 2022

自版本 7.16 起，Linux VDA 支持用于保护用户会话安全的 TLS 加密。默认情况下，TLS 加密处于禁用状态。

### 启用 **TLS** 加密

要启用用于保护用户会话安全的 TLS 加密，请在 Linux VDA 和 Delivery Controller (Controller) 上获取证书并启用 TLS 加密。

#### 获取证书

从可信证书颁发机构 (CA) 获取 PEM 格式的服务器证书和 CRT 格式的根证书。服务器证书包含以下部分：

- 证书
- 未加密的私钥
- 中间证书（可选）

服务器证书示例：

```
-----BEGIN CERTIFICATE-----
MIIDTTCCAragAwIBAgIJALluncpiqGXCMA0GCSqGSIb3DQEBBQUAMGcxCzAJBgNV
BAYTAlVLMRIwEAYDVQQIEwlDYW1icmlkZ2UxEjAQBgNVBAcTCUNhbWJvdXJuZTEU
MBIGA1UEChMLQ2l0cml4IFRlc3QxGjAYBgNVBAMTEWNhMDAxLmNpdHJpdGUubmV0
MB4XDTA4MDkzMDEwNTk1M1oXDTI4MDkyNTEwNTk1M1owgYoxCzAJBgNVBAYTAlVL
MRIwEAYDVQQIEwlDYW1icmlkZ2UxEjAQBgNVBAcTCUNhbWJvdXJuZTEUMBIGA1UE
ChMLQ2l0cml4IFRlc3QxGzAZBgNVBAsTElNlcnZlciBDZXJ0aWZpY2F0ZTEgMB4G
A1UEAxMXY2EwMDEtc2MwMDEuY2l0cml0ZS5uZXQwgZ8wDQYJKoZIhvcNAQEBBQAD
gYOAMIGJAoGBALCTTOdxcivbIOLOF66xgO5gkNeIGKVP+37pSKV8B66lWCVzr6p9
t72Fa+9oCcf2x/ue274NXFcg4fqGRDsrEwl3YxM6COyBf7L6psrsCDNnBP1q8TJH
4xoPIXUeaW4MVk/3PVyfhHKs4fz8yy1I4VDnXVHhw+OFQ2Bq3NhwsRhnAgMBAAGj
gdwwgdkwCQYDVR0TBAIwADAdBgNVHQ4EFgQUrLidzYot+CUXSh9xMfp1M+/O8yOw
gZkGA1UdIwSBkTCBjoAU85kN1EPJOcVhcOss1slseDQwGsKha6RpMGcxCzAJBgNV
BAYTAlVLMRIwEAYDVQQIEwlDYW1icmlkZ2UxEjAQBgNVBAcTCUNhbWJvdXJuZTEU
MBIGA1UEChMLQ2l0cml4IFRlc3QxGjAYBgNVBAMTEWNhMDAxLmNpdHJpdGUubmV0
ggkAy8nC8dcB32EwEQYJYIZIAYb4QgEBBAQDAgVgMAOGCSqGSIb3DQEBBQUAA4GB
AD5axBYHwIxJCJzNt2zdXnbp2O0yUToWElBwQe/9cGaP6CpjoxJ7FJa2/8IpaT68
Vel8u1SEYY1GKCGCw93pc7sPKqb8pGBRI5/dygb+geFk1Q7KyVbuOIjOtr3pkxAe
b6CFJtNLudHUrwF6lOrB72zbyz3PiIx+HEwt1jOj8Z4K
-----END CERTIFICATE-----
-----BEGIN RSA PRIVATE KEY-----
MIICXgIBAAKBgQCwkOzncXIr2yNC9BeusYDuYJDXiBilT/t+6UilfAeupVglc6+q
fbe9hWvvaAnH9sf7ntu+DVxXIOH6hkQ7KxMNd2MTOgjsgX+y+qbK7AgzZwT9avEy
R+MaDyF1HmluDFZP9z1cn4RyrOH8/MstSOFQ511R4cPtBUNgatzYcLEYZwIDAQAB
AoGBAKwBgZu/bkl8edgB8YPyU7diiBX89IOs4b/aPjM+JDmjxb8N96RsPO24p9Ea
FtUc9+iL8mEroLUbSicCXjsJFc+cxg9vVaNa6EEkkBj735oCUERqSxOYb/lAdck/
FXzUOtqytUe/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
EjAQBgNVBAgTCUNhbWJyaWRnZTE5MBAGA1UEBxMJQ2FtYm9lcmSlMRQwEgYDVQQK
EwtDaXRyaXggVGVzdDEaMBgGA1UEAxMRY2EwMDEuY2l0cml0ZS5uZXQwgZ8wDQYJ
KoZIhvcNAQEBBQADgYOAMIGJAoGBAKVZmF7Uj7uOnvO3Qwdfi0nr3QkNH2DXpWrZ
Zh8cI9Vv+UFRUiC6oB7izLtBMFn3fOUP7i2CfkHN3ZGJ17p89pdyjket1MslVeJw
acOqrYvD+fNNSvJjunTbaCywVtALjmFSfMHeZJXVSckrpEhnk0nkMS16tcrya/K/
osSlzvI3AgMBAAGjgcwwgckwDAYDVR0TBAUwAwEB/zAdBgNVHQ4EFgQU85kN1EPJ
OcVhcOss1slseDQwGsIwgZkGA1UdIwSBkTCBjoAU85kN1EPJOcVhcOss1slseDQw
GsKha6RpMGcxCzAJBgNVBAYTAlVLMRIwEAYDVQQIEwlDYW1icmlkZ2UxEjAQBgNV
BAcTCUNhbWJvdXJuZTEUMBIGA1UEChMLQ2l0cml4IFRlc3QxGjAYBgNVBAMTEWNh
MDAxLmNpdHJpdGUubmV0ggkAy8nC8dcB32EwDQYJKoZIhvcNAQEFBQADgYEAIZ4Z
gXLLXf12RNqh/awtSbd41Ugv8BIKAsg5zhNAiTiXbzz8Cl3ec53Fb6nigMWc5Tli
UNCLXwnxRUiD4O0tESLX9ACUNH3I94yxOgujkSOSBni21jjZTvfBB32Rmr5DByJg
UmKORn/hdqMlcqpe5wO6as6+HN4wUOi+hEtUMME=
-----END CERTIFICATE-----
```

启用 **TLS** 加密

在 **Linux VDA** 上启用 **TLS** 加密　在 Linux VDA 上，使用 `enable_vdassl.sh` 工具启用（或禁用）TLS 加密。该工具位于 **/opt/Citrix/VDA/sbin** 目录中。有关工具中可用的选项的信息，请运行 `/opt/Citrix/VDA/sbin/enable_vdassl.sh -help` 命令。

提示：必须在每台 Linux VDA 服务器上安装服务器证书，以及必须在每台 Linux VDA 服务器和客户机上安装根证书。

在 **Controller** 上启用 **TLS** 加密

> 注意：
>
> 只能对整个交付组启用 TLS 加密。不能为特定应用程序启用 TLS 加密。

在 Controller 上的 PowerShell 窗口中，按顺序运行以下命令以对目标交付组启用 TLS 加密。

1. `Add-PSSnapin citrix.*`
2. `Get-BrokerAccessPolicyRule -DesktopGroupName 'GROUPNAME'| Set-BrokerAccessPolicyRule -HdxSslEnabled $true`

> 注意：
>
> 要确保 ICA 会话文件中仅包含 VDA FQDN，还可以运行 `Set-BrokerSite -DnsResolutionEnabled $true` 命令。该命令将启用 DNS 解析。如果禁用了 DNS 解析，ICA 会话文件会显示 VDA IP 地址，并仅为与 TLS 相关的项目（例如 SSLProxyHost 和 UDPDTLSPort）提供 FQDN。

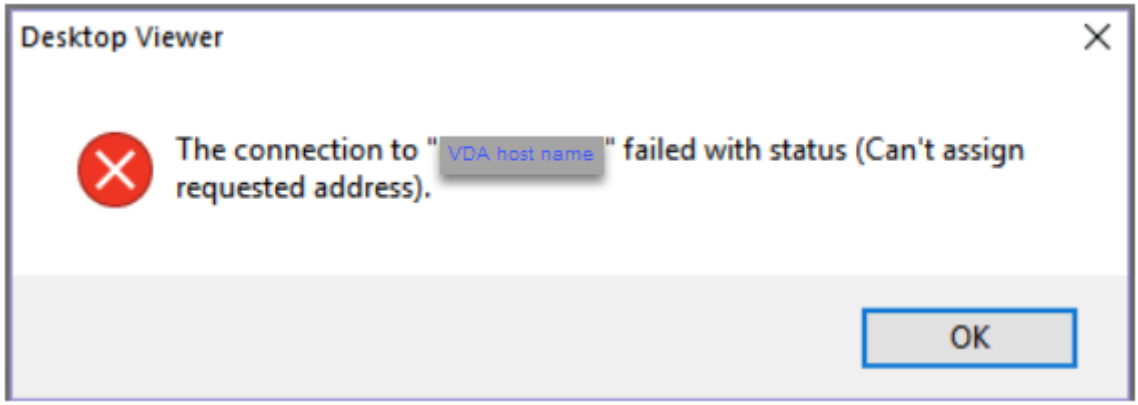
要在 Controller 上禁用 TLS 加密，请按顺序运行以下命令：

1. `Add-PSSnapin citrix.*`
2. `Get-BrokerAccessPolicyRule -DesktopGroupName 'GROUPNAME'| Set-BrokerAccessPolicyRule -HdxSslEnabled $false`
3. `Set-BrokerSite -DnsResolutionEnabled $false`

故障排除

尝试访问已发布的桌面会话时，适用于 Windows 的 Citrix Workspace 应用程序中可能会出现下面的“无法分配请求的地址”错误：




解决方法：向 **hosts** 文件添加一个条目，类似于：

```
<IP address of the Linux VDA>      <FQDN of the Linux VDA>
```

在 Windows 计算机上，**hosts** 文件通常位于 `C:\Windows\System32\drivers\etc\hosts` 中。

## 使用 DTLS 的安全用户会话

November 8, 2021

从 7.18 版本起，DTLS 加密是一项完全受支持的功能。默认情况下，此功能在 VDA 上处于启用状态。有关详细信息，请参阅传输层安全性。

### 启用 DTLS 加密

#### 验证自适应传输是否已启用

在 Citrix Studio 中，验证 **HDX** 自适应传输策略设置为首选还是诊断模式。

#### 在 Linux VDA 上启用 SSL 加密

在 Linux VDA 上，使用 `enable_vdassl.sh` 工具启用（或禁用）SSL 加密。该工具位于 **/opt/Citrix/VDA/sbin**。有关工具中可用的选项的信息，请运行 `/opt/Citrix/VDA/sbin/enable_vdassl.sh -h` 命令。

> 注意：
>
> 当前，Linux VDA 支持 DTLS 1.0 和 DTLS 1.2。DTLS 1.2 需要 Citrix Receiver for Windows 4.12 或适用于 Windows 的 Citrix Workspace 应用程序 1808 或更高版本。如果您的客户端仅支持 DTLS 1.0（例如，

> Citrix Receiver for Windows 4.11），请使用 `enable_vdassl.sh` 工具将 `SSLMinVersion` 设置 `TLS_1.0`，将 `SSLCipherSuite` 设置为 `COM` 或 `ALL`。

## 基于文本的会话水印

April 21, 2021

基于文本的会话水印有助于威慑和启用跟踪数据盗窃功能。这一可跟踪的信息在会话桌面上显示，对使用相机和屏幕截图盗取数据的数据盗窃行为具有威慑作用。可以指定一层文本水印，该水印将在整个会话屏幕上显示，但不会改变原始文档的内容。

> 重要：
>
> 基于文本的会话水印不属于安全功能。此解决方案不能完全阻止数据盗窃，但可以提供一定级别的威慑作用和可跟踪性。使用此功能时，我们不保证完整的信息可追溯性。但是，我们建议您将此功能与其他安全解决方案（如果适用）结合使用。

会话水印属于文本，不适用于向用户提供的会话。会话水印中包含用于跟踪数据盗窃的信息。最重要的数据是在其中创建了屏幕图像的当前会话的登录用户的身份。为了更有效地跟踪数据泄漏，请包括服务器或客户端 Internet 协议地址以及连接时间等其他信息。

要调整用户体验，请使用会话水印策略设置配置屏幕上的放置位置和水印外观。

### 限制

- 在使用浏览器内容重定向的会话中不支持会话水印。要使用会话水印功能，请务必禁用浏览器内容重定向。
- 如果会话在使用旧版 NVIDIA 驱动程序的全屏硬件加速 H.264 或 H.265 编码模式下运行（在本例中，NvCaptureType 在注册表中设置为 2），则不支持会话水印，并且不显示会话水印。
- 水印对会话重影功能不可见。
- 如果您按 Print Screen 键捕获屏幕，在 VDA 端捕获的屏幕将不包括水印。我们建议您采取措施避免复制屏幕捕获。

## 使用智能卡进行直通身份验证

March 11, 2022

登录 Linux Virtual Desktop 会话时，用户可以使用连接到客户端设备的智能卡进行身份验证。此功能是通过 ICA 智能卡虚拟通道进行的智能卡重定向实现的。用户还可以在会话中使用智能卡。用例包括向文档中添加数字签名、加密或解密电子邮件以及向要求智能卡身份验证的 Web 站点进行身份验证。

对于此功能，Linux VDA 使用与 Windows VDA 相同的配置。有关详细信息，请参阅本文中的配置智能卡环境部分。

使用智能卡进行直通身份验证的可用性取决于以下条件：

- Linux VDA 安装在 RHEL 7 或 RHEL 8 上。
- 使用 OpenSC 支持的智能卡。
- 使用 Citrix Workspace for Windows 的应用程序。

> 注意：
>
> 官方不支持在 Linux VDA 会话中使用映射的智能卡登录 Citrix Gateway。

### RHEL 8 支持智能卡

智能卡身份验证依赖于 pam_krb5 模块，该模块在 RHEL 8 上已弃用。要在 RHEL 8 上使用智能卡身份验证，请按如下所示构建 pam_krb5 模块：

1. 从 https://centos.pkgs.org/7/centos-x86_64/pam_krb5-2.4.8-6.el7.x86_64.rpm.html 下载 pam_krb5-2.4.8-6 源代码。
2. 在 RHEL 8 上构建并安装 pam_krb5 模块。

```
yum install -y opensc pcsc-lite pcsc-lite-libs pcsc-lite-ccid nss-tools
yum install gcc krb5-devel pam-devel autoconf libtool
rpm2cpio pam_krb5-2.4.8-6.el7.src.rpm | cpio -div
tar xvzf pam_krb5-2.4.8.tar.gz
cd pam_krb5-2.4.8
./configure --prefix=/usr
make
make install
```

3. 验证 pam_krb5.so 是否存在于 /usr/lib64/security/ 下。

```
ls -l /usr/lib64/security | grep pam_krb5
```

### 在 RHEL 7、RHEL 8 上安装 Linux VDA 软件

使用 RPM 软件包管理器或轻松安装功能安装 Linux VDA 软件，请参阅安装概述部分。

VDA 安装完成后，请验证 VDA 是否能够在 Delivery Controller 中注册，以及已发布的 Linux 桌面会话是否能够使用密码身份验证成功启动。

### 确保 OpenSC 支持智能卡

OpenSC 是 RHEL 7.4+ 上广泛使用的智能卡驱动程序。作为 CoolKey 的完全兼容替代品，OpenSC 支持多种类型的智能卡（请参阅 Red Hat Enterprise Linux 中的智能卡）。

在本文中，YubiKey 4 智能卡用作阐明配置的示例。YubiKey 4 是一个通用 USB CCID PIV 设备，可以很容易地从 Amazon 或其他零售供应商处购买。OpenSC 驱动程序支持 YubiKey 4。

如果贵组织需要某些其他更高级的智能卡，请准备一台安装了 RHEL 7 或 RHEL 8 以及 OpenSC 软件包的物理机。有关 OpenSC 安装的信息，请参阅安装智能卡驱动程序。插入智能卡，然后运行以下命令以确认 OpenSC 是否支持智能卡：

```
pkcs11-tool --module opensc-pkcs11.so --list-slots
```

## 配置

### 准备根证书

根证书用于验证智能卡上的证书。请执行以下操作以下载并安装根证书。

1. 获取 PEM 格式的根证书，通常从 CA 服务器中获取。

   可以运行类似如下的命令以将 DER 文件（*.crt、*.cer、*.der）转换为 PEM。在下面的命令示例中，**certnew.cer** 为 DER 文件。

   ```
   openssl x509 -inform der -in certnew.cer -out certnew.pem
   ```

2. 在 `openssl` 目录中安装根证书。**certnew.pem** 文件用作示例。

   ```
   cp certnew.pem <path where you install the root certificate>
   ```

   要创建安装根证书的路径，请运行 `sudo mdkir -p <path where you install the root certificate>`。

配置智能卡环境

可以使用 ctxsmartlogon.sh 脚本配置智能卡环境或手动执行配置。

- 使用 ctxsmartlogon.sh 脚本配置智能卡环境

> 注意：
>
> ctxsmartlogon.sh 脚本将 PKINIT 信息添加到默认领域。可以通过 **/etc/krb5.conf** 配置文件更改此设置。

首次使用智能卡之前，请运行 ctxsmartlogon.sh 脚本以配置智能卡环境。

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

结果类似于以下内容：

```
#*****************************************************************************
# 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.
```

要禁用智能卡，请执行以下操作：

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

结果类似于以下内容：

```
#*****************************************************************************
# 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.
```

- 手动配置智能卡环境

  Linux VDA 使用与 Windows VDA 相同的智能卡环境。在该环境中，必须配置多个组件，包括域控制器、Microsoft 证书颁发机构 (CA)、Internet Information Services、Citrix StoreFront 和 Citrix Workspace 应用程序。有关基于 YubiKey 4 智能卡的配置信息，请参阅知识中心文章 CTX206156。

  继续执行下个步骤之前，请确保所有组件都已正确配置，私钥和用户证书都已下载到智能卡，并且您能够使用智能卡成功登录 Windows VDA。

#### 安装 **PC/SC Lite** 软件包

PCSC Lite 是个人计算机/智能卡 (PC/SC) 规范在 Linux 中的实现。它提供与智能卡和读卡器进行通信的 Windows 智能卡接口。Linux VDA 中的智能卡重定向是在 PC/SC 级别实现的。

运行以下命令可安装 PC/SC Lite 软件包。

```
yum install pcsc-lite pcsc-lite-ccid pcsc-lite-libs
```

#### 安装智能卡读卡器

OpenSC 是 RHEL 上广泛使用的智能卡驱动程序。如果未安装 OpenSC，请运行以下命令以进行安装。

```
yum install opensc
```

#### 安装用于智能卡身份验证的 **PAM** 模块

运行以下命令可安装 pam_krb5 和 krb5-pkinit 模块。

RHEL 7:

```
yum install pam_krb5 krb5-pkinit
```

RHEL 8:

```
yum install krb5-pkinit
```

pam_krb5 模块是可插入身份验证模块，可识别 PAM 的应用程序可以使用它来检查密码以及从密钥发行中心 (KDC) 获取票据授予票据。krb5-pkinit 模块包含允许客户端使用私钥和证书从 KDC 获取初始凭据的 PKINIT 插件。

#### 配置 **pam_krb5** 模块

pam_krb5 模块与 KDC 交互以使用智能卡中的证书获取 Kerberos 票据。要在 PAM 中启用 pam_krb5 身份验证，请运行以下命令：

```
authconfig --enablekrb5 --update
```

在 **/etc/krb5.conf** 配置文件中，根据实际的领域添加 PKINIT 信息。

> 注意：
>
> **pkinit_cert_match** 选项指定客户端证书在用于尝试 PKINIT 身份验证之前必须匹配的规则。匹配规则的语法为：
>
> *[relation-operator] component-rule …*
>
> 其中 `relation-operator` 可以为 &&，表示所有组件规则必须匹配，也可以为 ||，表示只有一条组件规则必须匹配。

西面是一个通用 krb5.conf 文件的示例：

```
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

  }
```

添加 PKINIT 信息后，该配置文件将如下所示。

```
CTXDEV.LOCAL = {
    kdc = ctx-ad.ctxdev.local
    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
}
```

配置 **PAM** 身份验证

PAM 配置文件指示用于 PAM 身份验证的模块。要添加 pam_krb5 作为身份验证模块，请向 **/etc/pam.d/smartcard-auth** 文件中添加以下行：

```
auth [success=done ignore=ignore default=die] pam_krb5.so preauth_options
=X509_user_identity=PKCS11:/usr/lib64/pkcs11/opensc-pkcs11.so
```

如果使用 Winbind，则修改后该配置文件将如下所示。

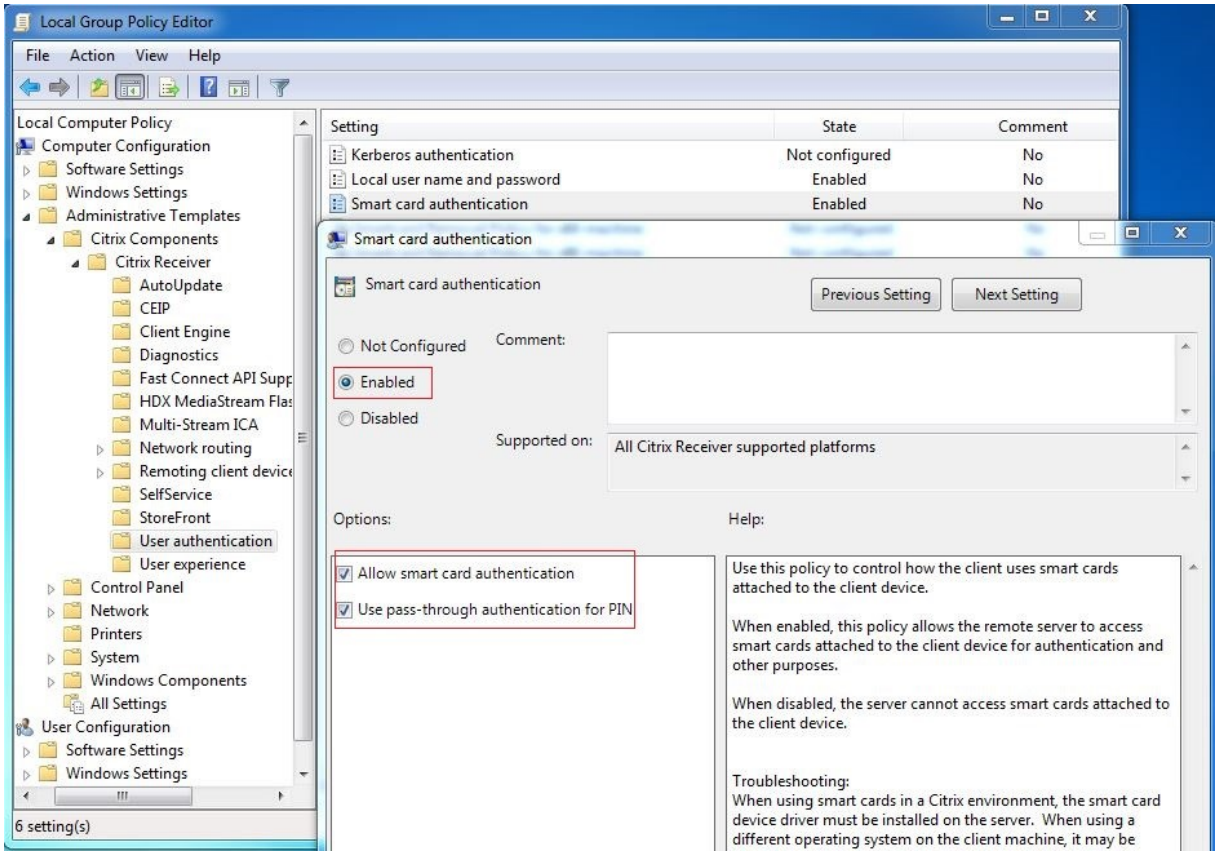
```
#%PAM-1.0
# This file is auto-generated.
# User changes will be destroyed the next time authconfig is run.
auth        required      pam_env.so
auth        [success=done ignore=ignore default=die]  pam_krb5.so preauth_options=X509_user_identity=PKCS11:/usr/lib64/pkcs11/libcoolkeypk11.so
auth        sufficient    pam_permit.so
auth        required      pam_deny.so

account     required      pam_unix.so broken_shadow
account     sufficient    pam_localuser.so
account     sufficient    pam_succeed_if.so uid < 1000 quiet
account     [default=bad success=ok user_unknown=ignore] pam_winbind.so
account     [default=bad success=ok auth_err=ignore user_unknown=ignore ignore=ignore]  pam_krb5.so
account     required      pam_permit.so

password    required      pam_pkcs11.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_winbind.so
session     optional      pam_krb5.so
```

（可选）使用智能卡进行单点登录

单点登录 (SSO) 是一项 Citrix 功能，用于实现对虚拟桌面和应用程序启动的直通身份验证。此功能降低了用户键入其 PIN 码的次数。要将 SSO 与 Linux VDA 结合使用，请配置 Citrix Workspace 应用程序。该配置与 Windows VDA 相同。有关详细信息，请参阅知识中心文章 CTX133982。

在 Citrix Workspace 应用程序中配置组策略时，请按如下所示启用智能卡身份验证。




快速智能卡登录

快速智能卡是对现有基于 HDX PC/SC 的智能卡重定向的改进。在高延迟 WAN 环境中使用智能卡时，可以提高性能。有关详细信息，请参阅智能卡。

Linux VDA 支持在以下版本的 Citrix Workspace 应用程序中使用快速智能卡：

- Citrix Receiver for Windows 4.12
- 适用于 Windows 的 Citrix Workspace 应用程序 1808 及更高版本

在客户端上启用快速智能卡登录　默认情况下，快速智能卡登录功能在 VDA 上处于启用状态，在客户端上处于禁用状态。在客户端，要启用快速智能卡登录，请将以下参数包含在关联 StoreFront 站点的 default.ica 文件中：

```
[WFClient]
SmartCardCryptographicRedirection=On
```

在客户端上禁用快速智能卡登录　要在客户端上禁用快速智能卡登录，请从关联的 StoreFront 站点的 default.ica 文件中删除 **SmartCardCryptographicRedirection** 参数。

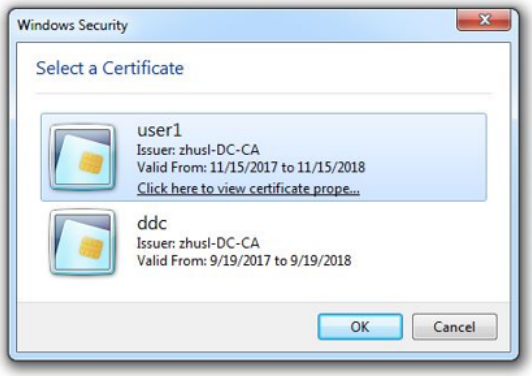
 

## 使用情况

### 使用智能卡登录 **Linux VDA**

用户可以在 SSO 和非 SSO 场景中使用智能卡登录 Linux VDA。

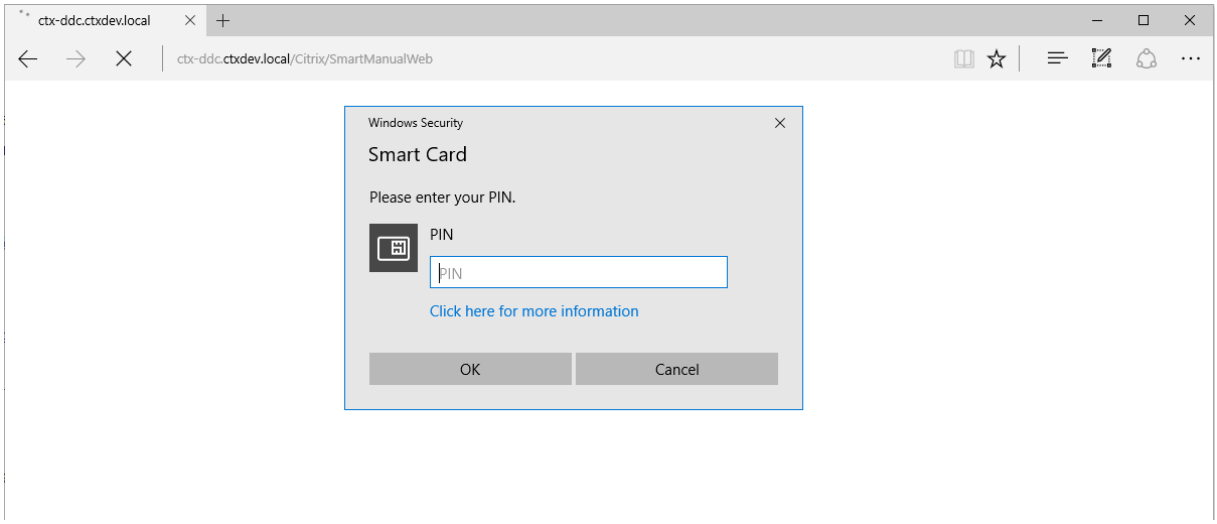
- 在 SSO 场景中，用户将使用缓存的智能卡证书和 PIN 码自动登录 StoreFront。用户在 StoreFront 中启动 Linux Virtual Desktop 会话时，PIN 码将传递到 Linux VDA 以进行智能卡身份验证。
- 在非 SSO 场景中，系统将提示用户选择证书并键入 PIN 码以登录 StoreFront。





用户在 StoreFront 中启动 Linux Virtual Desktop 会话时，将显示 Linux VDA 登录对话框，如下所示。用户名是从智能卡中的证书中提取的，用户必须重新键入 PIN 码以进行登录身份验证。

此行为与 Windows VDA 相同。




### 使用智能卡重新连接到会话

请务必将智能卡连接到客户端设备，才能重新连接到会话。否则，Linux VDA 端将显示一个灰色缓存窗口，并快速退出，因为重新身份验证在未连接智能卡的情况下失败。在这种情况下，系统不提供任何其他提示以提醒您连接智能卡。

但是，在 StoreFront 端，如果尝试重新连接到会话时未连接智能卡，StoreFront Web 可能会按如下所示提供警报。


限制

智能卡移除策略

现在，Linux VDA 对智能卡移除仅使用默认行为。成功登录 Linux VDA 后移除智能卡时，会话仍保持连接，并且会话屏幕不锁定。

支持其他智能卡和 **PKCS#11** 库

虽然我们的支持列表中仅列出了 OpenSC 智能卡，但是您可以尝试使用其他智能卡和 PKCS#11 库，因为 Citrix 会提供通用智能卡重定向解决方案。要切换到特定的智能卡或 PKCS#11 库，请执行以下操作：

1. 使用 PKCS#11 库替换所有 `opensc-pkcs11.so` 实例。
2. 要将 PKCS#11 库的路径设置为注册表，请运行以下命令：

   ```
   /opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\
       CurrentControlSet\Control\Citrix\VirtualChannels\Scard" -v "
       PKCS11LibPath" -d "PATH"
   ```

   其中 **PATH** 指向您的 PKCS#11 库，例如 /usr/lib64/pkcs11/opensc-pkcs11.so
3. 在客户端上禁用快速智能卡登录。

## 双跃点单点登录身份验证

November 8, 2021

此功能可将访问 StoreFront 应用商店需要输入的用户凭据注入到适用于 Linux 的 Citrix Workspace 应用程序和 Citrix Receiver for Linux 13.10 的 AuthManager 模块。注入后，您可以使用该客户端访问 Linux 虚拟桌面会话中的虚拟桌面和应用程序，而无需再次输入用户凭据。

> 注意：
>
> 适用于 Linux 的 Citrix Workspace 应用程序和 Citrix Receiver for Linux 13.10 支持此功能。

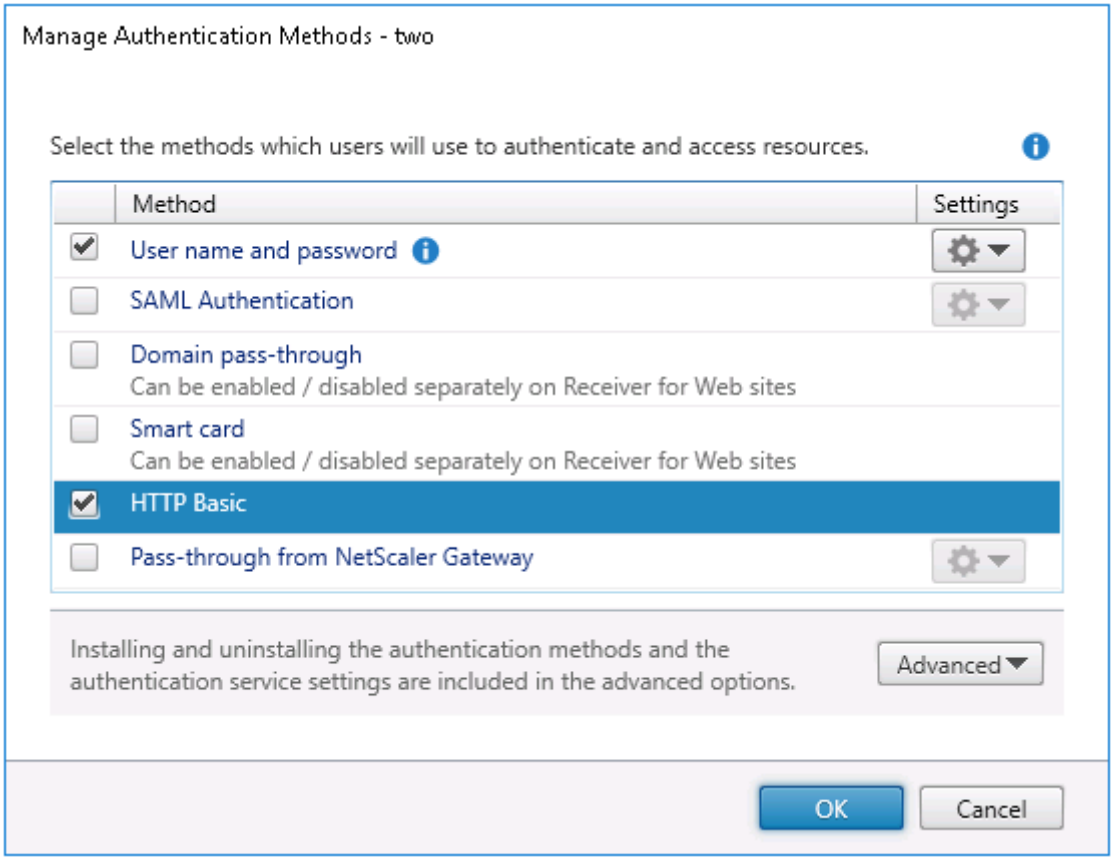
要启用此功能，请执行以下操作：

1. 在 Linux VDA 上，安装适用于 Linux 的 Citrix Workspace 应用程序或 Citrix Receiver for Linux 13.10。

   从 Citrix Workspace 应用程序或 Citrix Receiver 的 Citrix 下载页面下载相应应用程序。

   默认安装路径为 /opt/Citrix/ICAClient/。如果要将应用程序安装在其他路径中，请将 ICAROOT 环境变量设置为指向实际安装路径。

2. 在 Citrix StoreFront 管理控制台中，为目标应用商店添加 **HTTP** 基本身份验证方法。




3. 将以下注册表项添加到 AuthManager 配置文件 ($ICAROOT/config/AuthManConfig.xml) 以允许进行 HTTP 基本身份验证：

```
<Protocols>
     <HTTPBasic>
         <Enabled>True</Enabled>
    </HTTPBasic>
</Protocols>
```

4. 请运行以下命令以在指定目录中安装根证书。

```
cp rootcert.pem $ICAROOT/keystore/cacerts/
$ICAROOT/util/ctx_rehash  $ICAROOT/keystore/cacerts/
```

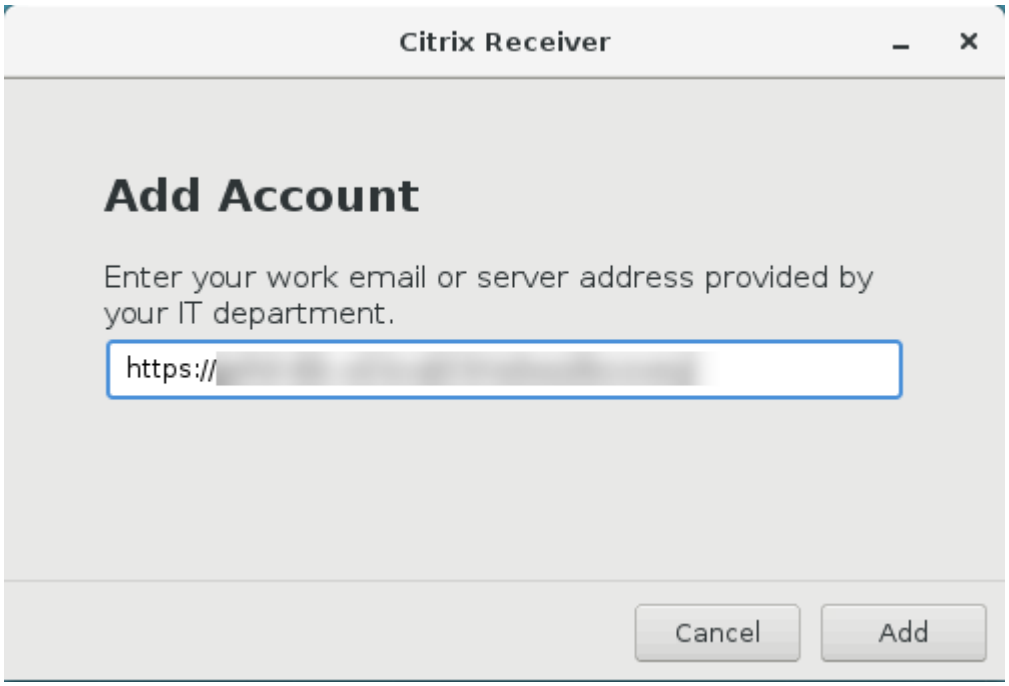
5. 运行以下命令以启用该功能：

```
/opt/Citrix/VDA/bin/ctxreg update -k "HKLM\System\
    CurrentControlSet\Control\Citrix" -v "LurSsonEnabled" -d "0
    x00000001"
```

6. 启动 Linux 虚拟桌面会话和该会话中的适用于 Linux 的 Citrix Workspace 应用程序或 Citrix Receiver for Linux 13.10。

   首次启动 Linux 虚拟桌面会话中的适用于 Linux 的 Citrix Workspace 应用程序或 Citrix Receiver for Linux 13.10 时，系统会提示您登录应用商店帐户。以后您可以自动登录到之前指定的应用商店。

   > 注意：
   >
   > 输入 HTTPS URL 作为您的应用商店帐户。




## 配置未经身份验证的会话

April 18, 2024

按照本文信息配置未经身份验证的会话。安装 Linux VDA 以使用此功能时无需特殊设置。

> 注意：
>
> 配置未经身份验证的会话时，请考虑会话预启动并不受支持。会话预启动在适用于 Android 的 Citrix Workspace 应用程序上也不受支持。

### 创建未经身份验证的应用商店

要在 Linux VDA 上支持未经身份验证的会话，请使用 StoreFront 创建未经身份验证的应用商店。

### 在交付组中启用未经身份验证的用户

在创建未经身份验证的应用商店后，在交付组中启用未经身份验证的用户以支持未经身份验证的会话。要在交付组中启用未经身份验证的用户，请按照 Citrix Virtual Apps and Desktops 文档中的说明进行操作。

### 设置未经身份验证的会话空闲时间

未经身份验证的会话的默认空闲超时时间是 10 分钟。此值是通过注册表设置 **AnonymousUserIdleTime** 进行配置。可以使用 **ctxreg** 工具更改此值。例如，将此注册表设置设为 5 分钟：

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

### 设置未经身份验证的用户的最大数量

要设置未经身份验证的用户的最大数量，请使用注册表项 **MaxAnonymousUserNumber**。此设置限制单个 Linux VDA 上同时运行的未经身份验证的会话数。可以使用 **ctxreg** 工具配置此注册表设置。例如，将该值设置为 32：

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

> 重要
>
> :
>
> 限制未经身份验证的会话数。如果同时启动太多会话，VDA 可能会出现问题，其中包括耗尽可用内存。

### 故障排除

配置未经身份验证的会话时，请考虑以下事项：

- 无法登录到未经身份验证的会话。

确认注册表是否已更新包含了以下内容（设置为 0）：

```
sudo /opt/Citrix/VDA/bin/ctxreg read -k "HKLM\System\CurrentControlSet\Control\Citrix" -v MaxAnonymousUserNumber
```

确认 **ncsd** 服务是否正在运行，且已配置为启用 **passwd** 缓存：

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

如果已启用，请将 **Passwd** 缓存变量设置为 **no** ，然后重新启动 **ncsd** 服务。更改此配置后您可能需要重新安装 Linux VDA。

- 使用 **KDE** 时未经身份验证的会话中显示锁屏按钮。

默认情况下未经身份验证的会话中禁用锁屏按钮和菜单。但是，它们仍可显示在 KDE 中。在 KDE 中，要对特定用户禁用锁屏按钮和菜单，请将以下行添加到配置文件 **$Home/.kde/share/config/kdeglobals** 中。例如：

```
[KDE Action Restrictions]
action/lock_screen=false
```

但是，如果在全局范围的 `kdeglobals` 文件（如 `/usr/share/kde-settings/kde-profile/default/share/config/kdeglobals`）中将 `KDE Action Restrictions` 参数配置为不可变，用户配置将不起作用。

要解决此问题，请修改系统范围的 `kdeglobals` 文件以删除 `[KDE Action Restrictions]` 部分中的 `[$i]` 标记，或直接使用系统范围的配置来禁用锁屏按钮和菜单。有关 KDE 配置的详细信息，请参阅 KDE System Administration/Kiosk/Keys 页面。

## 配置 LDAPS

November 8, 2021

安全 LDAP (LDAPS) 允许您为 Active Directory 管理的域启用安全轻型目录访问协议以提供通过 SSL（Secure Socket Layer，安全套接字层）/TLS（Transport Layer Security，传输层安全性）进行通信的功能。

默认不加密客户端与服务器应用程序之间的 LDAP 通信。通过使用 SSL/TLS 的 LDAP (LDAPS)，可以保护 Linux VDA 与 LDAP 服务器之间的 LDAP 查询内容。

以下 Linux VDA 组件在 LDAPS 上都具有依赖项：

- Broker 代理：Linux VDA 注册到 Delivery Controller 中
- 策略服务：策略评估

配置 LDAPS 涉及以下过程：

- 在 Active Directory (AD)/LDAP 服务器上启用 LDAPS
- 导出根 CA 以供客户端使用
- 在 Linux VDA 上启用/禁用 LDAPS
- 为第三方平台配置 LDAPS
- 配置 SSSD
- 配置 Winbind
- 配置 Centrify
- 配置 Quest

## 在 **AD/LDAP** 服务器上启用 **LDAPS**

可以通过安装 Microsoft 证书颁发机构 (CA) 或非 Microsoft CA 提供的格式正确的证书来启用通过 SSL 的 LDAP (LDAPS)。

> 提示：
>
> 在域控制器上安装企业根 CA 时将自动启用通过 SSL/TLS 的 LDAP (LDAPS)。

有关如何安装证书并验证 LDAPS 连接的详细信息，请参阅 Microsoft 支持站点上的 How to enable LDAP over SSL with a third-party certification authority（如何借助第三方证书颁发机构启用通过 SSL 的 LDAP）。

当您有一个多层（例如，两层或三层）证书颁发机构层次结构时，您在域控制器上不会自动拥有用于 LDAPS 身份验证的合适证书。

有关如何使用多层证书颁发机构层次结构为域控制器启用 LDAPS 的信息，请参阅 Microsoft TechNet 站点上的 LDAP over SSL (LDAPS) Certificate（通过 SSL 的 LDAP (LDAPS) 证书）一文。

### 启用根证书颁发机构以供客户端使用

客户端必须使用 LDAP 服务器信任的 CA 颁发的证书。要为客户端启用 LDAPS 身份验证，请将根 CA 证书导入到受信任的密钥库。

有关如何导出根 CA 的详细信息，请参阅 Microsoft 支持 Web 站点上的 How to export Root Certification Authority Certificate（如何导出根证书颁发机构）。

## 在 **Linux VDA** 上启用或禁用 **LDAPS**

要在 Linux VDA 上启用或禁用 LDAPS，请运行以下脚本（在以管理员身份登录时）：

此命令的语法包括以下内容：

- 通过提供的根 CA 证书启用通过 SSL/TLS 的 LDAP：

```
/opt/Citrix/VDA/sbin/enable_ldaps.sh -Enable pathToRootCA
```

- 利用通道绑定通过 SSL/TLS 启用 LDAP：

```
/opt/Citrix/VDA/sbin/enable_ldaps.sh -Enablecb pathToRootCA
```

> 注意：
>
> 用于通道绑定的根 CA 证书必须采用 PEM 格式。启用通道绑定之前，请务必创建一个 Python3 虚拟环境。有关详细信息，请参阅创建 Python3 虚拟环境。

- 回退到未启用 SSL/TLS 的 LDAP

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

专用于 LDAPS 的 Java 密钥库位于 **/etc/xdl/.keystore**。受影响的注册表项包括：

```
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
```

### 为第三方平台配置 **LDAPS**

除 Linux VDA 组件外，还有多个附着于可能也需要安全 LDAP 的 VDA 的第三方软件组件，例如 SSSD、Winbind、Centrify 和 Quest。以下各部分介绍了如何通过 LDAPS、STARTTLS 或 SASL 签名和封装配置安全 LDAP。

> 提示：
>
> 并非所有这些软件组件都优先使用 SSL 端口 636 来确保安全 LDAP。并且大多数时间 LDAPS（端口 636 上通过 SSL 的 LDAP）不能与端口 389 上的 STARTTLS 共存。

#### SSSD

根据选项在端口 636 或端口 389 上配置 SSSD 安全 LDAP 流量。有关详细信息，请参阅 SSSD LDAP Linux 手册页。

#### Winbind

Winbind LDAP 查询使用 ADS 方法。Winbind 在端口 389 上仅支持 StartTLS 方法。受影响的配置文件为 **/etc/openldap/ldap.conf** 下的 **ldap.conf** 和 **/etc/samba/smb.conf** 下的 **smb.conf**。按如下所示更改这些文件：

```
ldap.conf:

TLS_REQCERT never

smb.conf:

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

或者，安全 LDAP 可以通过 SASL GSSAPI 签名和封装进行配置，但不能与 TLS/SSL 共存。要使用 SASL 加密，请更改 **smb.conf** 配置：

```
smb.conf:

ldap ssl = off
ldap ssl ads = no
client ldap sasl wrapping = seal
```

### Centrify

Centrify 在端口 636 上不支持 LDAPS。但它在端口 389 上提供安全加密。有关详细信息，请参阅 Centrify 站点。

### Quest

Quest Authentication Service 在端口 636 上支持 LDAPS，但在端口 389 上使用其他方法提供安全加密。

#### 故障排除

使用此功能时可能会引发以下问题：

- **LDAPS** 服务可用性

  请确认 LDAPS 连接是否在 AD/LDAP 服务器上可用。默认情况下，端口为 636。

- 启用了 **LDAPS** 时 **Linux VDA** 注册失败

  验证 LDAP 服务器和端口是否已正确配置。请先检查根 CA 证书，确保其与 AD/LDAP 服务器匹配。

- 意外错误地更改注册表项

  如果在未使用 **enable_ldaps.sh** 的情况下意外更新 LDAPS 相关的注册表项，则可能会破坏 LDAPS 组件的依赖项。

- **LDAP** 流量不通过 **SSL/TLS** 从 **Wireshark** 或任何其他网络监视工具加密

  默认禁用 LDAPS。请运行 **/opt/Citrix/VDA/sbin/enable_ldaps.sh** 强制执行。

- 没有来自 **Wireshark** 或任何其他网络连接监视工具的 **LDAPS** 流量

  发生 Linux VDA 注册和组策略评估时会出现 LDAP/LDAPS 流量。

- 无法通过在 **AD** 服务器上运行 **ldp connect** 验证 **LDAPS** 可用性

  使用 AD FQDN 而非 IP 地址。

- 无法通过运行 **/opt/Citrix/VDA/sbin/enable_ldaps.sh** 脚本导入根 **CA** 证书

  请提供 CA 证书的完整路径，并确认根 CA 证书的类型是否正确。它应与受支持的大多数 Java Keytool 类型兼容。如果它未在支持列表中列出，您可以先转换类型。如果遇到证书格式问题，我们建议您使用 base64 编码的 PEM 格式。

- 无法通过 **Keytool -list** 显示根 **CA** 证书

  通过运行 `/opt/Citrix/VDA/sbin/enable_ldaps.sh` 启用 LDAPS 时，证书将被导入到 /etc/xdl/.keystore 中，并设置密码以保护密钥库。如果忘记了密码，可以重新运行该脚本以创建密钥库。

## 创建 **Python3** 虚拟环境

May 31, 2022

本文详细介绍了创建没有网络连接的 Python3 虚拟环境的必备条件和步骤。或者，如果您要连接到网络，请运行 `ctxsetup.sh` 以创建 Python3 虚拟环境。

### 必备条件

- 必须具有管理权限才能访问 `/opt/Citrix/VDA/sbin/ctxpython3` 目录。
- `Python3` 软件包的滚轮文件已准备就绪。可以从 https://pypi.org/ 下载滚轮文件。

### 创建 **Python3** 虚拟环境

请完成以下步骤以创建 Python3 虚拟环境：

1. 安装 Python3 依赖项。

   对于 **RHEL**：

   ```
   yum -y install python36-devel krb5-devel gcc
   ```

> 注意：
>
> 您可能必须启用特定存储库才能安装某些依赖项。对于 RHEL 7，请运行 `subscription-manager repos --enable rhel-7-server-optional-rpms` 命令。对于 RHEL 8，请运行 `subscription-manager repos --enable=rhel-8-for-x86_64-appstream-rpms` 命令。

对于 **Ubuntu**、**Debian**：

```
apt-get -y install python3-dev python3-pip python3-venv libkrb5-dev
```

对于 **SUSE**：

```
zypper -i -n install python3-devel python3-setuptools krb5-devel gcc libffi48-devel
```

> 注意：
>
> 您可能必须启用 `SUSE_Linux_Enterprise_Software_Development_Kit_12_SP5_x86_64` 存储库才能安装某些依赖项。

2. 创建 Python3 虚拟环境。

   对于 **RHEL**、**Ubuntu**、**Debian**：

```
sudo python3 -m venv /opt/Citrix/VDA/sbin/ctxpython3
```

   对于 **SUSE**：

```
sudo python3 -m venv /opt/Citrix/VDA/sbin/ctxpython3

sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m pip install setuptools==40.6.2
```

3. 安装 LDAPS 依赖项。

   对于 **RHEL**、**Ubuntu**、**Debian**：

```
sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m pip install cffi == 1.14.2 cryptography == 3.1 decorator == 4.4.2 gssapi ==1.6.2 ldap3==2.8.1 netifaces == 0.10.9 pg8000 == 1.17.0 psutil == 5.8.0 pyasn1 == 0.4.8 pycparser == 2.20 scramp == 1.2.0 six == 1.15.0 termcolor == 1.1.0
```

   对于 **SUSE**：

```
sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m easy_install cffi == 1.14.2 cryptography == 3.1 decorator == 4.4.2 gssapi ==1.6.2 ldap3==2.8.1 netifaces == 0.10.9 pg8000 == 1.17.0
```

```
psutil == 5.8.0 pyasn1 == 0.4.8 pycparser == 2.20 scramp ==
1.2.0 six == 1.15.0 termcolor == 1.1.0
```

4. 安装 XDPing 依赖项。

   对于 **RHEL**、**Ubuntu**、**Debian**:

```
sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m pip install
    cffi == 1.14.2 cryptography == 3.1 decorator == 4.4.2 gssapi
    ==1.6.2 ldap3==2.8.1 netifaces == 0.10.9 pg8000 == 1.17.0
    psutil == 5.8.0 pyasn1 == 0.4.8 pycparser == 2.20 scramp ==
    1.2.0 six == 1.15.0 termcolor == 1.1.0

sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m pip install /
    opt/Citrix/VDA/sbin/ctxpython3/packages/xdping-*.whl
```

   对于 **SUSE**:

```
sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m easy_install
    cffi == 1.14.2 cryptography == 3.1 decorator == 4.4.2 gssapi
    ==1.6.2 ldap3==2.8.1 netifaces == 0.10.9 pg8000 == 1.17.0
    psutil == 5.8.0 pyasn1 == 0.4.8 pycparser == 2.20 scramp ==
    1.2.0 six == 1.15.0 termcolor == 1.1.0

sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m easy_install /
    opt/Citrix/VDA/sbin/ctxpython3/packages/xdping-*.whl
```

## XDPing

November 8, 2021

### 说明

Linux XDPing 工具是一个基于命令行的应用程序，可自动执行检查 Linux VDA 环境中常见配置问题的过程。

Linux XDPing 工具在系统中执行 150 多个单独的测试，这些测试大致分类如下：

- 检查是否满足 Linux VDA 系统要求
- 识别和显示计算机信息，包括 Linux 发行版
- 检查 Linux 内核的兼容性
- 检查是否存在任何可能影响 Linux VDA 操作的已知 Linux 发行版问题
- 检查安全增强型 Linux (SELinux) 模式和兼容性
- 识别网络接口并检查网络设置
- 检查存储分区和可用的磁盘空间
- 检查计算机主机和域名配置

- 检查 DNS 配置并执行查找测试
- 识别基本虚拟机管理程序并检查虚拟机配置。支持：
  - Citrix Hypervisor
  - Microsoft HyperV
  - VMware vSphere
- 检查时间设置并检查网络时间同步是否可以运行
- 检查 PostgreSQL 服务是否正确配置和运行
- 检查防火墙是否已启用且所需端口是否已打开
- 检查 Kerberos 配置并执行身份验证测试
- 检查 LDAP 搜索环境中的组策略服务引擎
- 检查 Active Directory 集成是否已正确设置，以及当前计算机是否已加入域。支持：
  - Samba Winbind
  - Dell Quest Authentication Services
  - Centrify DirectControl
  - SSSD
- 检查 Active Directory 中 Linux 计算机对象的完整性
- 检查可插拔身份验证模块 (PAM) 配置
- 检查核心转储模式
- 检查是否安装了 Linux VDA 所需的软件包
- 识别 Linux VDA 软件包并检查安装的完整性
- 检查 PostgreSQL 注册表数据库的完整性
- 检查 Linux VDA 服务是否已正确配置和运行
- 检查 VDA 和 HDX 配置的完整性
- 探测配置的每个 Delivery Controller，以测试 Broker Service 是否可访问、运行且响应迅速
- 检查计算机是否已在 Delivery Controller 场中注册
- 检查每个活动或断开连接的 HDX 会话的状态
- 扫描日志文件中是否存在与 Linux VDA 相关的错误和警告
- 检查 Xorg 的版本是否适合

### 使用 **Linux XDPing** 工具

> 注意：
>
> 在使用 XDPing 之前，请务必创建一个 Python3 虚拟环境。

XDPing 附带从命令 shell 运行的名为 `xdping` 的单个可执行文件。

要显示命令行选项，请使用 `--help` 选项：

```
sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m xdping --help
```

要运行全套测试，请在不使用任何命令行选项的情况下运行 `xdping`：

```
sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m xdping
```

要在安装 Linux VDA 软件包之前检查环境，请运行 `pre-flight` 测试：

```
sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m xdping --preflight
```

要仅运行特定的测试类别（例如，时间测试和 Kerberos 测试），请使用 `-T` 选项：

```
sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m xdping -T time,
    kerberos
```

要探测特定的 XenDesktop 控制器，请执行以下操作：

```
sudo /opt/Citrix/VDA/sbin/ctxpython3/bin/python3 -m xdping -d myddc.
    domain.net
```

示例输出

下面是运行 Kerberos 测试的示例输出：

```
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.

      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]
```

# 配置 Xauthority

November 8, 2021

Linux VDA 支持使用 X11 显示功能（包括 `xterm` 和 `gvim`）进行交互式远程处理的环境。此功能提供必需的安全机制以确保 XClient 与 XServer 之间的通信安全。

可以通过两种方法确保此安全通信的权限安全：

- **Xhost**。默认情况下，Xhost 仅允许本地主机 XClient 与 XServer 进行通信。如果选择允许远程 XClient 访问 XServer，则必须运行 Xhost 命令授予对特定计算机的权限。或者，也可以使用 **xhost +** 以允许任意 XClient 连接到 XServer。
- **Xauthority**。可以在每个用户的主目录中找到 `.Xauthority` 文件。它用于将凭据存储在 xauth 使用的 cookie 中以用于对 XServer 进行身份验证。启动 XServer 实例 (Xorg) 后，该 cookie 将用于对与该特定显示的连接进行身份验证。

## 工作原理

Xorg 启动时，`.Xauthority` 文件将被传递到 Xorg。此 `.Xauthority` 文件包含以下元素：

- 显示数量
- 远程请求协议
- cookie 数量

可以使用 `xauth` 命令浏览此文件。例如：

```
# xauth -f ~/.Xauthority

# > list

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

如果 XClient 远程连接到 Xorg，则必须满足两个必备条件：

- 设置远程 XServer 的 **DISPLAY** 环境变量。
- 获取包含 Xorg 中的其中一个 cookie 数量的 `.Xauthority` 文件。

## 配置 Xauthority

要在 Linux VDA 上启用 Xauthority 以进行远程 X11 显示，必须创建下面两个注册表项：

```
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
```

启用 Xauthority 后，手动或通过挂载共享主目录将 `.Xauthority` 文件传递到 XClient：

- 手动将 `.Xauthority` 文件传递给 XClient

  启动 ICA 会话后，Linux VDA 将为 XClient 生成 `.Xauthority` 文件，并将该文件存储在登录用户的主目录中。可以将此 `.Xauthority` 文件复制到远程 XClient 计算机，并设置 `DISPLAY` 和 `XAUTHORITY` 环境变量。`DISPLAY` 是存储在 `.Xauthority` 文件中的显示编号，`XAUTHORITY` 是 Xauthority 的文件路径。例如，请查看以下命令：

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


  export XAUTHORITY={
  the file path of .Xauthority }
  ```

  > 注意：
  >
  > 如果未设置 `XAUTHORITY` 环境变量，则默认使用 `~/.Xauthority` 文件。

- 通过挂载共享主目录将 `.Xauthority` 文件传递到 XClient

  最便捷的方式是为登录用户装载共享主目录。当 Linux VDA 启动 ICA 会话时，将在登录用户的主目录下创建 `.Xauthority` 文件。如果此主目录与 XClient 共享，用户不需要手动将此 `.Xauthority` 文件传输到 XClient。正确设置 `DISPLAY` 和 `XAUTHORITY` 环境变量后，将自动在 XServer 桌面中显示 GUI。

### 故障排除

如果 Xauthority 不起作用，请按照故障排除步骤进行操作：

1. 以具有 root 权限的管理员身份获取所有 Xorg cookie：

   ```
   ps aux | grep -i xorg
   ```

   此命令将显示启动过程中传递到 Xorg 的 Xorg 进程和参数。另一个参数显示使用的 `.Xauthority` 文件。例如：

   ```
   /var/xdl/xauth/.Xauthority110
   ```

   使用 **Xauth** 命令显示 cookie：

   ```
   Xauth -f /var/xdl/xauth/.Xauthority110
   ```

2. 使用 `Xauth` 命令显示 `~/.Xauthority` 中包含的 cookie。如果显示编号相同，则显示的 cookie 必须与 Xorg 和 XClient 的 `.Xauthority` 文件中的 cookie 相同。
3. 如果 cookie 相同，请检查是否能够使用 Linux VDA 的 IP 地址（例如 10.158.11.11）访问远程显示端口，并检查已发布桌面显示数量（例如 160）。

   在 XClient 计算机上运行以下命令：

   ```
   telnet 10.158.11.11 6160
   ```

   端口号为 6000 + \< 显示数量\> 的总和。

   如果此 Telnet 操作失败，防火墙可能在阻止请求。

## 配置联合身份验证服务

May 8, 2023

Linux VDA 支持使用 FAS 登录您的 Citrix Virtual Apps and Desktops 环境。它使用与 Windows VDA 相同的 Windows 环境来执行 FAS 登录功能。有关为 FAS 配置 Windows 环境的信息，请参阅联合身份验证服务。本文提供了特定于 Linux VDA 的额外信息。

> 注意
>
> Linux VDA 不支持会话中行为策略。
>
> Linux VDA 使用短连接与 FAS 服务器传输数据。

### 在 **Linux VDA** 上配置 **FAS**

#### **RHEL 8/CentOS 8** 对 **FAS** 的支持

FAS 依赖于 pam_krb5 模块，该模块在 RHEL 8/CentOS 8 上已弃用。要在 RHEL 8/CentOS 8 上使用 FAS，请按如下所示构建 pam_krb5 模块：

1. 从以下 Web 站点下载 pam_krb5-2.4.8-6 源代码。

   https://centos.pkgs.org/7/centos-x86_64/pam_krb5-2.4.8-6.el7.x86_64.rpm.html。

2. 在 RHEL 8/CentOS 8 上构建并安装 pam_krb5 模块。

   ```
   yum install make gcc krb5-devel pam-devel autoconf libtool
   rpm2cpio pam_krb5-2.4.8-6.el7.src.rpm | cpio -div
   tar xvzf pam_krb5-2.4.8.tar.gz
   cd pam_krb5-2.4.8
   ./configure --prefix=/usr
   ```

```
make
make install
```

3. 验证 pam_krb5.so 是否存在于 /usr/lib64/security/ 下。

```
ls -l /usr/lib64/security | grep pam_krb5
```

设置 **FAS** 服务器

如果是全新安装 Linux VDA，要使用 FAS，请在执行 ctxinstall.sh 或 ctxsetup.sh 过程中看到要求提供 CTX_XDL_FAS_LIST 时，键入每个 FAS 服务器的 FQDN。由于 Linux VDA 不支持 AD 组策略，您可以改为提供以分号分隔的 FAS 服务器的列表。如果删除了任何服务器地址，请使用 **<none>** 文本字符串填充其空白，并且不要修改服务器地址的顺序。

如果是升级现有的 Linux VDA 安装，可以重新运行 `ctxsetup.sh` 来设置 FAS 服务器。或者，也可以运行以下命令来设置 FAS 服务器，并重新启动 `ctxvda` 服务以使设置生效。

```
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

service ctxjproxy restart

service ctxvda restart
```

要通过 `ctxreg` 更新 FAS 服务器，请运行以下命令：

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

service ctxjproxy restart

service ctxvda restart
```

安装证书

要验证用户的证书，请在 VDA 上安装根 CA 证书以及所有中间证书。例如，要安装根 CA 证书，请从前面的从 **Microsoft CA**（在 **AD** 中）检索 **CA** 证书步骤中获取 AD 根证书，或者从根 CA 服务器 `http://CA-SERVER/certsrv` 下载 DER 格式的该证书。

> 注意：
>
> 以下命令也适用于配置中间证书。

运行类似如下的命令将 DER 文件（.crt、.cer、.der）转换为 PEM：

```
sudo openssl x509 -inform der -in root.cer -out root.pem
```

然后，运行类似如下的命令将根 CA 证书安装到 `openssl` 目录：

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

> 注意：
>
> 请勿将根 CA 证书置于 **/root** 路径下。否则，FAS 无权读取根 CA 证书。

#### 运行 **ctxfascfg.sh**

运行 ctxfascfg.sh 脚本以配置 FAS 参数：

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

添加两个环境变量，以便 `ctxfascfg.sh` 可以在静默模式下运行：

- **CTX_FAS_ADINTEGRATIONWAY=winbind | sssd | centrify** –表示 Active Directory 集成方法，相当于在指定 `CTX_EASYINSTALL_ADINTEGRATIONWAY` 时使用 `CTX_EASYINSTALL_ADINTEGRATIONWAY`。如果未指定`CTX_EASYINSTALL_ADINTEGRATIONWAY`，`CTX_FAS_ADINTEGRATIONWAY` 将使用自己的值设置。
- **CTX_FAS_CERT_PATH =<certificate path>** - 指定存储根证书和所有中间证书的完整路径。

选择正确的 Active Directory 集成方法，然后键入正确的证书路径（例如 `/etc/pki/CA/certs/`）。

脚本随后安装 krb5-pkinit 和 pam_krb5 软件包，并设置相关的配置文件。

#### 限制

- FAS 支持有限的 Linux 平台和 AD 集成方法。请参阅以下列表：

| | Winbind | SSSD | Centrify |
|---|---|---|---|
| RHEL 8.3 | 是 | 是 | 是 |
| RHEL 8.2 | 是 | 是 | 是 |
| RHEL 8.1 | 是 | 是 | 是 |
| RHEL 7.9 | 是 | 是 | 是 |
| RHEL 7.8 | 是 | 是 | 是 |
| Ubuntu 20.04 | 是 | 否 | 是 |
| Ubuntu 18.04 | 是 | 否 | 是 |

| | Winbind | SSSD | Centrify |
|---|---|---|---|
| Ubuntu 16.04 | 是 | 否 | 是 |
| SLES 12.5 | 是 | 否 | 是 |

- FAS 尚不支持锁屏界面。如果在会话中单击锁定按钮，则无法使用 FAS 重新登录该会话。
- 此版本仅支持联合身份验证服务体系结构概述一文中概括的常见 FAS 部署，不包括 **Windows 10 Azure AD** 联接。

## 故障排除

在对 FAS 进行故障排除之前，请确保已安装并正确配置 Linux VDA，以便可以使用密码身份验证在常用应用商店中成功启动非 FAS 会话。

如果非 FAS 会话正常运行，请将登录类别的 HDX 日志级别设置为“VERBOSE”，将 VDA 日志级别设置为“TRACE”。有关如何为 Linux VDA 启用跟踪日志记录的信息，请参阅知识中心文章 CTX220130。

### **FAS** 服务器配置错误

从 FAS 应用商店启动会话失败。

检查 **/var/log/xdl/hdx.log** 并查找类似如下内容的错误日志：

```
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

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
```

解决方案　运行以下命令以确认 Citrix 注册表值“HKEY_LOCAL_MACHINE\SOFTWARE\Citrix\VirtualDesktopAgent\Auther
是否设置为 <Your-FAS-Server-List>。

```
sudo /opt/Citrix/VDA/bin/ctxreg dump | grep "UserCredentialService"
```

如果现有设置不正确，请按照前面的设置 FAS 服务器步骤操作以重新设置。

**CA** 证书配置错误

从 FAS 应用商店启动会话失败。将显示一个灰色窗口，并且几秒钟后消失。


检查 **/var/log/xdl/hdx.log** 并查找类似如下内容的错误日志：

```
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

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
```

解决方案　验证是否在 /etc/krb5.CONFs 中正确设置了存储根 CA 证书和所有中间证书的完整路径。完整路径如下所示：

```
 [realms]

EXAMPLE.COM = {


    ......

    pkinit_anchors = DIR:/etc/pki/CA/certs/

    ......

 }
```

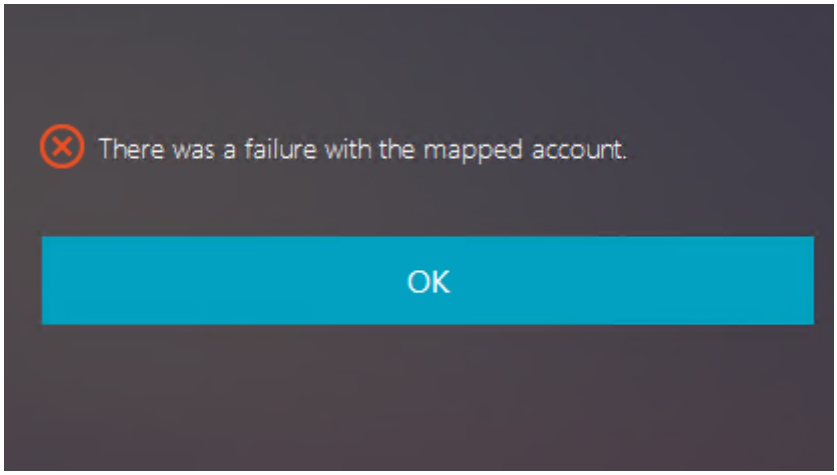
如果现有设置不正确，请按照前面的安装证书步骤重新对其进行设置。

或者，检查根 CA 证书是否有效。

重影帐户映射错误

FAS 配置了 SAML 身份验证。ADFS 用户在 ADFS 登录页面上输入用户名和密码后，可能会出现以下错误。

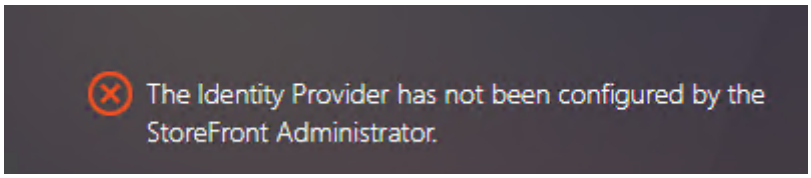



此错误表明已成功验证 ADFS 用户，但在 AD 上没有配置任何重影用户。

解决方案　在 AD 上设置重影帐户。

未配置 **ADFS**

尝试登录 FAS 应用商店过程中出现以下错误：




当 FAS 应用商店配置为使用 SAML 身份验证，但缺少 ADFS 部署时，会出现此问题。

解决方案　为联合身份验证服务部署 ADFS IdP。有关详细信息，请参阅联合身份验证服务 ADFS 部署。

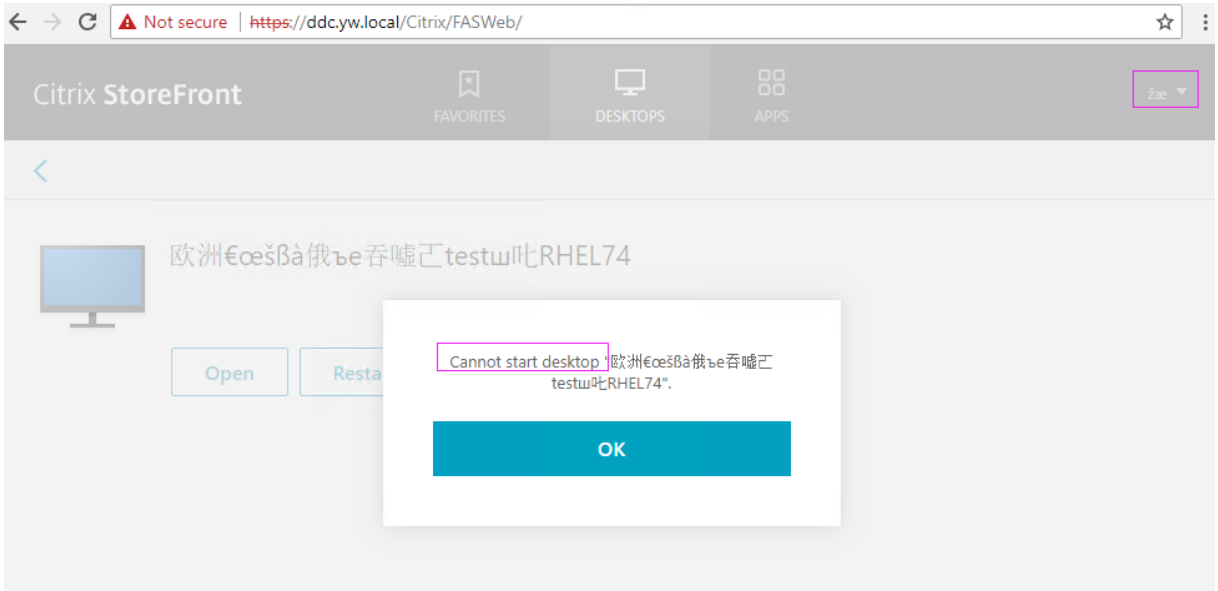
相关信息

- 通用 FAS 部署在联合身份验证服务体系结构概述一文中加以概括。
- 联合身份验证服务高级配置一章中引入了“操作方法”文章。

已知问题

如果正在使用 FAS，则尝试启动包含非英语字符的已发布桌面或应用程序会话时，可能会失败。




解决方法

在 CA 工具中右键单击 **Manage Templates**（管理模板）以将 **Citrix_SmartcardLogon** 模板从 **Build from this Active Directory information**（基于此 Active Directory 信息构建）更改为 **Supply in the request**（在请求中提供）：