

Einrichten von Horizon 7 for Linux-Desktops

Letzte Änderung am 15. September 2017
VMware Horizon 7 7.2



vmware®

Die aktuellste technische Dokumentation finden Sie auf der VMware-Website unter:

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

Falls Sie Anmerkungen zu dieser Dokumentation haben, senden Sie diese an:

docfeedback@vmware.com

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

VMware Global, Inc.
Zweigniederlassung Deutschland
Willy-Brandt-Platz 2
81829 München
Germany
Tel.: +49 (0) 89 3706 17 000
Fax: +49 (0) 89 3706 17 333
www.vmware.com/de

Copyright © 2016, 2017 VMware, Inc. Alle Rechte vorbehalten. [Urheberrechts- und Markenhinweise](#).

Inhalt

Einrichten von Horizon 7 for Linux-Desktops	5
1 Funktionen und Systemanforderungen	6
Funktionen von Horizon Linux-Desktops	6
Übersicht über die Konfigurationsschritte für Horizon 7 for Linux-Desktops	10
Systemanforderungen für Horizon 7 for Linux	11
Einstellungen der virtuellen Maschine für 2D- oder vSGA-Grafiken	17
2 Vorbereiten einer virtuellen Linux-Maschine für die Desktop-Bereitstellung	21
Erstellen einer virtuellen Maschine und Installieren von Linux	21
Vorbereiten einer Linux-Maschine für die Remote-Desktop-Bereitstellung	22
Installieren von Abhängigkeitspaketen für Horizon Agent	25
3 Einrichten der Active Directory-Integration für Linux-Desktops	27
Integrieren von Linux mit Active Directory	27
Einrichten von Single Sign-On und Smartcard-Umleitung	28
4 Einrichten von Grafiken für Linux-Desktops	31
Konfigurieren unterstützter RHEL-Distributionen für vGPU	31
Installieren des VIB für die NVIDIA GRID vGPU-Grafikkarte auf dem ESXi-Host	32
Konfigurieren eines gemeinsam genutzten PCI-Geräts für vGPU auf der virtuellen Linux-Maschine	33
Installieren des NVIDIA GRID vGPU-Anzeigetreibers	35
Überprüfen, ob der NVIDIA-Anzeigetreiber installiert ist	37
Konfigurieren von RHEL 6 für vDGA	38
Aktivieren von DirectPath I/O für NVIDIA GRID auf einem Host	38
Hinzufügen eines vDGA-Pass-Through-Geräts zu einer virtuellen RHEL 6-Maschine	39
Installieren des NVIDIA-Anzeigetreibers für vDGA	40
Überprüfen, ob der NVIDIA-Anzeigetreiber installiert ist	41
Konfigurieren von RHEL 7 für vSGA	42
Installieren des VIB für die NVIDIA-Grafikkarte für vSGA auf dem ESXi-Host	42
Konfigurieren von 3D-Funktionen für vSGA auf einer virtuellen Linux-Maschine	43
Überprüfen, ob vSGA auf einer virtuellen Linux-Maschine ausgeführt wird	44
5 Installieren von Horizon Agent	46
Installieren von Horizon Agent auf einer virtuellen Linux-Maschine	46
Befehlszeilenoptionen für install_viewagent.sh	47
Konfigurieren des Zertifikats für den Linux Agent	49

Durchführen eines Upgrades von Horizon Agent auf einer virtuellen Linux-Maschine	50
Durchführen eines Upgrades von Horizon Agent auf einer virtuellen Linux-Maschine	51
Deinstallieren von Horizon 7 für Linux-Maschinen	52

6 Konfigurationsoptionen für Linux-Desktops 54

Einstellen der Optionen in Konfigurationsdateien auf einem Linux-Desktop	54
Beispiel für Blast-Einstellungen für Linux-Desktops	63
Beispiel für Optionen der Clientlaufwerksumleitung für Linux-Desktops	64
Ausblenden der vSphere-Konsolenanzeige auf einem Linux-Desktop	65

7 Erstellen und Verwalten von Linux-Desktop-Pools 66

Erstellen eines manuellen Desktop-Pools für Linux	66
Verwalten eines Desktop-Pools für Linux	67
Erstellen eines automatisierten Full-Clone-Desktop-Pools für Linux	69
Broker-PowerCLI-Befehle	71

8 Massenbereitstellung von Horizon 7 für manuelle Desktop-Pools 74

Überblick über die Massenbereitstellung von Linux-Desktops	75
Überblick über die Massenaktualisierung von Linux-Desktops	76
Erstellen einer Vorlage für virtuelle Maschinen zum Klonen von Linux-Desktop-Maschinen	77
Eingabedatei für die PowerCLI-Beispielskripts zur Bereitstellung von Linux-Desktops	79
Beispielskript zum Klonen von virtuellen Linux-Maschinen	80
Beispielskript zum Hinzufügen geklonter virtueller Maschinen zu einer AD-Domäne	84
Beispielskript zum Hinzufügen geklonter virtueller Maschinen zu einer AD-Domäne mithilfe von SSH	87
Beispielskript zum Hochladen von Konfigurationsdateien zu virtuellen Linux-Maschinen	91
Beispielskript zum Hochladen von Konfigurationsdateien zu virtuellen Linux-Maschinen mithilfe von SSH	94
Beispielskript zur Durchführung eines Upgrades von Horizon Agent auf Linux-Desktop-Maschinen	98
Beispielskript zur Durchführung eines Upgrades von Horizon Agent auf virtuellen Linux-Maschinen mithilfe von SSH	103
Beispielskript zum Ausführen von Vorgängen auf virtuellen Linux-Maschinen	109

9 Fehlerbehebung bei Linux-Desktops 113

Ermitteln von Diagnoseinformationen für eine Horizon 7 for Linux-Maschine	113
Durchführen der Fehlerbehebung für das Kopieren/Einfügen zwischen Remote-Desktop und Client-Host	114
Konfigurieren der Linux-Firewall für das Akzeptieren eingehender TCP-Verbindungen	114
Fehler beim Trennen der Verbindung auf dem Horizon Client für ein iPad Pro durch View Agent	115
SLES 12 SP1-Desktop wird nach Ziehen und Ablegen nicht automatisch aktualisiert	115
Fehlerhafte SSO-Verbindung zu einem PowerOff-Agenten	115
Nicht erreichbare VM nach dem Erstellen eines manuellen Desktop-Pools für Linux	116

Einrichten von Horizon 7 for Linux-Desktops

Das Dokument *Einrichtung von Horizon 7 for Linux-Desktops* bietet Informationen zur Einrichtung einer virtuellen Linux-Maschine als VMware Horizon[®] 7-Desktop, inklusive der Vorbereitung des Linux-Gastbetriebssystems, der Installation von Horizon Agent auf der virtuellen Maschine und der Konfiguration der Maschine in View Administrator für die Verwendung in einer Horizon 7-Bereitstellung.

Zielgruppe

Diese Informationen richten sich an alle, die Remote-Desktops konfigurieren und verwenden möchten, die auf Linux-Gastbetriebssystemen ausgeführt werden. Die Informationen wurden für erfahrene Linux-Systemadministratoren verfasst, die mit der Technologie virtueller Maschinen sowie mit Datencenter-Vorgängen vertraut sind.

Funktionen und Systemanforderungen

1

Mit Horizon 6 oder höher können die Benutzer eine Verbindung zu Remote-Desktops herstellen, auf denen das Linux-Betriebssystem ausgeführt wird.

Dieses Kapitel enthält die folgenden Themen:

- [Funktionen von Horizon Linux-Desktops](#)
- [Übersicht über die Konfigurationsschritte für Horizon 7 for Linux-Desktops](#)
- [Systemanforderungen für Horizon 7 for Linux](#)

Funktionen von Horizon Linux-Desktops

Horizon 7 Version 7.2 enthält verschiedene neue Funktionen für Linux-Desktops.

Die folgenden neuen Funktionen werden für Horizon 7 Version 7.2 unterstützt.

- Clientlaufwerksumleitung (Client Drive Redirection, CDR)
- Unterstützung der USB-Umleitung
- Unterstützung der Audioausgabe für HTML Access
- Single Sign-On-Unterstützung auf RHEL 7 Workstation x64 und CentOS 7 x64
- Unterstützung von KDE (K Desktop Environment) auf CentOS 6 x64 und RHEL 6 x64
- Unterstützung von RHEL 6.9 x64 und CentOS 6.9 x64

In der folgenden Liste sind die wichtigsten Funktionen für Horizon Linux Desktops enthalten.

Automatisierter Full-Clone-Desktop-Pool	Ab Horizon 7 Version 7.0.2 können Sie automatisierte Full-Clone-Desktop-Pools für Linux-Desktops erstellen.
Manueller Desktop-Pool	Computerquelle. <ul style="list-style-type: none">■ Verwaltete virtuelle Maschine: Computerquelle der virtuellen vCenter-Maschine. Eine verwaltete virtuelle Maschine wird für eine neue und eine Upgrade-Bereitstellung unterstützt.

- Verwaltung der virtuellen Maschine aufheben: Computerquelle anderer Quellen. Eine nicht verwaltete virtuelle Maschine wird nur beim Upgrade von einer Bereitstellung mit aufgehobener Verwaltung der virtuellen Maschine unterstützt.

Hinweis Um die bestmögliche Leistung zu gewährleisten, sollten Sie keine nicht verwaltete virtuelle Maschine verwenden.

Mehrere Monitore

- vDGA/vGPU-Desktop unterstützt eine maximale Auflösung von 2560x1600 auf vier Monitoren.
- 2D/vSGA-Desktop auf vSphere 6.0 oder höher unterstützt eine maximale Auflösung von 2048x1536 auf vier Monitoren oder eine maximale Auflösung von 2560x1600 auf drei Monitoren.

Bei Ubuntu 14.04/16.04 müssen Sie für die Unterstützung mehrerer Monitore den Fenstermanager Compiz deaktivieren. Weitere Informationen finden Sie unter <http://kb.vmware.com/kb/2114809>.

Für SLES 12 SP1 müssen Sie das Standardpaket mit der Kernelebene kernel-default-3.12.49-11.1 verwenden. Wenn Sie ein Upgrade für das Paket durchgeführt haben, kann die Funktion für mehrere Monitore nicht benutzt werden. Der Desktop wird nur in einem Monitor angezeigt.

VMware Horizon HTML Access unterstützt die Funktion für mehrere Monitore nicht für Horizon 7 for Linux-Desktops.

Verlustfreier PNG-Modus

Bilder und Videos, die auf einem Desktop erzeugt werden, werden auf dem Clientgerät pixelgenau gerendert.

Software-H.264-Encoder

Die H.264-Decodierung kann die Blast Extreme-Leistung für Horizon Desktop verbessern, insbesondere bei Netzwerken mit niedriger Bandbreite. Wenn die Clientseite H.264 deaktiviert hat, wird Blast Extreme automatisch auf die Verwendung der JPEG-/PNG-Codierung zurückgesetzt.

Ab Horizon 7 Version 7.0.3 werden mehrere Monitore unterstützt.

3D-Grafiken

3D-Grafiken werden bei den folgenden Kombinationen von Linux-Versionen und Grafikkarten unterstützt:

- vSGA wird auf RHEL 7 Workstation x64 mit NVIDIA GRID K1- oder K2-Grafikkarten unterstützt.
- vDGA wird auf RHEL 6 Workstation x64 mit NVIDIA GRID K1- oder K2-Grafikkarten unterstützt.
- vGPU wird auf RHEL 6 Workstation x64 mit NVIDIA Maxwell M60-Grafikkarten unterstützt.

- vGPU wird auf RHEL 7 Workstation x64 mit NVIDIA Maxwell M60-Grafikkarten unterstützt.
- vGPU wird auf RHEL 6 Workstation x64 mit NVIDIA M6-Grafikkarten unterstützt.
- vGPU wird auf RHEL 7 Workstation x64 mit NVIDIA M6-Grafikkarten unterstützt.

Zwischenablagenumleitung

Mit der Zwischenablagenumleitung können Sie RTF- oder reinen Text zwischen einem Clienthost und einem Linux-Remote-Desktop kopieren und einfügen. Sie können mithilfe der Optionen von Horizon Agent die Richtung und die maximale Textgröße für das Kopieren/Einfügen festlegen. Diese Funktion ist standardmäßig aktiviert. Sie können die Funktion bei der Installation deaktivieren.

Single Sign-On

Single Sign-On wird von den folgenden Linux-Versionen unterstützt:

- RHEL 6/7 Workstation x64
- CentOS 6/7 x64
- SLED 11 SP3/SP4 x64
- Ubuntu 14.04/16.04 x64

Smartcard-Umleitung mit SSO

Die Smartcard-Umleitung wird auf RHEL 6 Workstation x64 unterstützt. Es werden Smartcards vom Typ PIV (Personal Identity Verification) und CAC (Common Access Card) unterstützt. Mac-Client wird nicht unterstützt.

Audio-Eingang

Die Umleitung des Audio-Eingangs von einem Clienthost zu einem Linux-Remote-Desktop wird unterstützt. Diese Funktion basiert nicht auf der Funktion der USB-Umleitung. Wenn diese Funktion aktiviert werden soll, müssen Sie diese bei der Installation auswählen. Gleichzeitig müssen Sie das standardmäßige System-Audio im Gerät „PulseAudio Server (lokal)“ in Ihrer Anwendung für den Audio-Eingang auswählen. Die Funktion wird für folgende Distributionen unterstützt:

- Ubuntu 14.04/16.04 x64
- CentOS 7 x64
- RHEL 7 Workstation x64

Audioausgabe

Die Umleitung der Audioausgabe wird unterstützt. Diese Funktion ist standardmäßig aktiviert. Um diese Funktion zu deaktivieren, müssen Sie für die Option `RemoteDisplay.allowAudio` **false** festlegen. Ab Horizon 7 Version 7.2 bietet VMware Horizon HTML Access bei Verwendung von Chrome- und Firefox-Browsern eine Unterstützung der Audio-Ausgabe für Linux-Desktops.

Clientlaufwerksumleitung

Die Funktion der Clientlaufwerksumleitung (Client Drive Redirection, CDR) ist ab Horizon 7 Version 7.2 verfügbar. Wenn Sie die CDR-Funktion aktivieren, können Sie auf die freigegebenen Ordner und Laufwerke Ihres lokalen Systems zugreifen. Sie verwenden dazu den `tsc\client`-Ordner in Ihrem Stammverzeichnis auf dem Remote-Linux-Desktop. Um diese Funktion verwenden zu können, müssen Sie die CDR-Komponenten installieren.

USB-Umleitung

Die Funktion der USB-Umleitung ist ab Horizon 7 Version 7.2 verfügbar. Mit der Funktion haben Sie von Linux-Remote-Desktops aus Zugriff auf lokal angeschlossene USB-Geräte. Sie müssen die Komponenten der USB-Umleitung und das USB-VHCI-Treiber-Kernelmodul installieren, um die USB-Funktion verwenden zu können. Stellen Sie sicher, dass Sie über ausreichend Rechte für die Verwendung des USB-Geräts verfügen, das Sie umleiten möchten.

Hinweis Das USB 3.0-Protokoll wird in Horizon 7 Version 7.2 unterstützt.

Synchronisierung von Tastaturlayout und Gebietsschema

Diese Funktion legt fest, ob das Systemgebietsschema und das aktuelle Tastaturlayout eines Clients mit den Horizon Linux Agent-Desktops synchronisiert werden sollen. Wenn diese Einstellung aktiviert wurde oder nicht konfiguriert ist, ist eine Synchronisierung zugelassen. Wenn diese Einstellung deaktiviert ist, ist eine Synchronisierung nicht erlaubt.

Diese Funktion wird nur für Horizon Client für Windows und für die Gebietsschemas Englisch, Französisch, Deutsch, Japanisch, Koreanisch, Spanisch, Chinesisch (vereinfacht) und Chinesisch (traditionell) unterstützt.

K Desktop Environment

In Horizon 7 Version 7.2 wird KDE (K Desktop Environment) auf den im Folgenden aufgeführten Linux-Distributionen unterstützt.

- CentOS 6 x64
- RHEL 6 x64

FIPS 140-2-Modus

Die Unterstützung des FIPS 140-2-Modus (Federal Information Processing Standard) ist zwar noch nicht NIST CMVP-validiert (Cryptographic Module Validation Program), ist jedoch schon ab Horizon 7 Version 7.2 verfügbar.

Horizon 7 Agent für Linux implementiert kryptografische Module, die auf FIPS 140-2-Kompatibilität ausgelegt sind. Diese Module wurden in im CMVP-Zertifikat 2839 und 2866 aufgelisteten Betriebsumgebungen überprüft und auf diese Plattform portiert. Allerdings muss die CAVP- und CMVP-Testanforderung, die neuen Betriebsumgebungen in den NIST CAVP- und CMVP-Zertifikaten von VMware aufzuführen, im Rahmen der Produkt-Roadmap noch erfüllt werden.

Hinweis Für die Unterstützung des FIPS 140-2-Modus ist das TLS-Protokoll (Transport Layer Security-Protokoll) Version 1.2 erforderlich.

Für Linux-Desktops und Desktop-Pools gelten die folgenden Einschränkungen:

- Virtueller Druck, standortbasiertes Drucken und Echtzeit-Video werden nicht unterstützt.

Hinweis Wird ein Sicherheitsserver verwendet, muss der Port 22443 in der internen Firewall geöffnet sein, damit ein Datenverkehr zwischen dem Sicherheitsserver und dem Linux-Desktop möglich ist.

Übersicht über die Konfigurationsschritte für Horizon 7 for Linux-Desktops

Wenn Sie Horizon 7 for Linux-Desktops installieren und konfigurieren, müssen Sie unterschiedliche Schritte durchführen, je nachdem, ob Sie 2D-Grafiken oder 3D-Grafiken auf den virtuellen Maschinen installieren.

2D-Grafiken – Überblick über die Konfigurationsschritte

Für 2D-Grafiken führen Sie die folgenden Schritte durch:

- 1 Überprüfen Sie die Systemanforderungen für die Einrichtung einer Horizon 7 for Linux-Bereitstellung. Siehe [Systemanforderungen für Horizon 7 for Linux](#).
- 2 Erstellen Sie in vSphere eine virtuelle Maschine und installieren Sie das Linux-Betriebssystem. Siehe [Erstellen einer virtuellen Maschine und Installieren von Linux](#).
- 3 Bereiten Sie das Gastbetriebssystem für eine Bereitstellung als Desktop in einer Horizon 7-Umgebung vor. Siehe [Vorbereiten einer Linux-Maschine für die Remote-Desktop-Bereitstellung](#).
- 4 Konfigurieren Sie das Linux-Gastbetriebssystem zur Authentifizierung mit Active Directory. Dieser Schritt wird mit einer externen Software auf der Basis der Anforderungen in Ihrer Umgebung durchgeführt und nicht in diesem Handbuch beschrieben.
- 5 Installieren Sie Horizon Agent auf der virtuellen Linux-Maschine. Siehe [Installieren von Horizon Agent auf einer virtuellen Linux-Maschine](#).
- 6 Erstellen Sie einen Desktop-Pool mit den konfigurierten virtuellen Linux-Maschinen. Siehe [Erstellen eines manuellen Desktop-Pools für Linux](#).

3D-Grafiken – Überblick über die Konfigurationsschritte

Sie müssen die Konfiguration von NVIDIA GRID vGPU, vDGA oder vSGA auf den virtuellen Linux-Maschinen abschließen, bevor Sie Horizon Agent auf den Maschinen installieren und einen Desktop-Pool in View Administrator bereitstellen.

- 1 Überprüfen Sie die Systemanforderungen für die Einrichtung einer Horizon 7 for Linux-Bereitstellung. Siehe [Systemanforderungen für Horizon 7 for Linux](#).
- 2 Erstellen Sie in vSphere eine virtuelle Maschine und installieren Sie das Linux-Betriebssystem. Siehe [Erstellen einer virtuellen Maschine und Installieren von Linux](#).
- 3 Bereiten Sie das Gastbetriebssystem für eine Bereitstellung als Desktop in einer Horizon 7-Umgebung vor. Siehe [Vorbereiten einer Linux-Maschine für die Remote-Desktop-Bereitstellung](#).
- 4 Konfigurieren Sie das Linux-Gastbetriebssystem zur Authentifizierung mit Active Directory. Dieser Schritt wird mit einer externen Software auf der Basis der Anforderungen in Ihrer Umgebung durchgeführt und nicht in diesem Handbuch beschrieben.
- 5 Konfigurieren Sie 3D-Funktionen auf Ihren ESXi-Hosts und der virtuellen Linux-Maschine. Folgen Sie den Vorgehensweisen für die 3D-Funktion, die Sie installieren möchten.
 - Siehe [Konfigurieren unterstützter RHEL-Distributionen für vGPU](#).
 - Siehe [Konfigurieren von RHEL 6 für vDGA](#).
 - Siehe [Konfigurieren von RHEL 7 für vSGA](#).
- 6 Installieren Sie Horizon Agent auf der virtuellen Linux-Maschine. Siehe [Installieren von Horizon Agent auf einer virtuellen Linux-Maschine](#).
- 7 Erstellen Sie einen Desktop-Pool mit den konfigurierten virtuellen Linux-Maschinen. Siehe [Erstellen eines manuellen Desktop-Pools für Linux](#).

Massenbereitstellung

Mit View Administrator können Sie nur virtuelle Linux-Maschinen in einem manuellen Desktop-Pool bereitstellen. vSphere PowerCLI bietet dagegen die Möglichkeit zur Entwicklung von Skripten, die die Bereitstellung eines Pools von Linux-Desktop-Maschinen automatisieren. Siehe [Kapitel 8 Massenbereitstellung von Horizon 7 für manuelle Desktop-Pools](#).

Systemanforderungen für Horizon 7 for Linux

Für die Installation von Horizon 7 for Linux muss Ihr Linux-System bestimmte Anforderungen in Bezug auf Betriebssystem, Horizon 7 und die vSphere-Plattform erfüllen.

Unterstützte Linux-Versionen für Horizon Agent

Die folgende Tabelle enthält eine Aufstellung der Linux-Betriebssysteme, die auf virtuellen Maschinen in einem Desktop-Pool unterstützt werden.

Tabelle 1-1. Unterstützte Linux-Betriebssysteme für View Agent

Linux-Distribution	Architektur
Ubuntu 14.04 und 16.04	x64
Hinweis Deaktivieren Sie Compiz, um die Leistungseinbußen zu vermeiden. Weitere Informationen finden Sie unter http://kb.vmware.com/kb/2114809 .	
Ubuntu 12.04	x64
RHEL 6.6, 6.7, 6.8, 6.9, 7.2 und 7.3	x64
CentOS 6.6, 6.7, 6.8, 6.9, 7.2 und 7.3	x64
NeoKylin 6 Update 1	x64
SLED 11 SP3/SP4, SLED 12 SP1/SP2	x64
SLES 12 SP1/SP2	x64

Hinweis Der Linux Agent verfügt bei einigen Linux-Distributionen über Abhängigkeitspakete. Weitere Informationen finden Sie unter [Installieren von Abhängigkeitspaketen für Horizon Agent](#).

Erforderliche Plattform- und Horizon 7-Softwareversionen

Um Horizon 7 for Linux installieren und anwenden zu können, muss Ihre Bereitstellung bestimmte Anforderungen für die vSphere-Plattform, Horizon 7 und die Horizon Client-Software erfüllen.

Tabelle 1-2. Erforderliche Plattform- und Horizon 7-Softwareversionen

Plattform und Software	Unterstützte Versionen
vSphere-Plattformversion	<ul style="list-style-type: none"> ■ vSphere 6.0 U2 oder eine höhere Version ■ vSphere 6.5 oder eine höhere Version
Horizon-Umgebung	<ul style="list-style-type: none"> ■ Horizon-Verbindungsserver 7.2
Horizon Client-Software	<ul style="list-style-type: none"> ■ Horizon Client 4.5.0 für Android ■ Horizon Client 4.5.0 für Windows ■ Horizon Client 4.5.0 für Linux ■ Horizon Client 4.5.0 für Mac OS X ■ Horizon Client 4.5.0 für iOS (iPad Pro) ■ HTML Access 4.5.0 mit Chrome, Firefox und InternetExplorer ■ Zero Clients werden nicht unterstützt

Von virtuellen Linux-Maschinen verwendete TCP-Ports

View Agent und Horizon Clients verwenden TCP-Ports für den Netzwerkzugriff untereinander und zwischen den verschiedenen View Server-Komponenten.

Tabelle 1-3. Von virtuellen Linux-Maschinen verwendete TCP-Ports

Quelle	Port	Ziel	Port	Protokoll	Beschreibung
Horizon Client	*	Linux Agent	22443	TCP	Blast, wenn Blast Security Gateway nicht verwendet wird
Sicherheitsserver, View-Verbindungsserver oder Access Point-Appliance	*	Linux Agent	22443	TCP	Blast, wenn Blast Security Gateway verwendet wird
View Agent	*	View-Verbindungsserver	4001, 4002	TCP	JMS-SSL-Datenverkehr.

Hinweis Weitere Informationen zu den von Clients verwendeten TCP- und UDP-Ports finden Sie unter „Von Clients und View verwendete TCP- und UDP-Ports“ im Dokument *Horizon Client- und View Agent-Sicherheit* in View.

Überprüfen des von virtuellen Linux-Maschinen verwendeten Linux-Kontos

Die nachfolgende Tabelle enthält den von virtuellen Linux-Maschinen verwendeten Kontonamen und den Kontotyp.

Tabelle 1-4. Kontoname und Kontotyp

Kontoname	Kontotyp	Verwendet von
Stammordner	Integriertes Linux-Betriebssystem	Java Standalone Agent, mksvchanserver, Shell-Skripts
vmwblast	erstellt durch das Linux Agent-Installationsprogramm	VMwareBlastServer
<Aktuell angemeldeter Benutzer>	Integriertes Linux-Betriebssystem oder AD-Benutzer oder LDAP-Benutzer	Python-Skript

Desktop-Umgebung

Horizon 7 for Linux Desktop unterstützt mehrere Desktop-Umgebungen auf unterschiedlichen Linux-Distributionen. Die folgende Tabelle enthält die Standard-Desktop-Umgebungen für jede Linux-Distribution, die von Horizon 7 for Linux Desktop unterstützt wird.

Tabelle 1-5. Unterstützte Desktop-Umgebungen

Linux-Distribution	Standard-Desktop Umgebung	Von Horizon 7 for Linux Desktop unterstützte Desktop-Umgebung
Ubuntu 14.04/16.04	Unity	Gnome Fallback/Flashback (Metacity)
Ubuntu 12.04	Unity	Unity
RHEL/CentOS 6	Gnome	Gnome, K Desktop Environment (KDE)

Linux-Distribution	Standard-Desktop	
	Umgebung	Von Horizon 7 for Linux Desktop unterstützte Desktop-Umgebung
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

Zur Installation der Gnome Fallback/Flashback-Desktop-Umgebung (Metacity) auf Ubuntu 14.04/16.04 finden Sie weitere Informationen im VMware-KB-Artikel <http://kb.vmware.com/kb/2114809>.

Beachten Sie die im Folgenden aufgeführten Hinweise zur Installation von KDE als Standard-Desktop-Umgebung für RHEL/CentOS 6-Distributionen.

Hinweis Die Single Sign-On-Anmeldung (SSO) für KDE funktioniert nur, wenn Sie diese mithilfe der nachfolgend beschriebenen Schritte installieren.

- 1 Installieren Sie das RHEL/CentOS 6-Betriebssystem mit der Standardeinstellung für die Desktop-Umgebung. Damit wird Gnome installiert.
- 2 Installieren Sie KDE mit dem im Folgenden aufgeführten Befehl.

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

Wenn Sie die SSO-Anmeldung für eine Linux-Distribution aktivieren, auf der mehrere Desktop-Umgebungen installiert sind, beachten Sie die nachfolgend aufgeführten Hinweise zur Auswahl der Desktop-Umgebung.

- Für Ubuntu-14.04/16.04 aktivieren Sie in der Datei `/etc/vmware/viewagent-custom.conf` die Option `UseGnomeFlashback`, um für den Endbenutzer die Verwendung von Gnome Fallback/Flashback (Metacity) verbindlich festzulegen.
- Für RHEL/CentOS 6 müssen die Endbenutzer den Desktop-Umgebungsnamen `kde` oder `gnome` in der Datei `~/.dmrc` angeben, um festzulegen, welche Desktop-Umgebung in der nächsten SSO-Anmeldungssitzung verwendet wird. Um beispielsweise KDE als Standard-Desktop-Umgebung verwenden zu können, muss der Endbenutzer die folgende Einstellung in seine Datei `~/.dmrc` aufnehmen:

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

Nach der ersten Einrichtung muss sich der Endbenutzer abmelden oder das Linux-System neu starten, um KDE im obigen Beispiel als Standard-Desktop in seiner nächsten Anmeldungssitzung festzulegen.

Wenn Sie die SSO-Anmeldung auf einer Linux-Distribution mit mehreren installierten Desktop-Umgebungen deaktiviert haben, muss der Endbenutzer die gewünschte Desktop-Umgebung auswählen, wenn er sich bei dieser Linux-Distribution anmeldet.

VHCI-Treiber für die USB-Umleitung

Die Funktion der USB-Umleitung wird ab Horizon 7 Version 7.1 für Linux-Desktops unterstützt. Die Funktion verfügt über eine Abhängigkeit vom Kernaltreiber des USB Virtual Host Controller Interface (VHCI) Sie müssen einen Patch für den VHCI-Treiber zur Unterstützung von USB 3.0 installieren.

Das Installationsprogramm für Horizon for Linux umfasst die Binärdatei des VHCI-Treibers für den Standardkernel der unterstützten Linux-Distributionen. Damit wird der VHCI-Treiber installiert, wenn die Funktion zur USB-Umleitung ausgewählt wird. Die nachfolgende Tabelle enthält die Standardkernelversionen, die vom Installationsprogramm für den Horizon for Linux Desktop installiert werden.

Tabelle 1-6. Standardkernelversionen

Linux-Distribution	Standardkernelversion
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

Wenn Ihr Linux-System eine andere Kernelversion verwendet, als die im Installationsprogramm von Horizon for Linux enthaltenen Standardversionen, müssen Sie den USB-VHCI-Quellcode von <https://sourceforge.net/projects/usb-vhci/files/linux%20kernel%20module/> herunterladen. Sie müssen dann den Quellcode des VHCI-Treibers kompilieren und die damit generierte Binärdatei auf Ihrem Linux-System installieren.

Hinweis Der VHCI-Treiber muss vor der Installation von Horizon for Linux installiert werden.

Wenn Sie die aktuelle Version des Quellcodes des USB-VHCI-Treibers heruntergeladen haben, kompilieren Sie diesen mit den nachfolgend aufgeführten Befehlen und installieren Sie ihn auf Ihrem Linux-System. Wenn Sie beispielsweise die Installationsdatei VMware-horizonagent-linux-x86_64-*<Version>-<Build-Nummer>.tar.gz* im Verzeichnis /install_tmp/ entpacken, ist /Install_tmp/VMware-horizonagent-linux-x86_64-*<Version>-<Build-Nummer>/resources/vhci/patch/vhci.patch* die *full-path-to-patch-file*. Der patch-Befehl, der verwendet werden muss, lautet:

```
# patch -p1 < /install_tmp/VMware-horizonagent-linux-x86_64-<Version>-<Build-Nummer>/resources/vhci/patch/vhci.patch
```

Tabelle 1-7. Kompilieren und Installieren der VHCI-USB-Treiber

Linux-Distribution	Schritte zum Kompilieren und Installieren der VHCI-USB-Treiber
Ubuntu 14.04 Ubuntu 16.04	<p>Kompilieren und installieren Sie die VHCI-Treiber.</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	<p>1 Installieren Sie die Abhängigkeitspakete.</p> <pre># yum install gcc-c++ # yum install kernel-devel-\$(uname -r) # yum install kernel-headers-\$(uname -r) # yum install patch</pre> <p>2 Kompilieren und installieren Sie die VHCI-Treiber.</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>
SUSE 11 SP4 SUSE 12 SP2	<p>1 Ermitteln Sie die Version des aktuellen Kernelpakets.</p> <pre># rpm -qa grep kernel-default-\$(echo \$(uname -r) cut -d '-' -f 1,2)</pre> <p>Die Ausgabe besteht aus dem Namen des aktuell installierten Kernelpakets. Wenn der Name des Pakets z. B. kernel-default-3.0.101-63.1 lautet, ist die aktuelle Version des Kernelpakets 3.0.101-63.1.</p> <p>2 Installieren Sie die Abhängigkeitspakete.</p> <p>a Bei SUSE 11 installieren Sie die Pakete kernel-source und kernel-default-devel, die dem aktuellen Kernel entsprechen, und die Pakete gcc und patch.</p> <pre>#zypper install --oldpackage kernel-source-<Kernelpaketversion> \ kernel-default-devel-<Kernelpaketversion> gcc patch</pre> <p>Beispiel:</p> <pre>#zypper install --oldpackage kernel-source-3.0.101-63.1 kernel-default- devel-3.0.101-63.1 gcc patch</pre> <p>b Bei SUSE 12 installieren Sie die Pakete kernel-devel, kernel-default-devel, kernel-macros und patch packages.</p> <pre>#zypper install --oldpackage kernel-devel-<Kernelpaketversion> \ kernel-default-devel-<Kernelpaketversion> kernel-macros-<Kernelpaketversion> patch</pre> <p>Beispiel:</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> <p>3 Kompilieren und installieren Sie die VHCI-Treiber.</p> <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</pre>

Wenn sich Ihre Linux-Kernelversion ändert, müssen Sie den VHCI-Treiber erneut kompilieren und installieren. Horizon for Linux muss aber nicht erneut installiert werden.

Einstellungen der virtuellen Maschine für 2D- oder vSGA-Grafiken

Wenn Sie bestimmte virtuelle Horizon 7 for Linux-Maschinen erstellen, müssen Sie die Einstellungen für den Arbeitsspeicher und die Konfigurationsparameter auf die empfohlenen Mindestwerte ändern.

Virtuelle Maschinen, die für die Verwendung von NVIDIA vDGA konfiguriert sind, benutzen die physische NVIDIA-Grafikkarte. Virtuelle Maschinen, die für die Verwendung von NVIDIA GRID vGPU konfiguriert sind, benutzen die virtuelle NVIDIA-Grafikkarte, die auf der Beschleunigung der physischen NVIDIA-Grafikkarte basiert. Sie müssen die Einstellungen für den Videospeicher (vRAM) und die Konfigurationsparameter für diese virtuellen Maschinen nicht ändern.

Virtuelle Maschinen, die für die Verwendung von 2D- oder vSGA-Grafiken konfiguriert sind, benutzen die virtuelle VMware-Grafikkarte. Für diese Art von virtuellen Maschinen müssen Sie folgende Einstellungen ändern:

- Videospeichereinstellungen (vRAM)
- Konfigurationsparameter
- Einstellungen für 3D-Arbeitsspeicher
- Einstellungen für vCPU und virtuellen Arbeitsspeicher zur Leistungsverbesserung

Videospeichereinstellungen (vRAM)

Wenn Sie im vSphere Client eine virtuelle Linux-Maschine erstellen, müssen Sie die vRAM-Größe wie in [Tabelle 1-8. Empfohlene vRAM-Einstellungen für 2D- oder vSGA-Grafiken](#) dargestellt konfigurieren. Legen Sie die vRAM-Größe fest, die für die Anzahl und Auflösung der von Ihnen für die virtuelle Maschine konfigurierten Monitore empfohlen wird.

Tabelle 1-8. Empfohlene vRAM-Einstellungen für 2D- oder vSGA-Grafiken

vRAM-Größe	Anzahl der Monitore	Maximale Auflösung
10 MB	1	1600x1200 oder 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

Diese vRAM-Werte stellen Mindestempfehlungen dar. Wenn auf der virtuellen Maschine mehr Ressourcen verfügbar sind, können Sie die vRAM-Größe für eine verbesserte Videoleistung erhöhen.

10 MB wird als Mindest-vRAM-Wert für eine mit einem Monitor mit der niedrigsten Auflösung konfigurierte Maschine empfohlen.

Um die Anzahl der Anzeigegeräte und die Größe des verwendeten Videospeichers (vRAM) wie in [Erstellen einer virtuellen Maschine und Installieren von Linux](#) beschrieben festzulegen, müssen Sie die virtuelle Maschine ausschalten.

Der Horizon-Verbindungsserver 7 konfiguriert die vRAM-Einstellungen auf virtuellen Linux-Maschinen nicht automatisch wie etwa auf virtuellen Windows-Maschinen. Sie müssen deswegen die vRAM-Einstellungen manuell im vSphere Client konfigurieren.

Wird eine virtuelle Linux-Maschine mit einer geringeren vRAM-Größe konfiguriert, treten eventuell folgende Probleme auf:

- Desktop-Sitzungen können unmittelbar nach der ersten Herstellung der Verbindung wieder getrennt werden.
- Die automatische Anpassung funktioniert nicht. Der Desktop wird dann nur in einem kleinen Bereich des Bildschirms dargestellt.

Wenn der Wert für **Anzahl der Anzeigegeräte** einer virtuellen Linux-Maschine kleiner ist als tatsächlich erforderlich, werden einer oder mehrere Monitore für den Desktop leer angezeigt.

Wenn ein Problem mit der automatischen Anpassung auftritt, können Sie eine höhere vRAM-Größe festlegen. vSphere Client ermöglicht eine maximale vRAM-Größe von 128 MB. Wenn Sie mehr als 128 MB für die vRAM-Größe festlegen möchten, müssen Sie die VMX-Konfigurationsdatei manuell ändern. Das folgende Beispiel legt eine vRAM-Größe von 256 MB fest:

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

Konfigurationsparameter

Um den Linux-Remote-Desktop auf mehreren Monitoren anzuzeigen, müssen Sie bestimmte Konfigurationsparameter für die virtuelle Maschine festlegen. Die Festlegung eines Konfigurationsparameters für eine virtuelle Maschine wird allgemein folgendermaßen vorgenommen:

- 1 Schalten Sie die virtuelle Maschine aus
- 2 Klicken Sie im vSphere Web Client mit der rechten Maustaste auf die virtuelle Maschine, und wählen Sie **Einstellungen bearbeiten** aus.
- 3 Klicken Sie auf die Registerkarte **VM-Optionen** und dann auf **Erweitert**.
- 4 Klicken Sie auf **Konfiguration bearbeiten** und dann auf **Zeile hinzufügen**.
- 5 Geben Sie den Namen und den Wert des Konfigurationsparameters ein.
- 6 Klicken Sie auf **OK**, um die Änderungen zu speichern.

Sie müssen die folgenden Konfigurationsparameter festlegen:

- Legen Sie für `svga.autodetect` „False“ fest.

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

- Berechnen Sie die Werte für `svga.maxWidth` und `svga.maxHeight` entsprechend der Anzahl und der Ausrichtung (horizontal oder vertikal) der Anzeigemonitore. Als allgemeine Regel gilt, dass die Werte für `svga.maxWidth` und `svga.maxHeight` groß genug sein müssen, um alle Bildschirme zu unterstützen. Beispielsweise müssen Sie zur Unterstützung von vier Anzeigegeräten mit einer maximalen Auflösung von 2560x1600 die folgenden Werte festlegen:

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

Wenn Sie über mehrere Monitore verfügen, ist die Festlegung dieser Konfigurationsparameter obligatorisch. Ansonsten können ein oder mehrere der folgenden Probleme auftreten:

- Der Desktop wird auf einigen Monitoren angezeigt, die anderen Monitore bleiben leer.
- Eine Tastatureingabe wird mehrfach angezeigt.
- Der Desktop wird verlangsamt.
- Der Desktop wird nur in einem kleinen Bereich des Bildschirms dargestellt.

Monitore mit auf 4096x4096 eingeschränkter Bildschirmgröße

Für RHEL 6.8/6.9/7.3, CentOS 6.8/6.9/7.3, Ubuntu 16.04, SLED 12 SP2, und SLES 12 SP2 beträgt die maximale Bildschirmgröße für 2D und vSGA 4096x4096. Wenn Sie den Befehl `xrandr` ausführen, zeigt die erste Zeile der Ausgabe `Maximum 4096x4096` an.

Für Ubuntu 14.04 gilt die Einschränkung auch, wenn Sie die aktuellen Patches der offiziellen Ubuntu-Repositorys installieren.

Für eine Verbindung mit mehreren Monitoren ist eventuell eine höhere Bildschirmgröße als 4096x4096 erforderlich. Um diese Einschränkung zu umgehen, verwenden Sie eine der folgenden Lösungen:

- Wenn Sie VMware Hardware Version 11 (HWv11) oder höher für Ihre virtuelle Maschine verwenden müssen, fügen Sie der VMX-Konfigurationsdatei der virtuellen Maschine folgende Zeile hinzu:

```
mks.enable3d = TRUE
```

Mit dieser Lösung kann das Linux-Betriebssystem 3D-Funktionen an Softwareanwendungen wie Chrome übermitteln. Allerdings wird dadurch eventuell die Leistung des Linux-Systems beeinträchtigt.

- Wenn Ihre virtuelle Maschine VMware Hardware Version 10 verwenden kann, fügen Sie die folgende Zeile in die VMX-Datei Ihrer virtuellen Maschine ein:

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

Diese Lösung ist nicht für RHEL 6.8 und CentOS 6.8 anwendbar.

Damit diese Lösung auf Ubuntu 16.04-Systemen wirksam werden kann, müssen Sie die aktuellen Patches der offiziellen Ubuntu-Repositorys installieren.

vCPU- und Arbeitsspeichereinstellungen

Wenn Sie die Leistung eines 2D- oder vSGA-Desktops verbessern möchten, können Sie mehr vCPUs und einen größeren Arbeitsspeicher für die virtuelle Linux-Maschine festlegen. So können Sie z. B. zwei vCPUs und zwei GB virtuellen Arbeitsspeicher verwenden.

Für einen großen Bildschirm mit mehreren Monitoren (z. B. vier Monitore) legen Sie 4 vCPUs und 4 GB virtuellen Arbeitsspeicher für die virtuelle Maschine fest.

Für die Videowiedergabe in einem 2D- oder vSGA-Desktop geben Sie 4 vCPUs und 4 GB virtuellen Arbeitsspeicher für die virtuelle Maschine an.

Einstellungen für 3D-Arbeitsspeicher

Um die Leistung in einer vSGA-Umgebung mit mehreren Monitoren zu verbessern, setzen Sie die Einstellung **3D-Arbeitsspeicher** auf 1 GB oder höher.

Vorbereiten einer virtuellen Linux-Maschine für die Desktop-Bereitstellung

2

Das Einrichten eines Linux-Desktops umfasst das Erstellen einer virtuellen Linux-Maschine und das Vorbereiten des Betriebssystems auf die Remote-Desktop-Bereitstellung.

Dieses Kapitel enthält die folgenden Themen:

- [Erstellen einer virtuellen Maschine und Installieren von Linux](#)
- [Vorbereiten einer Linux-Maschine für die Remote-Desktop-Bereitstellung](#)
- [Installieren von Abhängigkeitspaketen für Horizon Agent](#)

Erstellen einer virtuellen Maschine und Installieren von Linux

Sie erstellen eine neue virtuelle Maschine für jeden in Horizon 7 bereitgestellten Remote-Desktop in vCenter Server. Dazu müssen Sie Ihre Linux-Distribution auf der virtuellen Maschine installieren.

Voraussetzungen

- Stellen Sie sicher, dass Ihre Bereitstellung den Anforderungen für die Unterstützung von Linux-Desktops entspricht. Siehe [Systemanforderungen für Horizon 7 for Linux](#).
- Machen Sie sich mit den Schritten für das Erstellen virtueller Maschinen in vCenter Server und mit der Installation von Gastbetriebssystemen vertraut. Unter „Erstellen und Vorbereiten virtueller Maschinen“ im *Einrichten von virtuellen Desktops in Horizon 7*-Dokument finden Sie dazu Erläuterungen.
- Informieren Sie sich über die empfohlenen Videospeicherwerte (vRAM) für die Monitore, die Sie mit der virtuellen Maschine verwenden möchten. Siehe [Systemanforderungen für Horizon 7 for Linux](#).

Verfahren

- 1 Erstellen Sie im vSphere Web Client oder vSphere Client eine neue virtuelle Maschine.

2 Konfigurieren Sie die benutzerdefinierten Konfigurationsoptionen.

- a Klicken Sie mit der rechten Maustaste auf die virtuelle Maschine und klicken Sie auf **Einstellungen bearbeiten**.
- b Geben Sie die Anzahl der vCPUs und die Größe des vMemory-Arbeitsspeichers an.
Empfohlene Werte finden Sie im Installationshandbuch Ihrer Linux-Distribution.
Ubuntu 12.04 empfiehlt beispielsweise die Konfiguration von 2.048 MB für den vMemory-Arbeitsspeicher und von zwei vCPUs.
- c Wählen Sie **Grafikkarte** aus und geben Sie die Anzahl der Anzeigegeräte sowie den gesamten Videospeicher (vRAM) ein.

Legen Sie im vSphere Web Client die vRAM-Größe für virtuelle Maschinen mit 2D oder vSGA fest. Diese verwenden den VMware-Treiber. Die vRAM-Größe hat keinen Einfluss auf vDGA- oder NVIDIA GRID vGPU-Maschinen. Diese verwenden NVIDIA-Treiber.

Empfohlene Werte finden Sie unter *Systemanforderungen für Horizon 7 for Linux*. Verwenden Sie nicht die Videospeicherberechnung.

3 Schalten Sie die virtuelle Maschine ein und installieren Sie die Linux-Distribution.

4 Konfigurieren Sie die Desktop-Umgebung für die Verwendung der jeweiligen Linux-Distribution.

Weitere Informationen finden Sie im Abschnitt „Desktop-Umgebung“ in [Systemanforderungen für Horizon 7 for Linux](#).

5 Stellen Sie sicher, dass der Hostname des Systems als 127.0.0.1 aufgelöst werden kann.

Vorbereiten einer Linux-Maschine für die Remote-Desktop-Bereitstellung

Zur Vorbereitung einer Linux-Maschine für die Verwendung als Desktop in einer Horizon 7-Bereitstellung müssen Sie bestimmte Aufgaben durchführen.

Bevor eine Linux-Maschine durch Horizon 7 verwaltet werden kann, muss sie in der Lage sein, mit dem Verbindungsserver zu kommunizieren. Sie müssen die Netzwerkeinstellungen auf der Linux-Maschine konfigurieren, damit diese unter Verwendung des vollqualifizierten Domänennamens (FQDN) die Verbindungsserver-Instanz pingen kann.

Open VMware Tools (OVT) sind auf RHEL 7-, CentOS 7-, SLED 12- und SLES 12-Maschinen vorinstalliert. Wenn Sie eine dieser Maschinen für die Verwendung als Remote-Desktop vorbereiten, können Sie bei manueller Ausführung des Installationsprogramms in der nachfolgend beschriebenen Installation der VMware Tools die Schritte 1 bis 5 überspringen.

Wenn Sie eine Ubuntu16.04-Maschine verwenden, installieren Sie darauf OVT. Wenn Sie diese Maschine für eine Verwendung als Remote-Desktop vorbereiten, können Sie die Schritte 1 bis 5 im nachfolgend dargestellten Vorgang überspringen und OVT manuell auf Ihrer Ubuntu 16.04-Maschine mithilfe des folgenden Befehls installieren:

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

Voraussetzungen

- Stellen Sie sicher, dass eine neue virtuelle Maschine (VM) in vCenter Server erstellt und Ihre Linux-Distribution auf der Maschine installiert wurde.
- Machen Sie sich mit den Schritten zum Mounten und Installieren von VMware Tools auf einer Linux-VM vertraut. Erläuterungen dazu finden Sie unter „Manuelles Installieren oder Durchführen eines Upgrades der VMware Tools in einer virtuellen Linux-Maschine“ im Dokument *Verwaltung virtueller vSphere-Maschinen*.
- Machen Sie sich mit den Schritten zur Konfiguration Ihrer Linux-Maschine für deren Auflösung über das DNS vertraut. Die Schritte sind im Einzelnen von den jeweiligen Linux-Distributionen und -Versionen abhängig. Anleitungen dazu finden Sie in der Dokumentation Ihrer Linux-Distributionen und -Versionen.

Verfahren

- 1 Im vSphere Web Client oder vSphere Client mounten Sie die virtuelle Festplatte von VMware Tools auf der VM.
- 2 Klicken Sie mit der rechten Maustaste auf die Installationsdatei `VMwareTools-x.x.x-xxxx.tar.gz` von VMware Tools, klicken Sie im eingblendeten Kontextmenü auf **Extrahieren nach** und wählen Sie den Desktop für Ihre Linux-Distribution aus.

Der Ordner `vmware-tools-distrib` wird für den Desktop extrahiert.

- 3 Melden Sie sich auf der VM als Root an und öffnen Sie ein Terminalfenster.
- 4 Dekomprimieren Sie die TAR-Installationsdatei von VMware Tools.

Beispiel:

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

- 5 Führen Sie das Installationsprogramm aus und konfigurieren Sie VMware Tools.

Der exakte Befehl ist von den jeweiligen Linux-Distributionen abhängig. Beispiel:

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

In der Regel wird die Konfigurationsdatei `vmware-config-tools.pl` nach Ausführung der Installationsprogrammdatei ausgeführt.

- 6 Ordnen Sie den Hostnamen der Linux-Maschine zu 127.0.0.1 in der Datei `/etc/hosts` zu.

Für RHEL, CentOS, SLES und SLED müssen Sie den Hostnamen manuell zu 127.0.0.1 zuordnen, da dies nicht automatisch erfolgt. Für Ubuntu ist dieser Schritt nicht erforderlich, da die Zuordnung hier automatisch erfolgt. Dieser Schritt ist auch nicht erforderlich, wenn Sie die Massenbereitstellung von Desktops verwenden, weil der Klonvorgang diese Zuordnung hinzufügt.

Hinweis Wenn Sie den Hostnamen der Linux-Maschine nach der Installation von Horizon Agent ändern, müssen Sie den neuen Hostnamen zu 127.0.0.1 in der Datei `/etc/hosts` zuordnen. Andernfalls wird der alte Hostname weiterhin verwendet.

- 7 Für RHEL 7 und CentOS 7 müssen Sie sicherstellen, dass `virbr0` deaktiviert ist.

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

- 8 Stellen Sie sicher, dass die View-Verbindungsserver-Instanzen im Pod über das DNS aufgelöst werden können.

- 9 Konfigurieren Sie die Linux-Maschine für Runlevel 5 als Standard.

Für Linux-Desktops muss das Runlevel 5 zur ordnungsgemäßen Ausführung gültig sein.

- 10 Auf einer Ubuntu-Maschine, die für die Authentifizierung mit einem OpenLDAP-Server konfiguriert wurde, geben Sie den vollqualifizierten Domännennamen auf der Maschine an.

Dieser Schritt stellt sicher, dass die Informationen im Benutzerfeld auf der Seite „Sitzungen“ in View Administrator korrekt dargestellt werden. Bearbeiten Sie die Datei `/etc/hosts` wie folgt:

- a `# nano /etc/hosts`
- b Fügen Sie den vollqualifizierten Domännennamen hinzu. Beispiel: `127.0.0.1
hostname.domainname hostname.`
- c Beenden Sie und speichern Sie die Datei.

- 11 Für SUSE deaktivieren Sie „Hostnamen über DHCP ändern“. Legen Sie den Hostnamen oder den Domännennamen fest.

- a Klicken Sie in Yast auf **Netzwerkeinstellungen**.
- b Klicken Sie auf die Registerkarte **Hostname/DNS**.
- c Deaktivieren Sie **Hostnamen über DHCP ändern**
- d Geben Sie den Hostnamen und den Domännennamen ein.
- e Klicken Sie auf **OK**.

Wenn Sie nach der Installation von VMware Tools ein Upgrade für den Linux-Kernel durchführen, wird VMware Tools eventuell nicht mehr ausgeführt. Zur Lösung dieses Problems finden Sie Erläuterungen unter <http://kb.vmware.com/kb/2050592>.

Installieren von Abhängigkeitspaketen für Horizon Agent

Horizon Agent for Linux verfügt für jede Linux-Distribution über spezifische Abhängigkeitspakete. Sie müssen diese Pakete vor der Installation von Horizon Agent for Linux installieren.

Voraussetzungen

Stellen Sie sicher, dass eine neue virtuelle Maschine (VM) in vCenter Server erstellt und Ihre Linux-Distribution auf der Maschine installiert wurde.

Verfahren

- 1 Installieren Sie die obligatorischen Pakete, wenn diese noch nicht installiert oder nicht standardmäßig aktualisiert wurden. Die Installation wird abgebrochen, wenn ein Paket nicht den Anforderungen entspricht.

Tabelle 2-1. Obligatorische Abhängigkeitspakete

Linux-Distribution	Pakete
SLED 11 SP3/SP4 Upgrade von xorg-x11-server auf eine höhere Version als 7.4.27.111.1	<code>zypper install xorg-x11-server</code>
SLES 12 SP1/SLED 12 SP1 Upgrade von xf86-video-vmware auf eine höhere Version als 13.0.2-3.2 vom SUSE-Repository	<ol style="list-style-type: none"> 1 Registrieren Sie SUSE 12, um die SUSE-Repositorys zu aktivieren. <code>SUSEConnect -r <i>Registrierungscode</i> -e <i>E-Mail</i></code> 2 Aktualisieren Sie die Version xf86-video-vmware. <code>zypper install xf86-video-vmware</code>
SLES 12	<p>Eine python-gobject2-Installation ist für SLES 12 Linux-Desktop erforderlich, wenn Sie Horizon Agent installieren möchten.</p> <ol style="list-style-type: none"> 1 Registrieren Sie SUSE 12, um die SUSE-Repositorys zu aktivieren. <code>SUSEConnect -r <i>Registrierungscode</i> -e <i>E-Mail</i></code> 2 Installieren Sie python-gobject2. <code>zypper install python-gobject2</code>
Ubuntu 14.04 Upgrade von indicator-session auf 12.10.5+15.04.20150327 (verfügbar in https://launchpad.net/ubuntu/wily/amd64/indicator-session/12.10.5+15.04.20150327-0ubuntu1)	<code>wget http://launchpadlibrarian.net/201393830/indicator-session_12.10.5+15.04.20150327-0ubuntu1_amd64.deb</code> <code>sudo dpkg -i ./indicator-session_12.10.5+15.04.20150327-0ubuntu1_amd64.deb</code>
Ubuntu 16.04	<code>apt-get install python-dbus python-gobject</code>

2 Installieren Sie das optionale Paket für Horizon Agent.

- Standardmäßig ist in RHEL oder CentOS 6.7 glibc-2.12-1.166.el6.x86_64 installiert, was einen Stillstand nach sich ziehen kann. Daraufhin hängt die Desktopverbindung. Zur Beseitigung dieses Problems müssen Sie glibc auf die neueste Version aus einem Online-Repository aktualisieren.

```
sudo yum install glibc
```

- Auf Ubuntu 14.04-Desktops mit mehreren Monitoren ist ein Gnome-Sitzungs-Fallback zur Deaktivierung des Fenstermanagers Compiz für eine verbesserte Leistung erforderlich.

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

- Auf Ubuntu 16.04-Desktops mit mehreren Monitoren ist ein Gnome-Sitzungs-Flashback zur Deaktivierung Fenstermanagers Compiz für eine verbesserte Leistung erforderlich.

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

Einrichten der Active Directory-Integration für Linux-Desktops

3

View nutzt die vorhandene Microsoft Active Directory (AD)-Infrastruktur für die Benutzerauthentifizierung und -verwaltung. Sie können die Linux-Desktops mit Active Directory integrieren, sodass sich Benutzer mit ihrem Active Directory-Benutzerkonto bei einem Linux-Desktop anmelden können.

Dieses Kapitel enthält die folgenden Themen:

- [Integrieren von Linux mit Active Directory](#)
- [Einrichten von Single Sign-On und Smartcard-Umleitung](#)

Integrieren von Linux mit Active Directory

Für die Integration von Linux in Active Directory (AD) sind mehrere Lösungen verfügbar. Für Horizon 7 for Linux Desktop sind alle Lösungen anwendbar.

Die folgenden Lösungen können in einer Horizon 7 for Linux Desktop-Umgebung verwendet werden:

- OpenLDAP-Server-Pass-Through-Authentifizierung
- System Security Services Daemon (SSSD) LDAP-Authentifizierung bei Microsoft Active Directory
- Winbind-Domänenbeitritt

Auf einer höheren Ebene umfasst die Lösung mit der OpenLDAP-Pass-Through-Authentifizierung die folgenden Schritte:

- 1 Installieren Sie Zertifikatsdienste im Active Directory, um LDAPS (Lightweight Directory Access Protocol over SSL) zu aktivieren.
- 2 Richten Sie einen OpenLDAP-Server ein.
- 3 Synchronisieren Sie die Benutzerinformationen (außer Kennwort) von Active Directory mit dem OpenLDAP-Server.
- 4 Konfigurieren Sie den OpenLDAP-Server, um die Kennwortüberprüfung an einen separaten Prozess zu delegieren, z. B. an `ssslauthd`, der die Kennwortüberprüfung bei Active Directory durchführen kann.
- 5 Konfigurieren Sie die Linux-Desktops für die Verwendung eines LDAP-Clients zur Authentifizierung von Benutzern mit dem OpenLDAP-Server.

Die SSSD-LDAP-Authentifizierung bei der Microsoft Active Directory-Lösung umfasst die folgenden Schritte:

- 1 Installieren Sie die Zertifikatdienste in Active Directory zur Aktivierung von LDAPS.
- 2 Konfigurieren Sie SSSD auf dem Linux-Desktop für die direkte Verwendung der LDAP-Authentifizierung bei Microsoft Active Directory.

Die Winbind-Domänenbeitritts-Lösung umfasst die folgenden Schritte:

- 1 Installieren Sie die Winbind-, Samba- und Kerberos-Pakete auf dem Linux-Desktop.
- 2 Fügen Sie den Linux-Desktop zu Microsoft Active Directory hinzu.

Wenn Sie die LDAP-basierten Lösungen verwenden, müssen Sie die Konfiguration in einer Vorlagen-VM durchführen. In den geklonten virtuellen Maschinen sind keine zusätzlichen Schritte erforderlich.

Bei Verwendung der Winbind-Domänenbeitritts-Lösung oder anderer Kerberos-basierter Authentifizierungslösungen müssen Sie die Vorlagen-VM zu Active Directory hinzufügen und die geklonte virtuelle Maschine erneut Active Directory hinzufügen. Verwenden Sie dazu beispielsweise den folgenden Befehl:

```
sudo /usr/bin/net ads join -U <Domänenbenutzer="">%<Domänenkennwort="">
```

Verwenden Sie die folgenden Optionen, um den Befehl für den erneuten Beitritt zur Domäne auf einer geklonten virtuellen Maschine für die Winbind-Lösung auszuführen:

- Stellen Sie eine Remote-Verbindung wie SSH oder vSphere PowerCLI zu jeder virtuellen Maschine her und führen Sie den Befehl aus. Weitere Informationen zu Skripten finden Sie unter [Kapitel 8 Massenbereitstellung von Horizon 7 für manuelle Desktop-Pools](#).
- Fügen Sie den Befehl einem Shell-Skript hinzu und geben Sie den Skriptpfad zur Horizon Agent-Option RunOnceScript in der Datei /etc/vmware/viewagent-custom.conf an. Weitere Informationen finden Sie unter [Einstellen der Optionen in Konfigurationsdateien auf einem Linux-Desktop](#).

Hinweis Verwenden Sie zur Vereinfachung der Bereitstellung die SSSD-LDAP-Authentifizierung bei der Microsoft Active Directory-Lösung.

Einrichten von Single Sign-On und Smartcard-Umleitung

Zum Einrichten von Single Sign-On (SSO) und Smartcard-Umleitung müssen Sie einige Konfigurationsschritte durchführen.

Single Sign-On

Das Single Sign-On-Modul von Horizon View kommuniziert mit PAMs (Pluggable Authentication Modules) in Linux und ist nicht von der Methode abhängig, die Sie zur Integration von Linux mit Active Directory (AD) verwenden. Das Horizon View-SSO funktioniert bekanntermaßen mit den OpenLDAP- und Winbind-Lösungen, die für die Integration von Linux und AD sorgen.

Standardmäßig nimmt SSO an, dass das `sAMAccountName`-Attribut von AD die Anmelde-ID ist. Um sicherzustellen, dass die korrekte Anmelde-ID für SSO verwendet wird, müssen Sie die nachfolgend aufgeführten Konfigurationsschritte durchführen, wenn Sie die OpenLDAP- oder Winbind-Lösung verwenden.

- Für OpenLDAP setzen Sie `sAMAccountName` auf `uid`.
- Für Winbind fügen Sie die folgende Anweisung zur Konfigurationsdatei `/etc/samba/smb.conf` hinzu.

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

Wenn Benutzer den Domännennamen angeben müssen, um sich anzumelden, müssen Sie die `SSOUserFormat`-Option auf dem Linux-Desktop angeben. Weitere Informationen finden Sie unter [Einstellen der Optionen in Konfigurationsdateien auf einem Linux-Desktop](#). Achten Sie darauf, dass SSO immer die Kurzform des Domännennamens in Großbuchstaben verwendet. Wenn zum Beispiel die Domäne `mydomain.com` ist, wird `MYDOMAIN` von SSO als Domänenname verwendet. Aus diesem Grund müssen Sie `MYDOMAIN` angeben, wenn Sie `SSOUserFormat` festlegen. In Bezug auf die Kurz- und Langformen von Domännennamen gelten die folgenden Regeln:

- Für OpenLDAP müssen Sie die Kurzform der Domännennamen in Großbuchstaben verwenden.
- Winbind unterstützt sowohl die Langform als auch die Kurzform der Domännennamen.

AD unterstützt Sonderzeichen in Anmeldenamen, Linux jedoch nicht. Deshalb dürfen Sie keine Sonderzeichen in Anmeldenamen verwenden, wenn Sie SSO einrichten.

Wenn in AD das `UserPrincipalName`-Attribut (UPN) und das `sAMAccount`-Attribut eines Benutzers nicht übereinstimmen und der Benutzer sich mit dem UPN anmeldet, schlägt SSO fehl. Zur Umgehung des Problems kann sich der Benutzer mit dem Namen anmelden, der in `sAMAccount` gespeichert ist.

View erfordert nicht, dass beim Benutzernamen zwischen Groß- und Kleinschreibung unterschieden wird. Sie müssen sicherstellen, dass das Linux-Betriebssystem Benutzernamen verarbeiten kann, bei denen zwischen Groß- und Kleinschreibung unterschieden wird.

- Für Winbind wird beim Benutzernamen standardmäßig zwischen Groß- und Kleinschreibung unterschieden.
- Für OpenLDAP verwendet Ubuntu NSCD zur Authentifizierung von Benutzern, und es wird standardmäßig zwischen Groß- und Kleinschreibung unterschieden. RHEL und CentOS verwenden SSSD zur Authentifizierung von Benutzern, und es wird standardmäßig zwischen Groß- und Kleinschreibung unterschieden. Zum Ändern der Einstellung bearbeiten Sie die Datei `/etc/sss/sss.conf` und fügen Sie dem Abschnitt `[domain/default]` die folgende Zeile hinzu:

```
case_sensitive = false
```

Für Ubuntu 16.04 oder Ubuntu 14.04 konfigurieren Sie `UseGnomeFlashback=TRUE` in der Datei `/etc/vmware/viewagent-custom.conf`, um die GNOME Flashback (Metacity)-Desktop-Umgebung zu verwenden.

Smartcard-Umleitung

Zum Einrichten der Smartcard-Umleitung müssen Sie zunächst die Anweisungen des Linux-Distributors und des Smartcard-Anbieters befolgen. Führen Sie dann ein Upgrade auf pcsc-lite 1.7.4 durch. Führen Sie beispielsweise die folgenden Befehle aus:

```
#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-ipcdirdir=/var/run --disable-libusb --disable-serial --disable-usb
--disable-libudev
#make
#make install
#service pcscd start
```

Für Winbind fügen Sie die folgende Anweisung zur Konfigurationsdatei `/etc/samba/smb.conf` hinzu.

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

Wenn Sie Horizon Agent installieren, müssen Sie zuerst SELinux deaktivieren oder den Permissive Mode für SELinux aktivieren. Sie müssen auch die Smartcard-Umleitungskomponente speziell auswählen, weil die Komponente nicht standardmäßig ausgewählt wird. Weitere Informationen finden Sie unter [Befehlszeilenoptionen für install_viewagent.sh](#).

Smartcard-SSO wird in Horizon View 7.0.1 oder höher aktiviert. Außerdem gilt: Wenn die Funktion der Smartcard-Umleitung in einer virtuellen Maschine installiert ist, funktioniert die USB-Umleitung des vSphere-Clients mit dieser Smartcard nicht.

Die Smartcard-Umleitung unterstützt nur ein Smartcard-Lesegerät. Diese Funktion ist nicht funktionsfähig, wenn zwei oder mehr Lesegeräte an das Client-Gerät angeschlossen sind.

Die Smartcard-Umleitung unterstützt nur ein Zertifikat auf der Smartcard. Wenn sich auf der Smartcard mehrere Zertifikate befinden, wird das im ersten Slot befindliche Zertifikat verwendet; die anderen werden ignoriert. Dies ist eine Einschränkung durch Linux.

Hinweis

- Die Smartcard unterstützt den folgenden winbind-Wert. Andernfalls schlagen die Smartcard-SSO und manuelle Anmeldung fehl.

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

- Wenn Sie den Linux-Client für die Authentifizierung des Brokers an der PIV-Karte verwenden, was durch die Linux-Desktop-Smartcard-Umleitung unterstützt wird, müssen Sie die Konfiguration `view.sslProtocolString = "TLSv1.1"` für den Linux-Client bei `~/ .vmware/view-preferences` hinzufügen, um einen SSL-Fehler zu verhindern.
-

Einrichten von Grafiken für Linux-Desktops

4

Sie können die aktuell unterstützten RHEL-Distributionen entsprechend konfigurieren, um die Vorteile der NVIDIA-Funktionen auf einem ESXi-Host oder einem Gastbetriebssystem nutzen zu können.

Anforderungen für VM-Klone zur Einrichtung von 3D-Grafiken

Für die Einrichtung von 3D-Grafiken müssen die nachfolgend aufgeführten Anforderungen für VM-Klone erfüllt sein.

- Führen Sie für vGPU und vSGA die Grafikeinrichtung in der Basis-VM durch. Klonen Sie die virtuellen Maschinen (VMs). Die Grafikeinstellungen gelten für alle geklonten VMs. Es sind keine weiteren Einstellungen erforderlich.
- Führen Sie für vDGA die Grafikeinrichtung in der Basis-VM durch. Klonen Sie die virtuellen Maschinen (VMs). Allerdings müssen Sie vor dem Einschalten der geklonten VMs das vorhandene NVIDIA-Passthrough-PCI-Gerät aus der geklonten VM entfernen und der geklonten VM das neue NVIDIA-Passthrough-PCI-Gerät hinzufügen. Das NVIDIA-Passthrough-PCI-Gerät kann nicht von den VMs gemeinsam genutzt werden. Jede virtuelle Maschine verwendet ein dediziertes NVIDIA-Passthrough-PCI-Gerät .

Dieses Kapitel enthält die folgenden Themen:

- [Konfigurieren unterstützter RHEL-Distributionen für vGPU](#)
- [Konfigurieren von RHEL 6 für vDGA](#)
- [Konfigurieren von RHEL 7 für vSGA](#)

Konfigurieren unterstützter RHEL-Distributionen für vGPU

Sie können eine unterstützte RHEL-Distribution so einrichten, dass die NVIDIA vGPU-Funktionalität (gemeinsam genutzte GPU-Hardwarebeschleunigung) auf dem ESXi-Host zur Verfügung steht.

Sie müssen den NVIDIA Linux VM-Anzeigetreiber installieren, der zum GPU-Treiber (.vib) des ESXi-Hosts passt. Informationen zu Treiberpaketen finden Sie auf der NVIDIA-Website.

Wichtig NVIDIA vGPU wird auf NVIDIA Maxwell M60- und NVIDIA M6-Grafikkarten unterstützt. Diese Funktion ist nicht auf anderen NVIDIA-Grafikkarten wie GRID K1 oder K2 verwendbar.

Vorsicht Bevor Sie starten, stellen Sie sicher, dass Horizon Agent nicht auf der virtuellen Linux-Maschine installiert ist. Wenn Sie Horizon Agent vor der Konfiguration der Maschine zur Verwendung von NVIDIA vGPU installieren, werden die erforderlichen Parameter in der Datei `xorg.conf` überschrieben und NVIDIA vGPU funktioniert nicht. Sie müssen Horizon Agent nach dem Abschluss der NVIDIA vGPU-Konfiguration installieren.

Installieren des VIB für die NVIDIA GRID vGPU-Grafikkarte auf dem ESXi-Host

Sie müssen das VIB für Ihre NVIDIA GRID-Grafikkarte herunterladen und auf dem ESXi 6.0 U1-Host oder höher installieren.

NVIDIA bietet ein vGPU-Softwarepaket mit einem vGPU Manager, den Sie in diesem Vorgang auf dem ESXi-Host installieren, und einen Linux-Anzeigetreiber, den Sie auf der virtuellen Linux-Maschine in einem späteren Vorgang installieren.

Voraussetzungen

- Stellen Sie sicher, dass vSphere 6.0 U1 oder höher in Ihrer Umgebung installiert ist.
- Vergewissern Sie sich, dass die NVIDIA Maxwell M60- oder M6-GPU auf dem ESXi-Host installiert ist.

Verfahren

- 1 Laden Sie das VIB für Ihre NVIDIA GRID vGPU-Grafikkarte von der Site [NVIDIA Treiber Downloads](#) herunter.

Wählen Sie aus den Dropdown-Menüs die geeignete VIB-Version aus.

Option	Beschreibung
Produkttyp	GRID
Produktserie	Wählen Sie NVIDIA GRID vGPU aus.
Produkt	Wählen Sie die Version aus, die auf dem ESXi-Host installiert ist (z. B. GRID K2).
Betriebssystem	Wählen Sie die VMware vSphere ESXi-Version aus.

- 2 Dekomprimieren Sie die ZIP-Datei des vGPU-Softwarepakets.
- 3 Laden Sie den vGPU Manager-Ordner auf den ESXi 6.0 U1-Host hoch.

Hinweis Der Linux-Anzeigetreiber wird später auf der virtuellen Linux-Maschine installiert.

- 4 Schalten Sie alle virtuellen Maschinen auf dem ESXi-Host aus oder halten Sie diese an.

- 5 Stellen Sie mithilfe von SSH eine Verbindung zum ESXi-Host her.
- 6 Beenden Sie den xorg-Dienst.

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

- 7 Installieren Sie das NVIDIA VIB.

Beispiel:

```
# 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 Starten Sie den ESXi-Host neu oder aktualisieren Sie diesen.
 - ◆ Bei einem installierten ESXi-Host starten Sie den Host neu.
 - ◆ Bei einem zustandsfreien ESXi-Host aktualisieren Sie den Host mit den folgenden Schritten. (Diese Schritte können auch auf einem installierten Host durchgeführt werden.)

```
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 Stellen Sie sicher, dass der xorg-Dienst nach dem Neustart des Hosts ausgeführt wird.

Konfigurieren eines gemeinsam genutzten PCI-Geräts für vGPU auf der virtuellen Linux-Maschine

Zur Verwendung der NVIDIA vGPU müssen Sie ein gemeinsam genutztes PCI-Gerät für die virtuelle Linux-Maschine konfigurieren.

Voraussetzungen

- Stellen Sie sicher, dass die virtuelle Linux-Maschine für eine Verwendung als Desktop vorbereitet ist. Siehe [Erstellen einer virtuellen Maschine und Installieren von Linux](#) und [Vorbereiten einer Linux-Maschine für die Remote-Desktop-Bereitstellung](#).
- Stellen Sie sicher, dass Horizon Agent nicht auf der virtuellen Linux-Maschine installiert ist.
- Stellen Sie sicher, dass das NVIDIA VIB auf dem ESXi-Host installiert ist. Siehe [Installieren des VIB für die NVIDIA GRID vGPU-Grafikkarte auf dem ESXi-Host](#).

- Machen Sie sich mit den virtuellen GPU-Typen vertraut, die mit NVIDIA vGPU verfügbar sind und die mit der Einstellung **GPU-Profil** ausgewählt werden. Die virtuellen GPU-Typen bieten unterschiedliche Funktionen für die physischen GPUs, die auf dem ESXi-Host installiert sind. Siehe [Virtuelle GPU-Typen für NVIDIA](#).

Verfahren

- 1 Schalten Sie die virtuelle Maschine aus
- 2 Im vSphere Web Client wählen Sie die virtuelle Maschine aus und klicken auf der Registerkarte **VM-Hardware** auf **Einstellungen bearbeiten**.
- 3 Im Menü **Neues Gerät** wählen Sie **Gemeinsam genutztes PCI-Gerät** aus.
- 4 Klicken Sie auf **Hinzufügen** und wählen Sie **NVIDIA GRID vGPU** aus dem Dropdown-Menü aus.
- 5 Für die Einstellung **GPU-Profil** wählen Sie einen virtuellen GPU-Typ aus dem Dropdown-Menü aus.
- 6 Klicken Sie auf **Gesamten Arbeitsspeicher reservieren** und dann auf **OK**.

Um die GPU zur Unterstützung von NVIDIA GRID vGPU zu aktivieren, müssen Sie dafür den gesamten Arbeitsspeicher der virtuellen Maschine reservieren.

- 7 Schalten Sie die virtuelle Maschine ein.

Virtuelle GPU-Typen für NVIDIA

Mit der Einstellung **GPU-Profil** auf der Seite „Virtuelle Hardware“ im vSphere Web Client können Sie einen virtuellen GPU-Typ auswählen, der bestimmte Funktionen für den physischen NVIDIA-GPU auf dem ESXi-Host bereitstellt.

Auf virtuellen Linux-Maschinen wird NVIDIA GRID vGPU für NVIDIA Maxwell M60-GPUs oder NVIDIA M6-GPUs unterstützt.

Tabelle 4-1. Für NVIDIA GRID M60 vGPU auf virtuellen Linux-Maschinen verfügbare virtuelle GPU-Typen

Virtueller GPU-Typ	Physische Platine	Physische GPUs	FB pro virtuellem GPU	Anzeige-Heads	Maximale Auflösung	Maximale Anzahl virtueller GPUs pro physischem GPU	Maximale Anzahl virtueller GPUs pro physischer Platine
GRID M60-0q	GRID M60	Zwei	512M	2	2560x1600	16	32
GRID M60-1q	GRID M60	Zwei	1G	2	2560x1600	8	16
GRID M60-2q	GRID M60	Zwei	2G	4	2560x1600	4	8
GRID M60-4q	GRID M60	Zwei	4G	4	3840x2160	2	4
GRID M60-8q	GRID M60	Zwei	8G	4	3840x2160	1	2

Tabelle 4-2. Für NVIDIA GRID M6 vGPU auf virtuellen Linux-Maschinen verfügbare virtuelle GPU-Typen

Virtueller GPU-Typ	Physische Platine	Physische GPUs	FB pro virtuellem GPU	Anzeige-Heads	Maximale Auflösung	Maximale Anzahl virtueller GPUs pro physischem GPU	Maximale Anzahl virtueller GPUs pro physischer Platine
GRID M6-0q	GRID M6	1	512M	2	2560x1600	16	16
GRID M6-1q	GRID M6	1	1G	2	2560x1600	8	8
GRID M6-2q	GRID M6	1	2G	4	2560x1600	4	4
GRID M6-4q	GRID M6	1	4G	4	3840x2160	2	2
GRID M6-8q	GRID M6	1	8G	4	3840x2160	1	1

Installieren des NVIDIA GRID vGPU-Anzeigetreibers

Um den NVIDIA GRID vGPU-Anzeigetreiber zu installieren, müssen Sie den Standard-NVIDIA-Treiber deaktivieren, die NVIDIA-Anzeigetreiber herunterladen und das PCI-Gerät auf der virtuellen Maschine konfigurieren.

Voraussetzungen

- Stellen Sie sicher, dass Sie das vGPU-Softwarepaket von der NVIDIA-Download-Site heruntergeladen, das Paket dekomprimiert und den Linux-Anzeigetreiber (eine Paketkomponente) bereitgehalten haben. Siehe [Installieren des VIB für die NVIDIA GRID vGPU-Grafikkarte auf dem ESXi-Host](#).

Stellen Sie sicher, dass der virtuellen Maschine ein gemeinsam genutztes PCI-Gerät hinzugefügt wurde. Siehe [Konfigurieren eines gemeinsam genutzten PCI-Geräts für vGPU auf der virtuellen Linux-Maschine](#).

Verfahren

- 1 Deaktivieren Sie den Standard-NVIDIA Nouveau-Treiber und sperren Sie diesen für eine weitere Verwendung.

- a Bearbeiten Sie die Datei `grub.conf` oder `grub`.

Bei RHEL 6 ist das die Datei `/boot/grub/grub.conf`. Bei RHEL 7 handelt es sich um die Datei `/etc/default/grub`.

RHEL-Version	Befehl
6	<code>sudo vi /boot/grub/grub.conf</code>
7	<code>sudo vi /etc/default/grub</code>

- b Fügen Sie am Ende der Kernel-Optionen die Zeile `rdblacklist=nouveau` hinzu.

- c Bearbeiten Sie die Datei `blacklist.conf`.

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

- d Fügen Sie der Datei `blacklist.conf` die folgende Zeile an einer beliebigen Stelle hinzu.

```
blacklist nouveau
```

- 2 Starten Sie die virtuelle Maschine neu.

Die Art der Anzeige hat sich verändert.

- 3 (Optional) Stellen Sie sicher, dass der Nouveau-Treiber deaktiviert ist.

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

Dies ist dann der Fall, wenn die `grep`-Suche keine Ergebnisse ergibt.

- 4 Kopieren Sie den NVIDIA Linux-Anzeigetreiber in die virtuelle Maschine.

- 5 Öffnen Sie einen Remoteterminal für die virtuelle Maschine oder wechseln Sie durch Drücken von Strg-Alt-F2 zu einer Textkonsole, melden Sie sich als Root an und führen Sie dann den `init 3`-Befehl zur Deaktivierung von X Windows aus.

- 6 Installieren Sie weitere für den NVIDIA-Treiber erforderliche Komponenten.

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

- 7 Fügen Sie dem NVIDIA GRID vGPU-Treiberpaket ein ausführbares Attribut hinzu.

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

- 8 Starten Sie das NVIDIA GRID vGPU-Installationsprogramm.

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

- 9 Akzeptieren Sie die NVIDIA-Softwarelizenzvereinbarung und wählen Sie **Ja** für eine automatische Aktualisierung der X-Konfigurationseinstellungen aus.

Nächste Schritte

Installieren Sie Horizon Agent auf der virtuellen Linux-Maschine. Siehe [Installieren von Horizon Agent auf einer virtuellen Linux-Maschine](#).

Erstellen Sie einen Desktop-Pool mit den konfigurierten virtuellen Linux-Maschinen. Siehe [Erstellen eines manuellen Desktop-Pools für Linux](#).

Überprüfen, ob der NVIDIA-Anzeigetreiber installiert ist

Sie können prüfen, ob der NVIDIA-Anzeigetreiber auf einer virtuellen RHEL 6-Maschine installiert wurde, indem Sie die NVIDIA-Treiberausgabe in einer View-Desktop-Sitzung darstellen.

Voraussetzungen

- Überprüfen Sie, ob der NVIDIA-Anzeigetreiber installiert ist.
- Stellen Sie sicher, dass Horizon Agent auf der virtuellen Linux-Maschine installiert ist. Siehe [Installieren von Horizon Agent auf einer virtuellen Linux-Maschine](#).
- Stellen Sie sicher, dass die virtuelle Linux-Maschine in einem Desktop-Pool bereitgestellt wurde. Siehe [Erstellen eines manuellen Desktop-Pools für Linux](#).

Verfahren

- 1 Starten Sie die virtuelle Linux-Maschine neu.

Das Startskript für Horizon Agent initialisiert den X-Server und stellt die Topologie dar.

Die Anzeige der virtuellen Maschine erscheint nicht mehr in der vSphere-Konsole.

- 2 Von Horizon Client aus stellen Sie eine Verbindung zum Linux-Desktop her.
- 3 In der Linux-Desktop-Sitzung stellen Sie sicher, dass der NVIDIA-Anzeigetreiber installiert ist.

Öffnen Sie ein Terminalfenster und führen Sie den Befehl `glxinfo | grep NVIDIA` aus.

Die Ausgabe des NVIDIA-Treibers wird dargestellt. Beispiel:

```
[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
```

Der Benutzer kann auf die NVIDIA-Grafikfunktionen auf dem Remote-Desktop zugreifen.

Nachdem Sie sichergestellt haben, dass der NVIDIA-Anzeigetreiber installiert ist, führen Sie die folgenden Schritte durch, um eine korrekte Installation zu gewährleisten.

- Wenn Sie ein Upgrade für den Linux-Kernel durchführen, kann Horizon Agent eventuell nicht mehr mit dem View-Verbindungsserver kommunizieren. Um dieses Problem zu beheben, installieren Sie den NVIDIA-Treiber erneut.
- Richten Sie die NVIDIA GRID-Lizenzierung in der Linux-VM ein. Weitere Informationen finden Sie in der NVIDIA-Dokumentation. Der Linux-Desktop funktioniert ohne eingerichtete Lizenzierung nicht ordnungsgemäß. Beispielsweise ist dann keine automatische Anpassung möglich.

Konfigurieren von RHEL 6 für vDGA

Sie können durch entsprechende Einrichtung eines RHEL 6-Gastbetriebssystems die vDGA-Funktionalität auf dem ESXi-Host einem Horizon 7 for Linux-Desktop zur Verfügung stellen.

Vorsicht Bevor Sie starten, stellen Sie sicher, dass Horizon Agent nicht auf der virtuellen Linux-Maschine installiert ist. Wenn Sie Horizon Agent vor der Konfiguration der Maschine zur Verwendung von vDGA installieren, werden die erforderlichen Parameter in der Datei `xorg.conf` überschrieben und vDGA funktioniert nicht. Sie müssen Horizon Agent nach dem Abschluss der vDGA-Konfiguration installieren.

Aktivieren von DirectPath I/O für NVIDIA GRID auf einem Host

Bevor Sie eine virtuelle Linux-Maschine für die Verwendung von vDGA konfigurieren können, müssen Sie die NVIDIA GRID GPU-PCI-Geräte für den DirectPath I/O-Passthrough auf dem ESXi-Host verfügbar machen.

Voraussetzungen

- Stellen Sie sicher, dass vSphere 6.0 oder höher in Ihrer Umgebung installiert ist.
- Vergewissern Sie sich, dass die NVIDIA GRID K1- oder K2-Grafikkarte auf dem ESXi-Host installiert ist.

Verfahren

- 1 Im vSphere Web Client suchen Sie nach dem ESXi-Host.
- 2 Klicken Sie auf die Registerkarte **Verwalten** und anschließend auf **Einstellungen**.
- 3 Klicken Sie im Abschnitt „Hardware“ auf **PCI-Geräte**.
- 4 Um den DirectPath I/O-Passthrough für die NVIDIA GRID GPUs zu aktivieren, klicken Sie auf **Bearbeiten**.

Symbol	Beschreibung
Grünes Symbol	Das PCI-Gerät ist aktiv und kann aktiviert werden.
Oranges Symbol	Der Status des Geräts hat sich geändert. Sie müssen nun den Host neu starten, bevor Sie das Gerät verwenden können.

- 5 Wählen Sie die NVIDIA GRID GPUs aus und klicken Sie auf **OK**.

Die PCI-Geräte werden der Tabelle der für VMs verfügbaren DirectPath I/O-PCI-Geräte hinzugefügt.

- 6 Um die PCI-Geräte den virtuellen Linux-Maschinen zur Verfügung zu stellen, starten Sie den Host neu.

Hinzufügen eines vDGA-Pass-Through-Geräts zu einer virtuellen RHEL 6-Maschine

Um eine virtuelle RHEL 6-Maschine für die Verwendung von vDGA zu konfigurieren, müssen Sie der virtuellen Maschine das PCI-Gerät hinzufügen. Damit kann das physische Gerät auf dem ESXi-Host für eine Verwendung auf der virtuellen Maschine weitergeleitet werden.

Voraussetzungen

- Stellen Sie sicher, dass die virtuelle Linux-Maschine für eine Verwendung als Desktop vorbereitet ist. Siehe [Erstellen einer virtuellen Maschine und Installieren von Linux](#) und [Vorbereiten einer Linux-Maschine für die Remote-Desktop-Bereitstellung](#).
- Stellen Sie sicher, dass Horizon Agent nicht auf der virtuellen Linux-Maschine installiert ist.
- Vergewissern Sie sich, dass das NVIDIA GRID GPU-PCI-Gerät für den DirectPath I/O-Passthrough auf dem Host verfügbar ist. Siehe [Aktivieren von DirectPath I/O für NVIDIA GRID auf einem Host](#).

Verfahren

- 1 Melden Sie sich beim RHEL 6-Gastbetriebssystem als lokaler Benutzer mit konfigurierten Sudo-Rechten an.
- 2 Im vSphere Web Client wählen Sie die virtuelle Maschine aus und klicken auf der Registerkarte **VM-Hardware** auf **Einstellungen bearbeiten**.
- 3 Im Menü **Neues Gerät** wählen Sie **PCI-Gerät** aus.
- 4 Klicken Sie auf **Hinzufügen** und wählen Sie das PCI-Gerät aus dem Dropdown-Menü aus.
- 5 Klicken Sie auf **Gesamten Arbeitsspeicher reservieren** und dann auf **OK**.

Um die GPU zur Unterstützung von vDGA zu aktivieren, müssen Sie dafür den gesamten Arbeitsspeicher der virtuellen Maschine reservieren.

- 6 Schalten Sie die virtuelle Maschine ein und öffnen Sie die vSphere-Konsole, um eine Verbindung zur Maschine herzustellen.
- 7 Stellen Sie sicher, dass das NVIDIA GRID-Gerät an die virtuelle Maschine weitergeleitet wird.

Öffnen Sie ein Terminalfenster und führen Sie den folgenden Befehl aus:

```
lspci | grep NVIDIA
```

Der VGA-kompatible XX:00.0-Controller wird dargestellt. Beispiel:

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

Installieren des NVIDIA-Anzeigetreibers für vDGA

Um den NVIDIA-Anzeigetreiber für vDGA zu installieren, müssen Sie den Standard-NVIDIA-Treiber deaktivieren, die NVIDIA-Anzeigetreiber herunterladen und das PCI-Gerät auf der virtuellen Maschine konfigurieren.

Voraussetzungen

- Stellen Sie sicher, dass der virtuellen RHEL 6-Maschine das PCI-Gerät hinzugefügt wurde. Siehe [Hinzufügen eines vDGA-Pass-Through-Geräts zu einer virtuellen RHEL 6-Maschine](#).

Verfahren

- 1 Deaktivieren Sie den Standard-NVIDIA Nouveau-Treiber und sperren Sie diesen für eine weitere Verwendung.

- a Bearbeiten Sie die Datei `grub.conf`.

Bei RHEL 6 ist das die Datei `/boot/grub/grub.conf`.

RHEL-Version	Befehl
6	<code>sudo vi /boot/grub/grub.conf</code>

- b Fügen Sie am Ende der Kernel-Optionen die Zeile `rdblacklist=nouveau` hinzu.

- c Bearbeiten Sie die Datei `blacklist.conf`.

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

- d Fügen Sie der Datei `blacklist.conf` die folgende Zeile an einer beliebigen Stelle hinzu.

```
blacklist nouveau
```

- 2 Starten Sie die virtuelle Maschine neu.

Die Art der Anzeige hat sich verändert.

- 3 (Optional) Stellen Sie sicher, dass der Nouveau-Treiber deaktiviert ist.

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

Dies ist dann der Fall, wenn die `grep`-Suche keine Ergebnisse ergibt.

- 4 Laden Sie den NVIDIA-Treiber von der Site [NVIDIA Treiber Downloads](#) herunter.

Wählen Sie aus den NVIDIA-Dropdown-Menüs die geeignete Treiberversion aus:

Option	Beschreibung
Produkttyp	GRID
Produktserie	GRID Series

Option	Beschreibung
Produkt	Wählen Sie die Version aus, die auf dem ESXi-Host installiert ist (z. B. GRID K2).
Betriebssystem	Linux 64-Bit oder Linux 32-Bit

- 5 Öffnen Sie einen Remoteterminal für die virtuelle Maschine oder wechseln Sie durch Drücken von Strg-Alt-F2 zu einer Textkonsole, melden Sie sich als Root an und führen Sie dann den `init 3`-Befehl zur Deaktivierung von X Windows aus.
- 6 Installieren Sie weitere für den NVIDIA-Treiber erforderliche Komponenten.

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

- 7 Fügen Sie dem NVIDIA-Treiberpaket für vDGA ein ausführbares Attribut hinzu.

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

- 8 Starten Sie das NVIDIA-Installationsprogramm.

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

- 9 Akzeptieren Sie die NVIDIA-Softwarelizenzvereinbarung und wählen Sie **Ja** für eine automatische Aktualisierung der X-Konfigurationseinstellungen aus.

Nächste Schritte

Installieren Sie Horizon Agent auf der virtuellen Linux-Maschine. Siehe [Installieren von Horizon Agent auf einer virtuellen Linux-Maschine](#).

Erstellen Sie einen Desktop-Pool mit den konfigurierten virtuellen Linux-Maschinen. Siehe [Erstellen eines manuellen Desktop-Pools für Linux](#).

Überprüfen, ob der NVIDIA-Anzeigetreiber installiert ist

Sie können prüfen, ob der NVIDIA-Anzeigetreiber auf einer virtuellen RHEL 6-Maschine installiert wurde, indem Sie die NVIDIA-Treiberausgabe in einer View-Desktop-Sitzung darstellen.

Voraussetzungen

- Überprüfen Sie, ob der NVIDIA-Anzeigetreiber installiert ist.
- Stellen Sie sicher, dass Horizon Agent auf der virtuellen Linux-Maschine installiert ist. Siehe [Installieren von Horizon Agent auf einer virtuellen Linux-Maschine](#).
- Stellen Sie sicher, dass die virtuelle Linux-Maschine in einem Desktop-Pool bereitgestellt wurde. Siehe [Erstellen eines manuellen Desktop-Pools für Linux](#).

Verfahren

1 Starten Sie die virtuelle Linux-Maschine neu.

Das Startskript für Horizon Agent initialisiert den X-Server und stellt die Topologie dar.

Die Anzeige der virtuellen Maschine erscheint nicht mehr in der vSphere-Konsole.

2 Von Horizon Client aus stellen Sie eine Verbindung zum Linux-Desktop her.

3 In der Linux-Desktop-Sitzung stellen Sie sicher, dass der NVIDIA-Anzeigetreiber installiert ist.

Öffnen Sie ein Terminalfenster und führen Sie den Befehl `glxinfo | grep NVIDIA` aus.

Die Ausgabe des NVIDIA-Treibers wird dargestellt. Beispiel:

```
[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
```

Der Benutzer kann auf die NVIDIA-Grafikfunktionen auf dem Remote-Desktop zugreifen.

Nachdem Sie sichergestellt haben, dass der NVIDIA-Anzeigetreiber installiert ist, führen Sie die folgenden Schritte durch, um eine korrekte Installation zu gewährleisten.

- Wenn Sie ein Upgrade für den Linux-Kernel durchführen, kann Horizon Agent eventuell nicht mehr mit dem View-Verbindungsserver kommunizieren. Um dieses Problem zu beheben, installieren Sie den NVIDIA-Treiber erneut.
- Richten Sie die NVIDIA GRID-Lizenzierung in der Linux-VM ein. Weitere Informationen finden Sie in der NVIDIA-Dokumentation. Der Linux-Desktop funktioniert ohne eingerichtete Lizenzierung nicht ordnungsgemäß. Beispielsweise ist dann keine automatische Anpassung möglich.

Konfigurieren von RHEL 7 für vSGA

Sie können durch entsprechende Einrichtung eines RHEL 7-Gastbetriebssystems die vSGA-Funktionalität einem Horizon 7 for Linux-Desktop zur Verfügung stellen.

Installieren des VIB für die NVIDIA-Grafikkarte für vSGA auf dem ESXi-Host

Sie müssen das VIB für Ihre NVIDIA GRID-Grafikkarte herunterladen und auf dem ESXi 6.0 U1-Host oder höher installieren.

NVIDIA stellt einen VMware vSphere ESXi-Treiber für vSGA zur Verfügung. Für vSGA wird kein NVIDIA-Anzeigetreiber auf der virtuellen Linux-Maschine installiert.

Voraussetzungen

- Stellen Sie sicher, dass vSphere 6.0 U1 oder höher in Ihrer Umgebung installiert ist.

- Stellen Sie sicher, dass der NVIDIA-Treiber in Ihrer Umgebung installiert ist.
- Vergewissern Sie sich, dass die NVIDIA GRID K1- oder K2-GPU auf dem ESXi-Host installiert ist.

Verfahren

- 1 Laden Sie das VIB für Ihre NVIDIA GRID vGPU-Grafikkarte von der Site [NVIDIA Treiber Downloads](#) herunter.

Wählen Sie aus den Dropdown-Menüs die geeignete VIB-Version aus.

Option	Beschreibung
Produkttyp	GRID
Produktserie	Wählen Sie GRID-Serie aus.
Produkt	Wählen Sie die Version aus, die auf dem ESXi-Host installiert ist (z. B. GRID K2).
Betriebssystem	Wählen Sie die VMware vSphere ESXi-Version aus.

- 2 Laden Sie den VMware vSphere-ESXi-Treiber für vSGA auf den ESXi 6.0 U1-Host hoch.
- 3 Schalten Sie alle virtuellen Maschinen auf dem ESXi-Host aus oder halten Sie diese an.
- 4 Stellen Sie mithilfe von SSH eine Verbindung zum ESXi-Host her.
- 5 Beenden Sie den xorg-Dienst.

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

- 6 Installieren Sie das NVIDIA VIB.

Beispiel:

```
# 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 Starten Sie den xorg-Dienst neu, der für die GPU-Zuweisung verwendet wird:

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

- 8 Starten Sie den ESXi-Host neu.
- 9 Stellen Sie sicher, dass der xorg-Dienst nach dem Neustart des Hosts ausgeführt wird.

Konfigurieren von 3D-Funktionen für vSGA auf einer virtuellen Linux-Maschine

Um eine virtuelle RHEL 7-Maschine für die Verwendung von vSGA zu konfigurieren, müssen Sie die 3D-Einstellungen für die Grafikkarte der virtuellen Maschine im vSphere Web Client konfigurieren.

Voraussetzungen

- Stellen Sie sicher, dass die virtuelle Linux-Maschine für die Verwendung als Desktop vorbereitet wurde, dass Horizon Agent installiert ist und die Maschine in einem Desktop-Pool bereitgestellt wird.
- Stellen Sie sicher, dass das NVIDIA VIB auf dem ESXi-Host installiert ist. Siehe [Installieren des VIB für die NVIDIA-Grafikkarte für vSGA auf dem ESXi-Host](#).

Verfahren

- 1 Schalten Sie die virtuelle Maschine aus
- 2 Im vSphere Web Client wählen Sie die virtuelle Maschine aus und klicken auf der Registerkarte **VM-Hardware** auf **Einstellungen bearbeiten**.
- 3 In der Registerkarte „Virtuelle Hardware“ klicken Sie auf **Grafikkarte**, um die Menüeinstellungen zu erweitern.
- 4 Legen Sie für **Gesamter Videoarbeitsspeicher** den Wert 128 MB fest.
- 5 Für **3D-Grafiken** wählen Sie **3D-Unterstützung aktivieren** aus.
- 6 Für **3D-Renderer** wählen Sie **Hardware** aus dem Dropdown-Menü aus.
- 7 Für **3D-Arbeitsspeicher** wählen Sie einen geeigneten Wert gemäß den Anforderungen Ihrer Anwendung aus.

Wenn Ihre Benutzer mehr als drei Monitore angeschlossen haben, legen Sie einen Wert von mindestens 1024 MB fest.
- 8 Klicken Sie auf **OK**.
- 9 Schalten Sie die virtuelle Maschine ein.

Nächste Schritte

Stellen Sie sicher, dass vSGA auf der virtuellen Linux-Maschine ausgeführt wird.

Installieren Sie als Nächstes Horizon Agent auf der virtuellen Linux-Maschine. Siehe [Installieren von Horizon Agent auf einer virtuellen Linux-Maschine](#).

Überprüfen, ob vSGA auf einer virtuellen Linux-Maschine ausgeführt wird

Sie können überprüfen, ob vSGA auf einer virtuellen RHEL 7-Maschine ausgeführt wird, indem Sie die Protokolldatei der virtuellen Maschine und das Gastbetriebssystem durchsehen.

Verfahren

- 1 Öffnen Sie die Datei `vmware.log` für die virtuelle Maschine.

Wenn ein unterstützter GPU und ein unterstütztes NVIDIA VIB korrekt installiert sind, enthält die Protokolldatei Zeilen wie im folgenden Beispiel:

```
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"
```

Wenn ein unterstützter GPU und ein unterstütztes NVIDIA VIB nicht korrekt installiert sind, verwendet die virtuelle Maschine den Software-Renderer. Die Datei `vmware.log` enthält Zeilen wie im folgenden Beispiel:

```
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 Im Gastbetriebssystem auf der virtuellen Maschine geben Sie nachfolgend aufgeführten Befehl ein.

```
glxinfo|grep Gallium
```

Wenn vSGA funktioniert, gibt der Befehl folgenden Text zurück:

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

Wenn vSGA nicht korrekt funktioniert, gibt der Befehl folgenden Text zurück:

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

Nächste Schritte

Installieren Sie Horizon Agent auf der virtuellen Linux-Maschine. Siehe [Installieren von Horizon Agent auf einer virtuellen Linux-Maschine](#).

Installieren von Horizon Agent

Sie müssen Horizon Agent auf den Linux-Desktops installieren, sodass der View-Verbindungsserver mit den Desktops kommunizieren und sie verwalten kann.

Dieses Kapitel enthält die folgenden Themen:

- [Installieren von Horizon Agent auf einer virtuellen Linux-Maschine](#)
- [Konfigurieren des Zertifikats für den Linux Agent](#)
- [Durchführen eines Upgrades von Horizon Agent auf einer virtuellen Linux-Maschine](#)
- [Deinstallieren von Horizon 7 für Linux-Maschinen](#)

Installieren von Horizon Agent auf einer virtuellen Linux-Maschine

Bevor Sie eine virtuelle Linux-Maschine als Remote-Desktop bereitstellen können, müssen Sie auf dieser Horizon Agent installieren.

Ab Horizon 7.0.1 verwendet Horizon Agent for Linux durch vCenter verwaltete virtuelle Maschinen. Die verwalteten virtuellen Maschinen bieten die folgenden Erweiterungen.

- vCenter ist für die Linux-Desktop-Bereitstellung erforderlich.
- Für die Horizon Agent-Installation auf Linux ist keine Registrierung erforderlich.
- Bei einer Vielzahl an Linux-Desktop-Bereitstellungen können Sie den Horizon Agent auf der virtuellen Basismaschine installieren.

Vorsicht Wenn Sie NVIDIA GRID vGPU, vDGA oder vSGA verwenden möchten, müssen Sie diese 3D-Funktionen auf der virtuellen Linux-Maschine vor der Installation von Horizon Agent konfigurieren. Wenn Sie Horizon Agent zuerst installieren, werden erforderliche Parameter in der Datei `xorg.conf` überschrieben und die 3D-Grafikfunktionen können nicht verwendet werden.

Weitere Informationen finden Sie unter [Konfigurieren unterstützter RHEL-Distributionen für vGPU](#), [Konfigurieren von RHEL 6 für vDGA](#) oder [Konfigurieren von RHEL 7 für vSGA](#). Installieren Sie Horizon Agent erst nach dem Abschluss der 3D-Grafikkonfiguration.

Für die 2D-Grafikkonfiguration können Sie Horizon Agent nach dem Abschluss der unter [Vorbereiten einer Linux-Maschine für die Remote-Desktop-Bereitstellung](#) aufgeführten Schritte installieren.

Voraussetzungen

- Stellen Sie sicher, dass das Linux-Gastbetriebssystem für die Desktop-Verwendung vorbereitet ist. Siehe [Vorbereiten einer Linux-Maschine für die Remote-Desktop-Bereitstellung](#).
- Machen Sie sich mit dem Installationsskript von Horizon Agent für Linux vertraut. Siehe [Befehlszeilenoptionen für install_viewagent.sh](#).

Verfahren

- 1 Laden Sie die Installationsdatei für Horizon Agent für Linux von der VMware-Download-Site unter <https://my.vmware.com/web/vmware/downloads> herunter.

Unter „Desktop- und Endbenutzer-Computing“ wählen Sie den VMware Horizon 7-Download aus, der die Installationsdatei für Horizon Agent für Linux enthält.

Der Dateiname des Installationsprogramms lautet „VMware-horizonagent-linux-x86_64-y.y.y-xxxxxxx.tar.gz“ für 64 Bit, wobei y.y.y der Versionsnummer und xxxxxxx der Buildnummer entspricht.

- 2 Entpacken Sie das TAR-Archiv für Ihre Linux-Distribution auf dem Gastbetriebssystem.

Beispiel:

```
tar -xzf <View Agent-TAR-Archiv>
```

- 3 Wechseln Sie zum Ordner des TAR-Archivs.
- 4 Führen Sie das Skript `install_viewagent.sh` als Superuser aus.

Unter [Befehlszeilenoptionen für install_viewagent.sh](#) finden Sie eine Liste der Befehlszeilenoptionen.

Beispiel:

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

- 5 Geben Sie **Yes** ein, um die EULA-Vereinbarung anzunehmen, wenn Sie `install_viewagent.sh` ohne Angabe der Option `-A` ausführen.

Das Installationsprogramm wird ohne Annahme der EULA-Vereinbarung nicht ausgeführt.

- 6 Führen Sie einen Linux-Neustart durch, damit die Änderungen wirksam werden.

Nach der Installation wird der Dienst `viewagent` gestartet. Stellen Sie sicher, dass der Dienst mithilfe von `sudo service viewagent status` gestartet wird.

Nächste Schritte

Stellen Sie die virtuelle Maschine in einem Desktop-Pool bereit. Siehe [Erstellen eines manuellen Desktop-Pools für Linux](#).

Befehlszeilenoptionen für `install_viewagent.sh`

Das `install_viewagent.sh`-Skript installiert Horizon Agent auf einem Linux-Gastbetriebssystem.

Verwenden Sie für das `install_viewagent.sh`-Skript die folgende Syntax in einem Befehlsfenster der Gnome-Desktop-Umgebung.

```
install_viewagent.sh command_option argument [command_option argument] . . .
```

Das `install_viewagent.sh`-Skript enthält obligatorische und optionale Parameter.

Tabelle 5-1. `install_viewagent.sh` Optionaler, aber erforderlicher Parameter

Optionaler Parameter (erforderliche Informationen)	Beschreibung
-A yes	Annahme der Nutzungsbedingungen und der FIPS-Erklärung Um die Installation fortzusetzen zu können, müssen Sie yes angeben. Wird dieser Parameter nicht angegeben, werden Sie vom Installationsprogramm dazu aufgefordert.

Tabelle 5-2. Optionale Parameter für `install_viewagent.sh`

Optionale Parameter	Beschreibung
-a yes no	Installieren Sie die Unterstützung der Audio-Eingangsumleitung oder umgehen Sie diese. Die Standardeinstellung ist „yes“.
-f yes no	Installieren oder umgehen Sie die Unterstützung der kryptografischen Module, die auf FIPS 140-2-Kompatibilität (Federal Information Processing Standards) ausgelegt sind. Die Standardeinstellung ist no . Weitere Informationen finden Sie in der Beschreibung des FIPS 140-2-Modus unter Funktionen von Horizon Linux-Desktops .
-j	JMS SSL-Schlüsselspeicherkey. Standardmäßig wird vom Installationsprogramm eine zufällige Zeichenfolge generiert.
-m yes no	Installieren oder umgehen Sie die Unterstützung für die Smartcard-Umleitung. Die Standardeinstellung ist no .
-r yes no	Automatischer Neustart des Systems nach der Installation. Die Standardeinstellung ist no .
-s	Subject DN des selbst signierten Zertifikats. Standardmäßig wird vom Installationsprogramm „Blast“ verwendet.
-C yes no	Installieren oder umgehen Sie die Unterstützung für die Zwischenablagenumleitung. Die Standardeinstellung ist yes .
-F yes no	Installieren oder umgehen Sie die CDR-Unterstützung. Die Standardeinstellung ist yes .
-M yes no	Aktualisieren Sie Linux Agent auf verwalteten oder nicht verwalteten Agenten. Die Standardeinstellung ist yes .
-S yes no	Installieren oder umgehen Sie die Unterstützung für Single Sign-On-Anmeldung (SSO). Die Standardeinstellung ist yes .
-U yes no	Installieren oder umgehen Sie die USB-Unterstützung. Die Standardeinstellung ist no .

Tabelle 5-3. Beispiele für install_viewagent.sh -Parameter

Bedingung	Beispiele
Neue Installation	<pre>sudo ./install_viewagent.sh -A yes</pre> <p>Für eine neue Installation ist immer die Erstellung eines neuen Desktop-Pools erforderlich.</p>
Upgrade einer nicht verwalteten virtuellen Maschine unter Beibehaltung der Charakteristika einer nicht verwalteten virtuellen Maschine	<pre>sudo ./install_viewagent.sh -A yes-M no</pre> <p>Für diesen Upgrade-Typ ist die Erstellung eines neuen Desktop-Pools nicht erforderlich. Sie können den vorhandenen Desktop-Pool erneut verwenden.</p> <p>Hinweis Um die bestmögliche Leistung zu gewährleisten, sollten Sie keine nicht verwaltete virtuelle Maschine verwenden.</p>
Upgrade der Bereitstellung einer nicht verwalteten virtuellen Maschine und Umwandlung in Charakteristika einer verwalteten virtuellen Maschine. Für dieses Upgrade ist die Erstellung eines neuen Desktop-Pools auf dem Broker erforderlich.	<pre>sudo ./install_viewagent.sh -A yes</pre> <p>Für diesen Upgrade-Typ ist die Erstellung eines neuen Desktop-Pools erforderlich. Sie müssen den vorhandenen Desktop-Pool löschen.</p>

Konfigurieren des Zertifikats für den Linux Agent

Wenn Sie den Linux Agent installieren, erstellt das Installationsprogramm ein selbstsigniertes Zertifikat für VMwareBlastServer.

- Wenn das Blast Security Gateway auf dem Broker deaktiviert ist, übergibt VMwareBlastServer dieses Zertifikat an den Browser, der mithilfe von HTML Access eine Verbindung mit dem Linux-Desktop herstellt.
- Wenn das Blast Security Gateway auf dem Broker aktiviert ist, wird das Zertifikat von Blast Security Gateway an den Browser übergeben.

Um die Einhaltung von Branchen- oder Sicherheitsbestimmungen sicherzustellen, können Sie das selbstsignierte Zertifikat durch ein von einer Zertifizierungsstelle (CA, Certificate Authority) signiertes Zertifikat ersetzen .

Verfahren

- 1 Installieren Sie den privaten Schlüssel und das Zertifikat für VMwareBlastServer.
 - a Benennen Sie den privaten Schlüssel in `ru1.key` und das Zertifikat in `ru1.crt` um.
 - b Führen Sie `sudo chmod 550 /etc/vmware/ssl` aus.
 - c Kopieren Sie `ru1.crt` und `ru1.key` nach `/etc/vmware/ssl`.
 - d Führen Sie `chmod 440 /etc/vmware/ssl` aus.
- 2 Installieren Sie die Stamm- und Zwischenzertifizierungsstelle im Zertifizierungsstellen-Store des Linux-Betriebssystems.

Hinweis Erläuterungen zur Änderung der Linux-Systemeinstellungen finden Sie in der Dokumentation Ihrer Linux-Distribution.

Durchführen eines Upgrades von Horizon Agent auf einer virtuellen Linux-Maschine

Sie können das Upgrade von Horizon Agent auf einer virtuellen Linux-Maschine durchführen, indem Sie die aktuelle Version von Horizon Agent installieren.

Nicht verwaltete virtuelle Maschine: Das Agent-Installationsprogramm registriert die virtuelle Maschine am Broker. Dies erfordert Broker-Administratorinformationen. Der Assistent zur **Erstellung von Desktop-Pools** verwendet **Andere Quellen** auf der Seite „Computerquelle“, um die registrierte virtuelle Maschine auszuwählen.

Verwaltete virtuelle Maschine: Das Installationsprogramm kommuniziert nicht mit dem Broker. Der Assistent zur **Erstellung von Desktop-Pools** verwendet **Virtuelle vCenter-Maschinen** auf der Seite „Computerquelle“, um die virtuellen Maschinen über vCenter auszuwählen. Die Bereitstellung der verwalteten virtuellen Maschine unterstützt die folgenden Funktionen.

- Betriebsrichtlinie für Remote-Computer
- Benutzern das Zurücksetzen ihrer Computer gestatten

Hinweis Horizon Agent for Linux 7.0.0 und frühere Versionen funktionierten als nicht verwaltete virtuelle Maschinen. Der Horizon Agent for Linux 7.0.1 funktioniert als Unterstützung für verwaltete virtuelle Maschinen.

Sie können mit den folgenden Methoden eine Aktualisierung der Bereitstellung einer nicht verwalteten zur Bereitstellung einer verwalteten virtuellen Maschine vornehmen.

- Behalten Sie die Bereitstellung der nicht verwalteten virtuellen Maschine bei und nehmen Sie eine Aktualisierung auf die erforderliche Version vor. Für diesen Aktualisierungstyp sind keine Konfigurationsänderungen auf dem View-Verbindungsserver erforderlich.

- Nehmen Sie eine Aktualisierung von der Bereitstellung einer nicht verwalteten virtuellen Maschine auf die Bereitstellung einer verwalteten virtuellen Maschine einer beliebigen Version vor. Für diesen Aktualisierungstyp muss ein neuer Desktop-Pool auf dem View-Verbindungsserver erstellt werden.

Hinweis Für die Aktualisierung der Bereitstellung einer verwalteten virtuellen Maschine können Sie die Bereitstellung einer verwalteten virtuellen Maschine beibehalten und eine Aktualisierung auf die erforderliche Version vornehmen. Das Konvertieren der Bereitstellung der verwalteten virtuellen Maschine zur Bereitstellung der nicht verwalteten virtuellen Maschine während der Aktualisierung wird nicht unterstützt.

Die folgenden Parameter stehen für die Aktualisierung zur Verfügung.

Tabelle 5-4. Optionale Parameter für das Upgrade von Horizon Agent

Parameter	Beschreibung
-A yes	Annahme der Nutzungsbedingungen und der FIPS-Erklärung Um die Installation fortzusetzen zu können, müssen Sie yes angeben. Wird dieser Parameter nicht angegeben, werden Sie vom Installationsprogramm dazu aufgefordert.
-a yes no	Installieren Sie die Unterstützung der Audio-Eingangsumleitung oder umgehen Sie diese. Die Standardeinstellung ist „yes“.
-f yes no	Installieren oder umgehen Sie die Unterstützung der kryptografischen Module, die auf FIPS 140-2-Kompatibilität (Federal Information Processing Standards) ausgelegt sind. Die Standardeinstellung ist no . Weitere Informationen finden Sie in der Beschreibung des FIPS 140-2-Modus unter Funktionen von Horizon Linux-Desktops .
-m yes no	Installieren oder umgehen Sie die Unterstützung für die Smartcard-Umleitung. Die Standardeinstellung ist no .
-r yes no	Starten Sie das Betriebssystem nach der Installation neu. Die Standardeinstellung ist no .
-C yes no	Installieren oder umgehen Sie die Unterstützung für die Zwischenablagenumleitung. Die Standardeinstellung ist yes .
-F yes no	Installieren oder umgehen Sie die CDR-Unterstützung. Die Standardeinstellung ist yes .
-M yes no	Aktualisieren Sie den Linux Agent auf den Agent managed unmanaged. Der Standardwert lautet ja .
-S yes no	Installieren oder umgehen Sie die Single Sign-On (SSO)-Unterstützung. Die Standardeinstellung ist yes .
-U yes no	Installieren oder umgehen Sie die USB-Unterstützung. Die Standardeinstellung ist no .

Durchführen eines Upgrades von Horizon Agent auf einer virtuellen Linux-Maschine

Sie können das Upgrade von Horizon Agent auf einer Linux-Maschine durchführen, indem Sie die aktuelle Version von Horizon Agent installieren.

Voraussetzungen

- Stellen Sie sicher, dass der VMwareBlastServer-Prozess nicht ausgeführt wird.

Um diesen Prozess anzuhalten, stellen Sie sicher, dass der Benutzer von der Maschine abgemeldet ist und keine Desktop-Sitzung aktiv ist, oder starten Sie die Maschine neu.

Verfahren

- 1 Laden Sie die neueste Installationsdatei für Horizon Agent für Linux von der VMware-Download-Site unter <https://my.vmware.com/web/vmware/downloads> herunter.

Unter „Desktop- und Endbenutzer-Computing“ wählen Sie den VMware Horizon 7-Download aus, der die Installationsdatei für Horizon Agent für Linux enthält.

Der Dateiname des Installationsprogramms lautet „VMware-viewagent-linux-x86_64-y.y.y-xxxxxxx.tar.gz“ für 64 Bit, wobei y.y.y der Versionsnummer und xxxxxxx der Buildnummer entspricht.

- 2 Entpacken Sie das TAR-Archiv für Ihre Linux-Distribution auf dem Gastbetriebssystem.

Beispiel:

```
tar -xzf <View Agent-TAR-Archiv>
```

- 3 Wechseln Sie zum Ordner des TAR-Archivs.
- 4 Führen Sie das `install_viewagent.sh`-Skript aus, um nicht verwaltete virtuelle Maschinen mithilfe eines der folgenden Bereitstellungsszenarien zu aktualisieren:

Option	Beschreibung
Aktualisieren der Bereitstellung einer nicht verwalteten virtuellen Maschine und Beibehalten der Bereitstellung einer nicht verwalteten virtuellen Maschine	<pre>sudo ./install_viewagent.sh -A yes -M no</pre> <p>Hinweis Um die bestmögliche Leistung zu gewährleisten, sollten Sie keine nicht verwaltete virtuelle Maschine verwenden.</p>
Aktualisieren der Bereitstellung einer nicht verwalteten virtuellen Maschine und Ändern zur Bereitstellung einer verwalteten virtuellen Maschine	<pre>sudo ./install_viewagent.sh -A yes -M yes</pre> <p>Hinweis Löschen Sie in View Administrator den vorhandenen Desktop-Pool für die Bereitstellung der nicht verwalteten virtuellen Maschine und erstellen Sie einen neuen Desktop-Pool für die Bereitstellung der verwalteten virtuellen Maschine. Weitere Informationen finden Sie unter Erstellen eines manuellen Desktop-Pools für Linux.</p>
Aktualisieren der Bereitstellung einer verwalteten virtuellen Maschine	<pre>sudo ./install_viewagent.sh -A yes -M yes</pre> <p>Hinweis Nach dem Aktualisieren kann Ihr vorhandener Desktop-Pool erneut verwendet werden.</p>

Deinstallieren von Horizon 7 für Linux-Maschinen

Zum Deinstallieren von Horizon 7 für Linux auf einer virtuellen Maschine müssen Sie den Horizon Agent deinstallieren und die Konfigurationsdateien entfernen.

Voraussetzungen

Stellen Sie sicher, dass der VMwareBlastServer-Prozess nicht ausgeführt wird. Stellen Sie zum Beenden dieses Prozesses sicher, dass Sie sich von der Maschine abmelden und dass keine Desktop-Sitzung aktiv ist, oder starten Sie die Maschine neu.

Verfahren

- 1 Öffnen Sie auf der virtuellen Maschine ein Terminalfenster und führen Sie das Deinstallationsskript für Horizon Agent aus.

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

Das Skript beendet die Horizon Agent-Prozesse, löscht den Horizon Agent-Dienst und die Software aus dem Installationsverzeichnis `/usr/lib/vmware/viewagent`.

- 2 Löschen Sie die Horizon 7 for Linux-Konfigurationsdateien manuell aus dem Verzeichnis `/etc/vmware`.

Konfigurationsoptionen für Linux-Desktops

6

Mithilfe von Konfigurationsdateien können Sie verschiedene Optionen konfigurieren, um die Benutzererfahrung anzupassen.

Dieses Kapitel enthält die folgenden Themen:

- [Einstellen der Optionen in Konfigurationsdateien auf einem Linux-Desktop](#)
- [Beispiel für Blast-Einstellungen für Linux-Desktops](#)
- [Beispiel für Optionen der Clientlaufwerksumleitung für Linux-Desktops](#)
- [Ausblenden der vSphere-Konsolenanzeige auf einem Linux-Desktop](#)

Einstellen der Optionen in Konfigurationsdateien auf einem Linux-Desktop

Sie können verschiedene Optionen konfigurieren, indem Sie der Datei `/etc/vmware/config` oder `/etc/vmware/viewagent-custom.conf` Einträge hinzufügen.

Bei der Installation von View Agent oder Horizon Agent kopiert das Installationsprogramm die beiden Konfigurationsvorlagendateien `config.template` und `viewagent-custom.conf.template` in `/etc/vmware`. Außerdem kopiert das Installationsprogramm, falls die Dateien `/etc/vmware/config` und `/etc/vmware/viewagent-custom.conf` nicht vorhanden sind, `config.template` nach `config` und `viewagent-custom.conf.template` in `viewagent-custom.conf`. In den Vorlagendateien sind alle Konfigurationsoptionen aufgelistet und dokumentiert. Um eine Option einzustellen, entfernen Sie einfach den Kommentar und ändern Sie den Wert wie gewünscht.

So aktiviert beispielsweise die folgende Zeile in `/etc/vmware/config` den verlustfreien PNG-Modus.

```
RemoteDisplay.alwaysLossless=TRUE
```

Nachdem Sie Ihre Änderungen vorgenommen haben, müssen Sie Linux neu starten, damit die Änderungen wirksam werden.

Konfigurationsoptionen in /etc/vmware/config

VMwareBlastServer und seine zugehörigen Plug-ins verwenden die Konfigurationsdatei /etc/vmware/config.

Hinweis Die folgende Tabelle enthält Beschreibungen für alle von Agent erzwungenen Richtlinieneinstellungen für USB in der Horizon Agent-Konfigurationsdatei. Horizon Agent verwendet die Einstellungen, um zu entscheiden, ob der USB-Anschluss zur Host-Maschine umgeleitet werden kann. Horizon Agent übergibt die Einstellungen auch an Horizon Client zur Auswertung und Erzwingung je nachdem, ob Sie den merge(**m**)-Modifizierer zur Anwendung der Horizon Agent-Filterrichtlinieneinstellung zusätzlich zur Horizon Client-Filterrichtlinieneinstellung festlegen oder den (**o**)-Modifizierer zur Verwendung der Horizon Agent-Filterrichtlinieneinstellung anstelle der Horizon Client-Filterrichtlinieneinstellung überschreiben.

Tabelle 6-1. Konfigurationsoptionen in /etc/vmware/config

Option	Wert/Format	Standard	Beschreibung
RemoteDisplay.alwaysLossless	true oder false	false	Grafische Anwendungen und insbesondere grafische Anwendungen zur Bildbearbeitung erfordern ein pixelgenaues Rendering von Bildern in der Clientanzeige eines Linux-Desktops. Sie haben die Möglichkeit, einen speziellen verlustfreien PNG-Modus für Bilder und die Videowiedergabe zu konfigurieren, die auf einem Linux-Desktop generiert und auf dem Clientgerät gerendert werden. Diese Funktion verwendet zusätzliche Bandbreite zwischen dem Client und dem ESXi-Host.
mksVNCServer.useUIInputButton Mapping	true oder false	false	Legen Sie diese Option fest, um die Unterstützung einer linkshändigen Maus auf Ubuntu oder RHEL 7 zu aktivieren. CentOS und RHEL 6 unterstützen eine linkshändige Maus, und Sie müssen diese Option nicht festlegen.
RemoteDisplay.allowAudio	true oder false	true	Legen Sie diese Option fest, um die Audio-Ausgabe zu aktivieren/deaktivieren.
VVC.ScRedir.Enable	true oder false	true	Legen Sie diese Option fest, um die Smartcard-Umleitung zu aktivieren/deaktivieren.
VVC.logLevel	fatal error, warn, info, debug oder trace	info	Verwenden Sie diese Option zur Festlegung der Protokollebene des VVC-Proxy-Knotens.
VVC.RTAV.Enable	true oder false	true	Legen Sie diese Option fest, um die Audio-Eingabe zu aktivieren/deaktivieren.

Option	Wert/Format	Standard	Beschreibung
Clipboard.Direction	0, 1, 2, oder 3	2	<p>Durch diese Option wird die Richtlinie für die Zwischenablagenumleitung bestimmt.</p> <ul style="list-style-type: none"> ■ 0 - Zwischenablagenumleitung deaktivieren. ■ 1 - Zwischenablagenumleitung in beide Richtungen aktivieren. ■ 2 - Zwischenablagenumleitung nur vom Client zum Remote-Desktop aktivieren. ■ 3 - Zwischenablagenumleitung nur vom Remote-Desktop zum Client aktivieren.
cdrserver.logLevel	error, warn, info, debug, trace oder verbose	info	Verwenden Sie diese Option zur Festlegung der Protokollebene für <code>vmware-CDRserver.log</code> .
cdrserver.forcedByAdmin	true oder false	false	Legen Sie diese Option fest, um für den Client die gemeinsame Nutzung zusätzlicher Ordner auszuschließen oder zuzulassen, die nicht mit der Option <code>cdrserver.shareFolders</code> angegeben wurden.
cdrserver.shareFolders	<i>file_path1,R; file_path2,; file_path3,R; . . .</i>	Nicht definiert	<p>Geben Sie einen oder mehrere Dateipfade zu den Ordnern an, die der Client mit dem Linux-Desktop gemeinsam nutzen kann. Beispiel:</p> <ul style="list-style-type: none"> ■ Für einen Windows-Client: C:\spreadsheets,;D:\ebooks,R ■ Für einen Nicht-Windows-Client: /tmp/spreadsheets;/tmp/ebooks,;/home/finance,R
cdrserver.permissions	R	RW	<p>Verwenden Sie diese Option zur Anwendung zusätzlicher Lese/Schreib-Berechtigungen, über die Horizon Agent für die von Horizon Client freigegebenen Ordner verfügt. Beispiel:</p> <ul style="list-style-type: none"> ■ Wenn der von Horizon Client freigegebene Ordner über die Berechtigungen <code>read</code> und <code>write</code> verfügt und Sie cdrserver.permissions=R festlegen, verfügt Horizon Agent nur über <code>read</code>-Zugriffsberechtigungen. ■ Wenn der von Horizon Client freigegebene Ordner nur über <code>read</code>-Berechtigungen verfügt und Sie cdrserver.permissions=RW festlegen, verfügt Horizon Agent weiterhin nur über <code>read</code>-Zugriffsrechte. Für Horizon Agent ist es nicht möglich, das von Horizon Client festgelegte <code>Nur-read</code>-Attribut zu ändern. Die einzige Möglichkeit für Horizon Agent ist das Entfernen der Schreibzugriffsrechte. <p>Beispiele für eine typische Anwendung:</p> <ul style="list-style-type: none"> ■ cdrserver.permissions=R ■ #cdrserver.permissions=R (d. h. den Eintrag auskommentieren oder löschen)

Option	Wert/Format	Standard	Beschreibung
cdrserver.cacheEnable	true oder false	true	Legen Sie diese Option fest, um die Funktion des Schreibcache von der Agentseite zur Clientseite zu aktivieren oder zu deaktivieren.
UsbRedirPlugin.log.logLevel	error, warn, info, debug, trace oder verbose	info	Verwenden Sie diese Option zur Festlegung der Protokollebene des USB-Umleitungs-Plug-Ins.
UsbRedirServer.log.logLevel	error, warn, info, debug, trace oder verbose	info	Verwenden Sie diese Option zur Festlegung der Protokollebene des USB-Umleitungsservers.
viewusb.AllowAutoDeviceSplitting	{m o}: {true false}	Nicht definiert, entspricht false	Legen Sie diese Option fest, um das automatische Splitten von Composite USB-Geräten zuzulassen oder auszuschließen. Beispiel: m:true
viewusb.SplitExcludeVidPid	{m o}:vid-xxx1_pid-yyy1[;vid-xxx2_pid-yyy2;...]	Nicht definiert	Verwenden Sie diese Option, um ein bestimmtes Composite USB-Gerät für das Splitten nach Anbieter- und Produkt-IDs auszuschließen oder einzubeziehen. Das Format dieser Einstellung lautet vid-xxx1_pid-yyy1[;vid-xxx2_pid-yyy2;...] . ID-Nummern müssen in hexadezimaler Schreibweise angegeben werden. Sie können das Platzhalterzeichen (*) anstelle einzelner Ziffern in einer ID verwenden. Beispiel: m:vid-0f0f_pid-55**
viewusb.SplitVidPid	{m o}: vid-xxxx_pid-yyyy([exintf:zz[;exintf:ww]])[;...]	Nicht definiert	Legen Sie diese Option fest, um die Komponenten eines Composite USB-Gerätes, die durch Anbieter- und Produkt-IDs angegeben sind, als separate Geräte zu behandeln. Das Format dieser Einstellung lautet vid-xxxx_pid-yyyy(exintf:zz[;exintf:ww]) . Sie können mit dem Stichwort exintf Komponenten durch Angabe ihrer Schnittstellennummer von der Umleitung ausschließen. Sie müssen hexadezimale ID-Nummern und dezimale Schnittstellennummern einschließlich der 0 am Anfang angeben. Sie können das Platzhalterzeichen (*) anstelle einzelner Ziffern in einer ID verwenden. Beispiel: o:vid-0f0f_pid-***(exintf-01);vid-0781_pid-554c(exintf:01;exintf:02)
Hinweis			
Horizon schließt nicht automatisch die Komponenten ein, die Sie nicht explizit ausgeschlossen haben. Sie müssen eine Filterrichtlinie wie z. B. Include VidPid Device (VidPid-Gerät einbeziehen) angeben, um diese Komponenten einzubeziehen.			
viewusb.AllowAudioIn	{m o}: {true false}	Nicht definiert, entspricht true	Verwenden Sie diese Option, um die Umleitung für Audio-Eingabe-Geräte zuzulassen oder auszuschließen. Beispiel: o:false

Option	Wert/Format	Standard	Beschreibung
viewusb.AllowAudioOut	{m o}: {true false}	Nicht definiert, entspricht <code>false</code>	Legen Sie diese Option fest, um die Umleitung für Audio-Ausgabe-Geräte zuzulassen oder auszuschließen.
viewusb.AllowHIDBootable	{m o}: {true false}	Nicht definiert, entspricht <code>true</code>	Verwenden Sie diese Option, um die Umleitung anderer Eingabegeräte neben Tastatur und Maus, die zur Startzeit verfügbar sind (auch als „startfähige Eingabegeräte“ bezeichnet), zuzulassen oder auszuschließen.
viewusb.AllowDevDescFailsafe	{m o}: {true false}	Nicht definiert, entspricht <code>false</code>	Legen Sie diese Option fest, um die Umleitung für Geräte zuzulassen oder auszuschließen, auch wenn Horizon Client die Konfigurations-/Gerätebeschreibungen nicht abrufen kann. Um ein Gerät auch beim Scheitern des Abrufs der Konfigurations-/Gerätebeschreibungen zuzulassen, muss dieses in „Include“-Filter wie z. B. IncludeVidPid oder IncludePath eingeschlossen werden.
viewusb.AllowKeyboardMouse	{m o}: {true false}	Nicht definiert, entspricht <code>false</code>	Verwenden Sie diese Option, um die Umleitung von Tastaturen mit eingebauten Zeigegegeräten (Maus, Trackball oder Touchpad) zuzulassen oder auszuschließen.
viewusb.AllowSmartcard	{m o}: {true false}	Nicht definiert, entspricht <code>false</code>	Legen Sie diese Option fest, um die Umleitung für Smartcard-Geräte zuzulassen oder auszuschließen.
viewusb.AllowVideo	{m o}: {true false}	Nicht definiert, entspricht <code>true</code>	Verwenden Sie diese Option, um die Umleitung für Videogeräte zuzulassen oder auszuschließen.
viewusb.DisableRemoteConfig	{m o}: {true false}	Nicht definiert, entspricht <code>false</code>	Legen Sie diese Option fest, um die Verwendung von Horizon Agent-Einstellungen zuzulassen oder auszuschließen, wenn eine USB-Gerätefilterung durchgeführt wird.
viewusb.ExcludeAllDevices	{true false}	Nicht definiert, entspricht <code>false</code>	Verwenden Sie diese Option, um alle USB-Geräte von der Umleitung auszuschließen oder in die Umleitung einzubeziehen. Wenn für diese Einstellung true festgelegt ist, können Sie andere Richtlinieneinstellungen verwenden, um zuzulassen, dass bestimmte Geräte oder Gerätefamilien umgeleitet werden. Wenn für diese Einstellung false festgelegt ist, können Sie andere Richtlinieneinstellungen verwenden, um zu verhindern, dass bestimmte Geräte oder Gerätefamilien umgeleitet werden. Wenn Sie den Wert von ExcludeAllDevices in Horizon Agent auf true festlegen und diese Einstellung an Horizon Client übergeben wird, hat die Horizon Agent-Einstellung Vorrang vor der Horizon Client-Einstellung.

Option	Wert/Format	Standard	Beschreibung
viewusb.ExcludeFamily	<code>{m o}: family_name_1[; family_name_2;. ..]</code>	Nicht definiert	<p>Verwenden Sie diese Option, um Gerätefamilien von der Umleitung auszuschließen oder in die Umleitung einzubeziehen. Beispiel: m:bluetooth;smart-card</p> <p>Wenn Sie das automatische Gerätesplitten aktiviert haben, prüft Horizon die Gerätefamilie jeder Schnittstelle eines Composite USB-Gerätes, um zu entscheiden, welche Schnittstelle ausgeschlossen werden sollte. Wenn Sie das automatische Gerätesplitten deaktiviert haben, prüft Horizon die Gerätefamilie des gesamten Composite USB-Gerätes.</p> <p>Hinweis Allerdings sind Maus und Tastatur standardmäßig von der Umleitung ausgeschlossen und müssen deshalb nicht mit dieser Einstellung ausgeschlossen werden.</p>
viewusb.ExcludeVidPid	<code>{m o}:vid-xxx1 pid-yyy1[;vid- xxx2_pid- yyy2;..]</code>	Nicht definiert	<p>Legen Sie diese Option fest, um Geräte mit einer bestimmten Anbieter- oder Produkt-ID von der Umleitung auszuschließen. Sie müssen hexadezimale ID-Nummern angeben. Sie können das Platzhalterzeichen (*) anstelle einzelner Ziffern in einer ID verwenden.</p> <p>Beispiel: 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>	Nicht definiert	<p>Verwenden Sie diese Option, um Geräte an bestimmten Hub- oder Portpfaden von der Umleitung auszuschließen. Bus- und Portnummern müssen im hexadezimalen Format angegeben werden. Sie können das Platzhalterzeichen nicht in Pfaden verwenden.</p> <p>Beispiel: 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>	Nicht definiert	<p>Legen Sie diese Option fest, um Gerätefamilien in die Umleitung einzubeziehen.</p> <p>Beispiel: o:storage; smart-card</p>
viewusb.IncludePath	<code>{m o}:bus-x1[/ y1].../ port- z1[;bus-x2[/ y2].../ portz2;...]</code>	Nicht definiert	<p>Verwenden Sie diese Option, um Geräte an bestimmten Hub- oder Portpfaden in die Umleitung einzubeziehen. Bus- und Portnummern müssen im hexadezimalen Format angegeben werden. Sie können das Platzhalterzeichen nicht in Pfaden verwenden.</p> <p>Beispiel: 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>	Nicht definiert	<p>Legen Sie diese Option fest, um Geräte mit bestimmten Anbieter- oder Produkt-IDs in die Umleitung einzubeziehen. Sie müssen hexadezimale ID-Nummern angeben. Sie können das Platzhalterzeichen (*) anstelle einzelner Ziffern in einer ID verwenden.</p> <p>Beispiel: o:vid-***_pid-0001;vid-0561_pid-554c</p>

Option	Wert/Format	Standard	Beschreibung
mksVNCServer.useXExtButtonMapping	true oder false	false	Legen Sie diese Option fest, um die Unterstützung einer linkshändigen Maus auf SLED 11 SP3 zu aktivieren oder zu deaktivieren.
mksvhan.clipboardSize	Eine Ganzzahl	1024	Verwenden Sie diese Option, um die maximale Größe der Zwischenablage für das Kopieren und Einfügen anzugeben.
RemoteDisplay.maxBandwidthKbps	Eine Ganzzahl	4096000	Legt die maximale Bandbreite für eine VMware Blast-Sitzung in Kilobits pro Sekunde (KBit/s) fest. Die Bandbreite umfasst den gesamten Sitzungsdatenverkehr, Bilddarstellung, Audio, virtuelle Kanäle und VMware Blast-Steuerung eingeschlossen. Der maximale Wert lautet 4 GBit/s (4096000).
RemoteDisplay.maxFPS	Eine Ganzzahl	60	Legt die maximale Rate der Bildschirmaktualisierungen fest. Mit dieser Einstellung steuern Sie die durchschnittliche Bandbreite, die Benutzer in Anspruch nehmen. Der gültige Wert sollte zwischen 3 und 60 liegen. Die Standardeinstellung beträgt 60 Aktualisierungen pro Sekunde.
RemoteDisplay.enableStats	true oder false	false	Aktivieren oder deaktivieren Sie die Blast-Protokollstatistik im MKS-Protokoll, beispielsweise FPS, RTT usw.
RemoteDisplay.allowH264	true oder false	true	Legen Sie diese Option zum Aktivieren oder Deaktivieren der H.264-Codierung fest.
vdpservice.log.logLevel	fatal error, warn, info, debug oder trace	info	Verwenden Sie diese Option zum Festlegen der Protokollebene des vdp-service.
RemoteDisplay.qpmaxH264	Verfügbarer Wertebereich: 0–51	36	Verwenden Sie diese Option, um den Quantisierungsparameter „H264minQP“ festzulegen, der die für die H.264-Codierung konfigurierte beste Bildqualität angibt. Geben Sie einen Wert an, der größer ist als der für „RemoteDisplay.qpminH264“ festgelegte Wert.
RemoteDisplay.qpminH264	Verfügbarer Wertebereich: 0–51	10	Verwenden Sie diese Option, um den Quantisierungsparameter „H264maxQP“ festzulegen, der die für die H.264-Codierung konfigurierte geringste Bildqualität angibt. Geben Sie einen Wert an, der kleiner ist als der für „RemoteDisplay.qpmaxH264“ festgelegte Wert.
RemoteDisplay.minQualityJPEG	Verfügbarer Wertebereich: 1–100	25	Legt die Bildqualität für die Desktop-Anzeige für die JPEG/PNG-Codierung fest. Die Einstellungen für eine niedrige Bildqualität sind für Bereiche gedacht, die sich häufig ändern, z. B. durch einen Bildlauf.

Option	Wert/Format	Standard	Beschreibung
RemoteDisplay.midQualityJPEG	Verfügbarer Wertebereich: 1–100	35	Legt die Bildqualität für die Desktop-Anzeige für die JPEG/PNG-Codierung fest. Legt die Einstellungen für die mittlere Qualität der Desktop-Anzeige fest.
RemoteDisplay.maxQualityJPEG	Verfügbarer Wertebereich: 1–100	90	Legt die Bildqualität für die Desktop-Anzeige für die JPEG/PNG-Codierung fest. Die Einstellungen für eine hohe Bildqualität sind für eher statische Bereiche sinnvoll.

Konfigurationsoptionen in /etc/vmware/viewagent-custom.conf

Java Standalone Agent verwendet die Konfigurationsdatei /etc/vmware/viewagent-custom.conf.

Tabelle 6-2. Konfigurationsoptionen in /etc/vmware/viewagent-custom.conf

Option	Wert	Standard	Beschreibung
Subnet	NULL oder Netzwerkadresse und Maskierung in IP-Adresse/im CIDR-Format	NULL	<p>Wenn mehrere lokale IP-Adressen mit unterschiedlichen Subnetzen vorhanden sind, können Sie mit dieser Option das Subnetz festlegen, dass der Linux-Agent für den View-Verbindungsserver zur Verfügung stellen soll.</p> <p>Wenn mehrere Subnetzkonfigurationen auf einem Linux-Agent-Computer ermittelt wurden, müssen Sie mit dieser Option das Subnetz angeben, das vom Linux-Agent verwendet werden soll. Wenn Sie z. B. Docker auf einem Linux-Computer installiert haben, wird diese Plattform als ein virtueller Netzwerkadapter behandelt. Um zu verhindern, dass der Linux-Agent die Docker-Plattform als virtuellen Netzwerkadapter verwendet, müssen Sie diese Option zur Verwendung des physischen Netzwerkadapters festlegen.</p> <p>Sie müssen den Wert in der IP-Adresse/im CIDR-Format angeben. Beispiel: Subnet=192.168.1.0/24.</p> <p>Mit NULL wird festgelegt, dass der Linux-Agent die IP-Adresse per Zufallsauswahl bestimmt.</p>
SSOEnable	true oder false	true	Legen Sie diese Option fest, um Single Sign-On (SSO) zu aktivieren/deaktivieren.
SSOUserFormat	Eine Textzeichenfolge	[Benutzername]	<p>Verwenden Sie diese Option, um das Format des Anmeldenamens für das Single Sign-On anzugeben. Der Standard ist lediglich der Benutzername. Legen Sie diese Option fest, wenn auch der Domänenname erforderlich ist. Meist ist der Anmeldename der Domänenname plus einem Sonderzeichen, gefolgt vom Benutzernamen. Wenn das Sonderzeichen ein Rückschrägstrich ist, muss ein weiterer Rückschrägstrich als Escape-Zeichen verwendet werden. Beispiele für Formate von Anmeldenenamen:</p> <ul style="list-style-type: none"> ■ SSOUserFormat=[Domäne]\\[Benutzername] ■ SSOUserFormat=[Domäne]+[Benutzername] ■ SSOUserFormat=[Benutzername]@[Domäne]
CDREnable	true oder false	true	Legen Sie diese Option fest, um die Funktion der Clientlaufwerksumleitung (Client Drive Redirection, CDR) zu aktivieren oder zu deaktivieren.

Option	Wert	Standard	Beschreibung
USBEnable	true oder false	true	Legen Sie diese Option fest, um die Funktion der USB-Umleitung zu aktivieren oder zu deaktivieren.
KeyboardLayoutSync	true oder false	true	<p>Verwenden Sie diese Option, um festzulegen, ob das Systemgebietsschema und das aktuelle Tastaturlayout eines Clients mit den Horizon Agent for Linux-Desktops synchronisiert werden sollen.</p> <p>Wenn diese Einstellung aktiviert wurde oder nicht konfiguriert ist, ist eine Synchronisierung zugelassen. Wenn diese Einstellung deaktiviert ist, ist eine Synchronisierung nicht erlaubt.</p> <p>Diese Funktion wird nur für Horizon Client für Windows und für die Gebietsschemas Englisch, Französisch, Deutsch, Japanisch, Koreanisch, Spanisch, Chinesisch (vereinfacht) und Chinesisch (traditionell) unterstützt.</p>
StartBlastServerTimeout	Eine Ganzzahl	20	Diese Option legt die Zeit (in Sekunden) fest, die dem VMwareBlastServer-Prozess zur Initialisierung zur Verfügung steht. Wenn der Prozess nicht innerhalb dieses Timeout-Werts verfügbar ist, schlägt die Anmeldung des Benutzers fehl.
SSLCiphers	Eine Textzeich enfolge	!aNULL:kECDH +AESGCM:ECDH +AESGCM:RSA +AESGCM:kECDH +AES:ECDH+AES:RSA +AES	Verwenden Sie diese Option zum Festlegen der Liste der Verschlüsselungen. Sie müssen das Format verwenden, das in https://www.openssl.org/docs/manmaster/man1/ciphers.html definiert ist.
SSLProtocols	Eine Textzeich enfolge	TLSv1_1:TLSv1_2	Verwenden Sie diese Option zum Festlegen der Sicherheitsprotokolle. Die unterstützten Protokolle sind TLSv1.0, TLSv1.1 und TLSv1.2.
SSLCipherServerPreference	true oder false	true	Verwenden Sie diese Option, um die Option SSL_OP_CIPHER_SERVER_PREFERENCE zu aktivieren oder zu deaktivieren. Weitere Informationen finden Sie unter https://www.openssl.org/docs/manmaster/ssl/SSL_CTX_set_options.html .
UseGnomeFlashback	true oder false	false	<p>Diese Option legt fest, ob die GNOME Flashback (Metacity)-Desktop-Umgebung verwendet werden soll, wenn sie auf einem Ubuntu 14.04- oder Ubuntu16.04-System installiert ist. Diese Option ist unabhängig von der Aktivierung der SSO-Funktion wirksam.</p> <p>Wenn für diese Option TRUE festgelegt ist, wird immer die GNOME Flashback (Metacity)-Desktop-Umgebung anstelle der Standard-Desktop-Umgebung verwendet.</p> <p>Tipp Um die Leistung Ihres Systems zu verbessern, konfigurieren Sie UseGnomeFlashback=TRUE nach des GNOME Flashback (Metacity)-Desktops auf Ihrem Ubuntu 14.04- oder Ubuntu 16.04-System.</p>

Option	Wert	Standard	Beschreibung
LogCnt	Eine Ganzzahl	-1	Verwenden Sie diese Option zur Festlegung der Anzahl der reservierten Protokolle in /tmp/vmware-root. <ul style="list-style-type: none"> ■ -1 - alle beibehalten ■ 0 - alle löschen ■ > 0 - Anzahl der reservierten Protokolle.
RunOnceScript			Verwenden Sie diese Option, um die geklonte VM erneut zu AD beitreten zu lassen. Legen Sie das RunOnceScript fest, nachdem der Hostname geändert wurde. Das angegebene Skript wird nur einmal nach der ersten Änderung des Hostnamens ausgeführt. Das Skript wird als Stammberechtigung ausgeführt, wenn der Agentendienst gestartet wird und sich der Hostname seit der Agenteninstallation geändert hat. Zum Beispiel müssen Sie für die winbind-Lösung die Basis-VM AD mit winbind beitreten lassen und diese Option auf einen Skriptpfad festlegen. Diese muss den Befehl für den erneuten Beitritt zur Domäne /usr/bin/net ads join -U <ADUserName>%<ADUserPassword> enthalten. Nach dem VM-Klon ändert die Betriebssystemanpassung den Hostnamen. Wenn der Agentendienst gestartet wird, wird das Skript ausgeführt, damit die geklonte VM zu AD beitrifft.
RunOnceScriptTimeout		120	Verwenden Sie diese Option, um die Zeit bis zur Zeitüberschreitung in Sekunden für die Option „RunOnceScript“ festzulegen. Legen Sie z. B. RunOnceScriptTimeout=120 fest

Hinweis Die drei Sicherheitsoptionen SSLCiphers, SSLProtocols und SSLCipherServerPreference gelten für den VMwareBlastServer-Prozess. Beim Start des VMwareBlastServer-Prozesses durchläuft der Java Standalone Agent diese Optionen als Parameter. Wenn Blast Secure Gateway (BSG) aktiviert ist, wirken sich diese Optionen auf die Verbindung zwischen BSG und dem Linux-Desktop aus. Wenn BSG deaktiviert ist, wirken sich diese Optionen auf die Verbindung zwischen dem Client und dem Linux-Desktop aus.

Beispiel für Blast-Einstellungen für Linux-Desktops

Sie können die Bildqualität Ihrer Remote-Desktop-Anzeige anpassen, um die Benutzerfreundlichkeit zu verbessern. Durch die Anpassung der Bildqualität können Sie im Fall schlechter Netzwerkverbindungen eine durchgängige Benutzerfreundlichkeit sicherstellen.

Beispiel für VMware Blast Extreme-Protokolleinstellungen

VMwareBlastServer und seine zugehörigen Plug-ins verwenden die Konfigurationsdatei /etc/vmware/config.

Tabelle 6-3. Beispiel für Blast-Konfigurationsoptionen in /etc/vmware/config

Name der Option	Parameter	Hochgeschw. Hochgeschw. eits-LAN	LAN	Dedizierte s WAN	Breitband -WAN	Langsam es WAN	Extrem langsam
Bandbreiteneinstellungen	RemoteDisplay.maxBandwidthKbps	1000000 (1 GBit/s)	1000000 (1 GBit/s)	1000000 (1 GBit/s)	5000 (5 MBit/s)	2000 (2 MBit/s)	1000 (1 MBit/s)
Maximale FPS	RemoteDisplay.maxFPS	60	30	30	20	15	5
Audiowiedergabe	RemoteDisplay.allowAudio	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE
Anzeigequalität (JPEG/PNG)	RemoteDisplay.maxQualityJPEG	90	90	90	70	60	50
Anzeigequalität (JPEG/PNG)	RemoteDisplay.midQualityJPEG	35	35	35	35	35	35
Anzeigequalität (JPEG/PNG)	RemoteDisplay.minQualityJPEG	25	25	25	20	20	20
Anzeigequalität (H.264)	RemoteDisplay.qpmaxH264	28	36	36	36	36	42
Anzeigequalität (H.264)	RemoteDisplay.qpminH264	10	10	10	10	10	10

Beispiel für Optionen der Clientlaufwerksumleitung für Linux-Desktops

Durch Konfiguration der Optionen für die Clientlaufwerksumleitung (Client Drive Redirection, CDR) können Sie festlegen, ob auf die freigegebenen Ordner und Laufwerke eines lokalen Systems von Linux-Remote-Desktops aus zugegriffen werden kann.

Konfigurieren Sie CDR-Einstellungen durch Hinzufügen von Einträgen zur Datei /etc/vmware/config.

Mit dem nachfolgenden Konfigurationsbeispiel werden die Ordner D:\ebooks und C:\spreadsheets freigegeben, wobei beide Ordner schreibgeschützt sind. Außerdem können vom Client keine weiteren Ordner freigegeben werden.

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

Im vorhergehenden Beispiel wurde das Komma „**,**“ nach **ebooks** platziert. **spreadsheets** ist obligatorisch für eine korrekte Optionsanalyse.

Ein „**R**“ in der `cdserver.sharedFolders`-Option hat Auswirkungen auf alle in dieser Einstellung aufgeführten Ordner. Im folgenden Beispiel sind die Ordner **ebooks** und **spreadsheets** schreibgeschützt, auch wenn der **R**-Wert nur nach dem Ordnerpfad `/home/jsmith` platziert wird.

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

Ausblenden der vSphere-Konsolenanzeige auf einem Linux-Desktop

Wenn ein Benutzer eine Verbindung mit einem Linux-Desktop herstellt, kann der Desktop auch in der vSphere-Konsole der virtuellen Linux-Maschine dargestellt werden. Virtuelle Linux-Maschinen lassen sich so konfigurieren, dass die vSphere-Konsole leer ist, wenn Benutzer eine Verbindung mit ihren Desktops herstellen.

Verfahren

- ◆ Auf dem ESXi-Host fügen Sie der VMX-Datei der virtuellen Maschine die folgende Zeile hinzu.

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

Die vSphere-Konsolenanzeige bleibt dann leer, auch wenn Sie nach dem Abmelden des Benutzers vom Desktop eine Verbindung zu einer virtuellen Maschine herstellen.

Erstellen und Verwalten von Linux-Desktop-Pools

7

Um virtuelle Linux-Maschinen für die Verwendung als Remote-Desktops zu konfigurieren, müssen Sie einen Desktop-Pool mit virtuellen Linux-Maschinen erstellen.

Horizon für Linux unterstützt die folgenden Typen von Desktop-Pools:

- Manueller Desktop-Pool mit virtueller vCenter-Maschine
- Automatisierter Full-Clone-Desktop-Pool

Um einen manuellen Desktop-Pool mit einer virtuellen vCenter-Maschine zu erstellen, müssen Sie auf allen virtuellen Maschinen Horizon Agent installieren. Verwenden Sie dann den Verbindungsserver-Assistenten für das Erstellen von Desktop-Pools, um die virtuellen Maschinen zum Desktop-Pool hinzuzufügen. Informationen zum Klonen einer großen Anzahl virtueller Maschinen finden Sie unter [Überblick über die Massenbereitstellung von Linux-Desktops](#).

Um einen automatisierten Full-Clone-Desktop-Pool zu erstellen, müssen Sie Horizon Agent auf einer Linux-VM-Vorlage installieren. Verwenden Sie dann den Verbindungsserver-Assistenten für das Erstellen von Desktop-Pools, um die vollständig virtuellen Maschinen zu klonen.

Dieses Kapitel enthält die folgenden Themen:

- [Erstellen eines manuellen Desktop-Pools für Linux](#)
- [Verwalten eines Desktop-Pools für Linux](#)
- [Erstellen eines automatisierten Full-Clone-Desktop-Pools für Linux](#)
- [Broker-PowerCLI-Befehle](#)

Erstellen eines manuellen Desktop-Pools für Linux

Sie können einen manuellen Desktop-Pool für virtuelle Linux-Maschinen erstellen.

Voraussetzungen

- Stellen Sie sicher, dass Horizon View Agent auf dem Linux-Gastbetriebssystem installiert ist. Siehe [Installieren von Horizon Agent auf einer virtuellen Linux-Maschine](#).
- Prüfen Sie, ob vCenter Server zum Horizon-Verbindungsserver hinzugefügt wurde.

Verfahren

- 1 In View Administrator fügen Sie einen manuellen Desktop-Pool hinzu.
Wählen Sie **Katalog > Desktop-Pools > Hinzufügen** aus.
- 2 Wählen Sie **Manueller Desktop-Pool** aus.
- 3 Wählen Sie auf der Seite „Benutzerzuweisung“ dedizierte oder variable Benutzerzuweisungen für die Maschinen im Desktop-Pool aus und klicken Sie dann auf **Weiter**.
- 4 Wählen Sie auf der Seite „Computerquelle“ die Option **Virtuelle vCenter-Maschinen** aus und klicken Sie auf **Weiter**.
- 5 Wählen Sie auf der Seite „vCenter Server“ den entsprechenden vCenter Server aus und klicken Sie auf **Weiter**.
- 6 Geben Sie die **Desktop-Pool-ID** ein.
- 7 Legen Sie auf der Seite „Desktop-Pool-Einstellungen“ die folgenden Optionen fest.

Option	Beschreibung
Standardanzeigeprotokoll	VMware Blast
Benutzern die Wahl des Protokolls erlauben	Nein
3D-Renderer	Verwalten Sie vSphere Client für 2D-, vSGA- oder vDGA-Desktop und NVIDIA GRID vGPU für vGPU-Desktop.

Hinweis Die Pool-Einstellungen sind obligatorisch. Andernfalls treten bei der Verbindung zum Desktop möglicherweise Fehler auf und es wird ein Protokollfehler oder ein schwarzer Bildschirm angezeigt.

- 8 Wählen Sie auf der Seite „Virtuelle vCenter-Maschinen hinzufügen“ die virtuelle Linux-Maschine aus. Klicken Sie auf **Hinzufügen** und dann auf **Weiter**.

Hinweis Virtuelle Windows- und Linux-Maschinen dürfen nicht im selben Desktop-Pool erstellt werden.

- 9 Folgen Sie den Eingabeaufforderungen, um die Prozedur abzuschließen.
Erteilen Sie Benutzern Berechtigungen für die Maschinen im Desktop-Pool. In View Administrator wählen Sie den Desktop-Pool aus, wählen **Berechtigungen > Berechtigung hinzufügen** und fügen Benutzer oder Gruppen hinzu.

Die virtuellen Linux-Maschinen stehen dann für die Verwendung als Remote-Desktops in einer Horizon 7-Bereitstellung zur Verfügung.

Verwalten eines Desktop-Pools für Linux

Beim Erstellen eines manuellen Desktop-Pools und Hinzufügen von Linux-Maschinen zum Pool können Sie die manuellen Desktop-Pools verwalten, indem Sie die Einstellungen konfigurieren. Sie müssen nur

Linux-Gastbetriebssysteme zum manuellen Desktop-Pool hinzufügen. Wenn der Pool sowohl virtuelle Windows- als auch virtuelle Linux-Gastbetriebssysteme enthält, wird der Pool als Windows-Pool behandelt und kann nicht mit Linux-Desktops verbunden werden.

Unterstützung für das Verwalten von Vorgängen

- Deaktivieren oder Aktivieren eines Desktop-Pools
- Klonen eines automatisierten Desktop-Pools
- Löschen eines Desktop-Pools

Sie können virtuelle Maschinen aus View Manager entfernen oder vom Datenträger löschen

Unterstützung für Remote-Einstellungen

Tabelle 7-1. Remote-Einstellungen

Remote-Einstellung	Optionen
Betriebsrichtlinie für Remote-Computer	<ul style="list-style-type: none"> ■ Keine Betriebsaktion vornehmen ■ Computer müssen immer eingeschaltet sein ■ Anhalten ■ Ausschalten
Automatic logoff after disconnect (Nach Verbindungstrennung automatisch abmelden)	<ul style="list-style-type: none"> ■ Sofort ■ Nie ■ Nach n Minuten
Benutzern das Zurücksetzen/den Neustart ihrer Computer gestatten	<ul style="list-style-type: none"> ■ Ja ■ Nein
Benutzer darf separate Sitzungen von unterschiedlichen Client-Geräten aus starten	<ul style="list-style-type: none"> ■ Ja ■ Nein
Löschen der Maschine nach der Abmeldung für automatisierten Desktop-Pool mit Full-Clone und Floating	<ul style="list-style-type: none"> ■ Ja ■ Nein

Unterstützung für View Administrator-Vorgänge

- Sitzung trennen
- Logoff Session (Von Sitzung abmelden)
- Desktop neu starten/zurücksetzen
- Nachricht senden

Für einen dedizierten Desktop-Pool können Sie die Benutzerzuweisung für jede virtuelle Maschine hinzufügen oder entfernen. Bei einer großen Anzahl an Vorgängen müssen Sie View PowerCLI-Cmdlets verwenden.

- Update-UserOwnership

■ Remove-UserOwnership

Hinweis Lassen Sie die Einstellungen für das **Remote-Anzeigeprotokoll** unverändert. Diese Einstellung muss immer dieselbe sein wie die Desktop-Poolerstellung.

Einstellung	Option
Standardanzeigeprotokoll	VMware Blast
Benutzern die Wahl des Protokolls erlauben	Nein
3D-Renderer	<ul style="list-style-type: none"> ■ Verwaltung mithilfe von vSphere Client für 2D oder vSGA oder vDGA ■ NVIDIA GRID vGPU

Weitere Informationen finden Sie in der Anleitung *Administration von View*.

Erstellen eines automatisierten Full-Clone-Desktop-Pools für Linux

Sie können einen automatisierten Full-Clone-Desktop-Pool für Linux erstellen. Nachdem Sie den automatisierten Full-Clone-Desktop-Pool erstellt haben, können Sie die virtuellen Linux-Maschinen als Remote-Desktops in einer Horizon 7-Bereitstellung verwenden.

Voraussetzungen

- Stellen Sie sicher, dass Horizon View Agent auf dem Linux-Gastbetriebssystem installiert ist. Siehe [Installieren von Horizon Agent auf einer virtuellen Linux-Maschine](#).
- Wenn Sie die Winbind-Lösung nutzen, um die virtuelle Linux-Maschine zu Active Directory beitreten zu lassen, müssen Sie die Konfiguration der Winbind-Lösung in der Vorlage der virtuellen Maschine durchführen.
- Wenn Sie die Winbind-Lösung verwenden, müssen Sie auf der virtuellen Maschine den Befehl für den Beitritt zur Domäne ausführen. Fügen Sie den Befehl in ein Shell-Skript ein und geben Sie den Skriptpfad zur Horizon Agent-Option RunOnceScript in `/etc/vmware/viewagent-custom.conf` an. Weitere Informationen finden Sie unter [Einstellen der Optionen in Konfigurationsdateien auf einem Linux-Desktop](#).
- Prüfen Sie, ob vCenter Server zum Horizon-Verbindungsserver hinzugefügt wurde.

Verfahren

- 1 Erstellen Sie eine Anpassungsspezifikation für das Gastbetriebssystem.

Erläuterungen finden Sie unter „Erstellen einer Anpassungsspezifikation für Linux im vSphere Web Client“ im Dokument *Verwaltung virtueller vSphere-Maschinen*. Für die Erstellung einer Spezifikation müssen die folgenden Einstellungen korrekt festgelegt werden.

Einstellung	Wert
Betriebssystem der virtuellen Zielmaschine	Linux
Computernamen	Verwenden Sie den Namen der virtuellen Maschine.

Einstellung	Wert
Domäne	Geben Sie die Domäne der View-Umgebung an.
Netzwerkeinstellungen	Verwenden Sie die Standardnetzwerkeinstellungen.
Primäres DNS	Geben Sie eine gültige Adresse an.

Hinweis Weitere Informationen zur Unterstützungsmatrix für die Gastbetriebssystemanpassung finden Sie unter <http://partnerweb.vmware.com/programs/guestOS/guest-os-customization-matrix.pdf>.

- Wählen Sie in Horizon Administrator **Katalog > Desktop-Pools > Hinzufügen** aus.
- Wählen Sie **Automatisierter Desktop-Pool** aus und klicken Sie auf **Weiter**.
- Wählen Sie entweder dedizierte oder flexible Benutzerzuweisungen für die Maschinen im Desktop-Pool aus und klicken Sie auf **Weiter**.
- Wählen Sie auf der Seite „vCenter Server“ **Vollständig virtuelle Maschinen** und den entsprechenden vCenter Server aus und klicken Sie auf **Weiter**.
- Geben Sie auf der Seite für die Desktop-Poolidentifikation die Desktop-Pool-ID ein und klicken Sie auf **Weiter**.
- Legen Sie auf der Seite „Desktop-Pooleinstellungen“ die folgenden Optionen fest und klicken Sie auf **Weiter**.

Option	Beschreibung
Standardanzeigeprotokoll	VMware Blast
Benutzern die Wahl des Protokolls erlauben	Nein
3D-Renderer	Verwalten Sie vSphere Client für 2D-, vSGA- oder vDGA-Desktop und NVIDIA GRID vGPU für vGPU-Desktop.

Hinweis Die Pool-Einstellungen sind obligatorisch. Andernfalls treten bei der Verbindung zum Desktop möglicherweise Fehler auf und es wird ein Protokollfehler oder ein schwarzer Bildschirm angezeigt.

- Legen Sie auf der Seite für die Bereitstellungseinstellungen die Optionen für die **Benennung der virtuellen Maschine** fest und klicken Sie auf **Weiter**.

Option	Beschreibung
Namen manuell angeben	Geben Sie Namen manuell ein.
Benennungsmuster verwenden	Geben Sie z. B. LinuxVM-{n} an. Sie müssen außerdem die folgenden Optionen für die Größe des Desktop-Pools angeben: <ul style="list-style-type: none"> ■ Maximale Anzahl an Maschinen ■ Anzahl an eingeschalteten Reservemaschinen

- 9 Wählen Sie auf der Seite für die Speicheroptimierung eine Richtlinie für die Speicherverwaltung aus und klicken Sie auf **Weiter**.
- 10 Sie müssen auf der Seite „vCenter-Einstellungen“ auf „Durchsuchen“ klicken und nacheinander alle vCenter Server-Einstellungen auswählen. Klicken Sie anschließend auf **Weiter**.
Es ist nicht möglich, eine vCenter Server-Einstellung zu überspringen:
 - a Vorlage
 - b Speicherort des VM-Ordners
 - c Host or cluster (Host oder Cluster)
 - d Resource pool (Ressourcen-Pool)
 - e Datenspeicher
- 11 Wählen Sie auf der Seite „Erweiterte Speicheroptionen“ die entsprechenden Speicheroptionen aus und klicken Sie auf **Weiter**.
- 12 Wählen Sie auf der Seite für die Gastbetriebssystemanpassung Ihre Gastanpassung für Linux aus und klicken Sie auf **Weiter**.
- 13 Prüfen Sie auf der Abschlusseite die Details und wählen Sie **Benutzern nach Abschluss dieses Assistenten Berechtigungen verleihen** aus.
- 14 Klicken Sie auf **Fertig stellen**.
- 15 Um Benutzern Berechtigungen für die im Desktop-Pool enthaltenen Maschinen zu erteilen, wählen Sie den Desktop-Pool aus und klicken Sie auf **Berechtigungen > Berechtigung hinzufügen**. Fügen Sie die Benutzer und Gruppen hinzu.
- 16 Warten Sie, bis alle virtuellen Linux-Maschinen im Desktop-Pool verfügbar sind.

Broker-PowerCLI-Befehle

Die View PowerCLI-Cmdlets, die zum Ausführen verschiedener Verwaltungsaufgaben auf Verbindungsserver- und dem Windows-Desktop dienen, funktionieren auch für Linux-Desktop.

Erstellen eines manuellen Desktop-Pools

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

Für Linux-Desktop sind die folgenden Optionen und Werte obligatorisch.

- DefaultProtocol Blast
- AllowProtocolOverride \$false
- threadRender usevc|vgpu. Verwenden Sie für vGPU-Desktop „-threadRender vgpu“ und für 2D/vSGA/vDGA-Desktop „-threadRender usevc“.

Beispiele

- Erstellen Sie einen dynamischen Linux-Desktop-Pool mit dem Namen „LinuxDesktop“ mit einer virtuellen Maschine namens „LinuxVM-01“.

```
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
```

- Erstellen Sie einen dedizierten Linux-vGPU-Desktop-Pool mit dem Namen „LinuxDesktop“ mit allen VMs, die mit einem VM-Namen wie „LinuxVM-“ beginnen.

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

- Erstellen Sie einen dynamischen Linux-Desktop-Pool mit dem Namen „LinuxDesktop“ mit dem ersten RHEL 6 x64 VM.

```
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
```

Erstellen eines automatisierten Full-Clone-Desktop-Pools

```
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]
```

Für Linux-Desktop sind die folgenden Optionen und Werte obligatorisch.

- DefaultProtocol Blast
- AllowProtocolOverride \$false
- threadRender usevc|vgpu. Verwenden Sie für vGPU-Desktop „-threadRender vgpu“ und für 2D/vSGA-Desktop „-threadRender usevc“.

Beispiel

```
Add-AutomaticPool -DefaultProtocol Blast -AllowProtocolOverride $false -threadRender 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
```

Hinzufügen oder Entfernen von Desktop-Pool-Berechtigungen

- Gewähren Sie der Domänenbenutzergruppe der Domäne „mydomain.org“ die Berechtigung für „LinuxDesktop“.

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

- Entfernen Sie die Berechtigung der Domänenbenutzergruppe der Domäne „mydomain.org“ von „LinuxDesktop“.

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

Zuweisen eines Benutzers zur VM oder Entfernen eines Benutzers aus der VM in einem dedizierten Desktop-Pool

- Weisen Sie den Benutzer „myuser“ zur VM „LinuxVM-01“ zu, die sich in einem dedizierten Desktop-Pool befindet.

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

- Entfernen Sie den Benutzer „myuser“ aus der VM „LinuxVM-01“, die sich in einem dedizierten Desktop-Pool befindet.

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

Abmelden der Desktop-Verbindung

- Melden Sie sich von der Desktop-Sitzung von „myuser“ ab.

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

Weitere Informationen über das Broker-PowerCLI-Cmdlet finden Sie im Thema über das Verwenden von View PowerCLI in *Integration von View*.

Massenbereitstellung von Horizon 7 für manuelle Desktop-Pools

8

Mit View Administrator können Sie automatisch einen Pool von Windows-Desktop-Maschinen erstellen, nicht jedoch von Linux-Desktop-Maschinen. Sie können jedoch Skripts entwickeln, die die Bereitstellung eines Pools von Linux-Desktop-Maschinen automatisieren.

Die zur Verfügung gestellten Beispielskripts dienen nur der Veranschaulichung. VMware übernimmt keine Verantwortung für Probleme, die im Zusammenhang mit der praktischen Anwendung der Beispielskripts auftreten.

Dieses Kapitel enthält die folgenden Themen:

- [Überblick über die Massenbereitstellung von Linux-Desktops](#)
- [Überblick über die Massenaktualisierung von Linux-Desktops](#)
- [Erstellen einer Vorlage für virtuelle Maschinen zum Klonen von Linux-Desktop-Maschinen](#)
- [Eingabedatei für die PowerCLI-Beispielskripts zur Bereitstellung von Linux-Desktops](#)
- [Beispielskript zum Klonen von virtuellen Linux-Maschinen](#)
- [Beispielskript zum Hinzufügen geklonter virtueller Maschinen zu einer AD-Domäne](#)
- [Beispielskript zum Hinzufügen geklonter virtueller Maschinen zu einer AD-Domäne mithilfe von SSH](#)
- [Beispielskript zum Hochladen von Konfigurationsdateien zu virtuellen Linux-Maschinen](#)
- [Beispielskript zum Hochladen von Konfigurationsdateien zu virtuellen Linux-Maschinen mithilfe von SSH](#)
- [Beispielskript zur Durchführung eines Upgrades von Horizon Agent auf Linux-Desktop-Maschinen](#)
- [Beispielskript zur Durchführung eines Upgrades von Horizon Agent auf virtuellen Linux-Maschinen mithilfe von SSH](#)
- [Beispielskript zum Ausführen von Vorgängen auf virtuellen Linux-Maschinen](#)

Überblick über die Massenbereitstellung von Linux-Desktops

Das Bereitstellen von manuellen Desktops für Linux umfasst verschiedene Schritte. Wenn Sie mehr als nur eine Handvoll Desktops bereitstellen möchten, können Sie einige dieser Schritte mithilfe von PowerCLI-Skripts automatisieren.

Für einige Vorgänge können Sie auswählen, ob entweder PowerCLI oder SSH die Befehle auf der Linux-Maschine ausführen soll. Die folgende Tabelle beschreibt die Unterschiede zwischen den beiden Vorgehensweisen.

PowerCLI	SSH
Es ist nicht erforderlich, zusätzliche Tools zu installieren.	<ul style="list-style-type: none"> Für Ubuntu müssen Sie den SSH-Server mit dem Befehl <code>sudo apt-get install openssh-server</code> installieren. Für RHEL und CentOS wird <code>openssh-server</code> standardmäßig installiert. Sie müssen jedoch sicherstellen, dass die Firewall-Einstellungen SSH zulassen. Die SSH-Client-Anwendungen <code>pscp.exe</code> und <code>plink.exe</code> müssen heruntergeladen und im selben Ordner wie die PowerCLI-Skripts abgelegt werden.
Das Hochladen von Dateien und die Befehlsausführung gehen langsamer vonstatten.	Das Hochladen von Dateien und die Befehlsausführung gehen schneller vonstatten.
Die Administrator-Anmeldedaten des ESXi-Hosts müssen angegeben werden.	Die Administrator-Anmeldedaten des ESXi-Hosts müssen nicht angegeben werden.
Es können keine Sonderzeichen im Administrator-Kennwort (bei Ausführung des Skripts zur Installation von Horizon Agent) oder im Kennwort des AD-Benutzers (bei Ausführung des Skripts zum Hinzufügen der Domäne) verarbeitet werden.	Es können Sonderzeichen im Administrator-Kennwort (bei Ausführung des Skripts zur Installation von Horizon Agent) oder im Kennwort des AD-Benutzers (bei Ausführung des Skripts zum Hinzufügen der Domäne) verarbeitet werden.

Hinweis Sowohl PowerCLI-basierte als auch SSH-basierte Skripts können Sonderzeichen in den Kennwörtern für den vCenter Server-Administrator und den Linux-Administrator verarbeiten. PowerCLI-basierte Skripts können Sonderzeichen auch im Kennwort des ESXi-Host-Administrators verarbeiten. In all diesen Fällen ist kein Escape-Zeichen erforderlich.

Weitere Informationen zu vSphere PowerCLI finden Sie unter <https://www.vmware.com/support/developer/PowerCLI>.

Die Massenbereitstellung eines Pools von Linux-Desktops umfasst die folgenden Schritte:

- 1 Erstellen Sie eine Vorlage für die virtuelle Maschine und installieren Sie Horizon Agent auf der virtuellen Maschine.
Siehe [Erstellen einer Vorlage für virtuelle Maschinen zum Klonen von Linux-Desktop-Maschinen](#).
- 2 Erstellen Sie eine Anpassungsspezifikation für das Gastbetriebssystem.

Erläuterungen finden Sie unter „Erstellen einer Anpassungsspezifikation für Linux im vSphere Web Client“ im Dokument *Verwaltung virtueller vSphere-Maschinen*. Für die Erstellung einer Spezifikation müssen die folgenden Einstellungen korrekt festgelegt werden.

Einstellung	Wert
Betriebssystem der virtuellen Zielmaschine	Linux
Computername	Verwenden Sie den Namen der virtuellen Maschine.
Domäne	Geben Sie die Domäne der View-Umgebung an.
Netzwerkeinstellungen	Verwenden Sie die Standardnetzwerkeinstellungen.
Primäres DNS	Geben Sie eine gültige Adresse an.

Hinweis Weitere Informationen zur Unterstützungsmatrix für die Gastbetriebssystemanpassung finden Sie unter <http://partnerweb.vmware.com/programs/guestOS/guest-os-customization-matrix.pdf>.

- 3 Klonen Sie virtuelle Maschinen.

Siehe [Beispielskript zum Klonen von virtuellen Linux-Maschinen](#).

- 4 Lassen Sie die geklonten VMs zur Active Directory-Domäne beitreten, sofern Sie die winbind-Lösung verwenden. Sie können den Befehl für den Beitritt zur Domäne mit den folgenden Beispielskripts ausführen oder die Option `RunOnceScript` in `/etc/vmware/viewagent-custom.conf` verwenden, die in der Vorlage für die virtuelle Maschine konfiguriert ist.

Siehe [Beispielskript zum Hinzufügen geklonter virtueller Maschinen zu einer AD-Domäne](#) oder [Beispielskript zum Hinzufügen geklonter virtueller Maschinen zu einer AD-Domäne mithilfe von SSH](#).

- 5 Aktualisieren Sie die Konfigurationsoptionen in virtuellen Maschinen.

Siehe [Beispielskript zum Hochladen von Konfigurationsdateien zu virtuellen Linux-Maschinen](#) oder [Beispielskript zum Hochladen von Konfigurationsdateien zu virtuellen Linux-Maschinen mithilfe von SSH](#).

- 6 Erstellen Sie einen Desktop-Pool.

Siehe [Erstellen eines manuellen Desktop-Pools für Linux](#).

Unter [Beispielskript zum Ausführen von Vorgängen auf virtuellen Linux-Maschinen](#) erhalten Sie ein Beispielskript, das Vorgänge wie das Einschalten, Herunterfahren, Neustarten oder Löschen virtueller Maschinen durchführt. Mit diesem Skript lassen sich virtuelle Maschinen aus vCenter Server löschen.

Überblick über die Massenaktualisierung von Linux-Desktops

Die Massenaktualisierung von manuellen Desktops für Linux umfasst verschiedene Schritte. Sie können einige der Schritte mithilfe von PowerCLI-Skripts automatisieren.

Massenaktualisierung eines nicht verwalteten Desktops

Um die nicht verwaltete virtuelle Maschine per Massenaktualisierung auf eine verwaltete oder nicht verwaltete virtuelle Maschine zu aktualisieren, müssen Sie das Beispielaktualisierungsskript verwenden, um den neuen Horizon Agent auf die vorhandenen virtuellen Maschinen hochzuladen, und den Aktualisierungsbefehl ausführen.

- Wenn Sie die nicht verwaltete virtuelle Maschine beibehalten, kann Ihr vorhandener Desktop-Pool wiederverwendet werden.
- Wenn Sie eine Aktualisierung von einer nicht verwalteten virtuellen Maschine auf eine verwaltete virtuelle Maschine vornehmen, müssen Sie den vorhandenen Desktop-Pool löschen und einen neuen Desktop-Pool erstellen. Weitere Informationen finden Sie unter [Durchführen eines Upgrades von Horizon Agent auf einer virtuellen Linux-Maschine](#).

Massenaktualisierung eines verwalteten Desktops

Wählen Sie für die Massenaktualisierung der verwalteten virtuellen Maschine eine der folgenden Methoden aus.

Methode	Beschreibung
Installieren oder aktualisieren Sie in der Vorlagen-VM den neuen Horizon Agent und erstellen Sie einen Snapshot.	<ul style="list-style-type: none"> ■ Benutzerdaten und -profil gehen verloren, da die vorhandenen virtuellen Maschinen gelöscht wurden, außer die Benutzerdaten und das Benutzerprofil befinden sich auf dem Freigabeserver, zum Beispiel auf einem NFS-Server. ■ Nach der VM-Ersetzung fehlt möglicherweise der Status der virtuellen Maschine auf View Administrator. Sie müssen den Broker-Dienst neu starten, um dies zu beheben. ■ Bei Verwendung des verknüpften Klangs vermeidet diese Methode doppelte Daten auf jeder virtuellen Maschine.
Verwenden Sie das Beispielskript der Aktualisierung, um den neuen Horizon Agent auf die vorhandenen virtuellen Maschinen hochzuladen, und führen Sie den Aktualisierungsbefehl aus.	<ul style="list-style-type: none"> ■ Benutzerdaten und -profil werden beibehalten. ■ Bei Verwendung des verknüpften Klangs führt diese Methode zu doppelten Daten auf jeder virtuellen Maschine.

Erstellen einer Vorlage für virtuelle Maschinen zum Klonen von Linux-Desktop-Maschinen

Bevor Sie eine virtuelle Maschine klonen können, müssen Sie als Grundlage für die Klone eine Vorlage für virtuelle Maschinen erstellen.

Voraussetzungen

- Stellen Sie sicher, dass Ihre Bereitstellung den Anforderungen für die Unterstützung von Linux-Desktops entspricht. Siehe [Systemanforderungen für Horizon 7 for Linux](#).

- Machen Sie sich mit den Schritten für das Erstellen virtueller Maschinen in vCenter Server und mit der Installation von Gastbetriebssystemen vertraut. Unter „Erstellen und Vorbereiten virtueller Maschinen“ im *Einrichten von virtuellen Desktops in Horizon 7*-Dokument finden Sie dazu Erläuterungen.
- Informieren Sie sich über die empfohlenen Videospeicherwerte (vRAM) für die Monitore, die Sie mit der virtuellen Maschine verwenden möchten. Siehe [Systemanforderungen für Horizon 7 for Linux](#).
- Machen Sie sich mit den Schritten zur AD-Integration vertraut. Siehe [Kapitel 3 Einrichten der Active Directory-Integration für Linux-Desktops](#).
- Machen Sie sich mit den Schritten für das Installieren des Horizon Agent auf Linux vertraut. Siehe hierzu das Kapitel „Installieren von Horizon Agent und Verwalten von Linux-Desktops“.
- Machen Sie sich, falls erforderlich, mit den Schritten zur Konfiguration von Optionen mithilfe der Konfigurationsdateien von View vertraut. Siehe [Kapitel 6 Konfigurationsoptionen für Linux-Desktops](#).
- Wenn Sie planen, Grafiken einzurichten, machen Sie sich mit den entsprechenden Schritten vertraut. Siehe [Kapitel 4 Einrichten von Grafiken für Linux-Desktops](#).

Verfahren

- 1 Erstellen Sie im vSphere Web Client oder vSphere Client eine neue virtuelle Maschine.
- 2 Konfigurieren Sie die benutzerdefinierten Konfigurationsoptionen.
 - a Klicken Sie mit der rechten Maustaste auf die virtuelle Maschine und klicken Sie auf **Einstellungen bearbeiten**.
 - b Geben Sie die Anzahl der vCPUs und die Größe des vMemory-Arbeitsspeichers an.

Empfohlene Werte finden Sie im Installationshandbuch Ihrer Linux-Distribution.
Ubuntu 12.04 empfiehlt beispielsweise die Konfiguration von 2.048 MB für den vMemory-Arbeitsspeicher und von zwei vCPUs.
 - c Wählen Sie **Grafikkarte** aus und geben Sie die Anzahl der Anzeigegeräte sowie den gesamten Videospeicher (vRAM) ein.

Legen Sie im vSphere Web Client die vRAM-Größe für virtuelle Maschinen mit 2D oder vSGA fest. Diese verwenden den VMware-Treiber. Die vRAM-Größe hat keinen Einfluss auf vDGA- oder NVIDIA GRID vGPU-Maschinen. Diese verwenden NVIDIA-Treiber.

Empfohlene Werte finden Sie unter *Systemanforderungen für Horizon 7 for Linux*. Verwenden Sie nicht die Videospeicherberechnung.
- 3 Schalten Sie die virtuelle Maschine ein und installieren Sie die Linux-Distribution.
- 4 Erstellen Sie einen Benutzer mit Root-Rechten, z. B. „ViewBenutzer“. Mit diesem Benutzer wird nur Horizon Agent installiert und deinstalliert.

- 5 Bearbeiten Sie `/etc/sudoers` und fügen Sie die Zeile `ViewUser ALL=(ALL) NOPASSWD:ALL` hinzu.

Enthält `/etc/sudoers` diese Zeile ist kein Kennwort für die Ausführung von Sudo als „ViewBenutzer“ erforderlich. Wenn Sie für die Installation von Horizon Agent das in diesem Kapitel zur Verfügung gestellte Beispielskript verwenden, müssen Sie „ViewBenutzer“ als Eingabe angeben.

- 6 Bei Linux-Distributionen wie RHEL, CentOS oder NeoKylin bearbeiten Sie `/etc/sudoers` und kommentieren die folgenden Zeilen aus:

```
Defaults requiretty
Defaults !visiblepw
```

- 7 Wenn es sich bei der Linux-Distribution nicht um RHEL 7, CentOS 7, SLED 12 oder SLES 12 handelt, installieren Sie VMware Tools.

Für RHEL 7, CentOS 7, SLED 12 und SLES 12 wird Open VM Tools standardmäßig installiert.

- 8 Für die Linux-Distribution RHEL 7, CentOS 7 oder SLES 12 installieren Sie das Plug-In `deployPkg`.

Die Anweisungen dazu finden Sie unter <http://kb.vmware.com/kb/2075048>.

- 9 Für RHEL und CentOS aktivieren Sie die Netzwerkverbindungseinstellung **Automatisch verbinden**.

- 10 Führen Sie die Aufgaben zur AD-Integration aus.

- 11 Führen Sie die Schritte zum Einrichten von Grafiken durch.

- 12 Installieren Sie den Horizon Agent.

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

- 13 Führen Sie zusätzliche Konfigurationen mithilfe der Konfigurationsdateien von View durch.

- 14 Fahren Sie die virtuelle Maschine herunter und erstellen Sie einen Snapshot.

Eingabedatei für die PowerCLI-Beispielskripts zur Bereitstellung von Linux-Desktops

Die PowerCLI-Beispielskripts zur Bereitstellung von Linux-Desktops lesen eine Eingabedatei mit Informationen über die Desktop-Maschinen.

Die Eingabedatei ist vom Typ `csv` und enthält die folgenden Informationen:

- Name der virtuellen Desktop-Maschine
- Übergeordneter Name der virtuellen Maschine
- Anpassungsspezifikation des Gastbetriebssystems
- Datenspeicher mit den geklonten Desktop-Maschinen
- Host-ESXi-Server der Desktop-Maschine
- Snapshot der übergeordneten virtuellen Maschine für das Klonen

- Attribut für das Löschen der virtuellen Desktop-Maschine, wenn vorhanden

Das folgende Beispiel zeigt den möglichen Inhalt einer Eingabedatei.

```
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
```

Das Beispielskript geht davon aus, dass der Name dieser Eingabedatei `CloneVMs.csv` lautet und dass die Datei im selben Ordner wie die Skripts enthalten ist.

Beispielskript zum Klonen von virtuellen Linux-Maschinen

Sie können das im Folgenden aufgeführte Beispielskript für das Klonen einer beliebigen Anzahl virtueller Maschinen (VMs) anpassen und anwenden.

Für das Kopieren und Einfügen des Skriptinhalts ohne Seitenumbrüche verwenden Sie die HTML-Version dieses Themas, die auf der Dokumentationsseite von Horizon 7 unter https://www.vmware.com/support/pubs/view_pubs.html verfügbar ist.

Skripteingabe

Dieses Skript liest eine Eingabedatei, die im Kapitel [Eingabedatei für die PowerCLI-Beispielskripts zur Bereitstellung von Linux-Desktops](#) beschrieben ist. Es gibt verschiedene Eingabeaufforderungen für die folgenden Informationen aus:

- IP-Adresse von vCenter Server
- Anmeldenamen des Administrators für vCenter Server
- Kennwort des Administrators für vCenter Server
- Klontyp (verknüpft oder vollständig)
- Deaktivierung der vSphere-VM-Konsole

Skriptinhalt

```
<#
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
    $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBSTR($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

```

Skriptausführung

Die folgenden Meldungen resultieren aus einer Ausführung des Skripts:

```
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
```

Der für den Klonvorgang notwendige Zeitraum ist abhängig von der Anzahl der Desktop-Maschinen und kann von mehreren Minuten bis zu Stunden reichen. Um sicherzustellen, dass der Vorgang abgeschlossen ist, vergewissern Sie sich im vSphere Client, dass die letzte virtuelle Desktop-Maschine eingeschaltet ist und über ihren eigenen eindeutigen Hostnamen verfügt und dass VMware Tools ausgeführt wird.

Beispielskript zum Hinzufügen geklonter virtueller Maschinen zu einer AD-Domäne

Sie können das folgende Beispielskript anpassen und verwenden, um geklonte virtuelle Maschinen (VMs) zu einer Active Directory-Domäne (AD-Domäne) hinzuzufügen.

Sie müssen dieses Skript ausführen, wenn Sie die Winbind-Lösung für die AD-Integration verwenden, weil dabei der Schritt des Hinzufügens zur Domäne für geklonte VMs fehlschlägt. Dieses Skript führt einen Befehl zum Hinzufügen zur Domäne auf jeder VM aus. Sie müssen dieses Skript nicht ausführen, wenn Sie die OpenLDAP-Lösung verwenden.

Für das Kopieren und Einfügen des Skriptinhalts ohne Seitenumbrüche verwenden Sie die HTML-Version dieses Themas, die auf der Dokumentationsseite von Horizon 7 unter https://www.vmware.com/support/pubs/view_pubs.html verfügbar ist.

Skripteingabe

Dieses Skript liest eine Eingabedatei, die im Kapitel [Eingabedatei für die PowerCLI-Beispielskripts zur Bereitstellung von Linux-Desktops](#) beschrieben ist. Es gibt verschiedene Eingabeaufforderungen für die folgenden Informationen aus:

- IP-Adresse von vCenter Server
- Anmeldename des Administrators für vCenter Server
- Kennwort des Administrators für vCenter Server
- Anmeldename des Administrators für den ESXi-Host
- Kennwort des Administrators für den ESXi-Host
- Benutzeranmeldename für die Linux-VM
- Benutzerkennwort für die Linux-VM

- Anmeldenname eines AD-Benutzers, der autorisiert ist, Maschinen zur Domäne hinzuzufügen
- Kennwort des autorisierten AD-Benutzers

Skriptinhalt

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

.DESCRIPTION
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

```

Skriptausführung

Die folgenden Meldungen resultieren aus einer Ausführung des Skripts:

```

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: *****
```

Beispielskript zum Hinzufügen geklonter virtueller Maschinen zu einer AD-Domäne mithilfe von SSH

Sie können das folgende Beispielskript anpassen und verwenden, um geklonte virtuelle Maschinen (VMs) zu einer Active Directory-Domäne (AD-Domäne) hinzuzufügen. Dieses Skript verwendet SSH zur Ausführung von Befehlen auf den Linux-VMs.

Sie müssen dieses Skript ausführen, wenn Sie die Winbind-Lösung für die AD-Integration verwenden, weil dabei der Schritt des Hinzufügens zur Domäne für geklonte VMs fehlschlägt. Dieses Skript führt einen Befehl zum Hinzufügen zur Domäne auf jeder VM aus. Sie müssen dieses Skript nicht ausführen, wenn Sie die OpenLDAP-Lösung verwenden.

Für das Kopieren und Einfügen des Skriptinhalts ohne Seitenumbrüche verwenden Sie die HTML-Version dieses Themas, die auf der Dokumentationsseite von Horizon 7 unter https://www.vmware.com/support/pubs/view_pubs.html verfügbar ist.

Skripteingabe

Dieses Skript liest eine Eingabedatei, die im Kapitel [Eingabedatei für die PowerCLI-Beispielskripts zur Bereitstellung von Linux-Desktops](#) beschrieben ist. Es gibt verschiedene Eingabeaufforderungen für die folgenden Informationen aus:

- IP-Adresse von vCenter Server
- Anmeldename des Administrators für vCenter Server
- Kennwort des Administrators für vCenter Server
- Benutzeranmeldename für die Linux-VM
- Benutzerkennwort für die Linux-VM
- Anmeldename eines AD-Benutzers, der autorisiert ist, Maschinen zur Domäne hinzuzufügen
- Kennwort des autorisierten AD-Benutzers

Skriptinhalt

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

.DESCRPTION
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
""
""`nPlease 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
```

Skriptausführung

Die folgenden Meldungen resultieren aus einer Ausführung des Skripts:

```
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: *****
```

Beispielskript zum Hochladen von Konfigurationsdateien zu virtuellen Linux-Maschinen

Sie können das folgende Beispielskript anpassen und verwenden, um die Konfigurationsdateien config und viewagent-custom.conf zu mehreren virtuellen Linux-Maschinen (VMs) hochzuladen.

Für das Kopieren und Einfügen des Skriptinhalts ohne Seitenumbrüche verwenden Sie die HTML-Version dieses Themas, die auf der Dokumentationsseite von Horizon 7 unter https://www.vmware.com/support/pubs/view_pubs.html verfügbar ist.

Skripteingabe

Dieses Skript liest eine Eingabedatei, die im Kapitel [Eingabedatei für die PowerCLI-Beispielskripts zur Bereitstellung von Linux-Desktops](#) beschrieben ist. Es gibt verschiedene Eingabeaufforderungen für die folgenden Informationen aus:

- IP-Adresse von vCenter Server
- Anmeldename des Administrators für vCenter Server
- Kennwort des Administrators für vCenter Server
- Anmeldename des Administrators für den ESXi-Host
- Kennwort des Administrators für den ESXi-Host
- Benutzeranmeldename für die Linux-VM
- Benutzerkennwort für die Linux-VM

Skriptinhalt

```
<#
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

```

Skriptausführung

Die folgenden Meldungen resultieren aus einer Ausführung des Skripts:

```

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: *****

```

Beispielskript zum Hochladen von Konfigurationsdateien zu virtuellen Linux-Maschinen mithilfe von SSH

Sie können das folgende Beispielskript anpassen und verwenden, um die Konfigurationsdateien `config` und `viewagent-custom.conf` zu mehreren virtuellen Linux-Maschinen (VMs) hochzuladen. Dieses Skript verwendet SSH zur Ausführung von Befehlen auf den Linux-VMs.

Für das Kopieren und Einfügen des Skriptinhalts ohne Seitenumbrüche verwenden Sie die HTML-Version dieses Themas, die auf der Dokumentationsseite von Horizon 7 unter https://www.vmware.com/support/pubs/view_pubs.html verfügbar ist.

Skripteingabe

Dieses Skript liest eine Eingabedatei, die im Kapitel [Eingabedatei für die PowerCLI-Beispielskripts zur Bereitstellung von Linux-Desktops](#) beschrieben ist. Es gibt verschiedene Eingabeaufforderungen für die folgenden Informationen aus:

- IP-Adresse von vCenter Server
- Anmeldename des Administrators für vCenter Server
- Kennwort des Administrators für vCenter Server
- Benutzeranmeldename für die Linux-VM
- Benutzerkennwort für die Linux-VM

Skriptinhalt

```
<#
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

```

Skriptausführung

Die folgenden Meldungen resultieren aus einer Ausführung des Skripts:

```

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: *****

```

Beispielskript zur Durchführung eines Upgrades von Horizon Agent auf Linux-Desktop-Maschinen

Sie können das nachfolgend aufgeführte Beispielskript für ein Upgrade von Horizon Agent auf mehreren virtuellen Linux-Maschinen (VMs) anpassen und anwenden.

Dieses Skript lädt das TAR-Archiv des Installationsprogramms auf jede VM hoch, bevor Horizon Agent installiert wird. Die Upload-Aufgabe kann sehr viel Zeit in Anspruch nehmen, vor allem wenn eine große Anzahl von VMs beteiligt und die Netzwerkgeschwindigkeit langsam ist. Um Zeit zu sparen, können Sie das Skript ausführen, das SSH verwendet, oder das TAR-Archiv des Installationsprogramms an einem freigegebenen Speicherort ablegen, der für jede VM verfügbar ist, wodurch das Hochladen der Datei nicht erforderlich ist.

Für das Kopieren und Einfügen des Skriptinhalts ohne Seitenumbrüche verwenden Sie die HTML-Version dieses Themas, die auf der Dokumentationsseite von Horizon 7 unter https://www.vmware.com/support/pubs/view_pubs.html verfügbar ist.

Skripteingabe

Dieses Skript liest eine Eingabedatei, die im Kapitel [Eingabedatei für die PowerCLI-Beispielskripts zur Bereitstellung von Linux-Desktops](#) beschrieben ist. Es gibt verschiedene Eingabeaufforderungen für die folgenden Informationen aus:

- Annahme der Horizon Agent-Endbenutzerlizenzvereinbarung (EULA)
- IP-Adresse von vCenter Server
- Anmeldename des Administrators für vCenter Server
- Kennwort des Administrators für vCenter Server
- Anmeldename des Administrators für den ESXi-Host
- Kennwort des Administrators für den ESXi-Host
- Anmeldename des Benutzers für das Linux-Gastbetriebssystem
- Kennwort des Benutzers für das Linux-Gastbetriebssystem
- Pfad für das Horizon Agent-TAR-Archiv
- Aktualisieren auf verwaltete VM
- Installieren der Smartcard-Umleitungsfunktion

Skriptinhalt

```
<#
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

```

Skriptausführung

Die folgenden Meldungen resultieren aus einer Ausführung des Skripts:

```

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

```

Beispielskript zur Durchführung eines Upgrades von Horizon Agent auf virtuellen Linux-Maschinen mithilfe von SSH

Sie können das nachfolgend aufgeführte Beispielskript für ein Upgrade von Horizon Agent auf mehreren virtuellen Linux-Maschinen (VMs) anpassen und anwenden. Dieses Skript verwendet SSH zur Ausführung von Befehlen auf den Linux-VMs.

Für das Kopieren und Einfügen des Skriptinhalts ohne Seitenumbrüche verwenden Sie die HTML-Version dieses Themas, die auf der Dokumentationsseite von Horizon 7 unter https://www.vmware.com/support/pubs/view_pubs.html verfügbar ist.

Skripteingabe

Dieses Skript liest eine Eingabedatei, die im Kapitel [Eingabedatei für die PowerCLI-Beispielskripts zur Bereitstellung von Linux-Desktops](#) beschrieben ist. Es gibt verschiedene Eingabeaufforderungen für die folgenden Informationen aus:

- Annahme der Horizon Agent-Endbenutzerlizenzvereinbarung (EULA)
- IP-Adresse von vCenter Server
- Anmeldename des Administrators für vCenter Server

- Kennwort des Administrators für vCenter Server
- Anmeldenname des Administrators für den ESXi-Host
- Kennwort des Administrators für den ESXi-Host
- Anmeldenname des Benutzers für das Linux-Gastbetriebssystem
- Kennwort des Benutzers für das Linux-Gastbetriebssystem
- Pfad für das Horizon Agent-TAR-Archiv
- Aktualisieren auf verwaltete VM
- Installieren der Smartcard-Umleitungsfunktion

Skriptinhalt

```
<#
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
}
$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
"-----"
$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

```

Skriptausführung

Die folgenden Meldungen resultieren aus einer Ausführung des Skripts:

```

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

```

Beispielskript zum Ausführen von Vorgängen auf virtuellen Linux-Maschinen

Sie können das im Folgenden aufgeführte Beispielskript für das Ausführen von Vorgängen auf mehreren virtuellen Linux-Maschinen (VMs) anpassen und anwenden. Zu diesen Vorgängen gehört das Einschalten, Ausschalten, Herunterfahren, Neustarten und Löschen der virtuellen Maschinen.

Dieses Skript löscht virtuelle Maschinen aus vCenter Server, aber nicht aus View.

Für das Kopieren und Einfügen des Skriptinhalts ohne Seitenumbrüche verwenden Sie die HTML-Version dieses Themas, die auf der Dokumentationsseite von Horizon 7 unter https://www.vmware.com/support/pubs/view_pubs.html verfügbar ist.

Skripteingabe

Dieses Skript liest eine Eingabedatei, die im Kapitel [Eingabedatei für die PowerCLI-Beispielskripts zur Bereitstellung von Linux-Desktops](#) beschrieben ist. Es gibt verschiedene Eingabeaufforderungen für die folgenden Informationen aus:

- IP-Adresse von vCenter Server
- Anmeldename des Administrators für vCenter Server
- Kennwort des Administrators für vCenter Server
- Auszuführender Vorgang wie z. B. das Einschalten oder Ausschalten, das Herunterfahren des Gastbetriebssystems, das Neustarten der virtuellen Maschine, das Neustarten des VM-Gastbetriebssystems oder das Löschen einer virtuellen Maschine.
- Die Wartezeit (in Sekunden) zwischen den Vorgängen auf den VMs.

Skriptinhalt

```
<#
.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

```

Skriptausführung

Die folgenden Meldungen resultieren aus einer Ausführung des Skripts:

```

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

```

Für das Einschalten, das Neustarten einer VM oder das Neustarten eines VM-Gastbetriebssystems müssen Sie eine Wartezeit zwischen den virtuellen Maschinen von mindestens 20 Sekunden angeben, um einen „Boot Storm“ (Überlastung des Netzwerks durch zu viele Anmeldungen) zu vermeiden, durch den einige Vorgänge eventuell nicht ausgeführt werden könnten.

Fehlerbehebung bei Linux-Desktops

9

Bei der Verwaltung von Linux-Desktops können bestimmte Probleme auftreten. Zur Diagnose und Behebung dieser Probleme stehen Ihnen verschiedene Vorgehensweisen zur Verfügung.

Dieses Kapitel enthält die folgenden Themen:

- [Ermitteln von Diagnoseinformationen für eine Horizon 7 for Linux-Maschine](#)
- [Durchführen der Fehlerbehebung für das Kopieren/Einfügen zwischen Remote-Desktop und Client-Host](#)
- [Konfigurieren der Linux-Firewall für das Akzeptieren eingehender TCP-Verbindungen](#)
- [Fehler beim Trennen der Verbindung auf dem Horizon Client für ein iPad Pro durch View Agent](#)
- [SLES 12 SP1-Desktop wird nach Ziehen und Ablegen nicht automatisch aktualisiert](#)
- [Fehlerhafte SSO-Verbindung zu einem PowerOff-Agenten](#)
- [Nicht erreichbare VM nach dem Erstellen eines manuellen Desktop-Pools für Linux](#)

Ermitteln von Diagnoseinformationen für eine Horizon 7 for Linux-Maschine

Mit der Ermittlung von Diagnoseinformationen können Sie den technischen Support von VMware bei der Diagnose und Behandlung von Problemen mit einer Horizon 7 for Linux-Maschine unterstützen. Dazu erstellen Sie ein DCT-Bundle (Data Collection Tool, Datenerfassungstool), in dem die Informationen zur Konfiguration der Maschine zusammengestellt und in einem komprimierten TAR-Archiv protokolliert werden.

Verfahren

- 1 Melden Sie sich bei der virtuellen Linux-Maschine als Benutzer mit den erforderlichen Rechten an.
- 2 Öffnen Sie eine Eingabeaufforderung und führen Sie das `dct-debug.sh`-Skript aus.

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

Das Skript generiert ein TAR-Archiv mit dem DCT-Bundle. Beispiel:

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

Das TAR-Archiv wird in dem Verzeichnis generiert, in dem das Skript ausgeführt wurde (das aktuelle Arbeitsverzeichnis).

Durchführen der Fehlerbehebung für das Kopieren/Einfügen zwischen Remote-Desktop und Client-Host

Das Kopieren und Einfügen zwischen dem Remote-Desktop und dem Client-Host dauert mehr als drei Sekunden bei einer maximal unterstützten Datenmenge von 1 MB. Dieses Problem tritt nur beim Kopieren/Einfügen großen Datenmengen auf.

Problem

Wenn Sie eine vCPU und 1 GB Arbeitsspeicher für einen SLED 11 SP3/SP4-Desktop konfigurieren, kann das Kopieren und Einfügen von Daten zwischen den Remote-Desktop und dem lokalen Client-Host mehr als drei Sekunden dauern.

Ursache

Diese Verzögerung ist wahrscheinlich auf die alten Betriebssystem-APIs von SLED 11 SP3/SP4 zurückzuführen.

Lösung

- ◆ Konfigurieren Sie zwei vCPUs und 2 GB Arbeitsspeicher für SLED 11 SP3/SP4.

Konfigurieren der Linux-Firewall für das Akzeptieren eingehender TCP-Verbindungen

Damit Benutzer sich mit ihren Linux-Desktops verbinden können, müssen die Desktops eingehende TCP-Verbindungen von Horizon Client-Geräten, vom Sicherheitsserver und vom View-Verbindungsserver akzeptieren.

Bei Ubuntu- und Kylin-Distributionen wird standardmäßig die iptables-Firewall mit einer Eingaberichtlinie von ACCEPT konfiguriert.

Bei RHEL- und CentOS-Distributionen konfiguriert, wenn möglich, das Horizon Agent-Installationsskript die iptables-Firewall mit einer Eingaberichtlinie von ACCEPT.

Stellen Sie sicher, dass iptables auf einem RHEL- oder CentOS-Gastbetriebssystem über eine Eingaberichtlinie von ACCEPT für neue Verbindungen vom Blast-Port 22443 verfügt.

Wenn der BSG (Blast Secure Gateway) aktiviert ist, werden die Clientverbindungen von einem Horizon Client-Gerät über den BSG auf einem Sicherheitsserver oder einem View-Verbindungsserver zum Linux-Desktop hergestellt. Ist der BSG nicht aktiviert, werden die Verbindungen zum Linux-Desktop direkt vom Horizon Client-Gerät hergestellt.

Fehler beim Trennen der Verbindung auf dem Horizon Client für ein iPad Pro durch View Agent

Die SUSE View Agent-Verbindung kann nach einem Neustart oder Herunterfahren auf einem iPad Pro Horizon Client nicht getrennt werden.

Problem

Wenn Sie eine virtuelle SUSE-Maschine auf einem iPad Pro Horizon Client neu starten oder herunterfahren, antwortet der Desktop nicht. Fehler beim Trennen durch View Agent.

Ursache

Die SUSE-Maschine sendet nach dem Neustart oder Herunterfahren die Meldungen an Horizon Client möglicherweise nicht richtig.

Lösung

- ◆ Trennen Sie die Desktop-Verbindung manuell vom iPad Pro Horizon Client

SLES 12 SP1-Desktop wird nach Ziehen und Ablegen nicht automatisch aktualisiert

SLES 12 SP1 wird in einem Modus mit mehreren Monitoren nach dem Ziehen und Ablegen eines Gnome-Terminals nicht automatisch aktualisiert.

Problem

Wenn Sie SLES 12 SP1 in einem Modus mit mehreren Monitoren starten und zum Fenstermodus zurückkehren, wird der Desktop nicht automatisch aktualisiert, wenn Sie ein Gnome-Terminal ziehen und ablegen.

Ursache

Das Gnome-Terminal antwortet nicht auf das Ziehen und Ablegen.

Lösung

- 1 Beenden Sie die Gnome-Shell.



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

- 2 Starten Sie die Gnome-Shell erneut.

Fehlerhafte SSO-Verbindung zu einem PowerOff-Agenten

SSO stellt keine Verbindung zu einem PowerOff-Agenten her.

Problem

Wenn Sie sich als Broker anmelden und eine Verbindung zu einem Agenten herstellen, treten beim Verbinden von SSO zum PowerOff-Agenten Fehler auf.

Lösung

- ◆ Melden Sie sich manuell am Desktop an oder trennen Sie die Verbindung und stellen Sie eine erneute Verbindung zum Agenten her.

Nicht erreichbare VM nach dem Erstellen eines manuellen Desktop-Pools für Linux

Der Status der virtuellen Maschine antwortet nicht.

Problem

Nach dem Erstellen eines manuellen Desktop-Pools lautet der Status der virtuellen Maschine möglicherweise „Warten auf Agent“ oder „Nicht erreichbar“.

Ursache

Es liegen möglicherweise verschiedene Benutzerfehlerkonfigurations- oder Setup-Probleme dahingehend vor, dass der Status der virtuellen Maschine „Nicht erreichbar“ oder „Warten auf Agent“ lautet.

- Stellen Sie sicher, dass die Option `machine.id` in der VMX-Konfigurationsdatei der virtuellen Maschine vorhanden ist.

Wenn sie nicht vorhanden ist, müssen Sie sicherstellen, dass die virtuelle Maschine richtig zum Desktop-Pool hinzugefügt wurde. Erstellen Sie ansonsten den Desktop-Pool, um dem Broker zu ermöglichen, die Option zur VMX-Konfigurationsdatei neu zu schreiben.

- Stellen Sie sicher, dass das VMware-Tool oder Open VM-Tool richtig installiert ist.

Wenn die Schritte zum Installieren des VMware-Tools oder Open VM-Tools nicht richtig ausgeführt wurden, ist der `vmware-rpctool`-Befehl unter PATH auf der virtuellen Linux-Maschine möglicherweise nicht vorhanden. Sie müssen die Anleitung befolgen, um das VMware Tool oder Open VM-Tool zu installieren.

Führen Sie den Befehl nach dem Abschließen der Installation aus.

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

Die `machine.id`-Werte werden aus der VMX-Konfigurationsdatei der virtuellen Maschinen aufgeführt.

- Überprüfen Sie, ob der vollqualifizierte Domänenname des Brokers in einer IP-Adresse in der virtuellen Linux-Maschine des Agenten aufgelöst werden kann.