

設定 Horizon 7 for Linux 桌面平台

上次修改日期 2017 年 9 月 15 日
VMware Horizon 7 7.2



vmware®

您可以在 VMware 網站上找到最新的技術文件，網址如下：

<https://docs.vmware.com/tw/>

如果您對於本文件有任何意見，歡迎寄至：

docfeedback@vmware.com

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Copyright © 2016, 2017 VMware, Inc. 保留所有權利。 [版權與商標資訊](#)。

目錄

設定 Horizon 7 for Linux 桌面平台 5

1 功能和系統需求 6

Horizon Linux 桌面平台的功能 6

Horizon 7 for Linux 桌面平台的組態步驟概觀 9

Horizon 7 for Linux 的系統需求 10

2D 和 vSGA 圖形卡的虛擬機器設定 15

2 為桌面平台部署準備 Linux 虛擬機器 18

建立虛擬機器和安裝 Linux 18

準備用於遠端桌面平台部署的 Linux 機器 19

安裝 Horizon Agent 相依性套件 21

3 設定 Linux 桌面平台的 Active Directory 整合 23

整合 Linux 與 Active Directory 23

設定 Single Sign-On 和智慧卡重新導向 24

4 設定 Linux 桌面平台的圖形卡 27

針對 vGPU 設定支援的 RHEL 發行版 27

在 ESXi 主機上安裝適用於 NVIDIA GRID vGPU 圖形卡的 VIB 27

在 Linux 虛擬機器上針對 vGPU 設定共用 PCI 裝置 29

安裝 NVIDIA GRID vGPU 顯示驅動程式 30

確認已安裝 NVIDIA 顯示驅動程式 32

針對 vDGA 設定 RHEL 6 33

在主機上針對 NVIDIA GRID 啟用 DirectPath I/O 33

新增 vDGA 傳遞裝置到 RHEL 6 虛擬機器 33

安裝 vDGA 的 NVIDIA 顯示驅動程式 34

確認已安裝 NVIDIA 顯示驅動程式 36

針對 vSGA 設定 RHEL 7 37

安裝 NVIDIA 圖形卡的 VIB 以在 ESXi 主機上實現 vSGA 功能 37

在 Linux 虛擬機器上設定 vSGA 的 3D 功能 38

確認 vSGA 在 Linux 虛擬機器上執行 39

5 安裝 Horizon Agent 40

在 Linux 虛擬機器上安裝 Horizon Agent 40

install_viewagent.sh 命令列選項 41

設定 Linux 代理程式的憑證 42

- 升級 Linux 虛擬機器上的 Horizon Agent 43
- 升級 Linux 虛擬機器上的 Horizon Agent 44
- 解除安裝 Horizon 7 for Linux 機器 45

6 Linux 桌面平台的組態選項 46

- 在 Linux 桌面平台上設定組態檔中的選項 46
- Linux 桌面平台的 Blast 設定範例 53
- Linux 桌面平台用戶端磁碟機重新導向選項的範例 54
- 隱藏 Linux 桌面平台的 vSphere 主控台顯示 54

7 建立和管理 Linux 桌面平台集區 55

- 建立 Linux 的手動桌面平台集區 55
- 管理 Linux 的桌面平台集區 56
- 為 Linux 建立自動完整複製桌面平台集區 58
- 代理 PowerCLI 命令 60

8 針對手動桌面平台集區大量部署 Horizon 7 63

- Linux 桌面平台大量部署的概觀 63
- Linux 桌面平台大量升級的概觀 65
- 建立用於複製 Linux 桌面平台機器的虛擬機器範本 65
- 用於部署 Linux 桌面平台的範例 PowerCLI 指令碼輸入檔 67
- 用於複製 Linux 虛擬機器的範例指令碼 67
- 將複製的虛擬機器加入 AD 網域的範例指令碼 71
- 使用 SSH 將複製的虛擬機器加入 AD 網域的範例指令碼 74
- 上傳組態檔至 Linux 虛擬機器的範例指令碼 78
- 使用 SSH 上傳組態檔至 Linux 虛擬機器的範例指令碼 81
- 在 Linux 桌面平台機器上升級 Horizon Agent 的範例指令碼 85
- 使用 SSH 在 Linux 虛擬機器上升級 Horizon Agent 的範例指令碼 90
- 在 Linux 虛擬機器上執行作業的範例指令碼 95

9 疑難排解 Linux 桌面平台 100

- 收集 Horizon 7 for Linux 機器的診斷資訊 100
- 疑難排解遠端桌面平台和用戶端主機之間的複製和貼上 100
- 設定 Linux 防火牆以允許連入 TCP 連線 101
- View Agent 無法在 iPad Pro Horizon Client 上中斷連線 101
- SLES 12 SP1 桌面平台在拖放之後不會自動重新整理 101
- SSO 無法連線到關閉電源代理程式 102
- 在建立 Linux 的手動桌面平台集區後無法連線虛擬機器 102

設定 Horizon 7 for Linux 桌面平台

設定 Horizon 7 for Linux 桌面平台文件提供設定 Linux 虛擬機器以作為 VMware Horizon[®] 7 桌面平台的相關資訊，包括準備 Linux 客體作業系統、在虛擬機器上安裝 Horizon Agent，以及在 View Administrator 中設定機器以用於 Horizon 7 部署。

主要對象

本資訊適用對象是想要設定和使用執行於 Linux 客體作業系統的遠端桌面平台的任何使用者。本資訊是專為具有經驗且熟悉虛擬機器技術和資料中心操作的 Linux 系統管理員所撰寫。

功能和系統需求

在 Horizon 6 或更新版本中，使用者可以連線到執行 Linux 作業系統的遠端桌面平台。

本章節討論下列主題：

- [Horizon Linux 桌面平台的功能](#)
- [Horizon 7 for Linux 桌面平台的組態步驟概觀](#)
- [Horizon 7 for Linux 的系統需求](#)

Horizon Linux 桌面平台的功能

Horizon 7 (7.2 版) 為 Linux 桌面平台推出了數個新功能。

以下是 Horizon 7 (7.2 版) 所支援的新功能。

- 用戶端磁碟機重新導向 (CDR)
- USB 重新導向支援
- HTML Access 的音訊輸出支援
- RHEL 7 Workstation x64 和 CentOS 7 x64 上的 Single Sign-On 支援
- CentOS 6 x64 和 RHEL 6 x64 上的 K Desktop Environment (KDE) 支援
- RHEL 6.9 x64 和 CentOS 6.9 x64 的支援

下列清單顯示 Horizon Linux 桌面平台的主要功能。

自動完整複製桌面平台集區

從 Horizon 7 (7.0.2 版) 開始，您可以為 Linux 桌面平台建立自動完整複製桌面平台集區。

手動桌面平台集區

機器來源。

- 受管理的虛擬機器 - vCenter 虛擬機器的機器來源。新部署和升級部署可支援受管理的虛擬機器。
- 未受管理的虛擬機器 - 其他來源的機器來源。只有從未受管理的虛擬機器部署進行升級時，才可支援未受管理的虛擬機器。

備註 為確保能達到最佳效能，請勿使用未受管理的虛擬機器。

多監視器

- vDGA/vGPU 桌面平台可在四台監視器上支援最大解析度 2560x1600。

- vSphere 6.0 或更新版本上的 2D/vSGA 桌面平台可在四台監視器上支援最大解析度 2048x1536，或在三台監視器上支援最大解析度 2560x1600。

針對 Ubuntu 14.04/16.04，您必須停用 Compiz 才能支援多台監視器。如需詳細資訊，請參閱 <http://kb.vmware.com/kb/2114809>。

針對 SLES 12 SP1，您必須使用核心層級為 kernel-default-3.12.49-11.1 的預設套件。如果您已升級套件，則多監視器功能會失敗，而桌面平台會顯示在一台監視器中。

VMware Horizon HTML Access 不支援 Horizon 7 for Linux 桌面平台中的多監視器功能。

無失真 PNG

桌面平台上產生的影像和視訊會在用戶端裝置上以像素精準方式呈現。

軟體 H.264 編碼器

H.264 可改善 Horizon 桌面平台的 Blast Extreme 效能，在低頻寬網路下尤其如此。如果用戶端停用了 H.264，Blast Extreme 將會自動回復為 JPEG/PNG 編碼。

自 Horizon 7 (7.0.3 版) 起，支援多台監視器。

3D 圖形

下列組合的 Linux 版本和圖形卡支援 3D 圖形：

- 搭載 NVIDIA GRID K1 或 K2 圖形卡的 RHEL 7 Workstation x64 支援 vSGA。
- 搭載 NVIDIA GRID K1 或 K2 圖形卡的 RHEL 6 Workstation x64 支援 vDGA。
- 搭載 NVIDIA Maxwell M60 圖形卡的 RHEL 6 Workstation x64 支援 vGPU。
- 搭載 NVIDIA Maxwell M60 圖形卡的 RHEL 7 Workstation x64 支援 vGPU。
- 搭載 NVIDIA M6 圖形卡的 RHEL 6 Workstation x64 支援 vGPU。
- 搭載 NVIDIA M6 圖形卡的 RHEL 7 Workstation x64 支援 vGPU。

剪貼簿重新導向

透過剪貼簿重新導向功能，您可以在用戶端主機與遠端 Linux 桌面平台之間複製並貼上 RTF 文字或純文字。您可以使用 Horizon Agent 選項來設定複製/貼上方向和最大文字大小。此功能依預設為啟用狀態。您可以在安裝期間停用它。

Single Sign-on

下列 Linux 版本支援 Single Sign-on：

- RHEL 6/7 Workstation x64
- CentOS 6/7 x64
- SLED 11 SP3/SP4 x64

	<ul style="list-style-type: none"> ■ Ubuntu 14.04/16.04 x64
智慧卡重新導向與 SSO	RHEL 6 Workstation x64 支援智慧卡重新導向。支援個人身分驗證 (PIV) 卡和通用存取卡 (CAC)。不支援 Mac 用戶端。
Audio-in	<p>支援從用戶端主機到遠端 Linux 桌面平台的音訊輸入重新導向。此功能並非以 USB 重新導向功能為基礎。如果您要啟用此功能，則必須在安裝期間加以選取。您必須在您的應用程式中，選取系統預設音訊輸入裝置「PulseAudio 伺服器 (本機)」作為音訊輸入。支援此功能的平台如下：</p> <ul style="list-style-type: none"> ■ Ubuntu 14.04/16.04 x64 ■ CentOS 7 x64 ■ RHEL 7 Workstation x64
音訊輸出	支援音訊輸出重新導向。此功能依預設為啟用狀態。若要停用此功能，您必須將 <code>RemoteDisplay.allowAudio</code> 選項設為 false 。從 Horizon 7 (7.2 版) 開始，使用 Chrome 和 Firefox 瀏覽器存取的 VMware Horizon HTML Access 已可為 Linux 桌面平台提供音訊輸出支援。
用戶端磁碟機重新導向	從 Horizon 7 (7.2 版) 開始，用戶端磁碟機重新導向 (CDR) 功能已提供使用。當您啟用 CDR 功能時，本機系統的共用資料夾和磁碟機將變成可供您存取。您可以使用遠端 Linux 桌面平台上位於主目錄中的 <code>tsclient</code> 資料夾。若要使用此功能，您必須安裝 CDR 元件。
USB 重新導向	從 Horizon 7 (7.2 版) 開始，USB 重新導向功能已提供使用。此功能可讓您從遠端 Linux 桌面平台存取本機連結的 USB 裝置。若要使用 USB 功能，您必須安裝 USB 重新導向元件和 USB VHCI 驅動程式核心模組。確保您有足夠權限可使用所要重新導向的 USB 裝置。
備註 Horizon 7 (7.2 版) 中支援 USB 3.0 通訊協定。	
鍵盤配置和地區設定同步化	<p>此功能可指定是否將用戶端的系統地區設定和目前的鍵盤配置與 Horizon Linux Agent 桌面平台進行同步化。當此設定已啟用或未設定時，則允許進行同步化。當此設定已停用時，則不允許進行同步化。</p> <p>此功能僅支援 Windows 版 Horizon Client，並且僅適用於英文、法文、德文、日文、韓文、西班牙文、簡體中文和繁體中文地區設定。</p>
K Desktop Environment	<p>從 Horizon 7 (7.2 版) 開始，下列 Linux 發行版支援 K Desktop Environment (KDE)。</p> <ul style="list-style-type: none"> ■ CentOS 6 x64 ■ RHEL 6 x64
FIPS 140-2 模式	聯邦資訊處理標準 (FIPS) 140-2 模式雖然尚未向 NIST 密碼編譯模組驗證方案 (CMVP) 進行驗證，但從 Horizon 7 (7.2 版) 開始可提供支援。

Horizon 7 Agent for Linux 會實作專為 FIPS 140-2 符合性所設計的密碼編譯模組。這些模組會在 CMVP 憑證 #2839 和 #2866 所列出的作業環境中加以驗證，並移植到此平台。不過，在 VMware NIST CAVP 和 CMVP 憑證中納入新作業環境的 CAVP 和 CMVP 測試需求仍維持在產品藍圖上完成。

備註 要支援 FIPS 140-2 模式，必須使用傳輸層安全性 (TLS) 通訊協定 1.2 版。

Linux 桌面平台和桌面平台集區具有下列限制：

- 不支援虛擬列印、依據位置列印和即時視訊。

備註 使用安全伺服器時，必須在內部防火牆中開放連接埠 22443，才能在安全伺服器和 Linux 桌面平台之間允許流量。

Horizon 7 for Linux 桌面平台的組態步驟概觀

依據虛擬機器上安裝的是 2D 圖形或 3D 圖形，當您安裝和設定 Horizon 7 for Linux 桌面平台時，您必須依照不同的步驟順序。

2D 圖形 - 組態步驟概觀

對於 2D 圖形，執行下列步驟：

- 1 檢閱設定 Horizon 7 for Linux 部署的系統需求。請參閱 [Horizon 7 for Linux 的系統需求](#)。
- 2 在 vSphere 中建立虛擬機器並安裝 Linux 作業系統。請參閱[建立虛擬機器和安裝 Linux](#)。
- 3 為部署為 Horizon 7 環境的桌面平台準備客體作業系統。請參閱[準備用於遠端桌面平台部署的 Linux 機器](#)。
- 4 設定 Linux 客體作業系統以 Active Directory 驗證。根據環境需求，此步驟以第三方軟體實作，且未在本指南中說明。
- 5 在 Linux 虛擬機器上安裝 Horizon Agent。請參閱在 [Linux 虛擬機器上安裝 Horizon Agent](#)。
- 6 建立包含已設定 Linux 虛擬機器的桌面平台集區。請參閱[建立 Linux 的手動桌面平台集區](#)。

3D 圖形 - 組態步驟概觀

您必須先在 Linux 虛擬機器上完成 NVIDIA GRID vGPU、vDGA 或 vSGA 組態，才可在機器上安裝 Horizon Agent 並且在 View Administrator 中部署桌面平台集區。

- 1 檢閱設定 Horizon 7 for Linux 部署的系統需求。請參閱 [Horizon 7 for Linux 的系統需求](#)。
- 2 在 vSphere 中建立虛擬機器並安裝 Linux 作業系統。請參閱[建立虛擬機器和安裝 Linux](#)。
- 3 為部署為 Horizon 7 環境的桌面平台準備客體作業系統。請參閱[準備用於遠端桌面平台部署的 Linux 機器](#)。

- 4 設定 Linux 客體作業系統以 Active Directory 驗證。根據環境需求，此步驟以第三方軟體實作，且未在本指南中說明。
- 5 在 ESXi 主機和 Linux 虛擬機器上設定 3D 功能。依照您欲安裝的 3D 功能之程序進行。
 - 請參閱針對 vGPU 設定支援的 RHEL 發行版。
 - 請參閱針對 vDGA 設定 RHEL 6。
 - 請參閱針對 vSGA 設定 RHEL 7。
- 6 在 Linux 虛擬機器上安裝 Horizon Agent。請參閱在 Linux 虛擬機器上安裝 Horizon Agent。
- 7 建立包含已設定 Linux 虛擬機器的桌面平台集區。請參閱建立 Linux 的手動桌面平台集區。

大量部署

利用 View Administrator，您只能在手動桌面平台集區部署 Linux 虛擬機器。利用 vSphere PowerCLI，您可以開發能夠自動化部署 Linux 桌面平台機器集區的指令碼。請參閱第 8 章 針對手動桌面平台集區大量部署 Horizon 7。

Horizon 7 for Linux 的系統需求

若要安裝 Horizon 7 for Linux，您的 Linux 系統必須符合作業系統、Horizon 7 和 vSphere 平台的特定需求。

Horizon Agent 支援的 Linux 版本

下表列出了桌面平台集區中虛擬機器上支援的 Linux 作業系統。

表 1-1. View Agent 支援的 Linux 作業系統

Linux 發行版	架構
Ubuntu 14.04 和 16.04	x64
備註 停用 Compiz 以避免效能不佳。如需詳細資訊，請參閱 http://kb.vmware.com/kb/2114809 。	
Ubuntu 12.04	x64
RHEL 6.6、6.7、6.8、6.9、7.2 和 7.3	x64
CentOS 6.6、6.7、6.8、6.9、7.2 和 7.3	x64
NeoKylin 6 Update 1	x64
SLED 11 SP3/SP4、SLED 12 SP1/SP2	x64
SLES 12 SP1/SP2	x64

備註 Linux 代理程式在某些 Linux 發行版具有相依性套件。如需詳細資訊，請參閱安裝 Horizon Agent 相依性套件。

需要的平台和 Horizon 7 軟體版本

若要安裝及使用 Horizon 7 for Linux，您的部署必須符合 vSphere 平台、Horizon 7 和 Horizon Client 軟體的特定需求。

表 1-2. 需要的平台和 Horizon 7 軟體版本

平台和軟體	支援的版本
vSphere 平台版本	<ul style="list-style-type: none"> ■ vSphere 6.0 U2 或更新版本 ■ vSphere 6.5 或更新版本
Horizon 環境	<ul style="list-style-type: none"> ■ Horizon 連線伺服器 7.2
Horizon Client 軟體	<ul style="list-style-type: none"> ■ Android 版 Horizon Client 4.5.0 ■ Windows 版 Horizon Client 4.5.0 ■ Linux 版 Horizon Client 4.5.0 ■ Mac OS X 版 Horizon Client 4.5.0 ■ iOS 版 Horizon Client 4.5.0 (iPad Pro) ■ Chrome、Firefox 和 Internet Explorer 上的 HTML Access 4.5.0 ■ 不支援零用戶端

Linux 虛擬機器使用的 TCP 連接埠

View Agent 和 Horizon Client 在彼此之間與各種 View Server 元件之間的網路存取使用 TCP 連接埠。

表 1-3. Linux 虛擬機器使用的 TCP 連接埠

來源	連接埠	目標	連接埠	通訊協定	說明
Horizon Client	*	Linux 代理程式	22443	TCP	若未使用 Blast 安全閘道，則為 Blast
安全伺服器、View 連線伺服器或 Access Point 應用裝置	*	Linux 代理程式	22443	TCP	若使用 Blast 安全閘道，則為 Blast
View Agent	*	View 連線伺服器	4001、4002	TCP	JMS SSL 流量。

備註 如需有關用戶端使用之 TCP 和 UDP 連接埠的詳細資訊，請參閱 View 文件中《Horizon Client 和 View Agent 安全性》的〈用戶端和 View Agent 使用的 TCP 和 UDP 連接埠〉。

確認 Linux 虛擬機器使用的 Linux 帳戶

下表列出 Linux 虛擬機器使用的帳戶名稱和帳戶類型。

表 1-4. 帳戶名稱和帳戶類型

帳戶名稱	帳戶類型	使用者
根	Linux 作業系統內建	Java Standalone Agent、 mksvchanserver、Shell 指令碼
vmwblast	由 Linux 代理程式安裝程式建立	VMwareBlastServer
<目前登入的使用者>	Linux 作業系統內建、AD 使用者或 LDAP 使用者	Python 指令碼

桌面平台環境

Horizon 7 for Linux 桌面平台支援不同 Linux 發行版的多種桌面平台環境。下表列出 Horizon 7 for Linux 桌面平台對於各個 Linux 發行版支援的預設桌面平台環境。

表 1-5. 支援的桌面平台環境

Linux 發行版	預設桌面平台環境	Horizon 7 for Linux 桌面平台支援的桌面平台環境
Ubuntu 14.04/16.04	Unity	Gnome Fallback/Flashback (Metacity)
Ubuntu 12.04	Unity	Unity
RHEL/CentOS 6	Gnome	Gnome、K Desktop Environment (KDE)
RHEL/CentOS 7	Gnome	Gnome
SLED 11 SP4	Gnome	Gnome
SLED 12 SP1/SP2	Gnome	Gnome
SLES 12 SP1/SP2	Gnome	Gnome
NeoKylin 6 Update 1	Mate	Mate

若要在 Ubuntu 14.04/16.04 上安裝 Gnome Fallback/Flashback (Metacity) 桌面平台環境，請參閱 VMware 知識庫文章 <http://kb.vmware.com/kb/2114809> 以取得詳細資訊。

請使用下列資訊，將 KDE 安裝為 RHEL/CentOS 6 發行版上的預設桌面平台環境。

備註 KDE 的 Single Sign-On (SSO) 必須使用以下說明的步驟安裝才能正常運作。

- 1 使用預設桌面平台環境設定安裝 RHEL/CentOS 6 作業系統時，系統將會安裝 Gnome。
- 2 使用下列命令安裝 KDE。

```
sudo yum groupinstall "X Window System" "KDE Desktop"
```

如果您在安裝多個桌面平台環境的 Linux 發行版上啟用 SSO，請使用下列資訊來選取桌面平台環境。

- 針對 Ubuntu 14.04/16.04，請在 /etc/vmware/viewagent-custom.conf 檔案中啟用 UseGnomeFlashback 選項，以強制使用者使用 Gnome Fallback/Flashback (Metacity)。

- 針對 RHEL/CentOS 6，使用者必須在 `~/.dmrc` 檔案中指定桌面平台環境名稱 `kde` 或 `gnome`，以決定其下一個 SSO 登入工作階段中將使用的桌面平台環境。例如，若要以 KDE 作為預設桌面平台環境，使用者必須在其 `~/.dmrc` 檔案中包含下列設定。

```
[Desktop]
Session=kde
```

針對上述範例，使用者在初始設定之後必須登出或重新開機 Linux 系統，使 KDE 成為其下一個登入工作階段中的預設桌面平台。

如果您在安裝多個桌面平台環境的 Linux 發行版上停用 SSO，則使用者在登入該 Linux 發行版時，必須選取所需的桌面平台環境。

USB 重新導向的 VHCI 驅動程式

從 Horizon 7 for Linux (7.1 版) 桌面平台開始，已支援 USB 重新導向功能。此功能在 USB 虛擬主機控制器介面 (VHCI) 核心驅動程式上具有相依性。您必須修補 VHCI 驅動程式以支援 USB 3.0。

Horizon for Linux 安裝程式包含所支援 Linux 發行版之預設核心的 VHCI 驅動程式二進位檔。它會在選取 USB 重新導向功能時安裝 VHCI 驅動程式。下表列出 Horizon for Linux 桌面平台安裝程式將安裝的預設核心版本。

表 1-6. 預設核心版本

Linux 發行版	預設核心版本
RHEL/CentOS 6.9	2.6.32-696.el6.x86_64
SUSE 12 SP2	4.4.21-69-default
RHEL/CentOS 7.3	3.10.0-514.el7.x86_64
Ubuntu 14.04	3.13.0-24-generic
Ubuntu 16.04	4.4.0-21-generic

如果您的 Linux 系統所使用的核心版本不同於 Horizon for Linux 安裝程式隨附的預設版本，則必須從 <https://sourceforge.net/projects/usb-vhci/files/linux%20kernel%20module/> 下載 USB VHCI 原始程式碼。接著，您必須編譯 VHCI 驅動程式原始程式碼，並在您的 Linux 系統上安裝產生的二進位檔。

備註 安裝 Horizon for Linux 之前必須完成 VHCI 驅動程式安裝。

下載最新版的 USB VHCI 驅動程式原始程式碼之後，請使用下列命令加以編譯，並將其安裝在您的 Linux 系統上。例如，如果您將安裝檔案 `VMware-horizonagent-linux-x86_64-<version>-<build-number>.tar.gz` 解壓縮到 `/install_tmp/` 目錄下，則 `full-path-to-patch-file` 為 `/install_tmp/VMware-horizonagent-linux-x86_64-<version>-<buildnumber>/resources/vhci/patch/vhci.patch`，且要使用的 `patch` 命令為

```
# patch -p1 < /install_tmp/VMware-horizonagent-linux-x86_64-<version>-<build-number>/resources/vhci/patch/vhci.patch
```

表 1-7. 編譯及安裝 USB VHCI 驅動程式

Linux 發行版	編譯及安裝 USB VHCI 驅動程式的步驟
Ubuntu 14.04 Ubuntu 16.04	<p>編譯及安裝 VHCI 驅動程式。</p> <pre># tar -xzvf vhci-hcd-1.15.tar.gz # cd vhci-hcd-1.15 # patch -p1 < full-path_to_patch-file # make clean && make && make install</pre>
RHEL/CentOS 6.9 RHEL/CentOS 7.3	<ol style="list-style-type: none"> 安裝相依性套件。 <pre># yum install gcc-c++ # yum install kernel-devel-\$(uname -r) # yum install kernel-headers-\$(uname -r) # yum install patch</pre> 編譯及安裝 VHCI 驅動程式。 <pre># tar -xzvf vhci-hcd-1.15.tar.gz # cd vhci-hcd-1.15 # patch -p1 < full-path_to_patch-file # make clean && make && make install</pre>
SUSE 11 SP4 SUSE 12 SP2	<ol style="list-style-type: none"> 找出目前核心套件的版本 <pre># rpm -qa grep kernel-default-\$(echo \$(uname -r) cut -d '-' -f 1,2)</pre> <p>輸出結果為目前所安裝核心套件的名稱。例如，如果套件名稱為 kernel-default-3.0.101-63.1，則目前的核心套件版本將是 3.0.101-63.1。</p> 安裝相依性套件。 <ol style="list-style-type: none"> 針對 SUSE 11，請安裝符合目前核心的 kernel-source 和 kernel-default-devel 套件；即 gcc 和 patch 套件。 <pre>#zypper install --oldpackage kernel-source-<kernel-package-version> \ kernel-default-devel-<kernel-package-version> gcc patch</pre> <p>例如：</p> <pre>#zypper install --oldpackage kernel-source-3.0.101-63.1 kernel-default-devel-3.0.101-63.1 gcc patch</pre> 針對 SUSE 12，請安裝 kernel-devel、kernel-default-devel、kernel-macros 和 patch 套件。 <pre>#zypper install --oldpackage kernel-devel-<kernel-package-version> \ kernel-default-devel-<kernel-package-version> kernel-macros-<kernel-package-version> patch</pre> <p>例如：</p> <pre>#zypper install --oldpackage kernel-devel-4.4.21-90.1 kernel-default-devel-4.4.21-90.1 kernel-macros-4.4.21-90.1 patch</pre> 編譯及安裝 VHCI 驅動程式。 <pre># tar -xzvf vhci-hcd-1.15.tar.gz # cd vhci-hcd-1.15 # patch -p1 < full-path_to_patch-file # mkdir -p linux/\$(echo \$(uname -r) cut -d '-' -f 1)/drivers/usb/core # cp /lib/modules/\$(uname -r)/source/include/linux/usb/hcd.h linux/\$(echo \$(uname -r) cut -d '-' -f 1)/drivers/usb/core # make clean && make && make install</pre>

如果您的 Linux 核心版本已變更，則必須重新編譯並重新安裝 VHCI 驅動程式，但不需重新安裝 Horizon for Linux。

2D 和 vSGA 圖形卡的虛擬機器設定

當您建立特定的 Horizon 7 for Linux 虛擬機器時，您必須將記憶體設定和組態參數變更為建議的最小值。

設定為使用 NVIDIA vDGA 的虛擬機器會使用 NVIDIA 實體圖形卡。設定為使用 NVIDIA GRID vGPU 的虛擬機器會使用以 NVIDIA 實體圖形加速器為基礎的 NVIDIA 虛擬圖形卡。您不需要為這些虛擬機器變更視訊記憶體 (vRAM) 設定和組態參數。

設定為使用 2D 或 vSGA 圖形卡的虛擬機器會使用 VMware 虛擬圖形卡，且您必須為這些類型的虛擬機器變更下列設定：

- 視訊記憶體 (vRAM) 設定
- 組態參數
- 3D 記憶體設定
- 符合效能需求的 vCPU 和虛擬記憶體設定

視訊記憶體 (vRAM) 設定

在 vSphere Client 中建立 Linux 虛擬機器時，請依 [表 1-8. 2D 或 vSGA 圖形卡建議的 vRAM 設定](#) 所示設定 vRAM 大小。設定您為虛擬機器設定的顯示器數量和解析度所建議的 vRAM 大小。

表 1-8. 2D 或 vSGA 圖形卡建議的 vRAM 設定

vRAM 大小	顯示器數量	最大解析度
10 MB	1	1600x1200 或 1680x1050
12 MB	1	1920x1440
32 MB	1	2560x1600
48 MB	2	2048x1536
80 MB	2	2560x1600
128 MB	3	2560x1600
128 MB	4	2048x1536

這些 vRAM 大小是最低的建議值。如果虛擬機器上有更多資源可用，請將 vRAM 設定為更大的值，以獲得最佳的視訊效能。

對於設定為使用一部最低解析度監視器的機器，10 MB 是最低的 vRAM 大小建議值。

您必須依照[建立虛擬機器和安裝 Linux](#) 中的說明關閉虛擬機器的電源，以設定所要使用的顯示器數目和視訊記憶體數量。

Horizon 連線伺服器 7 不會像在 Windows 虛擬機器上一樣自動設定 Linux 虛擬機器上的 vRAM 設定。您必須手動在 vSphere Client 中設定 vRAM 設定。

如果 Linux 虛擬機器設定的 vRAM 大小比建議值還低，可能會發生下列問題：

- 桌面平台工作階段可能會在初始連線連上後立刻中斷連線。
- 自動調整可能無法運作。隨後桌面平台會顯示在螢幕的一小塊區域裡。

如果 Linux 虛擬機器的顯示器數目值小於實際的必要數量，則會有一或多台監視器顯示空白的桌面平台。

如果對建議的設定遇到自動調整的問題，您可以指定較大的 vRAM 大小。vSphere Client 允許的 vRAM 大小上限為 128 MB。如果指定的大小超出 128 MB，您必須手動修改 vmx 組態檔。下列範例指定 256 MB 的 vRAM 大小：

```
svga.vramSize = "268435456"
```

組態參數

若要在多台監視器上顯示 Linux 遠端桌面平台，您必須為虛擬機器設定特定的組態參數。為虛擬機器設定組態參數的一般步驟如下：

- 1 關閉虛擬機器電源。
- 2 在 vSphere Web Client 中，以滑鼠右鍵按一下虛擬機器，然後選取**編輯設定**。
- 3 按一下**虛擬機器選項**索引標籤，然後按一下**進階**。
- 4 按一下**編輯組態**，然後按一下**新增列**。
- 5 輸入組態參數名稱和值。
- 6 按一下**確定**以儲存變更。

您必須設定下列組態參數：

- 將 `svga.autodetect` 設為 `false`。

```
svga.autodetect="false"
```

- 根據顯示監視器的數量和方向 (水平或垂直) 來計算 `svga.maxWidth` 和 `svga.maxHeight` 值。一般規則是 `svga.maxWidth` 和 `svga.maxHeight` 值必須足夠大，以支援所有顯示器。例如，若要支援最大解析度為 2560x1600 的四個顯示器，您必須設定下列值：

```
svga.maxHeight="3200"
svga.maxWidth="10240"
```

如果您有多台監視器，則必須設定這些組態參數。否則，可能會遇到以下一或多個問題：

- 桌面平台會顯示在部分監視器上，而其他監視器是空白的。
- 按鍵輸入顯示多次。
- 桌面平台變得緩慢。
- 桌面平台會顯示在螢幕的一小部分區域中。

4096x4096 監視器的螢幕大小限制

針對 RHEL 6.8/6.9/7.3、CentOS 6.8/6.9/7.3、Ubuntu 16.04、SLED 12 SP2 和 SLES 12 SP2，2D 和 vSGA 的最大螢幕大小為 4096x4096。執行 `xrandr` 命令時，輸出的第一行具有 `maximum 4096x4096`。

針對 Ubuntu 14.04，如果您從官方 Ubuntu 存放庫安裝最新的修補程式，則也適用此限制。

多台監視器連線可能需要大於 4096x4096 的螢幕大小。若要略過此限制，請使用以下其中一個解決方案：

- 如果您的虛擬機器必須使用 VMware 硬體版本 11 (HWv11) 或更新版本，請將下列行新增至虛擬機器的 VMX 組態檔：

```
mks.enable3d = TRUE
```

利用此解決方案，Linux 作業系統即可將 3D 功能報告給軟體應用程式，例如 Chrome。不過，這麼做可能會影響 Linux 系統的效能。

- 如果您的虛擬機器可以使用 VMware 硬體版本 10，請在您的虛擬機器的 VMX 檔案中新增下列行：

```
virtualHW.version = "10"
```

此解決方案不適用於 RHEL 6.8 及 CentOS 6.8。

若要讓此解決方案在 Ubuntu 16.04 系統上正常運作，您必須從官方 Ubuntu 存放庫安裝最新的修補程式。

vCPU 和記憶體設定

若要改善 2D 或 vSGA 桌面平台的效能，請為 Linux 虛擬機器設定更多 vCPU 和虛擬記憶體。例如，設定 2 個 vCPU 和 2 GB 的虛擬記憶體。

對於多台監視器 (例如四台監視器) 的大型螢幕，請為虛擬機器設定 4 個 vCPU 和 4 GB 的虛擬記憶體。

對於 2D 或 vSGA 桌面平台中的視訊播放用途，請為虛擬機器設定 4 個 vCPU 和 4 GB 的虛擬記憶體。

3D 記憶體設定

若要改善 vSGA 多監視器環境中的效能，請將虛擬機器的 3D 記憶體設定設為 1 GB 或更高。

為桌面平台部署準備 Linux 虛擬機器

設定 Linux 桌面平台牽涉到建立 Linux 虛擬機器和為遠端桌面平台部署準備作業系統。

本章節討論下列主題：

- 建立虛擬機器和安裝 Linux
- 準備用於遠端桌面平台部署的 Linux 機器
- 安裝 Horizon Agent 相依性套件

建立虛擬機器和安裝 Linux

您會在 vCenter Server 中為 Horizon 7 中部署的每個遠端桌面平台建立新虛擬機器。您必須在虛擬機器上安裝 Linux 發行版。

必要條件

- 驗證您的部署符合支援 Linux 桌面平台的需求。請參閱 [Horizon 7 for Linux 的系統需求](#)。
- 熟悉在 vCenter Server 中建立虛擬機器和安裝客體作業系統的步驟。請參閱 在 Horizon 7 中設定虛擬桌面平台 文件中的「建立和準備虛擬機器」。
- 熟悉您將與虛擬機器搭配使用的顯示器建議的視訊記憶體 (vRAM) 值。請參閱 [Horizon 7 for Linux 的系統需求](#)。

程序

- 1 在 vSphere Web Client 或 vSphere Client 中，建立新虛擬機器。

2 設定自訂組態選項。

a 在虛擬機器上按一下滑鼠右鍵，然後按一下**編輯設定**。

b 指定 vCPU 和 vMemory 大小的數量。

如需建議的值，請遵循 Linux 發行版適用的安裝指南中的指示。

例如，Ubuntu 12.04 建議對 vMemory 設定 2048 MB 和 2 個 vCPU。

c 選取**視訊卡**，並指定顯示器數量和總視訊記憶體 (vRAM)。

針對使用 2D 或 vSGA 的虛擬機器 (使用 VMware 驅動程式)，在 vSphere Web Client 中設定 vRAM 大小。vRAM 大小在 vDGA 或 NVIDIA GRID vGPU 機器 (使用 NVIDIA 驅動程式) 上沒有作用。

如需建議的值，請遵循 Horizon 7 for Linux 的系統需求。請勿使用視訊記憶體計算機。

3 將虛擬機器開機並安裝 Linux 發行版。

4 設定要用於特定 Linux 發行版的桌面平台環境。

如需其他資訊，請參閱 [Horizon 7 for Linux 的系統需求](#) 中的〈桌面平台環境〉一節。

5 確保可將系統主機名稱解析為 127.0.0.1。

準備用於遠端桌面平台部署的 Linux 機器

您必須執行某些工作以準備要在 Horizon 7 部署中作為桌面平台使用的 Linux 機器。

Linux 機器必須能夠與連線伺服器通訊，Horizon 7 才能管理機器。您必須在 Linux 機器上設定網路，以便 Linux 機器可以使用其 FQDN (完整網域名稱) ping 連線伺服器執行個體。

RHEL 7、CentOS 7、SLED 12 和 SLES 12 機器上會預先安裝 Open VMware Tools (OVT)。如果您是要準備這些機器以用作遠端桌面平台，則可略過下列程序中的步驟 1 到 5，這些步驟說明如何以手動執行安裝程式來安裝 VMware Tools。

如果您正在使用 Ubuntu 16.04 機器，請在該機器上安裝 OVT。如果您正準備將此機器用作遠端桌面平台，則可略過下列程序中的步驟 1 到 5，並使用下列命令，在您的 Ubuntu 16.04 機器上手動安裝 OVT：

```
apt-get install open-vm-tools-desktop
```

必要條件

- 驗證已在 vCenter Server 中新建虛擬機器 (VM)，以及已在機器上安裝您的 Linux 發行版
- 自行熟悉在 Linux 虛擬機器上裝載和安裝 VMware Tools 的步驟。請參閱 vSphere 虛擬機器管理文件中的「在 Linux 虛擬機器中手動安裝或升級 VMware Tools」。
- 自行熟悉設定 Linux 機器以使機器可透過 DNS 解析的步驟。這些步驟可能因不同 Linux 發行版和版本而有所不同。如需指示，請查閱您的 Linux 發行版和版本的說明文件。

程序

1 在 vSphere Web Client 或 vSphere Client 中，在虛擬機器上裝載 VMware Tools 虛擬磁碟。

- 2 以滑鼠右鍵按一下 VMware Tools 安裝程式檔案 `VMwareTools.x.x.x-xxxx.tar.gz`，按一下解壓縮至，然後為您的 Linux 發行版選取桌面平台。

`vmware-tools-distrib` 資料夾隨即解壓縮到桌面平台。

- 3 在虛擬機器上，以 `root` 身分登入，並開啟終端機視窗。
- 4 解壓縮 VMware Tools tar 安裝程式檔案。

例如：

```
tar xzpf /mnt/cdrom/VMwareTools-x.x.x-yyyy.tar.gz
```

- 5 執行安裝程式並設定 VMware Tools。

不同 Linux 發行版中的命令可能有些微不同。例如：

```
cd vmware-tools-distrib
sudo ./vmware-install.pl -d
```

通常情況下，執行完安裝程式檔案之後會執行 `vmware-config-tools.pl` 組態檔案。

- 6 將 Linux 機器的主機名稱對應至 `/etc/hosts` 檔案中的 127.0.0.1。

針對 RHEL、CentOS、SLES 和 SLED，您必須手動將主機名稱對應至 127.0.0.1，因為它並不會自動對應。此步驟對 Ubuntu 來說並非必要，因為系統會依預設進行對應。此步驟在您大量部署桌面平台時也非必要，因為複製處理程序會新增此對應。

備註 如果您在安裝 Horizon Agent 之後變更了 Linux 機器的主機名稱，您必須將新的主機名稱對應至 `/etc/hosts` 檔案中的 127.0.0.1。否則，系統將會繼續使用舊的主機名稱。

- 7 對於 RHEL 7 和 CentOS 7，請確認已停用 `virbr0`。

```
virsh net-destroy default
virsh net-undefine default
service libvirtd restart
```

- 8 確保網繭中的 View 連線伺服器執行個體可透過 DNS 解析。

- 9 設定 Linux 機器，讓預設執行層級為 5。

執行層級必須是 5，Linux 桌面平台才能運作。

- 10 在設定以 OpenLDAP 伺服器驗證的 Ubuntu 機器上，設定機器的完整網域名稱。

此步驟可確保該資訊能在 View Administrator [工作階段] 頁面的 [使用者] 欄位中正確顯示。依照下列方法編輯 `/etc/hosts` 檔案：

- a `# nano /etc/hosts`
- b 新增完整網域名稱。例如：127.0.0.1 hostname.domainname hostname。
- c 結束並儲存檔案。

11 對於 SUSE，請停用 [透過 DHCP 變更主機名稱]。設定主機名稱或網域名稱。

- a 在 Yast 中，按一下網路設定。
- b 按一下主機名稱/DNS 索引標籤。
- c 取消選取透過 DHCP 變更主機名稱。
- d 輸入主機名稱和網域名稱。
- e 按一下確定。

安裝 VMware Tools 之後，如果您升級 Linux 核心，VMware Tools 可能停止執行。若要解決問題，請參閱 <http://kb.vmware.com/kb/2050592>。

安裝 Horizon Agent 相依性套件

Horizon Agent for Linux 具有某些 Linux 發行版特有的相依性套件。在安裝 Horizon Agent for Linux 之前，您必須先安裝這些套件。

必要條件

確認已在 vCenter Server 中建立新虛擬機器 (VM)，以及已在機器上安裝您的 Linux 發行版。

程序

1 安裝依預設不會進行安裝或升級的強制套件。如果有任何套件不符合需求，安裝程式便會中斷安裝。

表 2-1. 強制相依性套件

Linux 發行版	套件
SLED 11 SP3/SP4 將 xorg-x11-server 升級為比 7.4.27.111.1 更新的版本	<pre>zypper install xorg-x11-server</pre>
SLES 12 SP1/SLED 12 SP1 從 SUSE 存放庫將 xf86-video-vmware 升級為比 13.0.2-3.2 更新的版本	<ol style="list-style-type: none"> 登錄 SUSE 12 以啟用 SUSE 存放庫。 <pre>SUSEConnect -r 登錄代碼 -e 電子郵件</pre> 更新 xf86-video-vmware 版本。 <pre>zypper install xf86-video-vmware</pre>
SLES 12	<p>當您正在安裝 Horizon Agent 時，需要為 SLES 12 Linux 桌面平台安裝 python-gobject2。</p> <ol style="list-style-type: none"> 登錄 SUSE 12 以啟用 SUSE 存放庫。 <pre>SUSEConnect -r 登錄代碼 -e 電子郵件</pre> 安裝 python-gobject2。 <pre>zypper install python-gobject2</pre>

Linux 發行版	套件
Ubuntu 14.04 將 indicator-session 升級為 12.10.5+15.04.20150327，其可從 https://launchpad.net/ubuntu/wily/amd64/indicator-session/12.10.5+15.04.20150327-0ubuntu1 取得	<pre>wget http://launchpadlibrarian.net/201393830/indicator-session_12.10.5+15.04.20150327-0ubuntu1_amd64.deb</pre> <pre>sudo dpkg -i ./indicator-session_12.10.5+15.04.20150327-0ubuntu1_amd64.deb</pre>
Ubuntu 16.04	<pre>apt-get install python-dbus python-gobject</pre>

2 安裝 Horizon Agent 選用套件。

- 依預設，RHEL 或 CentOS 6.7 會安裝 `glibc-2.12-1.166.el6.x86_64`，而這可能導致鎖死問題。因此，桌面平台連線會中斷。若要克服此問題，您必須從線上存放庫將 `glibc` 升級為最新版本。

```
sudo yum install glibc
```

- 在具有多台監視器的 Ubuntu 14.04 桌面平台上，需要 `gnome-session-fallback` 才能停用 `Compiz` 以達到較佳的效能。

```
sudo apt-get install gnome-session-fallback
```

- 在具有多台監視器的 Ubuntu 16.04 桌面平台上，需要 `gnome-session-flashback` 才能停用 `Compiz` 以達到較佳的效能。

```
sudo apt-get install gnome-session-flashback
```

設定 Linux 桌面平台的 Active Directory 整合

3

View 使用現有的 Microsoft Active Directory (AD) 基礎結構進行使用者驗證及管理。您可以整合 Linux 桌面平台與 Active Directory，讓使用者可以使用自己的 Active Directory 使用者帳戶登入 Linux 桌面平台。

本章節討論下列主題：

- 整合 Linux 與 Active Directory
- 設定 Single Sign-On 和智慧卡重新導向

整合 Linux 與 Active Directory

您可以使用多個解決方案來整合 Linux 與 Active Directory (AD)，且 Horizon 7 for Linux 桌面平台與所用的解決方案不具有相依性。

以下是已知可在 Horizon 7 for Linux 桌面平台環境中運作的解決方案：

- OpenLDAP 伺服器傳遞驗證
- 對 Microsoft Active Directory 執行的系統安全服務精靈 (SSSD) LDAP 驗證
- Winbind 網域加入

OpenLDAP 傳遞驗證解決方案大致牽涉到下列步驟：

- 1 在 Active Directory 上安裝憑證服務以啟用 LDAPS (透過 SSL 的輕量型目錄存取通訊協定)。
- 2 設定 OpenLDAP 伺服器。
- 3 將使用者資訊 (密碼除外) 從 Active Directory 同步至 OpenLDAP 伺服器。
- 4 設定 OpenLDAP 伺服器以委派密碼驗證給個別處理程序 (例如 saslauthd)，以對 Active Directory 執行密碼驗證。
- 5 設定 Linux 桌面平台以使用 LDAP 用戶端對 OpenLDAP 伺服器驗證使用者。

對 Microsoft Active Directory 解決方案的 SSSD LDAP 驗證牽涉到下列步驟：

- 1 在 Active Directory 上安裝憑證服務以啟用 LDAPS。
- 2 在 Linux 桌面平台中設定 SSSD 以直接對 Microsoft Active Directory 使用 LDAP 驗證。

Winbind 網域加入解決方案牽涉到下列步驟：

- 1 在 Linux 桌面平台上安裝 Winbind、Samba 和 Kerberos 套件。

2 將 Linux 桌面平台加入至 Microsoft Active Directory。

如果您使用 LDAP 型解決方案，則僅需在範本虛擬機器中設定組態，而不需在複製的虛擬機器中執行其他步驟。

如果您使用 Winbind 網域加入解決方案或其他 Kerberos 驗證型解決方案，則需要將範本虛擬機器加入至 Active Directory，並將複製的虛擬機器重新加入至 Active Directory。例如，請使用下列命令：

```
sudo /usr/bin/net ads join -U <domain user>%<domain password>
```

請使用下列選項，對 Winbind 解決方案的已複製虛擬機器執行網域重新加入命令：

- 使用 SSH 或 vSphere PowerCLI 等工具遠端連線到每個虛擬機器，然後執行命令。如需指令碼的詳細資訊，請參閱 [第 8 章 針對手動桌面平台集區大量部署 Horizon 7](#)。
- 將命令包含在 Shell 指令碼中，然後在 `/etc/vmware/viewagent-custom.conf` 檔案中指定 Horizon Agent 選項 `RunOnceScript` 的指令碼路徑。如需詳細資訊，請參閱 [在 Linux 桌面平台上設定組態檔中的選項](#)。

備註 若要簡化部署程序，請對 Microsoft Active Directory 解決方案使用 SSSD LDAP 驗證。

設定 Single Sign-On 和智慧卡重新導向

若要設定 Single Sign-On (SSO) 和智慧卡重新導向，您必須執行一些組態步驟。

單一登入

Horizon View Single Sign-On 模組會連絡 Linux 中的 PAM (插入式驗證模組)，並且不依賴於您用來整合 Linux 與 Active Directory (AD) 的方法。Horizon View SSO 已知可與整合 Linux 與 AD 的 OpenLDAP 和 Winbind 解決方案搭配使用。

依預設，SSO 會假設 AD 的 `sAMAccountName` 屬性為登入識別碼。為確保對 SSO 使用正確的登入識別碼，如果您使用 OpenLDAP 或 Winbind 解決方案，則必須執行下列設定步驟：

- 針對 OpenLDAP，將 `sAMAccountName` 設定為 `uid`。
- 針對 Winbind，新增下列陳述式至組態檔 `/etc/samba/smb.conf`。

```
winbind use default domain = true
```

如果使用者必須指定要登入的網域名稱，您必須在 Linux 桌面平台上設定 `SSOUserFormat` 選項。如需詳細資訊，請參閱 [在 Linux 桌面平台上設定組態檔中的選項](#)。請注意，SSO 一律使用大寫的簡短網域名稱。例如，如果網域為 `mydomain.com`，SSO 將使用 `MYDOMAIN` 做為網域名稱。因此，設定 `SSOUserFormat` 選項時，您必須指定 `MYDOMAIN`。關於簡短和完整網域名稱，所適用的規則如下：

- 針對 OpenLDAP，您必須使用大寫的簡短網域名稱。
- Winbind 支援完整和簡短網域名稱。

AD 支援在登入名稱中使用特殊字元，但 Linux 不支援。因此，設定 SSO 時請勿在登入名稱中使用特殊字元。

在 AD 中，如果使用者的 `UserPrincipalName (UPN)` 屬性和 `sAMAccount` 屬性不符，而且使用者使用 UPN 登入，SSO 將會失敗。因應措施是讓使用者使用 `sAMAccount` 中儲存的名稱來登入。

View 不要求使用者名稱區分大小寫。您必須確保 Linux 作業系統可以處理不區分大小寫的使用者名稱。

- 針對 Winbind，使用者名稱預設不區分大小寫。
- 針對 OpenLDAP，Ubuntu 使用 NSCD 來驗證使用者，並且預設不區分大小寫。RHEL 和 CentOS 使用 SSSD 來驗證使用者，並且預設區分大小寫。若要變更設定，請編輯檔案 `/etc/sss/sss.conf`，並在 `[domain/default]` 區段新增下列程式行：

```
case_sensitive = false
```

針對 Ubuntu 16.04 或 14.04，請在 `/etc/vmware/viewagent-custom.conf` 檔案中設定 `UseGnomeFlashback=TRUE`，才能使用 GNOME Flashback (Metacity) 桌面平台環境。

智慧卡重新導向

若要設定智慧卡重新導向，請先遵循 Linux 經銷商與智慧卡廠商的指示。然後將 `pcsc-lite` 套件更新為 1.7.4。例如，執行下列命令：

```
#yum groupinstall "Development tools"
#yum install libudev-devel
#service pcscd stop
#wget https://alioth.debian.org/frs/download.php/file/3598/pcsc-lite-1.7.4.tar.bz2
#tar -xjvf pcsc-lite-1.7.4.tar.bz2
#cd ./pcsc-lite-1.7.4
#./configure --prefix=/usr/ --libdir=/usr/lib64/ --enable-usbdropdir=/usr/lib64/pcsc/drivers
--enable-confdir=/etc --enable-ipcdire=/var/run --disable-libusb --disable-serial --disable-usb
--disable-libudev
#make
#make install
#service pcscd start
```

針對 Winbind，新增下列陳述式至組態檔 `/etc/samba/smb.conf`。

```
winbind use default domain = true
```

安裝 Horizon Agent 時，您必須先停用 SELinux 或為 SELinux 啟用允許模式。您也必須特別選取智慧卡重新導向元件，因為該元件並非預設選取。如需詳細資訊，請參閱 [install_viewagent.sh 命令列選項](#)。

在 Horizon View 7.0.1 或更新版本中會啟用智慧卡 SSO。此外，如果在虛擬機器上安裝了智慧卡重新導向功能，則 vSphere Client 的 USB 重新導向無法與智慧卡搭配使用。

智慧卡重新導向僅支援一部智慧卡讀卡機。如果連接了兩部以上的讀卡機到用戶端裝置，此功能將無法運作。

智慧卡重新導向僅支援一張卡片上一個憑證。如果卡片上有多個憑證，則會使用第一個插槽中的憑證，並忽略其他憑證。此為 Linux 的限制。

備註

- 智慧卡支援下列 Winbind 值。若非如此，智慧卡 SSO 和手動登入將會失敗。

```
winbind use default domain=true
```

- 當您使用 Linux 用戶端透過 PIV 卡來驗證代理時 (Linux 桌面平台智慧卡重新導向可支援此功能)，您必須在 `~/.vmware/view-preferences` 上為 Linux 用戶端新增 `view.sslProtocolString = "TLSv1.1"` 組態，以避免發生 SSL 錯誤。
-

設定 Linux 桌面平台的圖形卡

您可以設定目前支援的 RHEL 發行版，以在 ESXi 主機或客體作業系統上利用 NVIDIA 功能。

設定 3D 圖形的虛擬機器複製需求

在設定 3D 圖形之前，您必須考慮下列虛擬機器複製需求。

- 對於 vGPU 和 vSGA，請在基礎虛擬機器中完成圖形設定。複製虛擬機器。圖形設定適用於複製的虛擬機器，且不需要進一步的設定。
- 對於 vDGA，請在基礎虛擬機器中完成圖形設定。複製虛擬機器。不過，在將複製的虛擬機器開啟電源之前，您必須先從複製的虛擬機器中移除現有的 NVIDIA 傳遞 PCI 裝置，並將新的 NVIDIA 傳遞 PCI 裝置新增至複製的虛擬機器。您無法在虛擬機器之間共用 NVIDIA 傳遞 PCI 裝置。每台虛擬機器都會使用專用的 NVIDIA 傳遞 PCI 裝置。

本章節討論下列主題：

- 針對 vGPU 設定支援的 RHEL 發行版
- 針對 vDGA 設定 RHEL 6
- 針對 vSGA 設定 RHEL 7

針對 vGPU 設定支援的 RHEL 發行版

您可以設定支援的 RHEL 發行版，以利用 ESXi 主機上的 NVIDIA vGPU (共用 GPU 硬體加速) 功能。

您必須使用符合 ESXi 主機 GPU 驅動程式 (.vib) 的 NVIDIA Linux 虛擬機器顯示驅動程式。請參閱 NVIDIA 網站以取得驅動程式套件的相關資訊。

重要 NVIDIA Maxwell M60 圖形卡和 NVIDIA M6 圖形卡支援 NVIDIA vGPU。此功能無法在其他 NVIDIA 圖形卡上使用，例如 GRID K1 或 K2。

注意 開始之前，請確認未在 Linux 虛擬機器上安裝 Horizon Agent。如果您在將機器設為使用 NVIDIA vGPU 之前先安裝 Horizon Agent，則會覆寫 xorg.conf 檔案中所需的組態參數，而且 NVIDIA vGPU 將無法運作。您必須在完成 NVIDIA vGPU 組態後才可安裝 Horizon Agent。

在 ESXi 主機上安裝適用於 NVIDIA GRID vGPU 圖形卡的 VIB

您必須為 ESXi 6.0 U1 或更新版本主機上的 NVIDIA GRID 圖形卡下載並安裝 VIB。

NVIDIA 提供包含 vGPU Manager 的 vGPU 軟體套件，也就是您在此程序中安裝於 ESXi 主機上的套件，以及 Linux 顯示驅動程式，這會在之後的程序中安裝於 Linux 虛擬機器。

必要條件

- 驗證您的環境中已安裝 vSphere 6.0 U1 或更新版本。
- 確認 ESXi 主機上已安裝 NVIDIA Maxwell M60 GPU 或 M6 GPU。

程序

- 1 從 [NVIDIA 驅動程式下載](#) 網站下載適用於 NVIDIA GRID vGPU 圖形卡的 VIB。
從下拉式功能表選取適當的 VIB 版本。

選項	描述
產品類型	GRID
產品系列	選取 NVIDIA GRID vGPU 。
產品	選取 ESXi 主機上安裝的版本 (例如 GRID K2)。
作業系統	選取 VMware vSphere ESXi 版本。

- 2 解壓縮 vGPU 軟體套件的 .zip 檔案。
- 3 將 vGPU Manager 資料夾上傳到 ESXi 6.0 U1 主機。

備註 您將會在之後的程序於 Linux 虛擬機器上安裝 Linux 顯示驅動程式。

- 4 關閉或暫停 ESXi 主機上的所有虛擬機器。
- 5 使用 SSH 連線到 ESXi 主機。
- 6 停止 xorg 服務。

```
# /etc/init.d/xorg stop
```

- 7 安裝 NVIDIA VIB。

例如：

```
# esxcli system maintenanceMode set --enable true
# esxcli software vib install -v /path-to-vib/NVIDIA-VIB-name.vib
# esxcli system maintenanceMode set --enable false
```

- 8 將 ESXi 主機重新開機或更新。
 - ◆ 對於已安裝的 ESXi 主機，將主機重新開機。
 - ◆ 對於無狀態 ESXi 主機，請執行下列步驟以更新主機。(這些步驟也可用於已安裝的主機。)

```
Update vmkdevmgr:
# kill -HUP $(cat /var/run/vmware/vmkdevmgr.pid)

Wait for the update to complete:
```

```
# localcli --plugin-dir /usr/lib/vmware/esxcli/int deviceInternal bind

This is a new requirement with the NVIDIA 352.* host driver:
# /etc/init.d/nvidia-vgpu start

Restart xorg, which is used for GPU assignment:
# /etc/init.d/xorg start
```

9 驗證 xorg 服務在重新啟動主機後開始執行。

在 Linux 虛擬機器上針對 vGPU 設定共用 PCI 裝置

若要使用 NVIDIA vGPU，您必須為 Linux 虛擬機器設定共用的 PCI 裝置。

必要條件

- 確認 Linux 虛擬機器已備妥可作為桌面平台。請參閱[建立虛擬機器和安裝 Linux](#)與準備用於遠端桌面平台部署的 [Linux 機器](#)。
- 確認未在 Linux 虛擬機器上安裝 Horizon Agent。
- 確認已在 ESXi 主機上安裝 NVIDIA VIB。請參閱 [在 ESXi 主機上安裝適用於 NVIDIA GRID vGPU 圖形卡的 VIB](#)。
- 自行熟悉可與 NVIDIA vGPU 使用的虛擬 GPU 類型，這些類型是您以 **GPU 設定檔**設定所做的選擇。虛擬 GPU 類型在安裝於 ESXi 主機的實體 GPU 上提供不同的功能。請參閱 [NVIDIA 虛擬 GPU 類型](#)。

程序

- 1 關閉虛擬機器電源。
- 2 在 vSphere Web Client 中，選取虛擬機器，然後在 **虛擬機器硬體**索引標籤下，按一下**編輯設定**。
- 3 在**新增裝置**功能表中，選取**共用 PCI 裝置**。
- 4 按一下**新增**，並從下拉式功能表選取 **NVIDIA GRID vGPU**。
- 5 針對 **GPU 設定檔**設定，從下拉式功能表選取虛擬 GPU 類型。
- 6 按一下**保留所有記憶體**，然後按**確定**。

您必須保留所有虛擬機器的記憶體，GPU 才能支援 NVIDIA GRID vGPU。

- 7 開啟虛擬機器電源。

NVIDIA 虛擬 GPU 類型

藉由 vSphere Web Client 虛擬硬體頁面的 **GPU 設定檔**設定，您可以選取在 ESXi 主機的實體 NVIDIA GPU 上提供特定功能的 GPU 類型。

在 Linux 虛擬機器上，NVIDIA Maxwell M60 GPU 或 NVIDIA M6 GPU 支援 NVIDIA GRID vGPU。

表 4-1. 在 Linux 虛擬機器上適用於 NVIDIA GRID M60 vGPU 的虛擬 GPU 類型

虛擬 GPU 類型	實體主機板	實體 GPU	每個虛擬 GPU 的畫面格緩衝區	顯示螢幕	最大解析度	每個實體 GPU 的虛擬 GPU 上限	每個實體主機板的虛擬 GPU 上限
GRID M60-0q	GRID M60	兩個	512M	2	2560x1600	16	32
GRID M60-1q	GRID M60	兩個	1G	2	2560x1600	8	16
GRID M60-2q	GRID M60	兩個	2G	4	2560x1600	4	8
GRID M60-4q	GRID M60	兩個	4G	4	3840x2160	2	4
GRID M60-8q	GRID M60	兩個	8G	4	3840x2160	1	2

表 4-2. 在 Linux 虛擬機器上適用於 NVIDIA GRID M6 vGPU 的虛擬 GPU 類型

虛擬 GPU 類型	實體主機板	實體 GPU	每個虛擬 GPU 的畫面格緩衝區	顯示螢幕	最大解析度	每個實體 GPU 的虛擬 GPU 上限	每個實體主機板的虛擬 GPU 上限
GRID M6-0q	GRID M6	一個	512M	2	2560x1600	16	16
GRID M6-1q	GRID M6	一個	1G	2	2560x1600	8	8
GRID M6-2q	GRID M6	一個	2G	4	2560x1600	4	4
GRID M6-4q	GRID M6	一個	4G	4	3840x2160	2	2
GRID M6-8q	GRID M6	一個	8G	4	3840x2160	1	1

安裝 NVIDIA GRID vGPU 顯示驅動程式

若要安裝 NVIDIA GRID vGPU 顯示驅動程式，您必須停用預設的 NVIDIA 驅動程式、下載 NVIDIA 顯示驅動程式，並在虛擬機器上設定 PCI 裝置。

必要條件

- 確認您已從 NVIDIA 下載網站下載 vGPU 軟體套件、解壓縮套件，並備妥 Linux 顯示驅動程式 (套件元件)。請參閱在 [ESXi 主機上安裝適用於 NVIDIA GRID vGPU 圖形卡的 VIB](#)。

另外確認共用 PCI 裝置已新增至虛擬機器。請參閱在 [Linux 虛擬機器上針對 vGPU 設定共用 PCI 裝置](#)

程序

- 1 停用並將預設的 NVIDIA Nouveau 驅動程式加入黑名單。

- a 編輯 `grub.conf` 或 `grub` 檔案。

針對 RHEL 6，該檔案為 `/boot/grub/grub.conf`。針對 RHEL 7，該檔案為 `/etc/default/grub`。

RHEL 版本	命令
6	<code>sudo vi /boot/grub/grub.conf</code>
7	<code>sudo vi /etc/default/grub</code>

- b 在核心選項尾端新增 `rdblacklist=nouveau` 行。
 - c 編輯 `blacklist.conf` 檔案。

```
sudo vi /etc/modprobe.d/blacklist.conf
```

- d 在 `blacklist.conf` 檔案的任意處新增下列行。

```
blacklist nouveau
```

- 2 重新啟動虛擬機器。

顯示的外觀與風格有變更。

- 3 (選擇性) 驗證已停用 Nouveau 驅動程式。

```
/sbin/lsmmod | grep nouveau
```

如果 `grep` 搜尋未傳回任何結果，則 Nouveau 驅動程式為停用。

- 4 將 NVIDIA Linux 顯示驅動程式複製到虛擬機器。
- 5 開啟虛擬機器的遠端終端機，或輸入 `Ctrl-Alt-F2`、登入為 `root` 身分並執行 `init 3` 命令來停用 X Windows，以切換到文字主控台。
- 6 安裝 NVIDIA 驅動程式所需的額外元件。

```
sudo yum install gcc-c++
sudo yum install kernel-devel-$(uname -r)
sudo yum install kernel-headers-$(uname -r)
```

- 7 將可執行檔旗標新增至 NVIDIA GRID vGPU 驅動程式套件。

```
chmod +x NVIDIA-Linux-x86_64-version-grid.run
```

8 啟動 NVIDIA GRID vGPU 安裝程式。

```
sudo ./NVIDIA-Linux-x86_64-version-grid.run
```

9 接受 NVIDIA 軟體授權合約並選取是來自動更新 X 組態設定。

後續步驟

在 Linux 虛擬機器上安裝 Horizon Agent。請參閱在 [Linux 虛擬機器上安裝 Horizon Agent](#)。

建立包含已設定 Linux 虛擬機器的桌面平台集區。請參閱[建立 Linux 的手動桌面平台集區](#)。

確認已安裝 NVIDIA 顯示驅動程式

您可以在 View 桌面平台工作階段中顯示 NVIDIA 驅動程式輸出，以確認 RHEL 6 虛擬機器上已安裝 NVIDIA 顯示驅動程式。

必要條件

- 檢查是否已安裝 NVIDIA 顯示驅動程式。
- 確認已在 Linux 虛擬機器上安裝 Horizon Agent。請參閱在 [Linux 虛擬機器上安裝 Horizon Agent](#)。
- 確認 Linux 虛擬機器已部署於桌面平台集區。請參閱[建立 Linux 的手動桌面平台集區](#)。

程序

1 重新啟動 Linux 虛擬機器。

Horizon Agent 啟動指令碼會初始化 X 伺服器並顯示拓撲。

您無法繼續在 vSphere 主控台中檢視虛擬機器顯示畫面。

2 從 Horizon Client，連接至 Linux 桌面平台。

3 在 Linux 桌面平台工作階段中，驗證已安裝 NVIDIA 顯示驅動程式。

開啟終端機視窗，並執行 `glxinfo | grep NVIDIA` 命令。

NVIDIA 驅動程式輸出隨即顯示。例如：

```
[root]# glxinfo | grep NVIDIA
server glx vendor string: NVIDIA Corporation
client glx vendor string: NVIDIA Corporation
OpenGL vendor string: NVIDIA Corporation
OpenGL version string: 4.5.0 NVIDIA 346.47
OpenGL shading language version string: 4.50 NVIDIA
```

使用者可以在遠端桌面平台上存取 NVIDIA 圖形功能。

在確認已安裝 NVIDIA 顯示驅動程式之後，請執行下列工作以讓安裝正確運作。

- 如果您升級 Linux 核心，Horizon Agent 可能無法與 View 連線伺服器通訊。若要解決問題，請重新安裝 NVIDIA 驅動程式。

- 在 Linux 虛擬機器中設定 NVIDIA GRID 授權。如需詳細資訊，請參閱 NVIDIA 說明文件。如果未設定授權，Linux 桌面平台將無法正確運作。例如，自動調整將無法運作。

針對 vDGA 設定 RHEL 6

您可以設定 RHEL 6 客體作業系統，讓 Horizon 7 for Linux 桌面平台得以在 ESXi 主機上利用 vDGA 功能。

注意 開始之前，請確認未在 Linux 虛擬機器上安裝 Horizon Agent。如果您在將機器設為使用 vDGA 之前先安裝 Horizon Agent，則會覆寫 `xorg.conf` 檔案中所需的組態參數，而且 vDGA 將無法運作。您必須在完成 vDGA 組態後才可安裝 Horizon Agent。

在主機上針對 NVIDIA GRID 啟用 DirectPath I/O

設定 Linux 虛擬機器以使用 vDGA 之前，您必須使 NVIDIA GRID GPU PCI 裝置可在 ESXi 主機上的 DirectPath I/O 傳遞使用。

必要條件

- 驗證您的環境中已安裝 vSphere 6.0 或更新版本。
- 驗證 ESXi 主機上已安裝 NVIDIA GRID K1 或 K2 圖形卡。

程序

- 1 在 vSphere Web Client 中，瀏覽至 ESXi 主機。
- 2 按一下**管理索引標籤**，然後按一下**設定**。
- 3 在 [硬體] 區段中，按一下 **PCI 裝置**。
- 4 若要啟用 NVIDIA GRID GPU 的 DirectPath I/O 傳遞，請按一下**編輯**。

圖示	描述
綠色圖示	PCI 裝置處於作用中狀態且可啟用。
橙色圖示	裝置的狀態已變更。您必須先將主機重新開機，然後才能使用該裝置。

- 5 選取 NVIDIA GRID GPU 並按一下**確定**。
PCI 裝置即新增至「可供 VM 使用的 DirectPath I/O PCI 裝置」表格。
- 6 將主機重新開機，以讓 PCI 裝置可供 Linux 虛擬機器使用。

新增 vDGA 傳遞裝置到 RHEL 6 虛擬機器

若要將 RHEL 6 虛擬機器設定為使用 vDGA，您必須新增 PCI 裝置到虛擬機器。藉由此步驟，即可傳遞 ESXi 主機上的實體裝置，以在虛擬機器上使用。

必要條件

- 確認 Linux 虛擬機器已備妥可作為桌面平台。請參閱[建立虛擬機器和安裝 Linux](#) 與準備用於遠端桌面平台部署的 [Linux 機器](#)。
- 確認未在 Linux 虛擬機器上安裝 Horizon Agent。
- 驗證 NVIDIA GRID GPU PCI 裝置可供主機上的 DirectPath I/O 傳遞使用。請參閱[在主機上針對 NVIDIA GRID 啟用 DirectPath I/O](#)。

程序

- 1 以 sudo 權限設定的本機使用者身分登入 RHEL 6 客體作業系統。
- 2 在 vSphere Web Client 中，選取虛擬機器，然後在 [虛擬機器硬體](#) 索引標籤下，按一下 [編輯設定](#)。
- 3 在 [新增裝置](#) 功能表中，選取 **PCI 裝置**。
- 4 按一下 **新增**，並從下拉式功能表選取 PCI 裝置。
- 5 按一下 **保留所有記憶體**，然後按 **確定**。

您必須保留所有虛擬機器的記憶體，GPU 才能支援 vDGA。

- 6 打開虛擬機器的電源，並開啟 vSphere 主控台來連線至機器。
- 7 驗證 NVIDIA GRID 裝置已傳遞到虛擬機器。

開啟終端機視窗，並執行下列命令：

```
lspci | grep NVIDIA
```

XX:00.0 VGA 相容控制器隨即顯示。例如：

```
NVIDIA Corporation GK104GL [GRID K2]
```

安裝 vDGA 的 NVIDIA 顯示驅動程式

若要安裝 vDGA 的 NVIDIA 顯示驅動程式，您必須停用預設的 NVIDIA 驅動程式、下載 NVIDIA 顯示驅動程式，並在虛擬機器上設定 PCI 裝置。

必要條件

- 驗證 PCI 裝置已新增至 RHEL 6 虛擬機器。請參閱[新增 vDGA 傳遞裝置到 RHEL 6 虛擬機器](#)。

程序

- 1 停用並將預設的 NVIDIA Nouveau 驅動程式加入黑名單。

- a 編輯 `grub.conf` 檔案。

針對 RHEL 6，該檔案為 `/boot/grub/grub.conf`。

RHEL 版本	命令
6	<code>sudo vi /boot/grub/grub.conf</code>

- b 在核心選項尾端新增 `rdblacklist=nouveau` 行。

- c 編輯 `blacklist.conf` 檔案。

```
sudo vi /etc/modprobe.d/blacklist.conf
```

- d 在 `blacklist.conf` 檔案的任意處新增下列行。

```
blacklist nouveau
```

- 2 重新啟動虛擬機器。

顯示的外觀與風格有變更。

- 3 (選擇性) 驗證已停用 Nouveau 驅動程式。

```
/sbin/lsmmod | grep nouveau
```

如果 `grep` 搜尋未傳回任何結果，則 Nouveau 驅動程式為停用。

- 4 從 [NVIDIA 驅動程式下載](#) 網站下載 NVIDIA 驅動程式。

從 NVIDIA 下拉式功能表選取適當的驅動程式版本：

選項	說明
產品類型	GRID
產品系列	GRID 系列
產品	選取 ESXi 主機上安裝的版本 (例如 GRID K2)。
作業系統	Linux 64 位元或 Linux 32 位元

- 5 開啟虛擬機器的遠端終端機，或輸入 `Ctrl-Alt-F2`、登入為 `root` 身分並執行 `init 3` 命令來停用 X Windows，以切換到文字主控台。
- 6 安裝 NVIDIA 驅動程式所需的額外元件。

```
sudo yum install gcc-c++
sudo yum install kernel-devel-$(uname -r)
sudo yum install kernel-headers-$(uname -r)
```

- 7 將可執行檔旗標新增至 vDGA 的 NVIDIA 驅動程式套件。

```
chmod +x NVIDIA-Linux-x86_64-version.run
```

- 8 啟動 NVIDIA 安裝程式。

```
sudo ./NVIDIA-Linux-x86_64-version.run
```

- 9 接受 NVIDIA 軟體授權合約並選取是來自動更新 X 組態設定。

後續步驟

在 Linux 虛擬機器上安裝 Horizon Agent。請參閱[在 Linux 虛擬機器上安裝 Horizon Agent](#)。

建立包含已設定 Linux 虛擬機器的桌面平台集區。請參閱[建立 Linux 的手動桌面平台集區](#)。

確認已安裝 NVIDIA 顯示驅動程式

您可以在 View 桌面平台工作階段中顯示 NVIDIA 驅動程式輸出，以確認 RHEL 6 虛擬機器上已安裝 NVIDIA 顯示驅動程式。

必要條件

- 檢查是否已安裝 NVIDIA 顯示驅動程式。
- 確認已在 Linux 虛擬機器上安裝 Horizon Agent。請參閱[在 Linux 虛擬機器上安裝 Horizon Agent](#)。
- 確認 Linux 虛擬機器已部署於桌面平台集區。請參閱[建立 Linux 的手動桌面平台集區](#)。

程序

- 1 重新啟動 Linux 虛擬機器。

Horizon Agent 啟動指令碼會初始化 X 伺服器並顯示拓撲。

您無法繼續在 vSphere 主控台中檢視虛擬機器顯示畫面。

- 2 從 Horizon Client，連接至 Linux 桌面平台。
- 3 在 Linux 桌面平台工作階段中，驗證已安裝 NVIDIA 顯示驅動程式。

開啟終端機視窗，並執行 `glxinfo | grep NVIDIA` 命令。

NVIDIA 驅動程式輸出隨即顯示。例如：

```
[root]# glxinfo | grep NVIDIA
server glx vendor string: NVIDIA Corporation
client glx vendor string: NVIDIA Corporation
OpenGL vendor string: NVIDIA Corporation
OpenGL version string: 4.5.0 NVIDIA 346.47
OpenGL shading language version string: 4.50 NVIDIA
```

使用者可以在遠端桌面平台上存取 NVIDIA 圖形功能。

在確認已安裝 NVIDIA 顯示驅動程式之後，請執行下列工作以讓安裝正確運作。

- 如果您升級 Linux 核心，Horizon Agent 可能無法與 View 連線伺服器通訊。若要解決問題，請重新安裝 NVIDIA 驅動程式。
- 在 Linux 虛擬機器中設定 NVIDIA GRID 授權。如需詳細資訊，請參閱 NVIDIA 說明文件。如果未設定授權，Linux 桌面平台將無法正確運作。例如，自動調整將無法運作。

針對 vSGA 設定 RHEL 7

您可以設定 RHEL 7 客體作業系統，讓 Horizon 7 for Linux 桌面平台得以利用 vSGA 功能。

安裝 NVIDIA 圖形卡的 VIB 以在 ESXi 主機上實現 vSGA 功能

您必須為 ESXi 6.0 U1 或更新版本主機上的 NVIDIA GRID 圖形卡下載並安裝 VIB。

NVIDIA 會提供 vSGA 的 VMware vSphere ESXi 驅動程式。對於 vSGA，NVIDIA 顯示驅動程式不會安裝在 Linux 虛擬機器上。

必要條件

- 驗證您的環境中已安裝 vSphere 6.0 U1 或更新版本。
- 確認您的環境中已安裝 NVIDIA 驅動程式。
- 確認 ESXi 主機上已安裝 NVIDIA GRID K1 或 K2 GPU。

程序

- 1 從 [NVIDIA 驅動程式下載](#) 網站下載適用於 NVIDIA GRID vGPU 圖形卡的 VIB。

從下拉式功能表選取適當的 VIB 版本。

選項	描述
產品類型	GRID
產品系列	選取 GRID 系列。
產品	選取 ESXi 主機上安裝的版本 (例如 GRID K2)。
作業系統	選取 VMware vSphere ESXi 版本。

- 2 將適用於 vSGA 的 VMware vSphere ESXi 驅動程式上傳到 ESXi 6.0 U1 主機。
- 3 關閉或暫停 ESXi 主機上的所有虛擬機器。
- 4 使用 SSH 連線到 ESXi 主機。
- 5 停止 xorg 服務。

```
# /etc/init.d/xorg stop
```

6 安裝 NVIDIA VIB。

例如：

```
# esxcli system maintenanceMode set --enable true
# esxcli software vib install -v /path-to-vib/NVIDIA-VIB-name.vib
# esxcli system maintenanceMode set --enable false
```

7 重新啟動用於 GPU 指派的 xorg：

```
# /etc/init.d/xorg start
```

8 將 ESXi 主機重新開機。

9 驗證 xorg 服務在重新啟動主機後開始執行。

在 Linux 虛擬機器上設定 vSGA 的 3D 功能

若要將 RHEL 7 虛擬機器設定為使用 vSGA，您必須在 vSphere Web Client 中設定虛擬機器視訊卡的 3D 設定。

必要條件

- 確認 Linux 虛擬機器已備妥，可作為桌面平台、已安裝 Horizon Agent，並且機器是部署在桌面平台集區中。
- 確認已在 ESXi 主機上安裝 NVIDIA VIB。請參閱[安裝 NVIDIA 圖形卡的 VIB 以在 ESXi 主機上實現 vSGA 功能](#)。

程序

- 1 關閉虛擬機器電源。
- 2 在 vSphere Web Client 中，選取虛擬機器，然後在 **虛擬機器硬體** 索引標籤下，按一下 **編輯設定**。
- 3 在虛擬硬體索引標籤上，按一下 **視訊卡** 以展開功能表設定。
- 4 將 **視訊記憶體總計** 設為 128 MB。
- 5 若為 **3D 圖形**，請選取 **啟用 3D 支援**。
- 6 若為 **3D 轉譯器**，請從下拉式功能表選取 **硬體**。
- 7 若為 **3D 記憶體**，請依應用程式需求選取適用的數值。

如果您的使用者連線至 3 部以上的監視器，請至少將此值設為 1024 MB。

8 按一下 **確定**。

9 開啟虛擬機器電源。

後續步驟

確認 vSGA 在 Linux 虛擬機器上執行。

接著，在 Linux 虛擬機器上安裝 Horizon Agent。請參閱在 [Linux 虛擬機器上安裝 Horizon Agent](#)。

確認 vSGA 在 Linux 虛擬機器上執行

您可以檢查虛擬機器記錄檔和客體作業系統，以確認 vSGA 在 RHEL 7 虛擬機器上執行。

程序

- 1 開啟虛擬機器的 `vmware.log` 檔案。

如果支援的 GPU 和 NVIDIA VIB 已正確安裝，記錄檔會顯示各行，如下列範例所示：

```
2015-06-24T22:19:25.259Z| mks| I120: OpenGL Version: "4.0.0 NVIDIA 346.69" (4.0.0)
2015-06-24T22:19:25.259Z| mks| I120: GLSL Version: "4.00 NVIDIA" (4.00.0)
2015-06-24T22:19:25.259Z| mks| I120: OpenGL Vendor: "NVIDIA Corporation"
2015-06-24T22:19:25.259Z| mks| I120: OpenGL Renderer: "Quadro 4000/PCIe/SSE2"
```

如果支援的 GPU 和 NVIDIA VIB 未正確安裝，虛擬機器會使用軟體轉譯器。`vmware.log` 檔案會顯示各行，如下列範例所示：

```
2015-07-06T17:09:26.423Z| vmx| I120: [msg.mks.noGPUResourceFallback] Hardware GPU resources are
not available. The virtual machine uses software rendering.
2015-07-06T17:09:26.423Z| vmx| I120: -----
2015-07-06T17:09:26.425Z| svga| I120: MKS-SWP: plugin started - llvmpipe (LLVM 3.3, 256 bits)
2015-07-06T17:09:26.426Z| svga| I120: Started Shim3D
2015-07-06T17:09:26.426Z| svga| I120: MKS-RenderMain: Starting SWRenderer
```

- 2 在虛擬機器的客體作業系統中，輸入下列命令。

```
glxinfo|grep Gallium
```

如果 vSGA 正在運作，命令會傳回下列文字：

```
OpenGL renderer string: Gallium 0.4 on SVGA3D; build : RELEASE;
```

如果 vSGA 未正常運作，命令會傳回下列文字：

```
OpenGL renderer string: Gallium 0.4 on llvmpipe (LLVM 3.3, 256 bits)
```

後續步驟

在 Linux 虛擬機器上安裝 Horizon Agent。請參閱[在 Linux 虛擬機器上安裝 Horizon Agent](#)。

安裝 Horizon Agent

您必須在 Linux 桌面平台上安裝 Horizon Agent，以便 View 連線伺服器可以與桌面平台通訊並管理桌面平台。

本章節討論下列主題：

- 在 Linux 虛擬機器上安裝 Horizon Agent
- 設定 Linux 代理程式的憑證
- 升級 Linux 虛擬機器上的 Horizon Agent
- 解除安裝 Horizon 7 for Linux 機器

在 Linux 虛擬機器上安裝 Horizon Agent

您必須先在 Linux 虛擬機器上安裝 Horizon Agent，之後才能夠將機器部署為遠端桌面平台。

從 Horizon 7.0.1 版開始，Horizon Agent for Linux 會使用 vCenter 管理的虛擬機器。受管理的虛擬機器可提供下列增強功能。

- vCenter 是 Linux 桌面平台部署的必要需求。
- 在 Linux 上安裝 Horizon Agent 時不需要登錄。
- 若要進行大規模的 Linux 桌面平台部署，您可以將 Horizon Agent 安裝在基礎虛擬機器上。

注意 如果您打算使用 NVIDIA GRID vGPU、vDGA 或 vSGA，請務必在安裝 Horizon Agent 之前先在 Linux 虛擬機器上設定這些 3D 功能。若先安裝 Horizon Agent，則會覆寫 xorg.conf 檔案中的所需參數，而且 3D 圖形功能將無法運作。

請參閱針對 vGPU 設定支援的 RHEL 發行版、針對 vDGA 設定 RHEL 6 或針對 vSGA 設定 RHEL 7。請在完成 3D 圖形組態之後再安裝 Horizon Agent。

對於 2D 圖形組態，您可以在完成準備用於遠端桌面平台部署的 Linux 機器中的步驟後再安裝 Horizon Agent。

必要條件

- 驗證 Linux 客體作業系統已準備就緒而可供桌面平台使用。請參閱準備用於遠端桌面平台部署的 Linux 機器。
- 自行熟悉 Linux 適用的 Horizon Agent 安裝程式指令碼。請參閱 `install_viewagent.sh` 命令列選項。

程序

- 1 從 VMware 下載網站下載 Linux 版 Horizon Agent 安裝程式檔案，網址為 <https://my.vmware.com/web/vmware/downloads>。

在「桌面平台與一般使用者運算」下，選取 VMware Horizon 7 下載，其包括 Linux 版 Horizon Agent 安裝程式。

64 位元 Linux 的安裝程式檔案名稱為 VMware-horizonagent-linux-x86_64-y.y.y-xxxxxxx.tar.gz，其中，y.y.y 是版本號碼，而 xxxxxxx 是組建編號。

- 2 在客體作業系統上解壓縮您的 Linux 發行版的 tarball。

例如：

```
tar -xzf <View Agent tar ball>
```

- 3 導覽至 tar ball 資料夾。
- 4 以 superuser 的身分執行 install_viewagent.sh 指令碼。

請參閱 [install_viewagent.sh 命令列選項](#) 以取得命令列選項的清單。

例如：

```
sudo ./install_viewagent.sh
```

- 5 如果您執行 install_viewagent.sh 而未指定 -A 選項，請輸入 **Yes** 來接受 EULA。

除非您接受 EULA，否則安裝程式不會執行。

- 6 將 Linux 重新開機以使變更生效。

安裝之後，會啟動 *viewagent* 服務。請使用 `sudo service viewagent status` 確認服務已啟動。

後續步驟

在桌面平台集區部署虛擬機器。請參閱[建立 Linux 的手動桌面平台集區](#)。

install_viewagent.sh 命令列選項

install_viewagent.sh 指令碼會在 Linux 客體作業系統上安裝 Horizon Agent。

在 gnome 桌面平台環境的命令提示視窗中，使用下列格式的 install_viewagent.sh 指令碼。

```
install_viewagent.sh command_option 引數 [command_option 引數] . . .
```

install_viewagent.sh 指令碼包括強制和選用參數。

表 5-1. install_viewagent.sh 選用但必要的參數

選用參數 (必要資訊)	說明
-A yes	EULA 和 FIPS 陳述式接受。您必須指定 yes ，安裝才能繼續。如果未指定此參數，安裝指令碼會出現提示，要求您輸入值。

表 5-2. install_viewagent.sh 選用參數

選用參數	說明
-a yes no	安裝或略過音訊輸入重新導向支援。預設值為 yes 。
-f yes no	安裝或略過專為聯邦資訊處理標準 (FIPS) 140-2 所設計的密碼編譯模組支援。預設值為 no 。如需詳細資訊，請參閱 Horizon Linux 桌面平台的功能 中的 FIPS 140-2 模式說明。
-j	JMS SSL 金鑰儲存區密碼。依預設，安裝程式會產生隨機字串。
-m yes no	安裝或略過智慧卡重新導向支援。預設值為 no 。
-r yes no	在安裝之後自動重新啟動系統。預設值為 no 。
-s	自我簽署憑證主體 DN。依預設，安裝程式會使用 Blast。
-C yes no	安裝或略過剪貼簿重新導向支援。預設值為 yes 。
-F yes no	安裝或略過 CDR 支援。預設值為 yes 。
-M yes no	將 Linux 代理程式升級為受管理或未受管理的代理程式。預設值為 yes 。
-S yes no	安裝或略過 Single Sign-On (SSO) 支援。預設值為 yes 。
-U yes no	安裝或略過 USB 支援。預設值為 no 。

表 5-3. install_viewagent.sh 參數範例

條件	範例
全新安裝	<pre>sudo ./install_viewagent.sh -A yes</pre> <p>全新安裝一律需要建立新的桌面平台集區。</p>
從未受管理的虛擬機器升級，並保留未受管理的虛擬機器樣式	<pre>sudo ./install_viewagent.sh -A yes-M no</pre> <p>此類型的升級不需要建立新的桌面平台集區。您可以重複使用現有的桌面平台集區。</p> <p>備註 為確保能達到最佳效能，請勿使用未受管理的虛擬機器。</p>
從未受管理的虛擬機器部署升級，並轉換為受管理的虛擬機器樣式。此升級需要在代理上建立新的桌面平台集區	<pre>sudo ./install_viewagent.sh -A yes</pre> <p>此類型的升級需要建立新的桌面平台集區。您必須刪除現有的桌面平台集區。</p>

設定 Linux 代理程式的憑證

當您安裝 Linux 代理程式時，安裝程式會產生 VMwareBlastServer 的自我簽署憑證。

- 當代理停用 Blast 安全閘道時，VMwareBlastServer 會向使用 HTML Access 來連線至 Linux 桌面平台的瀏覽器顯示此憑證。

- 當代理啟用 Blast 安全閘道時，Blast 安全閘道的憑證會向瀏覽器顯示憑證。

若要符合產業或安全法規，您可以將自我簽署憑證取代為憑證授權單位 (CA) 簽署的憑證。

程序

- 1 將私密金鑰和憑證安裝至 VMwareBlastServer。
 - a 將私密金鑰重新命名為 `rui.key`，並將憑證重新命名為 `rui.crt`。
 - b 執行 `sudo chmod 550 /etc/vmware/ssl`。
 - c 將 `rui.crt` 和 `rui.key` 複製到 `/etc/vmware/ssl`。
 - d 執行 `chmod 440 /etc/vmware/ssl`。
- 2 將根和中繼憑證授權單位安裝至 Linux 作業系統憑證授權單位存放區。

備註 如需變更 Linux 系統設定，請參閱 Linux 發行版說明文件。

升級 Linux 虛擬機器上的 Horizon Agent

您可以藉由安裝最新版本的 Horizon Agent，來升級 Linux 虛擬機器上的 Horizon Agent。

未受管理的虛擬機器：代理程式安裝程式會將虛擬機器登錄至需要代理管理資訊的代理。桌面平台集區建立精靈會使用 [機器來源] 頁面中的**其他來源**，選取已登錄的虛擬機器。

受管理的虛擬機器：安裝程式不會與代理通訊。桌面平台集區建立精靈會使用 [機器來源] 頁面中的**vCenter 虛擬機器**，透過 vCenter 選取虛擬機器。受管理的虛擬機器部署支援下列功能。

- 遠端機器電源原則
- 允許使用者重設其機器

備註 Horizon Agent for Linux 7.0.0 和更早版本的運作方式與未受管理的虛擬機器相同。Horizon Agent for Linux 7.0.1 的運作方式與受管理的虛擬機器支援相同。

您可以使用下列方法，從未受管理升級為受管理的虛擬機器部署。

- 保留未受管理的虛擬機器部署，並升級為必要版本。此類型的升級不需要在 View 連線伺服器中進行任何組態修改。
- 從未受管理的虛擬機器部署，升級為任何版本的受管理虛擬機器部署。此類型的升級需要在 View 連線伺服器上建立新的桌面平台集區。

備註 從受管理的虛擬機器部署升級時，您可以保留受管理的虛擬機器部署，並升級為必要版本。不過，在升級期間從受管理的虛擬機器部署轉換為未受管理的虛擬機器部署，則是不受支援的。

以下是可用於升級的參數。

表 5-4. 升級 Horizon Agent 的選用參數

參數	說明
-A yes	EULA 和 FIPS 陳述式接受。您必須指定 yes ，安裝才能繼續。如果未指定此參數，安裝指令碼會出現提示，要求您輸入值。
-a yes no	安裝或略過音訊輸入重新導向支援。預設值為 yes 。
-f yes no	安裝或略過專為聯邦資訊處理標準 (FIPS) 140-2 所設計的密碼編譯模組支援。預設值為 no 。如需詳細資訊，請參閱 Horizon Linux 桌面平台的功能 中的 FIPS 140-2 模式說明。
-m yes no	安裝或略過智慧卡重新導向支援。預設值為 no 。
-r yes no	安裝之後將作業系統重新開機。預設值為 no 。
-C yes no	安裝或略過剪貼簿重新導向支援。預設值為 yes 。
-F yes no	安裝或略過 CDR 支援。預設值為 yes 。
-M yes no	將 Linux 代理程式升級為受管理 未受管理的代理程式。預設值為 yes 。
-S yes no	安裝或略過 Single Sign-On (SSO) 支援。預設值為 yes 。
-U yes no	安裝或略過 USB 支援。預設值為 no 。

升級 Linux 虛擬機器上的 Horizon Agent

您可以藉由安裝最新版本的 Horizon Agent，來升級 Linux 機器上的 Horizon Agent。

必要條件

- 確認 VMwareBlastServer 處理程序未執行。

若要停止此處理程序，請確定使用者已登出機器，且沒有作用中的桌面平台工作階段，或是將機器重新開機。

程序

- 1 從 VMware 下載網站下載最新的 Linux 版 Horizon Agent 安裝程式檔案，網址為 <https://my.vmware.com/web/vmware/downloads>。

在「桌面平台與一般使用者運算」下，選取 VMware Horizon 7 下載，其包括 Linux 版 Horizon Agent 安裝程式。

64 位元 Linux 的安裝程式檔案名稱為 VMware-viewagent-linux-x86_64-y.y.y-xxxxxxx.tar.gz，其中，y.y.y 是版本號碼，而 xxxxxxx 是組建編號。

- 2 在客體作業系統上解壓縮您的 Linux 發行版的 tarball。

例如：

```
tar -xzf <View Agent tar ball>
```

- 3 導覽至 tar ball 資料夾。

- 4 執行 `install_viewagent.sh` 指令碼，以使用下列其中一個部署案例升級未受管理的虛擬機器：

選項	說明
升級未受管理的虛擬機器部署，並保留未受管理的虛擬機器部署	<pre>sudo ./install_viewagent.sh -A yes -M no</pre> <p>備註 為確保能達到最佳效能，請勿使用未受管理的虛擬機器。</p>
升級未受管理的虛擬機器部署，並將其變更為受管理的虛擬機器部署	<pre>sudo ./install_viewagent.sh -A yes -M yes</pre> <p>備註 在 View Administrator 中，刪除未受管理的虛擬機器部署現有的桌面平台集區，並為受管理的虛擬機器部署建立新的桌面平台集區。如需詳細資訊，請參閱建立 Linux 的手動桌面平台集區。</p>
升級受管理的虛擬機器部署	<pre>sudo ./install_viewagent.sh -A yes -M yes</pre> <p>備註 升級之後，您現有的桌面平台集區將可重複使用。</p>

解除安裝 Horizon 7 for Linux 機器

若要解除安裝虛擬機器上的 Horizon 7 for Linux，您必須解除安裝 Horizon Agent，並移除組態檔。

必要條件

確認 VMware BlastServer 處理程序未執行。若要停止此處理程序，請確定您已登出機器，且沒有作用中的桌面平台工作階段，或是將機器重新開機。

程序

- 1 在虛擬機器上開啟終端機視窗，並執行 Horizon Agent 解除安裝指令碼。

```
sudo /usr/lib/vmware/viewagent/bin/uninstall_viewagent.sh
```

此指令碼會停止 Horizon Agent 處理程序，並從安裝目錄 `/usr/lib/vmware/viewagent` 中刪除 Horizon Agent 服務和軟體。

- 2 手動刪除 `/etc/vmware` 目錄中的 Horizon 7 for Linux 組態檔。

Linux 桌面平台的組態選項

您可以使用組態檔設定各種選項來自訂使用者體驗。

本章節討論下列主題：

- 在 Linux 桌面平台上設定組態檔中的選項
- Linux 桌面平台的 Blast 設定範例
- Linux 桌面平台用戶端磁碟機重新導向選項的範例
- 隱藏 Linux 桌面平台的 vSphere 主控台顯示

在 Linux 桌面平台上設定組態檔中的選項

您可以透過新增項目到檔案 `/etc/vmware/config` 或 `/etc/vmware/viewagent-custom.conf` 來設定某些選項。

在 View Agent 或 Horizon Agent 的安裝期間，安裝程式會將兩個組態範本檔 `config.template` 和 `viewagent-custom.conf.template` 複製到 `/etc/vmware`。此外，如果檔案 `/etc/vmware/config` 和 `/etc/vmware/viewagent-custom.conf` 不存在，安裝程式會將 `config.template` 複製到 `config`，並將 `viewagent-custom.conf.template` 複製到 `viewagent-custom.conf`。範本檔中會列出並記載所有組態選項。若要設定選項，只需移除註解並適當變更值。

例如，`/etc/vmware/config` 中的下列程式行可啟用無失真 PNG 模式。

```
RemoteDisplay.alwaysLossless=TRUE
```

進行組態變更之後，請將 Linux 重新開機以讓變更生效。

`/etc/vmware/config` 中的組態選項

VMwareBlastServer 和其相關外掛程式使用組態檔 `/etc/vmware/config`。

備註 下表的說明包含 Horizon Agent 組態檔中由每個代理程式強制執行的 USB 原則設定。Horizon Agent 會使用這些設定來決定是否可以將某個 USB 轉送至主機。Horizon Agent 也會將設定傳遞至 Horizon Client 進行解譯和強制執行，並根據您是指定合併 (m) 修飾詞來套用 Horizon Agent 篩選原則設定以及 Horizon Client 篩選原則設定，或是指定覆寫 (o) 修飾詞以使用 Horizon Agent 篩選原則設定而非 Horizon Client 篩選原則設定。

表 6-1. /etc/vmware/config 中的組態選項

選項	值/格式	預設值	說明
RemoteDisplay.alwaysLossless	true 或 false	false	圖形應用程式，特別是圖形設計應用程式，會需要在 Linux 桌面平台的用戶端有像素精準的影像呈現。您可以針對 Linux 桌面平台上產生和在用戶端裝置上呈現的影像和視訊播放，設定無失真 PNG 模式。此功能會在用戶端和 ESXi 主機之間使用額外的頻寬。
mksVNCServer.useUIInputButton Mapping	true 或 false	false	設定此選項可在 Ubuntu 或 RHEL 7 上啟用左手使用滑鼠的支援。CentOS 和 RHEL 6 支援左手使用滑鼠，因此您不需要設定此選項。
RemoteDisplay.allowAudio	true 或 false	true	設定此選項可啟用/停用音訊輸出。
VVC.ScRdir.Enable	true 或 false	true	設定此選項可啟用/停用智慧卡重新導向。
VVC.logLevel	fatal error、warn、info、debug 或 trace	info	使用此選項以設定 VVC Proxy 節點的記錄層級。
VVC.RTAV.Enable	true 或 false	true	設定此選項可啟用/停用音訊輸入。
Clipboard.Direction	0, 1, 2, 或 3	2	此選項決定剪貼簿重新導向原則。 <ul style="list-style-type: none"> ■ 0 - 停用剪貼簿重新導向。 ■ 1 - 啟用雙向剪貼簿重新導向。 ■ 2 - 僅啟用從用戶端到遠端桌面平台的剪貼簿重新導向。 ■ 3 - 僅啟用從遠端桌面平台到用戶端的剪貼簿重新導向。
cdrserver.logLevel	error、warn、info、debug、trace 或 verbose	info	使用此選項，可設定 vmware-CDRserver.log 的記錄層級
cdrserver.forcedByAdmin	true 或 false	false	設定此選項，可防止或允許用戶端共用未使用 cdrserver.shareFolders 選項指定的其他資料夾。
cdrserver.sharedFolders	<i>file_path1,R; file_path2,; file_path3,R; . . .</i>	未定義	指定用戶端可以與 Linux 桌面平台共用的一或多個資料夾的檔案路徑。例如： <ul style="list-style-type: none"> ■ 針對 Windows 用戶端：C:\spreadsheets,;D:\ebooks,R ■ 針對非 Windows 用戶端：/tmp/spreadsheets;/tmp/ebooks,;/home/finance,R

選項	值/格式	預設值	說明
cdrserver.permissions	R	RW	<p>使用此選項，可在 Horizon Agent 所具備 Horizon Client 共用的資料夾上套用其他的讀取/寫入權限。例如：</p> <ul style="list-style-type: none"> ■ 如果 Horizon Client 共用的資料夾具有 read 和 write 權限，而您設定 cdrserver.permissions=R，則 Horizon Agent 僅具有 read 存取權限。 ■ 如果 Horizon Client 共用的資料夾僅具有 read 權限，而您設定 cdrserver.permissions=RW，則 Horizon Agent 仍將僅具有 read 存取權限。Horizon Agent 無法變更 Horizon Client 設定的唯一 read 屬性。Horizon Agent 僅能移除寫入存取權限 <p>一般的使用方式為：</p> <ul style="list-style-type: none"> ■ cdrserver.permissions=R ■ #cdrserver.permissions=R (例如註解排除或刪除該項目)
cdrserver.cacheEnable	true 或 false	true	設定此選項，可啟用或停用從代理程式到用戶端的寫入快取功能。
UsbRedirPlugin.log.logLevel	error、warn、info、debug、trace 或 verbose	info	使用此選項，可設定 USB 重新導向外掛程式的記錄層級。
UsbRedirServer.log.logLevel	error、warn、info、debug、trace 或 verbose	info	使用此選項，可設定 USB 重新導向伺服器的記錄層級。
viewusb.AllowAutoDeviceSplitting	{m o}: {true false}	未定義，其相當於 false	<p>設定此選項，可允許或不允許複合 USB 裝置的自動分割。</p> <p>範例: m:true</p>
viewusb.SplitExcludeVidPid	{m o}:vid-xxx1_pid-yyy1[;vid-xxx2_pid-yyy2;...]	未定義	<p>使用此選項，可排除或包含指定的複合 USB 裝置，以決定是否根據廠商和產品識別碼進行分割。設定的格式為 vid-xxx1_pid-yyy1[;vid-xxx2_pid-yyy2;...]。您必須以十六進位格式指定識別碼。您可以在識別碼中使用萬用字元 (*) 以取代個別數字。</p> <p>範例: m:vid-0f0f_pid-55**</p>

選項	值/格式	預設值	說明
viewusb.SplitVidPid	{m o}: vid-xxxx_pid-yyyy([exintf:zz[;exintf:ww]]);...	未定義	<p>設定此選項，可將根據廠商和產品識別碼指定的複合 USB 裝置元件視為個別裝置。設定的格式為 vid-xxxx_pid-yyyy(exintf:zz[;exintf:ww])。您可以使用 exintf 關鍵字，藉由指定它們的介面號碼來將元件自重新導向清單中排除。您必須以十六進位指定識別碼，及以十進位指定介面號碼，包括任何前置的 0。您可以在識別碼中使用萬用字元 (*) 以取代個別數字。</p> <p>範例： o:vid-0f0f_pid-*** (exintf-01);vid-0781_pid-554c(exintf:01;exintf:02)</p> <p>備註</p> <p>Horizon 不會自動包含您未明確排除的元件。您必須指定篩選原則，例如納入 VidPid 裝置，以納入那些元件。</p>
viewusb.AllowAudioIn	{m o}: {true false}	未定義，其相當於 true	使用此選項，可允許或不允許將音訊輸入裝置重新導向。範例: o:false
viewusb.AllowAudioOut	{m o}: {true false}	未定義，其相當於 false	設定此選項，可允許或不允許音訊輸出裝置的重新導向。
viewusb.AllowHIDBootable	{m o}: {true false}	未定義，其相當於 true	使用此選項以允許或不允許將鍵盤或滑鼠以外可在開機時使用的輸入裝置 (又稱為 HID 可開機裝置) 重新導向。
viewusb.AllowDevDescFailsafe	{m o}: {true false}	未定義，其相當於 false	設定此選項後，即便 Horizon Client 無法取得組態或裝置描述元時，仍可允許或不允許將裝置重新導向。若要在即使無法取得組態或裝置描述元的情形下也允許裝置，請將它納入在 Include 篩選器當中，例如 IncludeVidPid 或 IncludePath 。
viewusb.AllowKeyboardMouse	{m o}: {true false}	未定義，其相當於 false	使用此選項，可以允許或不允許將具備整合式指向裝置 (例如滑鼠、軌跡球或觸控板) 的鍵盤重新導向。
viewusb.AllowSmartcard	{m o}: {true false}	未定義，其相當於 false	設定此選項，可允許或不允許將智慧卡裝置重新導向。
viewusb.AllowVideo	{m o}: {true false}	未定義，其相當於 true	使用此選項，可允許或不允許將視訊裝置重新導向。
viewusb.DisableRemoteConfig	{m o}: {true false}	未定義，其相當於 false	設定此選項，可停用或啟用在執行 USB 裝置篩選時使用 Horizon Agent 設定。
viewusb.ExcludeAllDevices	{true false}	未定義，其相當於 false	使用此選項，可排除或包含所有 USB 裝置，以決定是否進行重新導向。如果設定為 true ，您可以使用其他原則設定，以允許將特定裝置或裝置系列重新導向。如果設定為 false ，您可以使用其他原則設定，以避免將特定裝置或裝置系列重新導向。如果在 Horizon Agent 上將 ExcludeAllDevices 的值設為 true ，且此設定傳遞至 Horizon Client，則 Horizon Agent 設定會覆寫 Horizon Client 設定。

選項	值/格式	預設值	說明
viewusb.ExcludeFamily	<code>{m o}:family_name_1[;family_name_2;...]</code>	未定義	<p>使用此選項，可排除裝置系列以避免進行重新導向。例如：m:bluetooth;smart-card</p> <p>如果您已經啟用自動裝置分割功能，Horizon 便會檢驗複合 USB 裝置每個介面的裝置系列，以確認應該排除的介面。如果您已經停用自動裝置分割功能，Horizon 便會檢驗整個複合 USB 裝置的裝置系列。</p> <p>備註 不過，依預設系統會排除滑鼠和鍵盤不進行重新導向，因此不需使用此設定來排除。</p>
viewusb.ExcludeVidPid	<code>{m o}:vid-xxx1_pid-yyy1[;vid-xxx2_pid-yyy2;...]</code>	未定義	<p>設定此選項，可排除具有特定廠商和產品識別碼的裝置，以避免進行重新導向。您必須以十六進位指定識別碼。您可以在識別碼中使用萬用字元 (*) 以取代個別數字。</p> <p>例如：o:vid-0781_pid-****;vid-0561_pid-554c</p>
viewusb.ExcludePath	<code>{m o}:bus-x1[/y1].../port-z1[;bus-x2[/y2].../port-z2;...]</code>	未定義	<p>使用此選項，可排除位於指定集線器或連接埠路徑上的裝置，以避免進行重新導向。您必須以十六進位指定匯流排和連接埠號碼。您不能在路徑中使用萬用字元。</p> <p>例如：m:bus-1/2/3_port-02;bus-1/1/1/4_port-ff</p>
viewusb.IncludeFamily	<code>{m o}:family_name_1[;family_name_2]...</code>	未定義	<p>設定此選項，可包含能夠重新導向的裝置系列。</p> <p>例如：o:storage; smart-card</p>
viewusb.IncludePath	<code>{m o}:bus-x1[/y1].../port-z1[;bus-x2[/y2].../portz2;...]</code>	未定義	<p>使用此選項，可包含指定集線器或連接埠路徑上能夠重新導向的裝置。您必須以十六進位指定匯流排和連接埠號碼。您不能在路徑中使用萬用字元。</p> <p>例如：m:bus-1/2_port-02;bus-1/7/1/4_port-0f</p>
viewusb.IncludeVidPid	<code>{m o}:vid-xxx1_pid-yyy1[;vid-xxx2_pid-yyy2;...]</code>	未定義	<p>設定此選項，可包含具有指定廠商和產品識別碼且能夠重新導向的裝置。您必須以十六進位指定識別碼。您可以在識別碼中使用萬用字元 (*) 以取代個別數字。</p> <p>例如：o:vid-***_pid-0001;vid-0561_pid-554c</p>
mksVNCServer.useXExtButtonMapping	true 或 false	false	設定此選項以啟用或停用對 SLED 11 SP3 上左手使用滑鼠的支援。
mksvhan.clipboardSize	整數	1024	使用此選項，可指定所要複製及貼上的剪貼簿大小上限。
RemoteDisplay.maxBandwidthKbps	整數	4096000	指定 VMware Blast 工作階段的最大頻寬 (單位為千位元/秒，即 kbps)。頻寬包括所有映像處理、音訊、虛擬通道和 VMware Blast 控制流量。最大值為 4 Gbps (4096000)。
RemoteDisplay.maxFPS	整數	60	指定畫面更新的最大速率。使用此設定，可管理使用者所使用的平均頻寬。有效值應介於 3 與 60 之間。預設值為每秒 60 次更新。
RemoteDisplay.enableStats	true 或 false	false	在 mks 記錄中啟用或停用 Blast 通訊協定統計資料，例如頻寬、FPS、RTT 等。
RemoteDisplay.allowH264	true 或 false	true	設定此選項，可啟用或停用 H.264 編碼。

選項	值/格式	預設值	說明
vdpservice.log.logLevel	fatal error、warn、info、debug 或 trace	info	使用此選項，可設定 vdpservice 的記錄層級。
RemoteDisplay.qpmaxH264	可用值範圍：0-51	36	使用此選項來設定 H264minQP 量化參數，其指定設定為使用 H.264 編碼的遠端顯示的最佳影像畫質。將該值設定為大於針對 RemoteDisplay.qpminH264 設定的值。
RemoteDisplay.qpminH264	可用值範圍：0-51	10	使用此選項來設定 H264maxQP 量化參數，其指定設定為使用 H.264 編碼的遠端顯示的最低影像畫質。將該值設定為小於針對 RemoteDisplay.qpmaxH264 設定的值。
RemoteDisplay.minQualityJPEG	可用值範圍：1-100	25	指定用於 JPEG/PNG 編碼的桌面平台顯示的影像畫質。低畫質設定用於經常變動的畫面區域，例如，在執行捲動時。
RemoteDisplay.midQualityJPEG	可用值範圍：1-100	35	指定用於 JPEG/PNG 編碼的桌面平台顯示的影像畫質。用來設定桌面平台顯示的中等畫質設定。
RemoteDisplay.maxQualityJPEG	可用值範圍：1-100	90	指定用於 JPEG/PNG 編碼的桌面平台顯示的影像畫質。高畫質設定用於較為靜態的畫面區域，可產生較佳影像畫質。

/etc/vmware/viewagent-custom.conf 中的組態選項

Java Standalone Agent 使用組態檔 `/etc/vmware/viewagent-custom.conf`。

表 6-2. /etc/vmware/viewagent-custom.conf 中的組態選項

選項	值	預設值	說明
子網路	NULL 或使用 IP 位址/CIDR 格式的網路位址和遮罩	NULL	<p>如果有多個本機 IP 位址具有不同的子網路，請使用此選項，設定 Linux 代理程式提供給 View 連線伺服器的子網路。</p> <p>在 Linux 代理程式機器上偵測到多個子網路組態時，必須要使用此選項，指定 Linux 代理程式所應使用的正確子網路。例如，如果您已在 Linux 機器上安裝 Docker，它將會導入為虛擬網路介面卡。若要避免 Linux 代理程式以 Docker 作為虛擬網路介面卡，您必須設定此選項，以使用實際的實體網路介面卡。</p> <p>您必須使用 IP 位址/CIDR 格式來指定值。例如，Subnet=192.168.1.0/24。</p> <p>NULL 表示 Linux 代理程式會隨機選取 IP 位址。</p>
SSOEnable	true 或 false	true	設定此選項可啟用/停用單一登入 (SSO)。

選項	值	預設值	說明
SSOUserFormat	文字字串	[username]	<p>使用此選項來指定 Single Sign-On 登入名稱的格式。預設值為僅使用者名稱。如果也需要網域名稱，請設定此選項。一般來說，登入名稱為網域名稱加上特殊字元並接著使用者名稱。如果特殊字元為反斜線，您必須使用另一個反斜線來逸出。登入名稱格式的範例：</p> <ul style="list-style-type: none"> ■ SSOUserFormat=[domain]\\[username] ■ SSOUserFormat=[domain]+[username] ■ SSOUserFormat=[username]@[domain]
CDREnable	true 或 false	true	設定此選項，可啟用或停用用戶端磁碟機重新導向 (CDR) 功能。
USBEnable	true 或 false	true	設定此選項，可啟用或停用 USB 重新導向功能。
KeyboardLayoutSync	true 或 false	true	<p>使用此選項，可指定是否將用戶端的系統地區設定清單和目前的鍵盤配置與 Horizon Agent for Linux 桌面平台同步。當此設定已啟用或未設定時，則允許進行同步化。當此設定已停用時，則不允許進行同步化。</p> <p>此功能僅支援 Windows 版 Horizon Client，並且僅適用於英文、法文、德文、日文、韓文、西班牙文、簡體中文和繁體中文地區設定。</p>
StartBlastServerTime out	整數	20	此選項決定 VMwareBlastServer 處理程序可進行初始化的時間長度 (以秒為單位)。如果處理程序未在這個逾時值內就緒，使用者的登入將會失敗。
SSLCiphers	文字字串	!aNULL:kECDH+AESGCM:ECDH+AESGCM:RSA+AESGCM:kECDH+AES:ECDH+AES:RSA+AES	使用此選項來指定加密清單。您必須使用 https://www.openssl.org/docs/manmaster/man1/ciphers.html 中定義的格式。
SSLProtocols	文字字串	TLSv1_1:TLSv1_2	使用此選項來指定安全性通訊協定。支援的通訊協定為 TLSv1.0、TLSv1.1 和 TLSv1.2。
SSLCipherServerPreference	true 或 false	true	<p>使用此選項來啟用或停用選項 SSL_OP_CIPHER_SERVER_PREFERENCE。如需詳細資訊，請參閱 https://www.openssl.org/docs/manmaster/ssl/SSL_CTX_set_options.html。</p>
UseGnomeFlashback	true 或 false	false	<p>此選項可判斷是否使用安裝在 Ubuntu 14.04 或 Ubuntu 16.04 系統中的 GNOME Flashback (Metacity) 桌面平台環境。不論 SSO 功能是否啟用，此選項皆會生效。</p> <p>將此選項設為 TRUE 之後，系統一律會使用 GNOME Flashback (Metacity) 桌面平台環境，而非預設的桌面平台環境。</p> <p>提示 若要改善系統的效能，在您的 Ubuntu 14.04 或 Ubuntu 16.04 系統上安裝 GNOME Flashback (Metacity) 桌面平台之後，請設定 UseGnomeFlashback=TRUE。</p>

選項	值	預設值	說明
LogCnt	整數	-1	使用此選項以設定 /tmp/vmware-root 中保留的記錄檔計數。 <ul style="list-style-type: none"> ■ -1 - 全部保留 ■ 0 - 全部刪除 ■ > 0 - 保留的記錄檔計數。
RunOnceScript			使用此選項，可將複製的虛擬機器重新加入 AD。 設定在主機名稱有所變更後執行一次的指令碼。指定的指令碼只會在第一次主機名稱變更後執行一次。當代理程式服務啟動，且主機名稱在代理程式安裝後有所變更時，即會以根權限執行指令碼。 以 Winbind 解決方案為例，您必須將基礎虛擬機器加入含有 Winbind 的 AD，並將此選項設定為指令碼路徑。其中必須包含網域重新加入命令 /usr/bin/net ads join -U <ADUserName>%<ADUserPassword>。在虛擬機器複製之後，作業系統自訂會變更主機名稱。當代理程式服務啟動時，指令碼即會執行，而將複製的虛擬機器加入 AD。
RunOnceScriptTimeout		120	使用此選項來為 RunOnceScript 選項設定逾時時間 (秒)。 例如，設定 RunOnceScriptTimeout=120

備註 SSLCiphers、SSLProtocols 和 SSLCipherServerPreference 這三個安全性選項是用於 VMwareBlastServer 處理程序。啟動 VMwareBlastServer 處理程序時，Java Standalone Agent 會將這些選項傳入為參數。啟用 Blast 安全閘道 (BSG) 時，這些選項會影響 BSG 與 Linux 桌面平台之間的連線。停用 BSG 時，這些選項會影響用戶端與 Linux 桌面平台之間的連線。

Linux 桌面平台的 Blast 設定範例

您可以調整遠端桌面平台顯示的影像畫質，以改善使用者體驗。當網路連線不佳時，改善影像畫質有助於保持一致的使用者體驗。

VMware Blast Extreme 通訊協定設定範例

VMwareBlastServer 和其相關外掛程式使用組態檔 /etc/vmware/config。

表 6-3. /etc/vmware/config 中的 Blast 組態選項範例

選項名稱	參數	高速 LAN	LAN	專用 WAN	寬頻 WAN	低速 WAN	極低速
頻寬設定	RemoteDisplay.maxBandwidthKbps	1000000 (1 Gbps)	1000000 (1 Gbps)	1000000 (1 Gbps)	5000 (5 Mbps)	2000 (2 Mbps)	1000 (1 Mbps)
最大 FPS	RemoteDisplay.maxFPS	60	30	30	20	15	5
音訊播放	RemoteDisplay.allowAudio	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE
顯示品質 (JPEG/PNG)	RemoteDisplay.maxQualityJPEG	90	90	90	70	60	50

選項名稱	參數	高速 LAN	LAN	專用 WAN	寬頻 WAN	低速 WAN	極低速
顯示品質 (JPEG/PNG)	RemoteDisplay.midQualityJPEG	35	35	35	35	35	35
顯示品質 (JPEG/PNG)	RemoteDisplay.minQualityJPEG	25	25	25	20	20	20
顯示品質 (H.264)	RemoteDisplay.qpmaxH264	28	36	36	36	36	42
顯示品質 (H.264)	RemoteDisplay.qpminH264	10	10	10	10	10	10

Linux 桌面平台用戶端磁碟機重新導向選項的範例

設定用戶端磁碟機重新導向 (CDR) 選項，以判斷是否可從遠端 Linux 桌面平台存取本機系統的共用資料夾和磁碟機。

將項目新增至 `/etc/vmware/config` 檔案來進行 CDR 設定。

下列組態範例會共用 `C:\ebooks` 和 `C:\spreadsheets` 資料夾、將這兩個資料夾設為唯讀，並防止用戶端共用更多資料夾。

```
cdserver.forcedByAdmin=true
cdserver.sharedFolders=d:\ebooks,;c:\spreadsheets,
cdserver.permissions=R
```

在先前的範例中，逗號 "," 放置在 **ebooks** 之後，而 **spreadsheets** 對於正確選項剖析則是強制使用。

包含在 `cdserver.sharedFolders` 選項中的任何 "R" 會影響該設定中所列出的所有資料夾。在下列範例中，即便 R 僅僅放在 `/home/jsmith` 資料夾路徑之後，**ebooks** 和 **spreadsheets** 資料夾仍皆為唯讀。

```
cdserver.sharedFolders=d:\ebooks,;c:\spreadsheets,;/home/jsmith,R
```

隱藏 Linux 桌面平台的 vSphere 主控台顯示

當使用者連線至 Linux 桌面平台時，也可以在 Linux 虛擬機器的 vSphere 主控台中顯示桌面平台。您可以設定 Linux 虛擬機器，以確保當使用者連線至其桌面平台時，vSphere 主控台是空白的。

程序

- ◆ 在 ESXi 主機上，新增下列行至 Linux 虛擬機器的 `vmx` 檔案。

```
RemoteDisplay.maxConnections = "0"
```

vSphere 主控台顯示仍保持空白，即使在使用者登出桌面平台時，您已連線至虛擬機器也是如此。

建立和管理 Linux 桌面平台集區

若要設定 Linux 虛擬機器以用作遠端桌面平台，您必須使用 Linux 虛擬機器來建立桌面平台集區。

Horizon for Linux 支援下列桌面平台集區類型：

- 具有 vCenter 虛擬機器的手動桌面平台集區
- 自動完整複製桌面平台集區

若要使用 vCenter 虛擬機器來建立手動桌面平台集區，您必須在所有虛擬機器上安裝 Horizon Agent。然後使用連線伺服器桌面平台集區建立精靈，將虛擬機器新增至桌面平台集區。若要複製大量虛擬機器，請參閱 [Linux 桌面平台大量部署的概觀](#)。

若要建立自動完整複製桌面平台集區，您必須在 Linux 虛擬機器範本上安裝 Horizon Agent。然後使用連線伺服器桌面平台集區建立精靈來複製完整的虛擬機器。

本章節討論下列主題：

- [建立 Linux 的手動桌面平台集區](#)
- [管理 Linux 的桌面平台集區](#)
- [為 Linux 建立自動完整複製桌面平台集區](#)
- [代理 PowerCLI 命令](#)

建立 Linux 的手動桌面平台集區

您可以為 Linux 虛擬機器建立手動桌面平台集區。

必要條件

- 驗證 Linux 客體作業系統上已安裝 Horizon Agent。請參閱 [在 Linux 虛擬機器上安裝 Horizon Agent](#)。
- 驗證 vCenter Server 已新增至 Horizon 連線伺服器。

程序

- 1 在 View Administrator 中，新增手動桌面平台集區。
選取類別目錄 > 桌面平台集區 > 新增。
- 2 選取手動桌面平台集區。
- 3 在 [使用者指派] 頁面上，為桌面平台集區中的機器選取專用或浮動使用者指派，然後按下一步。
- 4 在 [機器來源] 頁面上，選取 **vCenter 虛擬機器**，然後按下一步。

- 5 在 [vCenter Server] 頁面上，選取適當的 vCenter Server，然後按下一步。
- 6 輸入桌面平台集區識別碼。
- 7 在 [桌面平台集區設定] 頁面上，設定下列選項。

選項	說明
預設顯示通訊協定	VMware Blast
允許使用者選擇通訊協定	否
3D 轉譯器	使用 vSphere Client 管理 (適用於 2D、vSGA 或 vDGA 桌面平台)，針對 vGPU 桌面平台則使用 NVIDIA GRID vGPU

備註 集區設定是必要的。若未設定，您可能無法連線到桌面平台，而會遇到通訊協定錯誤或黑色畫面。

- 8 在 [新增 vCenter 虛擬機器] 頁面上，選取 Linux 虛擬機器。按一下**新增**，然後按下一步。

備註 請不要將 Windows 和 Linux 虛擬機器建立在相同的桌面平台集區中。

- 9 依照提示完成程序。

授權使用者使用桌面平台集區中的機器。在 **View Administrator** 中，選取桌面平台集區、選取**權利 > 新增權利**，並新增使用者或群組。

Linux 虛擬機器已準備就緒，可在 Horizon 7 部署中作為遠端桌面平台。

管理 Linux 的桌面平台集區

當您建立手動桌面平台集區並將 Linux 機器新增至集區時，您可以藉由設定來管理手動桌面平台集區。您必須只將 Linux 客體作業系統新增至手動桌面平台集區。如果集區同時包含 Windows 和 Linux 客體作業系統，則會將集區視為 Windows 集區，而您將無法連線到 Linux 桌面平台。

管理作業的支援

- 停用或啟用桌面平台集區
- 複製自動桌面平台集區
- 刪除桌面平台集區

您可以從 **View Manager** 中移除虛擬機器，或是從磁碟中刪除虛擬機器

遠端設定的支援

表 7-1. 遠端設定

遠端設定	選項
遠端機器電源原則	<ul style="list-style-type: none"> ■ 不執行電源動作 ■ 確認機器電源永遠開啟 ■ 暫止 ■ 關閉電源
中斷連線後自動登出	<ul style="list-style-type: none"> ■ 立即 ■ 永不 ■ 在 n 分鐘後
允許使用者重設/重新啟動其機器	<ul style="list-style-type: none"> ■ 是 ■ 否
允許使用者從不同的用戶端裝置起始單獨的工作階段	<ul style="list-style-type: none"> ■ 是 ■ 否
「登出後刪除機器」，用於具有完整複製和浮動的自動桌面平台集區	<ul style="list-style-type: none"> ■ 是 ■ 否

View Administrator 作業的支援

- 中斷工作階段連線
- 登出工作階段
- 重設/重新啟動桌面平台
- 傳送訊息

對於專用桌面平台集區，您可以新增或移除每個虛擬機器的使用者指派。對於大量作業，則必須使用 **View PowerCLI Cmdlet**。

- Update-UserOwnership
- Remove-UserOwnership

備註 請勿變更遠端顯示通訊協定設定。此設定必須一律與桌面平台集區建立相同。

設定	選項
預設顯示通訊協定	VMware Blast
允許使用者選擇通訊協定	否
3D 轉譯器	<ul style="list-style-type: none"> ■ 針對 2D、vSGA 或 vDGA，使用 vSphere Client 進行管理 ■ NVIDIA GRID vGPU

如需詳細資訊，請參閱《View 管理》指南。

為 Linux 建立自動完整複製桌面平台集區

您可以為 Linux 虛擬機器建立自動完整複製桌面平台集區。建立自動完整複製桌面平台集區之後，您可以使用 Linux 虛擬機器作為 Horizon 7 部署中的遠端桌面平台。

必要條件

- 驗證 Linux 客體作業系統上已安裝 Horizon Agent。請參閱 [在 Linux 虛擬機器上安裝 Horizon Agent](#)。
- 如果您使用 Winbind 解決方案來將 Linux 虛擬機器加入 Active Directory，您必須在虛擬機器範本中完成 Winbind 解決方案的設定。
- 如果您使用 Winbind 解決方案，則必須在虛擬機器上執行網域加入命令。在 Shell 指令碼中併入命令，並指定 `/etc/vmware/viewagent-custom.conf` 中 Horizon Agent 選項 `RunOnceScript` 的指令碼路徑。如需詳細資訊，請參閱 [在 Linux 桌面平台上設定組態檔中的選項](#)。
- 驗證 vCenter Server 已新增至 Horizon 連線伺服器。

程序

1 建立客體自訂規格。

請參閱 vSphere 虛擬機器管理文件中的「在 vSphere Web Client 中建立 Linux 的自訂規格」。建立規格時，請確定您正確指定下列設定。

設定	值
目標虛擬機器作業系統	Linux
電腦名稱	使用虛擬機器名稱。
網域	指定 View 環境的網域。
網路設定	使用標準網路設定。
主要 DNS	指定有效位址。

備註 如需客體作業系統自訂支援對照表的詳細資訊，請參閱 <http://partnerweb.vmware.com/programs/guestOS/guest-os-customization-matrix.pdf>。

- 2 在 Horizon Administrator 中，選取**類別目錄 > 桌面平台集區 > 新增**。
- 3 選取**自動桌面平台集區**，然後按**下一步**。
- 4 為桌面平台集區中的機器選取專用或浮動使用者指派，然後按**下一步**。
- 5 在 [vCenter Server] 頁面上，選取**完整的虛擬機器**並選取適當的 vCenter 伺服器，然後按**下一步**。
- 6 在 [桌面平台集區識別] 頁面上，輸入桌面平台集區識別碼，然後按**下一步**。

- 7 在 [桌面平台集區設定] 頁面上，設定下列選項，然後按下一步。

選項	說明
預設顯示通訊協定	VMware Blast
允許使用者選擇通訊協定	否
3D 轉譯器	使用 vSphere Client 管理 (適用於 2D、vSGA 或 vDGA 桌面平台)，針對 vGPU 桌面平台則使用 NVIDIA GRID vGPU

備註 集區設定是必要的。若未設定，您可能無法連線到桌面平台，而會遇到通訊協定錯誤或黑色畫面。

- 8 在 [佈建設定] 頁面上，設定**虛擬機器命名**選項，然後按下一步。

選項	說明
手動指定名稱	手動輸入名稱。
使用命名模式	例如，指定 LinuxVM-{n}。 您也必須指定下列桌面平台集區大小調整選項： <ul style="list-style-type: none"> ■ 機器數目上限 ■ 備用、電源開啟機器數目

- 9 在 [儲存空間最佳化] 頁面上，選取儲存空間管理原則，然後按下一步。

- 10 在 [vCenter 設定] 頁面上，您必須按一下 [瀏覽] 並按順序選取 vCenter Server 設定，然後按下一步。

您無法略過 vCenter Server 設定：

- a 範本
- b 虛擬機器資料夾位置
- c 主機或叢集
- d 資源集區
- e 資料存放區

- 11 在 [進階儲存選項] 頁面上，選取適當的儲存選項，然後按下一步。

- 12 在 [客體自訂] 頁面上，選取針對 Linux 的客體自訂，然後按下一步。

- 13 在 [即將完成] 頁面上，檢閱詳細資料並選取**此精靈完成後賦予使用者權利**。

- 14 按一下**完成**。

- 15 若要授權使用者使用桌面平台集區中的機器，請選取桌面平台集區，然後按一下**權利 > 新增權利**，並新增使用者和群組。

- 16 等候直到桌面平台集區中的所有 Linux 虛擬機器都變得可用。

代理 PowerCLI 命令

會在連線伺服器和 Windows 桌面平台上執行各種管理工作的 View PowerCLI Cmdlet，也可用於 Linux 桌面平台。

建立手動桌面平台集區

```
Add-ManualPool -DefaultProtocol Blast -AllowProtocolOverride $false -threadRender usevc|vgpu -
Pool_id <pool id> [more parameters]
```

對於 Linux 桌面平台，必須要使用下列選項和值。

- DefaultProtocol Blast
- AllowProtocolOverride \$false
- threadRender usevc|vgpu。對於 vGPU 桌面平台，請使用 -threadRender vgpu，2D/vSGA/vDGA 桌面平台則使用 -threadRender usevc。

範例

- 建立名為 LinuxDesktop、具有虛擬機器 LinuxVM-01 的浮動 Linux 桌面平台集區。

```
Add-ManualPool -DefaultProtocol Blast -AllowProtocolOverride $false -threadRender usevc -Pool_id
LinuxDesktop -Id (Get-DesktopVM -Name LinuxVM-01).id -Persistence NonPersistent -Vc_name
myvc.myorg.org
```

- 建立名為 LinuxDesktop 的專用 Linux vGPU 桌面平台集區，所有虛擬機器皆以 LinuxVM- 做為虛擬機器名稱的開頭。

```
Get-DesktopVM | Where-Object {$_.Name.StartsWith("LinuxVM-")} | Add-ManualPool -DefaultProtocol
Blast -AllowProtocolOverride $false -Persistence Persistent -threadRender vgpu -Pool_id
LinuxDesktop
```

- 建立具有第一個 RHEL 6 x64 虛擬機器的浮動 Linux 桌面平台集區 LinuxDesktop。

```
Get-DesktopVM | Where-Object {$_.GuestID -eq "rhel6_64Guest"} | Select-Object -Index 0 | Add-
ManualPool -DefaultProtocol Blast -AllowProtocolOverride $false -Persistence NonPersistent -
threadRender usevc -Pool_id LinuxDesktop
```

建立完整複製自動桌面平台集區

```
Add-AutomaticPool -DefaultProtocol Blast -AllowProtocolOverride $false -threadRender usevc|vgpu `
-Pool_id <pool id> -Vc_id <vCenter id> `
-NamePrefix <VM Name Prefix>" `
-templatePath <Virtual Machine Template Path> `
-VmFolderPath <Virtual Machine Folder Path> `
-ResourcePoolPath <Resource Pool Path> `
-dataStorePaths <Datastore Path> `
-customizationSpecName <Customization Specification Name> `
[more parameters]
```

對於 Linux 桌面平台，必須要使用下列選項和值。

- DefaultProtocol Blast
- AllowProtocolOverride \$false
- threedRender usevc|vgpu。對於 vGPU 桌面平台，請使用 -threedRender vgpu，2D/vSGA 桌面平台則使用 -threedRender usevc。

範例

```
Add-AutomaticPool -DefaultProtocol Blast -AllowProtocolOverride $false -threedrender usevc`
-pool_id FullClone-Linux `
-Vc_id (Get-ViewVC -serverName myvc.myorg.org).vc_id `
-NamePrefix "FullClone-{n:fixed=3}" `
-Persistence NonPersistent -deletePolicy DeleteOnUse `
-VmFolderPath "/LinuxVDI/vm/FullClone" `
-ResourcePoolPath "/LinuxVDI/host/LinuxVDICluster/Resources" `
-templatePath "/LinuxVDI/vm/LinuxTemplate" `
-dataStorePaths "/LinuxVDI/host/LinuxVDICluster/datastore" `
-customizationSpecName "linux-spec" `
-maximumCount 100
```

新增或移除桌面平台集區權利

- 為網域 mydomain.org 的網域使用者群組授與 LinuxDesktop 的權利。

```
Add-PoolEntitlement -Pool_id LinuxDesktop -Sid (Get-User -Name "domain user" -Domain
"mydomain.org").sid
```

- 移除 mydomain.org 網域的網域使用者群組對 LinuxDesktop 的權利。

```
Remove-PoolEntitlement -Pool_id LinuxDesktop -Sid (Get-User -Name "domain user" -Domain
"mydomain.org").sid
```

對專用桌面平台集區中的虛擬機器指派或移除使用者

- 將 myuser 使用者指派給位於專用桌面平台集區中的 LinuxVM-01 虛擬機器。

```
Update-UserOwnership -Machine_id (Get-DesktopVM -Name "LinuxVM-01").machine_id -Sid (Get-User -
Name "myuser" | Where-Object {$_.cn -eq "myuser"}).sid
```

- 從位於專用桌面平台集區中的 LinuxVM-01 虛擬機器移除 myuser 使用者。

```
Remove-UserOwnership -Machine_id (Get-DesktopVM -Name "LinuxVM-01").machine_id
```

登出桌面平台連線

- 從 myuser 的桌面平台工作階段中登出。

```
Get-RemoteSession -Username "mydomain.org\myuser" | Send-SessionLogoff
```

如需代理 PowerCLI Cmdlet 的詳細資訊，請參閱《View 整合》中的〈使用 View PowerCLI〉。

針對手動桌面平台集區大量部署 Horizon 7

8

利用 View Administrator，您可以自動建立 Windows (而非 Linux) 桌面平台機器集區。不過，您可以開發能夠自動化部署 Linux 桌面平台機器集區的指令碼。

提供的範例指令碼僅供說明之用。對於使用範例指令碼時可能產生的問題，VMware 概不負責。

本章節討論下列主題：

- [Linux 桌面平台大量部署的概觀](#)
- [Linux 桌面平台大量升級的概觀](#)
- [建立用於複製 Linux 桌面平台機器的虛擬機器範本](#)
- [用於部署 Linux 桌面平台的範例 PowerCLI 指令碼輸入檔](#)
- [用於複製 Linux 虛擬機器的範例指令碼](#)
- [將複製的虛擬機器加入 AD 網域的範例指令碼](#)
- [使用 SSH 將複製的虛擬機器加入 AD 網域的範例指令碼](#)
- [上傳組態檔至 Linux 虛擬機器的範例指令碼](#)
- [使用 SSH 上傳組態檔至 Linux 虛擬機器的範例指令碼](#)
- [在 Linux 桌面平台機器上升級 Horizon Agent 的範例指令碼](#)
- [使用 SSH 在 Linux 虛擬機器上升級 Horizon Agent 的範例指令碼](#)
- [在 Linux 虛擬機器上執行作業的範例指令碼](#)

Linux 桌面平台大量部署的概觀

要部署 Linux 的手動桌面平台，必須執行數個步驟。如果計劃部署多個桌面平台，您可以使用 PowerCLI 指令碼將一些步驟自動化。

對於部分作業，您可以選擇讓 PowerCLI 或 SSH 在 Linux 機器上執行命令。下表說明這兩個方法之間的差異。

PowerCLI	SSH
不需安裝額外工具。	<ul style="list-style-type: none"> ■ 針對 Ubuntu，您必須使用 <code>sudo apt-get install openssh-server</code> 命令安裝 SSH 伺服器。針對 RHEL 和 CentOS，預設會安裝 <code>openssh-server</code>，但您必須確保防火牆設定可允許 <code>ssh</code>。 ■ 必須下載 SSH 用戶端應用程式 <code>pscp.exe</code> 和 <code>plink.exe</code>，並將它們放在與 PowerCLI 指令碼相同的資料夾中。
上傳檔案和命令執行較緩慢。	上傳檔案和命令執行較快速。
必須提供 ESXi 主機的管理員認證。	不須提供 ESXi 主機的管理員認證。
無法處理執行指令碼來安裝 Horizon Agent 時管理員密碼中的特殊字元，或執行指令碼來加入網域時 AD 使用者密碼中的特殊字元。	可以處理執行指令碼來安裝 Horizon Agent 時管理員密碼中的特殊字元，或執行指令碼來加入網域時 AD 使用者密碼中的特殊字元。

備註 PowerCLI 式和 SSH 式指令碼均可處理 vCenter Server 管理員和 Linux 管理員密碼中的特殊字元。PowerCLI 式指令碼也可以處理 ESXi 主機管理員密碼中的特殊字元。在上述所有情況下，不需使用逸出字元。

如需 vSphere PowerCLI 的詳細資訊，請參閱 <https://www.vmware.com/support/developer/PowerCLI>。

大量部署一組 Linux 桌面平台的程序涉及下列步驟：

- 1 建立虛擬機器範本，然後在虛擬機器上安裝 Horizon Agent。

請參閱 [建立用於複製 Linux 桌面平台機器的虛擬機器範本](#)。

- 2 建立客體自訂規格。

請參閱 vSphere 虛擬機器管理文件中的「在 vSphere Web Client 中建立 Linux 的自訂規格」。建立規格時，請確定您正確指定下列設定。

設定	值
目標虛擬機器作業系統	Linux
電腦名稱	使用虛擬機器名稱。
網域	指定 View 環境的網域。
網路設定	使用標準網路設定。
主要 DNS	指定有效位址。

備註 如需客體作業系統自訂支援對照表的詳細資訊，請參閱 <http://partnerweb.vmware.com/programs/guestOS/guest-os-customization-matrix.pdf>。

- 3 複製虛擬機器。

請參閱 [用於複製 Linux 虛擬機器的範例指令碼](#)。

- 4 如果您使用 Winbind 解決方案，請將複製的虛擬機器加入 Active Directory (AD) 網域中。您可以使用下面的範例指令碼來執行網域加入命令，或使用 `/etc/vmware/viewagent-custom.conf` 中的選項 `RunOnceScript` (設定於範本虛擬機器中) 來執行。

請參閱[將複製的虛擬機器加入 AD 網域的範例指令碼](#)或使用 [SSH](#) 將複製的虛擬機器加入 AD 網域的範例指令碼。

5 更新虛擬機器中的組態選項。

請參閱[上傳組態檔至 Linux 虛擬機器的範例指令碼](#)或使用 [SSH](#) 上傳組態檔至 Linux 虛擬機器的範例指令碼。

6 建立桌面集區。

請參閱 [建立 Linux 的手動桌面平台集區](#)。

如需執行開機、關機、重新啟動或刪除虛擬機器的範例指令碼，請參閱在 [Linux 虛擬機器上執行作業的範例指令碼](#)。此指令碼可從 vCenter Server 刪除虛擬機器。

Linux 桌面平台大量升級的概觀

要大量升級 Linux 的手動桌面平台，必須執行數個步驟。您可以使用 PowerCLI 指令碼將部分步驟自動化。

大量升級未受管理的桌面平台

若要將未受管理的虛擬機器大量升級為受管理或未受管理的虛擬機器，您必須使用範例升級指令碼將新的 Horizon Agent 上傳至現有的虛擬機器，然後執行升級命令。

- 如果您保留未受管理的虛擬機器，則可重複使用您現有的桌面平台集區。
- 如果您從未受管理的虛擬機器升級為受管理的虛擬機器，您必須刪除現有的桌面平台集區，然後建立新的桌面平台集區。如需詳細資訊，請參閱 [升級 Linux 虛擬機器上的 Horizon Agent](#)。

大量升級受管理的桌面平台

若要大量升級受管理的虛擬機器，請選取下列其中一個方法。

方法	說明
在範本虛擬機器中，安裝或升級新的 Horizon Agent，並建立快照。	<ul style="list-style-type: none"> ■ 現有的虛擬機器已刪除，因此使用者資料和設定檔將會遺失，除非使用者資料和設定檔位於共用伺服器上，例如 NFS 伺服器。 ■ 完成虛擬機器取代後，View Administrator 上的虛擬機器狀態可能會是「遺失」。您必須重新啟動代理服務加以修正。 ■ 如果您使用連結複製，此方法將可避免每個虛擬機器上出現重複資料。
使用升級的範例指令碼將新的 Horizon Agent 上傳至現有的虛擬機器，然後執行升級命令。	<ul style="list-style-type: none"> ■ 使用者資料和設定檔會保留下來。 ■ 如果您使用連結複製，此方法將導致每個虛擬機器上出現重複資料。

建立用於複製 Linux 桌面平台機器的虛擬機器範本

執行虛擬機器複製之前，您必須建立複製所依據的虛擬機器範本。

必要條件

- 驗證您的部署符合支援 Linux 桌面平台的需求。請參閱 [Horizon 7 for Linux 的系統需求](#)。

- 熟悉在 vCenter Server 中建立虛擬機器和安裝客體作業系統的步驟。請參閱 在 Horizon 7 中設定虛擬桌面平台 文件中的「建立和準備虛擬機器」。
- 熟悉您將與虛擬機器搭配使用的顯示器建議的視訊記憶體 (vRAM) 值。請參閱 [Horizon 7 for Linux 的系統需求](#)。
- 請熟悉 AD 整合的步驟。請參閱第 3 章 設定 Linux 桌面平台的 Active Directory 整合。
- 熟悉在 Linux 上安裝 Horizon Agent 的步驟。請參閱〈安裝 Horizon Agent 和管理 Linux 桌面平台〉一章
- 如果需要，請熟悉使用 View 組態檔設定選項的步驟。請參閱第 6 章 Linux 桌面平台的組態選項。
- 若您計劃安裝圖形卡，請熟悉相關步驟。請參閱第 4 章 設定 Linux 桌面平台的圖形卡。

程序

- 1 在 vSphere Web Client 或 vSphere Client 中，建立新虛擬機器。
- 2 設定自訂組態選項。
 - a 在虛擬機器上按一下滑鼠右鍵，然後按一下**編輯設定**。
 - b 指定 vCPU 和 vMemory 大小的數量。

如需建議的值，請遵循 Linux 發行版適用的安裝指南中的指示。

例如，Ubuntu 12.04 建議對 vMemory 設定 2048 MB 和 2 個 vCPU。
 - c 選取**視訊卡**，並指定顯示器數量和總視訊記憶體 (vRAM)。

針對使用 2D 或 vSGA 的虛擬機器 (使用 VMware 驅動程式)，在 vSphere Web Client 中設定 vRAM 大小。vRAM 大小在 vDGA 或 NVIDIA GRID vGPU 機器 (使用 NVIDIA 驅動程式) 上沒有作用。

如需建議的值，請遵循 Horizon 7 for Linux 的系統需求。請勿使用視訊記憶體計算機。
- 3 將虛擬機器開機並安裝 Linux 發行版。
- 4 建立具備 root 權限的使用者，例如：ViewUser。這個使用者僅用來安裝和解除安裝 Horizon Agent。
- 5 編輯 /etc/sudoers 和新增行 ViewUser ALL=(ALL) NOPASSWD:ALL。

透過 /etc/sudoers 中的這行，不需要密碼即可以 ViewUser 的身分執行 sudo。執行本章節中提供的範例指令碼來安裝 Horizon Agent 時，您會指定 ViewUser 作為輸入。
- 6 如果 Linux 發行版為 RHEL、CentOS 或 NeoKylin，請編輯 /etc/sudoers 並註解排除下列行：

```
Defaults requiretty
Defaults !visiblepw
```
- 7 如果 Linux 發行版不是 RHEL 7、CentOS 7、SLED 12 或 SLES 12，請安裝 VMware Tools。

RHEL 7、CentOS 7、SLED 12 和 SLES 12 依預設會安裝 Open VM Tools。
- 8 如果 Linux 發行版為 RHEL 7、CentOS 7 或 SLES 12，請安裝 deployPkg 外掛程式。

指示位於 <http://kb.vmware.com/kb/2075048>。

- 9 針對 RHEL 和 CentOS，請啟用 [網路連線] 設定自動連線。
- 10 執行 AD 整合工作。
- 11 執行安裝圖形卡的步驟。
- 12 安裝 Horizon Agent

```
sudo ./install_viewagent.sh -A yes
```

- 13 使用 View 組態檔執行其他組態。
- 14 關閉虛擬機器並建立快照。

用於部署 Linux 桌面平台的範例 PowerCLI 指令碼輸入檔

部署 Linux 桌面平台的範例 PowerCLI 指令碼會讀取包含桌面平台機器相關資訊的輸入檔。

輸入檔的類型為 csv，並且包含下列資訊：

- 桌面平台虛擬機器名稱
- 父虛擬機器名稱
- 客體自訂規格
- 複製的桌面平台機器所在的資料存放區
- 管理桌面平台機器的 ESXi 伺服器
- 用於複製的父虛擬機器快照
- 指出在桌面平台虛擬機器存在時是否加以刪除的旗標

下列範例顯示輸入檔可能包含的內容。

```
VMName,Parentvm,CustomSpec,Datastore,Host,FromSnapshot,DeleteIfPresent  
linux-001,Ubuntu1204x64,linuxagent,datastore1,10.117.44.172,snapshot1,TRUE  
linux-002,Ubuntu1204x64,linuxagent,datastore1,10.117.44.172,snapshot1,TRUE  
linux-003,Ubuntu1204x64,linuxagent,datastore1,10.117.44.172,snapshot1,TRUE  
linux-004,Ubuntu1204x64,linuxagent,datastore1,10.117.44.172,snapshot1,TRUE  
linux-005,Ubuntu1204x64,linuxagent,datastore1,10.117.44.172,snapshot1,TRUE
```

範例指令碼假設此輸入檔的名稱是 CloneVMs.csv，而檔案位在與指令碼相同的資料夾中。

用於複製 Linux 虛擬機器的範例指令碼

您可以自訂和使用下列範例指令碼，以複製任意數量的虛擬機器 (VM)。

若要複製和貼上不含分頁符號的指令碼內容，請使用本主題的 HTML 版本，您可以從 Horizon7 文件頁面取得，網址是：https://www.vmware.com/support/pubs/view_pubs.html。

指令碼輸入

此指令碼會讀取一個輸入檔，在[用於部署 Linux 桌面平台的範例 PowerCLI 指令碼輸入檔](#)中有加以描述。此指令碼也會互動地要求下列資訊：

- vCenter Server 的 IP 位址
- vCenter Server 的管理員登入名稱
- vCenter Server 的管理員密碼
- 複製類型，可以是連結式或完整
- 是否停用 vSphere 虛擬機器主控台

指令碼內容

```
<#
Create Clones from a Master VM

The Tool supports creation of Full clone and linked clone from Master VM.
The parent VM is required for the linked-clone to work and the parent VMs file cannot be renamed or
moved.
#>
#----- Functions -----
function GetInput
{
    Param($prompt, $IsPassword = $false)
    $prompt = $prompt + ": "
    Write-Host $prompt -NoNewLine
    [Console]::ForegroundColor = "Blue"
    if ($IsPassword)
    {
        $input = Read-Host -AsSecureString
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBS
TR($input))
    }
    else
    {
        $input = Read-Host
    }

    [Console]::ResetColor()
    return $input
}

function IsVMExists ()
{
    Param($VMExists)
    Write-Host "Checking if the VM $VMExists already Exists"
    [bool]$Exists = $false

    #Get all VMS and check if the VMs is already present in VC
    $listvm = Get-vm
```

```

    foreach ($lvm in $listvm)
    {
        if($VMExists -eq $lvm.Name )
        {
            $Exists = $true
        }
    }
    return $Exists
}
function Disable_VM_Console()
{
    Param($VMToDisableConsole)
    $vmConfigSpec = New-Object VMware.Vim.VirtualMachineConfigSpec
    $extra = New-Object VMware.Vim.optionvalue
    $extra.Key="RemoteDisplay.maxConnections"
    $extra.Value="0"
    $vmConfigSpec.extraconfig += $extra
    $vm = Get-VM $VMToDisableConsole | Get-View
    $vm.ReconfigVM($vmConfigSpec)
}

function Delete_VM()
{
    Param($VMToDelete)
    Write-Host "Deleting VM $VMToDelete"
    Get-VM $VMToDelete | where { $_.PowerState -eq "PoweredOn" } | Stop-VM -confirm:$false
    Get-VM $VMToDelete | Remove-VM -DeleteFromDisk -confirm:$false
}

#----- Main Script -----

SvcAddress = GetInput -prompt "Your vCenter address" -IsPassword $false
SvcAdmin = GetInput -prompt "Your vCenter admin user name" -IsPassword $false
SvcPassword = GetInput -prompt "Your vCenter admin user password" -IsPassword $true
$cloneType = GetInput -prompt 'Clone Type ("linked" or "full")' -IsPassword $false
$disableVMConsole = GetInput -prompt 'Disable vSphere VM Console ("yes" or "no", recommend "yes")' -
IsPassword $false
"-----"
$csvFile = '.\CloneVMs.csv'

# Check that user passed only linked or full clone
if (($CloneType.length > 0) -and ($CloneType -ne "linked" -or $CloneType -ne "full"))
{
    write-host -ForegroundColor Red "Clone type supports only 'linked' or 'full' (case sensitive)"
    exit
}
if (($disableVMConsole.length > 0) -and ($disableVMConsole -ne "yes" -or $disableVMConsole -ne "no"))
{
    write-host -ForegroundColor Red "Disable vSphere VM Console supports only 'yes' or 'no' (case
sensitive)"
    exit
}

#check if file exists

```

```

if (!(Test-Path $csvFile))
{
    write-host -ForegroundColor Red "CSV File $CSVFile not found"
    exit
}

# Connect to the VC (Parameterize VC)
#Connect to vCenter
$VC_Conn_State = Connect-VIServer $vcAddress -user $vcAdmin -password $vcPassword
if([string]::IsNullOrEmpty($VC_Conn_State))
{
    Write-Host 'Exit since failed to login vCenter'
    exit
}
else
{
    Write-Host 'vCenter is connected'
}

#Read input CSV file
$csvData = Import-CSV $csvFile
#$csvData = Import-CSV $csvFile -
header("VMName","Parentvm","CustomSpec","Datastore","Host","FromSnapshot","DeleteIfPresent")
foreach ($line in $csvData)
{
    "`n-----"
    $VMName = $line.VMName
    write-host -ForegroundColor Yellow "VM: $VMName`n"

    $destVMName=$line.VMName
    $srcVM = $line.Parentvm
    $cSpec = $line.CustomSpec
    $targetDSName = $line.Datastore
    $destHost = $line.Host
    $srcSnapshot = $line.FromSnapshot
    $deleteExisting = $line.DeleteIfPresent
    if (IsVMExists ($destVMName))
    {
        Write-Host "VM $destVMName Already Exists in VC $vcAddress"
        if($deleteExisting -eq "TRUE")
        {
            Delete_VM ($destVMName)
        }
        else
        {
            Write-Host "Skip clone for $destVMName"
            continue
        }
    }
    $vm = get-vm $srcvm -ErrorAction Stop | get-view -ErrorAction Stop
    $cloneSpec = new-object VMware.VIM.VirtualMachineCloneSpec
    $cloneSpec.Location = new-object VMware.VIM.VirtualMachineRelocateSpec
    if ($CloneType -eq "linked")
    {
        $cloneSpec.Location.DiskMoveType =

```

```
[VMware.VIM.VirtualMachineRelocateDiskMoveOptions]::createNewChildDiskBacking
}
Write-Host "Using Datastore $targetDSName"
$newDS = Get-Datastore $targetDSName | Get-View
$CloneSpec.Location.Datastore = $newDS.summary.Datastore
Set-VM -vm $srcVM -snapshot (Get-Snapshot -vm $srcVM -Name $srcSnapshot) -confirm:$false
$cloneSpec.Snapshot = $vm.Snapshot.CurrentSnapshot
$cloneSpec.Location.Host = (get-vmhost -Name $destHost).Extensiondata.MoRef
$CloneSpec.Location.Pool = (Get-ResourcePool -Name Resources -Location (Get-VMHost -Name
$destHost)).Extensiondata.MoRef
# Start the Clone task using the above parameters
$task = $vm.CloneVM_Task($vm.parent, $destVMName, $cloneSpec)
# Get the task object
$task = Get-Task | where { $_.id -eq $task }
#Wait for the taks to Complete
Wait-Task -Task $task

$newvm = Get-vm $destVMName
$customSpec = Get-OSCustomizationSpec $cSpec
Set-vm -OSCustomizationSpec $cSpec -vm $newvm -confirm:$false
if ($disableVMConsole -eq "yes")
{
    Disable_VM_Console($destVMName)
}
# Start the VM
Start-VM $newvm
}
Disconnect-VIServer $vcAddress -Confirm:$false
exit
```

指令碼執行

下列訊息是來自指令碼的執行：

```
PowerCLI C:\scripts> .\CloneVMs.ps1
Your vCenter address: 10.117.44.17
Your vCenter admin user name: administrator
Your vCenter admin user password: *****
Clone Type<"linked" or "Full"> : linked
Disable vSphere VM Console ("yes" or "no", recommend "yes") : yes
```

複製程序耗費的時間視桌面平台機器的數量而定，並且範圍從數分鐘到幾個小時不等。若要驗證程序已完成，請從 **vSphere Client**，確定已將最新的桌面平台虛擬機器開機、具有自己的唯一主機名稱，而且 **VMware Tools** 正在執行。

將複製的虛擬機器加入 AD 網域的範例指令碼

您可以自訂和使用下列範例指令碼，以將複製的虛擬機器 (VM) 加入 **Active Directory (AD)** 網域。

如果您對 **AD** 整合使用 **Winbind** 解決方案，則必須執行此指令碼，因為加入網域的步驟對複製的虛擬機器將會失敗。此指令碼會在每個虛擬機器上執行命令來加入網域。如果您使用 **OpenLDAP** 解決方案，則不需要執行此指令碼。

若要複製和貼上不含分頁符號的指令碼內容，請使用本主題的 HTML 版本，您可以從 Horizon7 文件頁面取得，網址是：https://www.vmware.com/support/pubs/view_pubs.html。

指令碼輸入

此指令碼會讀取一個輸入檔，在用於部署 Linux 桌面平台的範例 **PowerCLI 指令碼輸入檔** 中有加以描述。此指令碼也會互動地要求下列資訊：

- vCenter Server 的 IP 位址
- vCenter Server 的管理員登入名稱
- vCenter Server 的管理員密碼
- ESXi 主機的管理員登入名稱
- ESXi 主機的管理員密碼
- Linux 虛擬機器的使用者登入名稱
- Linux 虛擬機器的使用者密碼
- 獲得授權可將機器加入網域的 AD 使用者的登入名稱
- 獲得授權之 AD 使用者的密碼

指令碼內容

```
<#
.SYNOPSIS
run command "sudo /usr/bin/net ads join"

.DESRIPTION
The tool is to run the command "sudo /usr/bin/net ads join" to join Linux to AD

.NOTES
#>
#----- Functions -----
function GetInput
{
    Param($prompt, $IsPassword = $false)
    $prompt = $prompt + ": "
    Write-Host $prompt -NoNewLine
    [Console]::ForegroundColor = "Blue"
    if ($IsPassword)
    {
        $input = Read-Host -AsSecureString
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBS
TR($input))
    }
    else
    {
        $input = Read-Host
    }
}
```



```

    [Console]::ResetColor()
    return $input
}
#----- Handle input -----
"-----"
SvcAddress = GetInput -prompt "Your vCenter address" -IsPassword $false
SvcAdmin = GetInput -prompt "Your vCenter admin user name" -IsPassword $false
SvcPassword = GetInput -prompt "Your vCenter admin user password" -IsPassword $true
"-----"
$hostAdmin = GetInput -prompt 'Your ESXi host admin user name, such as root' -IsPassword $false
$hostPassword = GetInput -prompt "Your ESXi admin user password" -IsPassword $true
"-----"
$guestUser = GetInput -prompt 'Your VM guest OS user name' -IsPassword $false
$guestPassword = GetInput -prompt 'Your VM guest OS user password' -IsPassword $true
"-----"
$adUser = GetInput -prompt 'Type the AD user name to join the AD' -IsPassword $false
""
"Please type the AD user password."
"Plase note that special character in password may not work with the script"
$adUserPassword = GetInput -prompt 'Your AD user password' -IsPassword $true
"-----"

#$csvFile = Read-Host 'Csv File '
$csvFile = '.\CloneVMs.csv'

#----- Main Script -----

#Connect to vCenter
#Connect to vCenter
$VC_Conn_State = Connect-VIServer $vcAddress -user $vcAdmin -password $vcPassword
if([string]::IsNullOrEmpty($VC_Conn_State))
{
    Write-Host 'Exit since failed to login vCenter'
    exit
}
else
{
    Write-Host 'vCenter is connected'
}

#Read input CSV file
$csvData = Import-CSV $csvFile

$destFolder = "/home/$guestUser/"

#Handle VMs one by one
foreach ($line in $csvData)
{
    "`n-----"
    $VMName = $line.VMName
    write-host -ForegroundColor Yellow "VM: $VMName`n"

    $cmd = "sudo /usr/bin/net ads join -U $adUser%$adUserPassword"
    Write-Host "Run cmd 'sudo /usr/bin/net ads join' in VM '$VMName' with user '$guestUser'"
}

```

```
Invoke-VMScript -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -GuestUser
$guestUser -GuestPassword $guestPassword -Confirm:$false -ScriptType Bash -ScriptText $cmd
}

Disconnect-VIServer $vcAddress -Confirm:$false
exit
```

指令碼執行

下列訊息是來自指令碼的執行：

```
PowerCLI C:\scripts> .\ClonedVMs_JoinDomain.ps1

-----
Your vCenter address: 10.117.44.17
Your vCenter admin user name: administrator
Your vCenter admin user password: *****

-----
Your ESXi host admin user name, such as root: root
Your ESXi host admin user password: *****

-----
Your VM guest OS user name: ViewUser
Your VM guest OS user password: *****

-----
Type the AD user name to join the AD: viewadmin
Please type the AD user password.
Please note that special character in password may not work with the script.
Your AD user password: *****
```

使用 SSH 將複製的虛擬機器加入 AD 網域的範例指令碼

您可以自訂和使用下列範例指令碼，以將複製的虛擬機器 (VM) 加入 Active Directory (AD) 網域。此指令碼使用 SSH 在 Linux 虛擬機器上執行命令。

如果您對 AD 整合使用 Winbind 解決方案，則必須執行此指令碼，因為加入網域的步驟對複製的虛擬機器將會失敗。此指令碼會在每個虛擬機器上執行命令來加入網域。如果您使用 OpenLDAP 解決方案，則不需要執行此指令碼。

若要複製和貼上不含分頁符號的指令碼內容，請使用本主題的 HTML 版本，您可以從 Horizon7 文件頁面取得，網址是：https://www.vmware.com/support/pubs/view_pubs.html。

指令碼輸入

此指令碼會讀取一個輸入檔，在[用於部署 Linux 桌面平台的範例 PowerCLI 指令碼輸入檔](#)中有加以描述。此指令碼也會互動地要求下列資訊：

- vCenter Server 的 IP 位址
- vCenter Server 的管理員登入名稱
- vCenter Server 的管理員密碼
- Linux 虛擬機器的使用者登入名稱

- Linux 虛擬機器的使用者密碼
- 獲得授權可將機器加入網域的 AD 使用者的登入名稱
- 獲得授權之 AD 使用者的密碼

指令碼內容

```
<#
.SYNOPSIS
run command "sudo /usr/bin/net ads join" via SSH

.DESCRIPTION
The tool is to run the command "sudo /usr/bin/net ads join" to join Linux machine to AD via SSH

.NOTES
#>
#----- Functions -----
function GetInput
{
    Param($prompt, $IsPassword = $false)
    $prompt = $prompt + ": "
    Write-Host $prompt -NoNewLine
    [Console]::ForegroundColor = "Blue"
    if ($IsPassword)
    {
        $input = Read-Host -AsSecureString
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBS
TR($input))
    }
    else
    {
        $input = Read-Host
    }

    [Console]::ResetColor()
    return $input
}

function Check_SSH_Client
{
    Param($IsPlink, $IsPSCP)
    if ($IsPlink)
    {
        if (Test-Path ".\plink.exe")
        {
            write-host -ForegroundColor Yellow 'SSH client "plink.exe" found'
        }
        else
        {
            write-host -ForegroundColor Red 'SSH client "plink.exe" not found, please download from
its official web site'
            exit
        }
    }
}
```

```

}
if ($IsPSCP)
{
    if (Test-Path ".\pscp.exe")
    {
        write-host -ForegroundColor Yellow 'SSH client "pscp.exe" found'
    }
    else
    {
        write-host -ForegroundColor Red 'SSH client "pscp.exe" not found, please download from its
official web site'
        exit
    }
}
}

function RunCmdViaSSH
{
    Param($VM_Name, $User, $Password, $Cmd, $returnOutput = $false)

    $VM= Get-VM $VM_Name
    $IP = $VM.guest.IPAddress[0]
    write-host "Run cmd on $VM_Name ($IP)"
    if($returnOutput)
    {
        $command = "echo yes | .\plink.exe -ssh -l $user -pw $password $IP " + "'" + $cmd + "'"
        $output = Invoke-Expression $command
        return $output
    }
    else
    {
        echo yes | .\plink.exe -ssh -l $user -pw $password $IP "$cmd"
    }
}

function UploadFileViaSSH
{
    Param($VM_Name, $User, $Password, $LocalPath, $DestPath)

    $VM= Get-VM $VM_Name
    $IP = $VM.guest.IPAddress[0]
    $command = "echo yes | .\pscp.exe -l $User -pw $Password $LocalPath $IP" + ":" + "$DestPath"
    write-host "Upload file: $command"
    Invoke-Expression $command
}

#----- Handle input -----
"-----"
Check_SSH_Client -IsPlink $true -IsPSCP $false
"-----"
$vcAddress = GetInput -prompt "Your vCenter address" -IsPassword $false
$vcAdmin = GetInput -prompt "Your vCenter admin user name" -IsPassword $false
$vcPassword = GetInput -prompt "Your vCenter admin user password" -IsPassword $true
"-----"

```

```

$guestUser = GetInput -prompt 'Your VM guest OS user name' -IsPassword $false
$guestPassword = GetInput -prompt 'Your VM guest OS user password' -IsPassword $true
"-----"
$adUser = GetInput -prompt 'Type the AD user name to join the AD' -IsPassword $false
""
"Please type the AD user password."
[Console]::ForegroundColor = "Yellow"
"Plase note that special character should be escaped. For example, $ should be \$\"
[Console]::ResetColor()
$adUserPassword = GetInput -prompt 'Your AD user password' -IsPassword $true
"-----"

#$csvFile = Read-Host 'Csv File '
$csvFile = '.\CloneVMs.csv'

#----- Main Script -----

#Connect to vCenter
$VC_Conn_State = Connect-VIServer $vcAddress -user $vcAdmin -password $vcPassword
if([string]::IsNullOrEmpty($VC_Conn_State))
{
    Write-Host 'Exit since failed to login vCenter'
    exit
}
else
{
    Write-Host 'vCenter is connected'
}

#Read input CSV file
$csvData = Import-CSV $csvFile

$destFolder = "/home/$guestUser/"

#Handle VMs one by one
foreach ($line in $csvData)
{
    "-----"
    $VMName = $line.VMName
    write-host -ForegroundColor Yellow "VM: $VMName`n"

    $cmd = "sudo /usr/bin/net ads join -U $adUser%$adUserPassword"
    Write-Host "Run cmd 'sudo /usr/bin/net ads join' in VM '$VMName' with user '$guestUser'"
    RunCmdViaSSH -VM_Name $VMName -User $guestUser -Password $guestPassword -Cmd $cmd
}

Disconnect-VIServer $vcAddress -Confirm:$false
exit

```

指令碼執行

下列訊息是來自指令碼的執行：

```
PowerCLI C:\scripts> .\ClonedVMs_JoinDomain_SSH.ps1

-----
Your vCenter address: 10.117.44.17
Your vCenter admin user name: administrator
Your vCenter admin user password: *****

-----
Your VM guest OS user name: ViewUser
Your VM guest OS user password: *****

-----
Type the AD user name to join the AD: viewadmin
Please type the AD user password.
Please note that special character should be escaped. For example, $ should be \$
Your AD user password: *****
```

上傳組態檔至 Linux 虛擬機器的範例指令碼

您可以自訂和使用下列範例指令碼，將組態檔 `config` 和 `viewagent-custom.conf` 上傳到多個 Linux 虛擬機器 (VM)。

若要複製和貼上不含分頁符號的指令碼內容，請使用本主題的 HTML 版本，您可以從 Horizon7 文件頁面取得，網址是：https://www.vmware.com/support/pubs/view_pubs.html。

指令碼輸入

此指令碼會讀取一個輸入檔，在用於部署 Linux 桌面平台的範例 **PowerCLI** 指令碼輸入檔中有加以描述。此指令碼也會互動地要求下列資訊：

- vCenter Server 的 IP 位址
- vCenter Server 的管理員登入名稱
- vCenter Server 的管理員密碼
- ESXi 主機的管理員登入名稱
- ESXi 主機的管理員密碼
- Linux 虛擬機器的使用者登入名稱
- Linux 虛擬機器的使用者密碼

指令碼內容

```
<#
Upload the configuration files config and viewagent-custom.conf to Linux VMs
#>
#----- Functions -----
function GetInput
{
```

```

    Param($prompt, $IsPassword = $false)
    $prompt = $prompt + ": "
    Write-Host $prompt -NoNewLine
    [Console]::ForegroundColor = "Blue"
    if ($IsPassword)
    {
        $input = Read-Host -AsSecureString
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBS
TR($input))
    }
    else
    {
        $input = Read-Host
    }

    [Console]::ResetColor()
    return $input
}

#----- Handle Input -----
"-----"
write-host -ForegroundColor Blue 'Please ensure your config file and viewagent-custom.conf file are
in current working directory'
$vcAddress = GetInput -prompt "Your vCenter address" -IsPassword $false
$vcAdmin = GetInput -prompt "Your vCenter admin user name" -IsPassword $false
$vcPassword = GetInput -prompt "Your vCenter admin user password" -IsPassword $true
"-----"
$hostAdmin = GetInput -prompt 'Your ESXi host admin user name, such as root' -IsPassword $false
$hostPassword = GetInput -prompt "Your ESXi admin user password" -IsPassword $true
"-----"
$guestUser = GetInput -prompt 'Your VM guest OS user name' -IsPassword $false
$guestPassword = GetInput -prompt 'Your VM guest OS user password' -IsPassword $true
"-----"

$csvFile = '.\CloneVMs.csv'
$setConfig = $false
$setCustomConf = $false
$config_File = "config"
$customConf_File = "viewagent-custom.conf"

#check if config file exists
if(Test-Path $config_File)
{
    $setConfig = $true
    write-host -ForegroundColor Yellow '"config" file found'
}
else
{
    write-host -ForegroundColor Yellow '"config" file not found, skip it'
}

if(Test-Path $customConf_File)
{
    $setCustomConf = $true

```

```

        write-host -ForegroundColor Yellow '"viewagent-custom.conf" file found'
    }
else
{
    write-host -ForegroundColor Yellow '"viewagent-custom.conf" file not found, skip it'
}

if (($setConfig -eq $false)-AND ($setCustomConf -eq $false))
{
    write-host -ForegroundColor Red 'Both file not found, exit'
    exit
}

#Connect to vCenter
$VC_Conn_State = Connect-VIServer $vcAddress -user $vcAdmin -password $vcPassword
if([string]::IsNullOrEmpty($VC_Conn_State))
{
    Write-Host 'Exit since failed to login vCenter'
    exit
}
else
{
    Write-Host 'vCenter is connected'
}

#Read input CSV file
$csvData = Import-CSV $csvFile

$destFolder = "/home/$guestUser/"

#Handle VMs one by one
foreach ($line in $csvData)
{
    "`n-----"
    $VMName = $line.VMName
    write-host -ForegroundColor Yellow "VM: $VMName`n"

    #Try to delete the configuration file from home folder on destination VM
    $cmd = "rm -rf config viewagent-custom.conf"
    Write-Host "Run cmd '$cmd' in VM '$VMName' with user '$guestUser'"
    Invoke-VMScript -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -GuestUser
$guestUser -GuestPassword $guestPassword -Confirm:$false -ScriptType Bash -ScriptText $cmd

    if ($setConfig)
    {
        Write-Host "Upload File '$config_File' to '$destFolder' of VM '$VMName' with user '$guestUser'"
        Copy-VMGuestFile -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -GuestUser
$guestUser -GuestPassword $guestPassword -Confirm:$false -LocalToGuest -Destination $destFolder -
Source $config_File

        $cmd = "sudo mv ./ $config_File /etc/vmware/";
        Write-Host "Move configuraton file: $cmd"
        Invoke-VMScript -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -GuestUser
$guestUser -GuestPassword $guestPassword -Confirm:$false -ScriptType Bash -ScriptText $cmd
    }
}

```



```

}

if ($setCustomConf)
{
    Write-Host "Upload File '$customConf_File' to '$destFolder' of VM '$VMName' with user '$guestUser'"
    Copy-VMGuestFile -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -GuestUser $guestUser -GuestPassword $guestPassword -Confirm:$false -LocalToGuest -Destination $destFolder -Source $customConf_File

    $cmd = "sudo mv ./ $customConf_File /etc/vmware/";
    Write-Host "Move configuraton file: $cmd"
    Invoke-VMScript -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -GuestUser $guestUser -GuestPassword $guestPassword -Confirm:$false -ScriptType Bash -ScriptText $cmd
}
}
Disconnect-VIServer $vcAddress -Confirm:$false
exit

```

指令碼執行

下列訊息是來自指令碼的執行：

```

PowerCLI C:\scripts> .\UpdateOptionFile.ps1
-----
Please ensure your config file and view-agent.conf file are in current working directory.
Your vCenter address: 10.117.44.17
Your vCenter admin user name: administrator
Your vCenter admin user password: *****
-----
Your ESXi host admin user name, such as root: root
Your ESXi host admin user password: *****
-----
Your VM guest OS user name: ViewUser
Your VM guest OS user password: *****

```

使用 SSH 上傳組態檔至 Linux 虛擬機器的範例指令碼

您可以自訂和使用下列範例指令碼，將組態檔 `config` 和 `viewagent-custom.conf` 上傳到多個 Linux 虛擬機器 (VM)。此指令碼使用 SSH 在 Linux 虛擬機器上執行命令。

若要複製和貼上不含分頁符號的指令碼內容，請使用本主題的 HTML 版本，您可以從 Horizon7 文件頁面取得，網址是：https://www.vmware.com/support/pubs/view_pubs.html。

指令碼輸入

此指令碼會讀取一個輸入檔，在用於部署 Linux 桌面平台的範例 [PowerCLI 指令碼輸入檔](#) 中有加以描述。此指令碼也會互動地要求下列資訊：

- vCenter Server 的 IP 位址
- vCenter Server 的管理員登入名稱

- vCenter Server 的管理員密碼
- Linux 虛擬機器的使用者登入名稱
- Linux 虛擬機器的使用者密碼

指令碼內容

```
<#
Upload the configuration files config and viewagent-custom.conf to Linux VMs using SSH
#>
#----- Functions -----
function GetInput
{
    Param($prompt, $IsPassword = $false)
    $prompt = $prompt + ": "
    Write-Host $prompt -NoNewLine
    [Console]::ForegroundColor = "Blue"
    if ($IsPassword)
    {
        $input = Read-Host -AsSecureString
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBS
TR($input))
    }
    else
    {
        $input = Read-Host
    }

    [Console]::ResetColor()
    return $input
}
function Check_SSH_Client
{
    Param($IsPlink, $IsPSCP)
    if ($IsPlink)
    {
        if (Test-Path ".\plink.exe")
        {
            write-host -ForegroundColor Yellow 'SSH client "plink.exe" found'
        }
        else
        {
            write-host -ForegroundColor Red 'SSH client "plink.exe" not found, please download from
its official web site'
            exit
        }
    }
    if ($IsPSCP)
    {
        if (Test-Path ".\pscp.exe")
        {
            write-host -ForegroundColor Yellow 'SSH client "pscp.exe" found'
        }
    }
}
```

```

        else
        {
            write-host -ForegroundColor Red 'SSH client "pscp.exe" not found, please download from its
official web site'
            exit
        }
    }
}

function RunCmdViaSSH
{
    Param($VM_Name, $User, $Password, $Cmd, $returnOutput = $false)

    $VM= Get-VM $VM_Name
    $IP = $VM.guest.IPAddress[0]
    write-host "Run cmd on $VM_Name ($IP)"
    if($returnOutput)
    {
        $command = "echo yes | .\plink.exe -ssh -l $user -pw $password $IP " + "'" + $cmd + "'"
        $output = Invoke-Expression $command
        return $output
    }
    else
    {
        echo yes | .\plink.exe -ssh -l $user -pw $password $IP "$cmd"
    }
}

function UploadFileViaSSH
{
    Param($VM_Name, $User, $Password, $LocalPath, $DestPath)

    $VM= Get-VM $VM_Name
    $IP = $VM.guest.IPAddress[0]
    $command = "echo yes | .\pscp.exe -l $User -pw $Password $LocalPath $IP" + ":" + "$DestPath"
    write-host "Upload file: $command"
    Invoke-Expression $command
}

#----- Handle Input -----
"-----"
Check_SSH_Client -IsPlink $true -IsPSCP $true
"-----"
write-host -ForegroundColor Blue 'Please ensure your config file and viewagent-custom.conf file are
in current working directory'
$vcAddress = GetInput -prompt "Your vCenter address" -IsPassword $false
$vcAdmin = GetInput -prompt "Your vCenter admin user name" -IsPassword $false
$vcPassword = GetInput -prompt "Your vCenter admin user password" -IsPassword $true
"-----"
$guestUser = GetInput -prompt 'Your VM guest OS user name' -IsPassword $false
$guestPassword = GetInput -prompt 'Your VM guest OS user password' -IsPassword $true
"-----"

$csvFile = '.\CloneVMs.csv'

```

```

$setConfig = $false
$setCustomConf = $false
$config_File = "config"
$customConf_File = "viewagent-custom.conf"

#check if config file exists
if(Test-Path $config_File)
{
    $setConfig = $true
    write-host -ForegroundColor Yellow '"config" file found'
}
else
{
    write-host -ForegroundColor Yellow '"config" file not found, skip it'
}

if(Test-Path $customConf_File)
{
    $setCustomConf = $true
    write-host -ForegroundColor Yellow '"viewagent-custom.conf" file found'
}
else
{
    write-host -ForegroundColor Yellow '"viewagent-custom.conf" file not found, skip it'
}

if (($setConfig -eq $false)-AND ($setCustomConf -eq $false))
{
    write-host -ForegroundColor Red 'Both file not found, exit'
    exit
}

#Connect to vCenter
$VC_Conn_State = Connect-VIServer $vcAddress -user $vcAdmin -password $vcPassword
if([string]::IsNullOrEmpty($VC_Conn_State))
{
    Write-Host 'Exit since failed to login vCenter'
    exit
}
else
{
    Write-Host 'vCenter is connected'
}

#Read input CSV file
$csvData = Import-CSV $csvFile

$destFolder = "/home/$guestUser/"

#Handle VMs one by one
foreach ($line in $csvData)
{
    "`n-----"
    $VMName = $line.VMName
    write-host -ForegroundColor Yellow "VM: $VMName`n"
}

```

```
#Try to delete the configuration file from home folder on destination VM
$cmd = "rm -rf config viewagent-custom.conf"
Write-Host "Run cmd '$cmd' in VM '$VMName' with user '$guestUser'"
RunCmdViaSSH -VM_Name $VMName -User $guestUser -Password $guestPassword -Cmd $cmd

if ($setConfig)
{
    Write-Host "Upload File '$config_File' to '$destFolder' of VM '$VMName' with user '$guestUser'"
    UploadFileViaSSH -VM_Name $VMName -User $guestUser -Password $guestPassword -LocalPath
$config_File -DestPath $destFolder

    $cmd = "sudo mv ./ $config_File /etc/vmware/";
    Write-Host "Move configuraton file: $cmd"
    RunCmdViaSSH -VM_Name $VMName -User $guestUser -Password $guestPassword -Cmd $cmd
}

if ($setCustomConf)
{
    Write-Host "Upload File '$customConf_File' to '$destFolder' of VM '$VMName' with user
'$guestUser'"
    UploadFileViaSSH -VM_Name $VMName -User $guestUser -Password $guestPassword -LocalPath
$customConf_File -DestPath $destFolder

    $cmd = "sudo mv ./ $customConf_File /etc/vmware/";
    Write-Host "Move configuraton file: $cmd"
    RunCmdViaSSH -VM_Name $VMName -User $guestUser -Password $guestPassword -Cmd $cmd
}
}
Disconnect-VIServer $vcAddress -Confirm:$false
exit
```

指令碼執行

下列訊息是來自指令碼的執行：

```
PowerCLI C:\scripts> .\UpdateOptionFile.ps1
-----
Please ensure your config file and view-agent.conf file are in current working directory.
Your vCenter address: 10.117.44.17
Your vCenter admin user name: administrator
Your vCenter admin user password: *****
-----
Your VM guest OS user name: ViewUser
Your VM guest OS user password: *****
```

在 Linux 桌面平台機器上升級 Horizon Agent 的範例指令碼

您可以自訂和使用下列範例指令碼，在多個 Linux 虛擬機器 (VM) 上升級 Horizon Agent。

此指令碼會在安裝 Horizon Agent 之前上傳安裝程式 tar ball 到每個虛擬機器。上傳工作可能很費時，特別是牽涉到大量虛擬機器且網路速度緩慢時。為了節省時間，您可以執行使用 SSH 的指令碼，或將安裝程式 tar ball 放置在每個虛擬機器可使用的共用位置上，以便不需要上傳檔案。

若要複製和貼上不含分頁符號的指令碼內容，請使用本主題的 HTML 版本，您可以從 Horizon7 文件頁面取得，網址是：https://www.vmware.com/support/pubs/view_pubs.html。

指令碼輸入

此指令碼會讀取一個輸入檔，在[用於部署 Linux 桌面平台的範例 PowerCLI 指令碼輸入檔](#)中有加以描述。此指令碼也會互動地要求下列資訊：

- 接受 Horizon Agent EULA (使用者授權合約)
- vCenter Server 的 IP 位址
- vCenter Server 的管理員登入名稱
- vCenter Server 的管理員密碼
- ESXi 主機的管理員登入名稱
- ESXi 主機的管理員密碼
- Linux 客體作業系統的使用者登入名稱
- Linux 客體作業系統的使用者密碼
- Horizon Agent tar ball 路徑
- 升級為受管理的虛擬機器
- 安裝智慧卡重新導向功能

指令碼內容

```
<#
Upload the Linux Agent installer tar ball and re-install
#>

#-----
Functions-----
function GetInput
{
    Param($prompt, $IsPassword = $false)
    $prompt = $prompt + ": "
    Write-Host $prompt -NoNewLine
    [Console]::ForegroundColor = "Blue"
    if ($IsPassword)
    {
        $input = Read-Host -AsSecureString
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBS
TR($input))
    }
}
```

```

    else
    {
        $input = Read-Host
    }

    [Console]::ResetColor()
    return $input
}

#-----Handle
input-----
"-----"
$acceptEULA = GetInput -prompt 'Accept Linux View Agent EULA in tar bundle ("yes" or "no")' -
IsPassword $false
if ($acceptEULA -ne "yes")
{
    write-host -ForegroundColor Red "You need accept the EULA with 'yes'(case sensitive)"
    exit
}
$svcAddress = GetInput -prompt "Your vCenter address" -IsPassword $false
$svcAdmin = GetInput -prompt "Your vCenter admin user name" -IsPassword $false
$svcPassword = GetInput -prompt "Your vCenter admin user password" -IsPassword $true
"-----"
$hostAdmin = GetInput -prompt 'Your ESXi host admin user name, such as root' -IsPassword $false
$hostPassword = GetInput -prompt "Your ESXi admin user password" -IsPassword $true
"-----"
$guestUser = GetInput -prompt 'Your VM guest OS user name' -IsPassword $false
$guestPassword = GetInput -prompt 'Your VM guest OS user password' -IsPassword $true
"-----"
$agentInstaller = GetInput -prompt 'Type the View Agent tar ball path' -IsPassword $false
"-----"
$UpgradeToManagedVM = GetInput -prompt 'Upgrade to managed VM ("yes" or "no")' -IsPassword $false
if (($UpgradeToManagedVM -ne "yes") -AND $UpgradeToManagedVM -ne "no")
{
    write-host -ForegroundColor Red "You need select 'yes' or 'no'(case sensitive)"
    exit
}
$installSmartcard = GetInput -prompt 'Install the Smartcard redirection feature ("yes" or "no")' -
IsPassword $false
if (($installSmartcard -ne "yes") -AND $installSmartcard -ne "no")
{
    write-host -ForegroundColor Red "You need select 'yes' or 'no'(case sensitive)"
    exit
}
"-----"

#$csvFile = Read-Host 'Csv File '
$csvFile = '.\CloneVMs.csv'

#check if file exists
if (!(Test-Path $agentInstaller))
{
    write-host -ForegroundColor Red "installer File not found"
    exit
}

```

```

#check if file exists
if (!(Test-Path $csvFile))
{
write-host -ForegroundColor Red "CSV File not found"
exit
}
#-----
Functions-----
function GetSourceInstallerMD5()
{
    $agentInstallerPath = Convert-Path $agentInstaller;
    $md5 = New-Object -TypeName System.Security.Cryptography.MD5CryptoServiceProvider;
    $md5HashWithFormat =
[System.BitConverter]::ToString($md5.ComputeHash([System.IO.File]::ReadAllBytes($agentInstallerPath)))
;
    $md5Hash = ($md5HashWithFormat.replace("-", "")).ToLower();
    return $md5Hash;
}

#-----
Main-----
#Get installer MD5Sum
$installerMD5Hash = GetSourceInstallerMD5;

#Connect to vCenter
$VC_Conn_State = Connect-VIServer $vcAddress -user $vcAdmin -password $vcPassword
if([string]::IsNullOrEmpty($VC_Conn_State))
{
    Write-Host 'Exit since failed to login vCenter'
    exit
}
else
{
    Write-Host 'vCenter is connected'
}

#Read input CSV file
$csvData = Import-CSV $csvFile

$destFolder = "/home/$guestUser/"

#Handle VMs one by one
foreach ($line in $csvData)
{
    "`n-----"
    $VMName = $line.VMName
    write-host -ForegroundColor Yellow "VM: $VMName`n"

    $cmd = "rm -rf VMware-*linux-*"
    Write-Host "Run cmd '$cmd' in VM '$VMName' with user '$guestUser'"
    Invoke-VMScript -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -GuestUser
$guestUser -GuestPassword $guestPassword -Confirm:$false -ScriptType Bash -ScriptText $cmd

    #Upload installer tar ball to Linux VM
    Write-Host "Upload File '$agentInstaller' to '$destFolder' of VM '$VMName' with user '$guestUser'"
}

```



```

Copy-VMGuestFile -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -GuestUser
$guestUser -GuestPassword $guestPassword -Confirm:$false -LocalToGuest -Destination $destFolder -
Source $agentInstaller

#Check the uploaded installer md5sum
$cmd = "md5sum VMware-*linux-*"
Write-Host "Run cmd '$cmd' in VM '$VMName' with user '$guestUser'"
$output = Invoke-VMScript -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -
GuestUser $guestUser -GuestPassword $guestPassword -Confirm:$false -ScriptType Bash -ScriptText $cmd

if($output.Contains($installerMD5Hash))
{
    Write-Host $VMName": Uploaded installer's MD5Sum matches the local installer's MD5Sum";
    Write-Host $VMName": Extract the installer and do installation";
    $cmd = "tar -xzf VMware-*linux-*.tar.gz"
    Write-Host "Run cmd '$cmd' in VM '$VMName' with user '$guestUser'"
    Invoke-VMScript -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -GuestUser
$guestUser -GuestPassword $guestPassword -Confirm:$false -ScriptType Bash -ScriptText $cmd

    $cmd = "sudo setenforce 0";
    Write-Host "Set the selinux to permissive mode: $cmd"
    Invoke-VMScript -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -GuestUser
$guestUser -GuestPassword $guestPassword -Confirm:$false -ScriptType Bash -ScriptText $cmd

    $cmd = "sudo killall /usr/lib/vmware/viewagent/VMwareBlastServer/VMwareBlastServer"
    Write-Host "Stop VMwareBlastServer before upgrading: $cmd"
    Invoke-VMScript -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -GuestUser
$guestUser -GuestPassword $guestPassword -Confirm:$false -ScriptType Bash -ScriptText $cmd

    #Run the upgrade command.
    $cmd = "cd VMware-*linux-* && sudo ./install_viewagent.sh -A yes -m $installSmartcard -M
$UpgradeToManagedVM"
    Write-Host "Run upgrade cmd in VM '$VMName' with user '$guestUser': $cmd"
    Invoke-VMScript -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -GuestUser
$guestUser -GuestPassword $guestPassword -Confirm:$false -ScriptType Bash -ScriptText $cmd

    $cmd = "sudo shutdown -r +1&"
    Write-Host "Reboot to apply the View Agent installation"
    Invoke-VMScript -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -GuestUser
$guestUser -GuestPassword $guestPassword -Confirm:$false -ScriptType Bash -ScriptText $cmd
}
else
{
    Write-Host $VMName": Uploaded installer's MD5Sum does NOT match the local installer's MD5Sum";
    Write-Host $VMName": Skip the installation. Please check your network and VMware Tools
status";
    exit;
}
}
Disconnect-VIServer $vcAddress -Confirm:$false
exit

```

指令碼執行

下列訊息是來自指令碼的執行：

```
PowerCLI C:\scripts> .\InstallAgent.ps1
-----
Accept Linux Horizon Agent EULA in tar bundle ("yes" or "no"): yes
Your vCenter address: 10.117.44.17
Your vCenter admin user name: administrator
Your vCenter admin user password: *****
-----
Your ESXi host admin user name, such as root: root
Your ESXi host admin user password: *****
-----
Your VM guest OS user name: ViewUser
Your VM guest OS user password: *****
-----
Type the Horizon Agent tar ball path. Please take care of the installer arch: .\VMware-viewagent-
linux-x86_64-x.y.z-1234567.tar.gz
-----
Upgrade to managed VM ("yes" or "no"): yes
Install the Smartcard redirection feature ("yes" or "no"): no
```

使用 SSH 在 Linux 虛擬機器上升級 Horizon Agent 的範例指令碼

您可以自訂和使用下列範例指令碼，在多個 Linux 虛擬機器 (VM) 上升級 Horizon Agent。此指令碼使用 SSH 在 Linux 虛擬機器上執行命令。

若要複製和貼上不含分頁符號的指令碼內容，請使用本主題的 HTML 版本，您可以從 Horizon7 文件頁面取得，網址是：https://www.vmware.com/support/pubs/view_pubs.html。

指令碼輸入

此指令碼會讀取一個輸入檔，在用於部署 Linux 桌面平台的範例 PowerCLI 指令碼輸入檔中有加以描述。此指令碼也會互動地要求下列資訊：

- 接受 Horizon Agent EULA (使用者授權合約)
- vCenter Server 的 IP 位址
- vCenter Server 的管理員登入名稱
- vCenter Server 的管理員密碼
- ESXi 主機的管理員登入名稱
- ESXi 主機的管理員密碼
- Linux 客體作業系統的使用者登入名稱
- Linux 客體作業系統的使用者密碼
- Horizon Agent tar ball 路徑

- 升級為受管理的虛擬機器
- 安裝智慧卡重新導向功能

指令碼內容

```
<#
Upload the Linux Agent installer tar ball and re-install
#>

#-----
Functions-----
function GetInput
{
    Param($prompt, $IsPassword = $false)
    $prompt = $prompt + ": "
    Write-Host $prompt -NoNewLine
    [Console]::ForegroundColor = "Blue"
    if ($IsPassword)
    {
        $input = Read-Host -AsSecureString
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBS
TR($input))
    }
    else
    {
        $input = Read-Host
    }

    [Console]::ResetColor()
    return $input
}
function Check_SSH_Client
{
    Param($IsPlink, $IsPSCP)
    if ($IsPlink)
    {
        if (Test-Path ".\plink.exe")
        {
            write-host -ForegroundColor Yellow 'SSH client "plink.exe" found'
        }
        else
        {
            write-host -ForegroundColor Red 'SSH client "plink.exe" not found, please download from
its official web site'
            exit
        }
    }
    if ($IsPSCP)
    {
        if (Test-Path ".\pscp.exe")
        {

```

```

        write-host -ForegroundColor Yellow 'SSH client "pscp.exe" found'
    }
    else
    {
        write-host -ForegroundColor Red 'SSH client "pscp.exe" not found, please download from its
official web site'
        exit
    }
}

function RunCmdViaSSH
{
    Param($VM_Name, $User, $Password, $Cmd, $returnOutput = $false)

    $VM= Get-VM $VM_Name
    $IP = $VM.guest.IPAddress[0]
    write-host "Run cmd on $VM_Name ($IP)"
    if($returnOutput)
    {
        $command = "echo yes | .\plink.exe -ssh -l $user -pw $password $IP " + "'" + $cmd + "'"
        $output = Invoke-Expression $command
        return $output
    }
    else
    {
        echo yes | .\plink.exe -ssh -l $user -pw $password $IP "$cmd"
    }
}

function UploadFileViaSSH
{
    Param($VM_Name, $User, $Password, $LocalPath, $DestPath)

    $VM= Get-VM $VM_Name
    $IP = $VM.guest.IPAddress[0]
    $command = "echo yes | .\pscp.exe -l $User -pw $Password $LocalPath $IP" + ":" + "$DestPath"
    write-host "Upload file $LocalPath to VM $VM_Name with user $User"
    Invoke-Expression $command
}

#-----Handle
input-----
"-----"
Check_SSH_Client -IsPlink $true -IsPSCP $true
"-----"
$acceptEULA = GetInput -prompt 'Accept Linux View Agent EULA in tar bundle ("yes" or "no")' -
IsPassword $false
if ($acceptEULA -ne "yes")
{
    write-host -ForegroundColor Red "You need accept the EULA with 'yes'(case sensitive)"
    exit
}
$vcAddress = GetInput -prompt "Your vCenter address" -IsPassword $false

```

```

SvcAdmin = GetInput -prompt "Your vCenter admin user name" -IsPassword $false
SvcPassword = GetInput -prompt "Your vCenter admin user password" -IsPassword $true
"-----"
$guestUser = GetInput -prompt 'Your VM guest OS user name' -IsPassword $false
$guestPassword = GetInput -prompt 'Your VM guest OS user password' -IsPassword $true
"-----"
$agentInstaller = GetInput -prompt 'Type the View Agent tar ball path' -IsPassword $false
"-----"
$UpgradeToManagedVM = GetInput -prompt 'Upgrade to managed VM ("yes" or "no")' -IsPassword $false
if (($UpgradeToManagedVM -ne "yes") -AND $UpgradeToManagedVM -ne "no")
{
    write-host -ForegroundColor Red "You need select 'yes' or 'no'(case sensitive)"
    exit
}
$installSmartcard = GetInput -prompt 'Install the Smartcard redirection feature ("yes" or "no")' -
IsPassword $false
if (($installSmartcard -ne "yes") -AND $installSmartcard -ne "no")
{
    write-host -ForegroundColor Red "You need select 'yes' or 'no'(case sensitive)"
    exit
}
"-----"

#csvFile = Read-Host 'Csv File '
$csvFile = '.\CloneVMs.csv'

#check if file exists
if (!(Test-Path $agentInstaller))
{
    write-host -ForegroundColor Red "installer File not found"
    exit
}

#check if file exists
if (!(Test-Path $csvFile))
{
    write-host -ForegroundColor Red "CSV File not found"
    exit
}
#-----
Functions-----
function GetSourceInstallerMD5()
{
    $agentInstallerPath = Convert-Path $agentInstaller;
    $md5 = New-Object -TypeName System.Security.Cryptography.MD5CryptoServiceProvider;
    $md5HashWithFormat =
[System.BitConverter]::ToString($md5.ComputeHash([System.IO.File]::ReadAllBytes($agentInstallerPath)))
;
    $md5Hash = ($md5HashWithFormat.replace("-", "")).ToLower();
    return $md5Hash;
}

#-----
Main-----
#Get installer MD5Sum

```

```

$installerMD5Hash = GetSourceInstallerMD5;

#Connect to vCenter
$VC_Conn_State = Connect-VIServer $vcAddress -user $vcAdmin -password $vcPassword
if([string]::IsNullOrEmpty($VC_Conn_State))
{
    Write-Host 'Exit since failed to login vCenter'
    exit
}
else
{
    Write-Host 'vCenter is connected'
}

#Read input CSV file
$csvData = Import-CSV $csvFile

$destFolder = "/home/$guestUser/"

#Handle VMs one by one
foreach ($line in $csvData)
{
    "`n-----"
    $VMName = $line.VMName
    write-host -ForegroundColor Yellow "VM: $VMName`n"

    $cmd = "rm -rf VMware-*linux-*"
    Write-Host "Run cmd '$cmd' in VM '$VMName' with user '$guestUser'"
    RunCmdViaSSH -VM_Name $VMName -User $guestUser -Password $guestPassword -Cmd $cmd

    #Upload installer tar ball to Linux VM
    Write-Host "Upload File '$agentInstaller' to '$destFolder' of VM '$VMName' with user '$guestUser'"
    UploadFileViaSSH -VM_Name $VMName -User $guestUser -Password $guestPassword -LocalPath
$agentInstaller -DestPath $destFolder

    #Check the uploaded installer md5sum
    $cmd = "md5sum VMware-*linux-*"
    Write-Host "Run cmd '$cmd' in VM '$VMName' with user '$guestUser'"
    $output = RunCmdViaSSH -VM_Name $VMName -User $guestUser -Password $guestPassword -Cmd $cmd -
$returnOutput $true

    if($output.Contains($installerMD5Hash))
    {
        Write-Host $VMName": Uploaded installer's MD5Sum matches the local installer's MD5Sum";
        Write-Host $VMName": Extract the installer and do installation";

        $cmd = "tar -xzf VMware-*linux-*.tar.gz"
        Write-Host "Run cmd '$cmd' in VM '$VMName' with user '$guestUser'"
        RunCmdViaSSH -VM_Name $VMName -User $guestUser -Password $guestPassword -Cmd $cmd

        $cmd = "sudo setenforce 0";
        Write-Host "Set the selinux to permissive mode: $cmd"
        RunCmdViaSSH -VM_Name $VMName -User $guestUser -Password $guestPassword -Cmd $cmd
    }
}

```

```

    $cmd = "sudo killall /usr/lib/vmware/viewagent/VMwareBlastServer/VMwareBlastServer"
    Write-Host "Stop VMwareBlastServer before upgrading: $cmd"
    RunCmdViaSSH -VM_Name $VMName -User $guestUser -Password $guestPassword -Cmd $cmd

    #Run the upgrade command.
    $cmd = "cd VMware-*linux-* && sudo ./install_viewagent.sh -r yes -A yes -m $installSmartcard
-M $UpgradeToManagedVM"
    Write-Host "Run upgrade cmd in VM '$VMName' with user '$guestUser': $cmd"
    RunCmdViaSSH -VM_Name $VMName -User $guestUser -Password $guestPassword -Cmd $cmd
    Write-Host -ForegroundColor Yellow "Linux Agent installer will reboot the Linux VM after
upgrade, and you may hit the ssh connection closed error message, which is expectation"
}
else
{
    Write-Host $VMName": Uploaded installer's MD5Sum does NOT match the local installer's MD5Sum";
    Write-Host $VMName": Skip the installation. Please check your network and VMware Tools
status";
    exit;
}
}
Disconnect-Viserver $vcAddress -Confirm:$false
exit

```

指令碼執行

下列訊息是來自指令碼的執行：

```

PowerCLI C:\scripts> .\InstallAgent.ps1
-----
Accept Linux Horizon Agent EULA in tar bundle ("yes" or "no"): yes
Your vCenter address: 10.117.44.17
Your vCenter admin user name: administrator
Your vCenter admin user password: *****
-----
Your VM guest OS user name: ViewUser
Your VM guest OS user password: *****
-----
Type the Horizon Agent tar ball path. Please take care of the installer arch: .\VMware-viewagent-
linux-x86_64-x.y.z-1234567.tar.gz
-----
-----
Upgrade to managed VM ("yes" or "no"): yes
Install the Smartcard redirection feature ("yes" or "no"): no

```

在 Linux 虛擬機器上執行作業的範例指令碼

您可以自訂和使用下列範例指令碼，在多個 Linux 虛擬機器 (VM) 上執行作業。該作業包含開啟電源、關閉電源、關機、重新啟動和刪除虛擬機器。

此指令碼可從 vCenter Server 刪除虛擬機器，但無法從 View 刪除。

若要複製和貼上不含分頁符號的指令碼內容，請使用本主題的 HTML 版本，您可以從 Horizon7 文件頁面取得，網址是：https://www.vmware.com/support/pubs/view_pubs.html。

指令碼輸入

此指令碼會讀取一個輸入檔，在[用於部署 Linux 桌面平台的範例 PowerCLI 指令碼輸入檔](#)中有加以描述。此指令碼也會互動地要求下列資訊：

- vCenter Server 的 IP 位址
- vCenter Server 的管理員登入名稱
- vCenter Server 的管理員密碼
- 要執行的動作，可以是開機、關機、關閉客體、重新啟動虛擬機器、重新啟動虛擬機器客體或刪除虛擬機器。
- 虛擬機器上各作業的等候間隔時間 (以秒為單位)。

指令碼內容

```
<#
.DESCRIPTION
The Tool supports:
1. Power off VMs
2. Power on VMs
3. Shutdown VMs
4. Restart VMs
5. Restart VM guest
6. Delete VMs from Disk
.NOTES
#>

#----- Functions -----

function GetInput
{
    Param($prompt, $IsPassword = $false)
    $prompt = $prompt + ": "
    Write-Host $prompt -NoNewLine
    [Console]::ForegroundColor = "Blue"
    if ($IsPassword)
    {
        $input = Read-Host -AsSecureString
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBS
TR($input))
    }
    else
    {
        $input = Read-Host
    }

    [Console]::ResetColor()
    return $input
}

function IsVMExists ($VMExists)
```



```

{
    Write-Host "Checking if the VM $VMExists Exists"
    [bool]$Exists = $false

    #Get all VMS and check if the VMs is already present in VC
    $listvm = Get-vm
    foreach ($lvm in $listvm)
    {
        if($VMExists -eq $lvm.Name )
        {
            $Exists = $true
            Write-Host "$VMExists is Exist"
        }
    }
    return $Exists
}

function Delete_VM($VMToDelete)
{
    Write-Host "Deleting VM $VMToDelete"
    Get-VM $VMToDelete | where { $_.PowerState -eq "PoweredOn" } | Stop-VM -confirm:$false
    Get-VM $VMToDelete | Remove-VM -DeleteFromDisk -confirm:$false
}

#----- Handle input -----
"-----"
$vcAddress = GetInput -prompt "Your vCenter address" -IsPassword $false
$vcAdmin = GetInput -prompt "Your vCenter admin user name" -IsPassword $false
$vcPassword = GetInput -prompt "Your vCenter admin user password" -IsPassword $true
"-----"

$action = GetInput -prompt 'Select action: 1). Power On 2). Power Off 3) Shutdown VM Guest 4).
Restart VM 5). Restart VM Guest 6). Delete VM' -IsPassword $false
$sleepTime = GetInput -prompt 'Wait time (seconds) between each VM' -IsPassword $false
"-----"

[Console]::ForegroundColor = "Yellow"
switch ($action)
{
    1
    {
        "Your selection is 1). Power On"
    }
    2
    {
        "Your selection is 2). Power Off"
    }
    3
    {
        "Your selection is 3) Shutdown"
    }
    4
    {
        "Your selection is 4). Restart VM"
    }
    5
    {

```

```

        "Your selection is 5). Restart VM Guest"
    }
    6
    {
        "Your selection is 6). Delete VM"
    }
    default
    {
        "Invalid selection for action: $action"
        exit
    }
}
[Console]::ResetColor()
$csvFile = '.\CloneVMs.csv'

#check if file exists
if (!(Test-Path $csvFile))
{
    write-host -ForegroundColor Red "CSV File not found"
    exit
}
"-----"

#----- Main -----
#Read input CSV file
Disconnect-VIServer $vcAddress -Confirm:$false
#Connect-VIServer $vcAddress -ErrorAction Stop -user $vcAdmin -password $vcPassword
Connect-VIServer $vcAddress -user $vcAdmin -password $vcPassword
$csvData = Import-CSV $csvFile

foreach ($line in $csvData)
{
    $VMName = $line.VMName
    switch ($action)
    {
        1
        {
            Get-VM $VMName | Start-VM -Confirm:$false
        }
        2
        {
            Get-VM $VMName | Stop-VM -Confirm:$false
        }
        3
        {
            Get-VM $VMName | Shutdown-VMGuest -Confirm:$false
        }
        4
        {
            Get-VM $VMName | Restart-VM -Confirm:$false
        }
        5
        {
            Get-VM $VMName | Restart-VMGuest -Confirm:$false
        }
    }
}

```

```

        6
        {
            if (IsVMExists ($VMName))
            {
                Delete_VM ($VMName)
            }
        }
        default{}
    }
    Start-Sleep -s $sleepTime
}

Disconnect-VIServer $vcAddress -Confirm:$false
exit

```

指令碼執行

下列訊息是來自指令碼的執行：

```

PowerCLI C:\scripts> .\VMOperations.ps1
Your vCenter address: 10.117.44.17
Your vCenter admin user name: administrator
Your vCenter admin user password: *****

-----
Select action: 1). Power On 2). Power Off 3) Shutdown VM Guest 4). Restart VM 5). Restart VM Guest
6). Delete VM: 1
Wait time (seconds) between each VM: 20
-----

Your selection is 6). Delete VM

```

針對開機、重新啟動虛擬機器和重新啟動虛擬機器客體作業，請指定虛擬機器之間的等候時間，至少 20 秒來避免開機風暴情況，該情況可能造成某些作業失敗。

疑難排解 Linux 桌面平台

當您管理 Linux 桌面平台時可能會發生某些問題。您可以遵循各種程序來診斷和修正問題。

本章節討論下列主題：

- 收集 Horizon 7 for Linux 機器的診斷資訊
- 疑難排解遠端桌面平台和用戶端主機之間的複製和貼上
- 設定 Linux 防火牆以允許連入 TCP 連線
- View Agent 無法在 iPad Pro Horizon Client 上中斷連線
- SLES 12 SP1 桌面平台在拖放之後不會自動重新整理
- SSO 無法連線到關閉電源代理程式
- 在建立 Linux 的手動桌面平台集區後無法連線虛擬機器

收集 Horizon 7 for Linux 機器的診斷資訊

您可以收集診斷資訊，協助 VMware 技術支援診斷並解決 Horizon 7 for Linux 機器的問題。您將建立 Data Collection Tool (DCT) 組合，該組合會取得機器的組態資訊並記錄至壓縮的 tarball。

程序

- 1 以具備所需權限的使用者身分登入 Linux 虛擬機器。
- 2 開啟命令提示字元，並執行 `dct-debug.sh` 指令碼。

```
sudo /usr/lib/vmware/viewagent/bin/dct-debug.sh
```

指令碼會產生包含 DCT 組合的 tarball。例如：

```
ubuntu-12-vdm-sdct-20150201-0606-agent.tgz
```

tarball 是在執行指令碼所在的目錄 (目前的工作目錄) 中產生。

疑難排解遠端桌面平台和用戶端主機之間的複製和貼上

在遠端桌面平台和用戶端主機之間複製和貼上需要超過三秒的時間，且最多支援 1 MB 的資料。如果您複製和貼上的資料大小較小，就不會發生此問題。

問題

為 SLED 11 SP3/SP4 桌面平台設定 1 個 vCPU 和 1 GB 記憶體時，在遠端桌面平台和本機用戶端主機之間進行複製和貼上可能需要超過三秒的時間。

原因

複製和貼上之所以會延遲，原因可能是 SLED 11 SP3/SP4 的作業系統 API 過舊。

解決方案

- ◆ 請為 SLED 11 SP3/SP4 設定兩個 vCPU 和 2 GB 記憶體。

設定 Linux 防火牆以允許連入 TCP 連線

若要允許使用者連接至其 Linux 桌面平台，桌面平台必須可以接受來自 Horizon Client 裝置、安全伺服器 and View 連線伺服器的連入 TCP 連線。

在 Ubuntu 和 Kylin 發行版上，iptables 防火牆預設會設定為使用「接受」的輸入原則。

在 RHEL 和 CentOS 發行版上，如有可能，Horizon Agent 安裝程式指令碼會將 iptables 防火牆設定為使用「接受」的輸入原則。

確定 RHEL 或 CentOS 客體作業系統上的 iptables 對來自 Blast 連接埠 22443 的新連線具有「接受」的輸入原則。

啟用 BSG 時，用戶端連線會從 Horizon Client 裝置透過安全伺服器或 View 連線伺服器上的 BSG 導向 Linux 桌面平台。未啟用 BSG 時，則是直接從 Horizon Client 裝置對 Linux 桌面平台連線。

View Agent 無法在 iPad Pro Horizon Client 上中斷連線

在 iPad Pro Horizon Client 上重新啟動或關閉後，SUSE View Agent 連線無法中斷。

問題

當您在 iPad Pro Horizon Client 上重新啟動或關閉 SUSE 虛擬機器時，桌面平台沒有回應。View Agent 無法中斷連線。

原因

在重新啟動或關閉作業後，SUSE 機器可能無法正確地將訊息傳送至 Horizon Client。

解決方案

- ◆ 從 iPad Pro Horizon Client 手動中斷桌面平台連線。

SLES 12 SP1 桌面平台在拖放之後不會自動重新整理

在您拖放 gnome 終端機後，SLES 12 SP1 不會在多監視器模式中自動重新整理。

問題

當您以多監視器模式啟動 SLES 12 SP1，然後回到視窗模式時，在您拖放 **gnome** 終端機後，桌面平台不會自動重新整理。

原因

gnome 終端機不會回應拖放作業。

解決方案

- 1 結束 **gnome-shell**。



```
kill -9 `pidof gnome-shell`
```

- 2 重新啟動 **gnome-shell**。

SSO 無法連線到關閉電源代理程式

SSO 不會連線到關閉電源代理程式。

問題

當您以代理的身分登入並連線到代理程式時，SSO 無法連線到關閉電源代理程式。

解決方案

- ◆ 手動登入桌面平台，或在中斷連線後重新連線到代理程式。

在建立 Linux 的手動桌面平台集區後無法連線虛擬機器

虛擬機器狀態為沒有回應。

問題

在您建立手動桌面平台集區後，虛擬機器的狀態可能是 [正在等待代理程式] 或 [無法連線]。

原因

有若干使用者錯誤組態或設定可能會導致虛擬機器處於 [無法連線] 或 [正在等待代理程式] 狀態。

- 確認選項 **machine.id** 存在於虛擬機器 **vmx** 組態檔中。

如果不存在，請確認虛擬機器已正確新增至桌面平台集區。若非如此，請重新建立桌面平台集區，讓代理將該選項重新寫入至 **vmx** 組態檔。

- 確認 **VMware Tool** 或 **Open VM Tool** 已正確安裝。

如果安裝 **VMware Tool** 或 **Open VM Tool** 的步驟未正確執行，則 **vmware-rpctool** 命令可能不會存在於 Linux 虛擬機器中的 **PATH** 下。您必須遵循指南來安裝 **VMware Tool** 或 **Open VM Tool**。

完成安裝後，請執行命令。

```
#vmware-rpctool "machine.id.get"
```

machine.id 值會從虛擬機器 vmx 組態檔中列出。

- 確認代理的 FQDN 是否可在代理程式 Linux 虛擬機器中解析為 IP 位址。