

# Horizon 7 for Linux 데 스크톱 설정

마지막 수정 날짜: 2017년 9월 15일  
VMware Horizon 7 7.2

VMware 웹 사이트 (<https://docs.vmware.com/kr/>) 에서 최신 기술 문서를 확인할 수 있습니다.  
또한 VMware 웹 사이트에서 최신 제품 업데이트를 제공합니다.  
이 문서에 대한 의견이 있으면 [docfeedback@vmware.com](mailto:docfeedback@vmware.com)으로 사용자 의견을 보내주십시오.

Copyright © 2016, 2017 VMware, Inc. 판권 소유. [저작권 및 상표 정보](#).

**VMware, Inc.**  
3401 Hillview Ave.  
Palo Alto, CA 94304  
[www.vmware.com](http://www.vmware.com)

# 목차

Horizon 7 for Linux 데스크톱 설정	5
<b>1</b> 기능 및 시스템 요구 사항	7
Horizon Linux 데스크톱의 기능	7
Horizon 7 for Linux 데스크톱의 구성 단계 개요	10
Horizon 7 for Linux에 대한 시스템 요구 사항	11
<b>2</b> 데스크톱 배포를 위해 Linux 가상 시스템 준비	19
가상 시스템 생성 및 Linux 설치	19
원격 데스크톱 배포를 위한 Linux 시스템 준비	20
Horizon Agent에 대한 종속성 패키지 설치	22
<b>3</b> Linux 데스크톱의 Active Directory 통합 설정	25
Linux와 Active Directory 통합	25
Single Sign-On 및 스마트 카드 리디렉션 설정	26
<b>4</b> Linux 데스크톱의 그래픽 설정	29
vGPU에 대해 지원되는 RHEL 배포 구성	29
vDGA용 RHEL 6 구성	34
vSGA용 RHEL 7 구성	38
<b>5</b> Horizon Agent 설치	41
Linux 가상 시스템에 Horizon Agent 설치	41
Linux Agent용 인증서 구성	43
Linux 가상 시스템에서 Horizon Agent 업그레이드	44
Horizon 7 for Linux 시스템 제거	46
<b>6</b> Linux 데스크톱용 구성 옵션	47
Linux 데스크톱의 구성 파일에서 옵션 설정	47
Linux 데스크톱용 Blast 설정 예	55
Linux 데스크톱에 대한 클라이언트 드라이브 리디렉션 옵션의 예	55
Linux 데스크톱의 vSphere 콘솔 디스플레이 표시하지 않음	56
<b>7</b> Linux 데스크톱 풀 생성 및 관리	57
Linux용 수동 데스크톱 풀 생성	57
Linux용 데스크톱 풀 관리	58
Linux용 자동화된 전체 클론 데스크톱 풀 생성	59
브로커 PowerCLI 명령	61
<b>8</b> 수동 데스크톱 풀의 Horizon 7 대량 배포	65
Linux 데스크톱의 대량 배포 개요	65

- Linux 데스크톱의 대량 업그레이드 개요 67
- Linux 데스크톱 시스템의 복제를 위한 가상 시스템 템플릿 만들기 67
- Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일 69
- Linux 가상 시스템을 복제하기 위한 샘플 스크립트 69
- 복제된 가상 시스템을 AD 도메인에 연결하기 위한 샘플 스크립트 73
- SSH를 사용하여 복제된 가상 시스템을 AD 도메인에 연결하기 위한 샘플 스크립트 76
- 구성 파일을 Linux 가상 시스템에 업로드하기 위한 샘플 스크립트 79
- SSH를 사용하여 구성 파일을 Linux 가상 시스템에 업로드하기 위한 샘플 스크립트 83
- Linux 데스크톱 시스템에서 Horizon Agent를 업그레이드하기 위한 샘플 스크립트 87
- SSH를 사용하여 Linux 가상 시스템에서 Horizon Agent를 업그레이드하기 위한 샘플 스크립트 91
- Linux 가상 시스템에서 작업을 수행하기 위한 샘플 스크립트 97

## 9 Linux 데스크톱 문제 해결 101

- Horizon 7 for Linux 시스템에 대한 진단 정보 수집 101
- 원격 데스크톱과 클라이언트 호스트 간의 복사 및 붙여넣기 문제 해결 102
- 들어오는 TCP 연결을 허용하도록 Linux 방화벽 구성 102
- View Agent에서 iPad Pro Horizon Client에 연결하지 못함 102
- SLES 12 SP1 데스크톱은 드래그 앤 드롭 이후 자동 새로 고침을 수행하지 않음 103
- SSO에서 PowerOff 에이전트에 연결하지 못함 103
- Linux용 수동 데스크톱 풀을 생성한 후에 VM에 연결할 수 없음 103

색인 105

# Horizon 7 for Linux 데스크톱 설정

---

Horizon 7 for Linux 데스크톱 설정 문서는 Linux 게스트 운영 체제 준비, 가상 시스템에 Horizon Agent 설치 및 Horizon 7 배포에서 사용할 수 있게 View Administrator에서 시스템을 구성하는 방법과 VMware Horizon® 7 데스크톱으로 사용할 수 있게 Linux 가상 시스템을 설정하는 방법에 대한 정보를 제공합니다.

## 대상

이 정보는 Linux 게스트 운영 체제에서 실행되는 원격 데스크톱을 구성 및 사용하려는 모든 사용자를 위해 작성되었습니다. 이 정보는 가상 시스템 기술과 데이터센터 운영에 익숙하고 경험 많은 Linux 시스템 관리자를 대상으로 작성되었습니다.



## 기능 및 시스템 요구 사항

Horizon 6 이상에서는 사용자가 Linux 운영 체제를 실행하는 원격 데스크톱에 연결할 수 있습니다. 이 장에서는 다음 주제에 대해 설명합니다.

- “Horizon Linux 데스크톱의 기능,” (7 페이지)
- “Horizon 7 for Linux 데스크톱의 구성 단계 개요,” (10 페이지)
- “Horizon 7 for Linux에 대한 시스템 요구 사항,” (11 페이지)

### Horizon Linux 데스크톱의 기능

Horizon 7 버전 7.2에서는 Linux 데스크톱에 대해 몇 가지 새 기능이 도입되었습니다.

다음 새 기능은 Horizon 7 버전 7.2 릴리스에서 지원됩니다.

- CDR(클라이언트 드라이브 리디렉션)
- USB 리디렉션 지원
- HTML Access에 대한 오디오 출력 지원
- RHEL 7 Workstation x64 및 CentOS 7 x64의 Single Sign-On 지원
- CentOS 6 x64 및 RHEL 6 x64의 KDE(K 데스크톱 환경) 지원
- RHEL 6.9 x64 및 CentOS 6.9 x64 지원

다음 목록에서는 Horizon Linux 데스크톱용 핵심 기능을 보여줍니다.

**자동화된 전체 클론 데스크톱 풀** Horizon 7 버전 7.0.2 릴리스부터 Linux 데스크톱용으로 자동화된 전체 클론 데스크톱 풀을 생성할 수 있습니다.

#### 수동 데스크톱 풀

#### 시스템 소스

- 관리되는 가상 시스템 - vCenter 가상 시스템의 시스템 소스입니다. 관리되는 가상 시스템은 새 배포 및 업그레이드 배포에 지원됩니다.
- 관리되지 않는 가상 시스템 - 기타 소스의 시스템 소스입니다. 관리되지 않는 가상 시스템은 관리되지 않는 가상 시스템 배포에서 업그레이드할 때만 지원됩니다.

**참고** 가능한 최고의 성능을 유지하려면 관리되지 않는 가상 시스템을 사용하지 마십시오.

#### 다중 모니터

- vDGA/vGPU 데스크톱은 4대의 모니터에서 최대 2560x1600의 해상도를 지원합니다.

- vSphere 6.0 이상에서의 2D/vSGA 데스크톱은 4대의 모니터에서 최대 2048x1536의 해상도를 지원하고, 3대의 모니터에서 최대 2560x1600의 해상도를 지원합니다.

Ubuntu 14.04/16.04에서는 다중 모니터를 지원하기 위해 Compiz를 사용하지 않도록 설정해야 합니다. 자세한 내용은 <http://kb.vmware.com/kb/2114809>에 나와 있습니다.

SLES 12 SP1의 경우 커널 수준 kernel-default-3.12.49-11.1의 기본 패키지를 사용해야 합니다. 패키지를 업그레이드한 경우 다중 모니터 기능이 실패하며 데스크톱이 한 대의 모니터에만 표시됩니다.

Horizon 7 for Linux 데스크톱에서는 VMware Horizon HTML Access가 다중 모니터 기능을 지원하지 않습니다.

### 무손실 PNG

데스크톱에서 생성되는 이미지 및 비디오가 클라이언트 디바이스에서 픽셀 단위로 정확하게 렌더링됩니다.

### 소프트웨어 H.264 인코더

H.264는 Horizon 데스크톱에 대해 Blast Extreme 성능을 향상시킬 수 있습니다. 이러한 효과는 저대역폭 네트워크에서 특히 두드러집니다. 클라이언트 측에서 H.264가 사용되지 않도록 설정되면 Blast Extreme은 자동으로 JPEG/PNG 인코딩으로 변경됩니다.

Horizon 7 버전 7.0.3 릴리스에서는 다중 모니터가 지원됩니다.

### 3D 그래픽

다음과 같은 Linux 버전 및 그래픽 카드 조합에서 3D 그래픽이 지원됩니다.

- vSGA는 NVIDIA GRID K1 또는 K2 그래픽 카드가 장착된 RHEL 7 Workstation x64에서 지원됩니다.
- vDGA는 NVIDIA GRID K1 또는 K2 그래픽 카드가 장착된 RHEL 6 Workstation x64에서 지원됩니다.
- vGPU는 NVIDIA Maxwell M60 그래픽 카드가 장착된 RHEL 6 Workstation x64에서 지원됩니다.
- vGPU는 NVIDIA Maxwell M60 그래픽 카드가 장착된 RHEL 7 Workstation x64에서 지원됩니다.
- vGPU는 NVIDIA M6 그래픽 카드가 장착된 RHEL 6 Workstation x64에서 지원됩니다.
- vGPU는 NVIDIA M6 그래픽 카드가 장착된 RHEL 7 Workstation x64에서 지원됩니다.

### 클립보드 리디렉션

클립보드 리디렉션 기능을 사용하면 클라이언트 호스트와 원격 Linux 데스크톱 간에 서식 있는 텍스트 또는 일반 텍스트를 복사한 후 붙여넣을 수 있습니다. Horizon Agent 옵션을 사용하여 복사/붙여넣기 방향 및 최대 텍스트 크기를 설정할 수 있습니다. 이 기능은 기본적으로 사용하도록 설정됩니다. 설치 중에 사용하지 않도록 설정할 수 있습니다.

### Single Sign-On

Single Sign-On은 다음 Linux 버전에서 지원됩니다.

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



<b>SSO를 사용한 스마트 카드 리더렉션</b>	스마트 카드 리더렉션은 RHEL 6 Workstation x64에서 지원됩니다. PIV(Personal Identity Verification) 카드 및 CAC(Common Access Card)가 지원됩니다. Mac 클라이언트는 지원되지 않습니다.
<b>오디오 입력</b>	클라이언트 호스트에서 원격 Linux 데스크톱으로의 오디오 입력 리더렉션이 지원됩니다. 이 기능은 USB 리더렉션 기능을 기반으로 하지 않습니다. 이 기능을 사용하도록 설정하려면 설치 중에 선택해야 합니다. 오디오 입력을 위해서는 애플리케이션의 디바이스 "PulseAudio 서버(로컬)"에서 시스템 기본 오디오를 선택해야 합니다. 이 기능은 다음에서 지원됩니다. <ul style="list-style-type: none"> <li>■ Ubuntu 14.04/16.04 x64</li> <li>■ CentOS 7 x64</li> <li>■ RHEL 7 Workstation x64</li> </ul>
<b>오디오 출력</b>	오디오 출력 리더렉션이 지원됩니다. 이 기능은 기본적으로 사용하도록 설정됩니다. 이 기능을 사용하지 않도록 설정하려면 RemoteDisplay.allowAudio 옵션을 <b>false</b> 로 설정해야 합니다. Horizon 7 버전 7.2 릴리스부터 Chrome 및 Firefox 브라우저를 사용해 VMWare Horizon HTML Access에 액세스하면 Linux 데스크톱에 대해 오디오 출력 지원이 제공됩니다.
<b>클라이언트 드라이브 리더렉션</b>	CDR(클라이언트 드라이브 리더렉션) 기능은 Horizon 7 버전 7.2 릴리스부터 사용할 수 있습니다. CDR 기능을 사용하도록 설정하면 로컬 시스템의 공유 폴더 및 드라이브에 액세스할 수 있게 됩니다. 원격 Linux 데스크톱의 홈 디렉토리에 있는 tsclient 폴더를 사용합니다. 이 기능을 사용하려면 CDR 구성 요소를 설치해야 합니다.
<b>USB 리더렉션</b>	USB 리더렉션 기능은 Horizon 7 버전 7.2 릴리스부터 사용할 수 있습니다. 이 기능을 사용하면 원격 Linux 데스크톱에서 로컬로 연결된 USB 디바이스에 액세스할 수 있습니다. USB 기능을 사용하려면 USB 리더렉션 구성 요소 및 USB VHCI 드라이버 커널 모듈을 설치해야 합니다. 리더렉션하려는 USB 디바이스를 사용하기 위한 충분한 권한이 부여되었는지 확인합니다.
	<b>참고</b> USB 3.0 프로토콜은 Horizon 7 버전 7.2 릴리스에서 지원됩니다.
<b>키보드 레이아웃 및 로캘 동기화</b>	이 기능은 클라이언트 시스템 로캘 및 현재 키보드 레이아웃을 Horizon Linux Agent 데스크톱과 동기화할지 여부를 지정합니다. 이 설정이 사용되도록 설정되거나 구성되지 않은 경우, 동기화가 허용됩니다. 이 설정이 사용되지 않도록 설정되면 동기화가 허용되지 않습니다.  이 기능은 Windows용 Horizon Client에서만 지원되고 영어, 프랑스어, 독일어, 일본어, 한국어, 스페인어, 중국어 간체 및 중국어 번체 로캘에서만 지원됩니다.
<b>K 데스크톱 환경</b>	Horizon 7 버전 7.2 릴리스에서는 KDE(K 데스크톱 환경)가 다음 Linux 배포에서 지원됩니다. <ul style="list-style-type: none"> <li>■ CentOS 6 x64</li> <li>■ RHEL 6 x64</li> </ul>
<b>FIPS 140-2 모드</b>	FIPS(Federal Information Processing Standard) 140-2 모드 지원은 아직 NIST CMVP(Cryptographic Module Validation Program)를 통해 유효성이 검사되지 않았지만 Horizon 7 버전 7.2부터 사용할 수 있습니다.

Linux용 Horizon 7 Agent는 FIPS 140-2 규정 준수를 위해 설계된 암호화 모듈을 구현합니다. 이러한 모듈은 CMVP 인증서 #2839 및 #2866에 나열된 운영 환경에서 유효성이 검사된 후 이 플랫폼에 이식되었습니다. 그렇지만 VMware의 NIST CAVP 및 CMVP 인증서에 새 운영 환경을 포함한다는 CAVP 및 CMVP 테스트 요구 사항은 제품 로드맵에서 완료될 항목으로 유지됩니다.

---

**참고** FIPS 140-2 모드를 지원하는 데 TLS(Transport Layer Security) 프로토콜 버전 1.2가 필요합니다.

---

Linux 데스크톱 및 데스크톱 풀에는 다음과 같은 제한이 있습니다.

- 가상 인쇄, 위치 기반 인쇄 및 실시간 비디오는 지원되지 않습니다.

---

**참고** 보안 서버가 사용되면 보안 서버와 Linux 데스크톱 간의 트래픽을 허용하기 위해 내부 방화벽에서 포트 22443이 열려 있어야 합니다.

---

## Horizon 7 for Linux 데스크톱의 구성 단계 개요

Horizon 7 for Linux 데스크톱을 설치하고 구성할 때에는 가상 시스템에 2D 그래픽과 3D 그래픽 중 어느 것을 설치하는지에 따라 다른 단계 순서를 따라야 합니다.

### 2D 그래픽 - 구성 단계 개요

2D 그래픽의 경우에는 다음 단계를 수행합니다.

- 1 Horizon 7 for Linux 배포를 설정하기 위한 시스템 요구 사항을 검토합니다. [“Horizon 7 for Linux에 대한 시스템 요구 사항,”](#) (11 페이지)를 참조하십시오.
- 2 vSphere에서 가상 시스템을 생성하고 Linux 운영 체제를 설치합니다. [“가상 시스템 생성 및 Linux 설치,”](#) (19 페이지)를 참조하십시오.
- 3 Horizon 7 환경에서 데스크톱으로 배포할 게스트 운영 체제를 준비합니다. [“원격 데스크톱 배포를 위한 Linux 시스템 준비,”](#) (20 페이지)를 참조하십시오.
- 4 Linux 게스트 운영 체제를 Active Directory로 인증하도록 구성합니다. 이 단계는 환경의 요구 사항을 기반으로 타사 소프트웨어에서 구현되며, 이 가이드에서는 설명하지 않습니다.
- 5 Linux 가상 시스템에 Horizon Agent를 설치합니다. [“Linux 가상 시스템에 Horizon Agent 설치,”](#) (41 페이지)를 참조하십시오.
- 6 구성된 Linux 가상 시스템을 포함하는 데스크톱 풀을 생성합니다. [“Linux용 수동 데스크톱 풀 생성,”](#) (57 페이지)를 참조하십시오.

### 3D 그래픽 - 구성 단계 개요

시스템에서 Horizon Agent를 설치하고 View Administrator에 데스크톱 풀을 배포하려면 먼저 Linux 가상 시스템에서 NVIDIA GRID vGPU, vDGA 또는 vSGA 구성을 완료해야 합니다.

- 1 Horizon 7 for Linux 배포를 설정하기 위한 시스템 요구 사항을 검토합니다. [“Horizon 7 for Linux에 대한 시스템 요구 사항,”](#) (11 페이지)를 참조하십시오.
- 2 vSphere에서 가상 시스템을 생성하고 Linux 운영 체제를 설치합니다. [“가상 시스템 생성 및 Linux 설치,”](#) (19 페이지)를 참조하십시오.
- 3 Horizon 7 환경에서 데스크톱으로 배포할 게스트 운영 체제를 준비합니다. [“원격 데스크톱 배포를 위한 Linux 시스템 준비,”](#) (20 페이지)를 참조하십시오.
- 4 Linux 게스트 운영 체제를 Active Directory로 인증하도록 구성합니다. 이 단계는 환경의 요구 사항을 기반으로 타사 소프트웨어에서 구현되며, 이 가이드에서는 설명하지 않습니다.

- 5 ESXi 호스트와 Linux 가상 시스템에 3D 기능을 구성합니다. 설치하려는 3D 기능의 절차를 수행합니다.
  - “vGPU에 대해 지원되는 RHEL 배포 구성,” (29 페이지)를 참조하십시오.
  - “vDGA용 RHEL 6 구성,” (34 페이지)를 참조하십시오.
  - “vSGA용 RHEL 7 구성,” (38 페이지)를 참조하십시오.
- 6 Linux 가상 시스템에 Horizon Agent를 설치합니다. “Linux 가상 시스템에 Horizon Agent 설치,” (41 페이지)를 참조하십시오.
- 7 구성된 Linux 가상 시스템을 포함하는 데스크톱 풀을 생성합니다. “Linux용 수동 데스크톱 풀 생성,” (57 페이지)를 참조하십시오.

## 대량 배포

View Administrator에서는 수동 데스크톱 풀에만 Linux 가상 시스템을 배포할 수 있습니다. vSphere PowerCLI를 사용하여 Linux 데스크톱 시스템 풀의 배포를 자동화하는 스크립트를 개발할 수 있습니다. 8장, “수동 데스크톱 풀의 Horizon 7 대량 배포,” (65 페이지)를 참조하십시오.

## Horizon 7 for Linux에 대한 시스템 요구 사항

Horizon 7 for Linux를 설치하려면 Linux 시스템이 운영 체제, Horizon 7 및 vSphere 플랫폼에 대한 특정 요구 사항을 충족해야 합니다.

### Horizon Agent에 지원되는 Linux 버전

다음 표에는 데스크톱 풀의 가상 시스템에서 지원되는 Linux 운영 체제가 나와 있습니다.

**표 1-1.** View Agent에 대한 지원되는 Linux 운영 체제

Linux 배포	아키텍처
Ubuntu 14.04 및 16.04	x64
<b>참고</b> 성능 저하를 피하려면 Compiz를 사용하지 않도록 설정합니다. 자세한 내용은 <a href="http://kb.vmware.com/kb/2114809">http://kb.vmware.com/kb/2114809</a> 에 나와 있습니다.	
Ubuntu 12.04	x64
RHEL 6.6, 6.7, 6.8, 6.9, 7.2 및 7.3	x64
CentOS 6.6, 6.7, 6.8, 6.9, 7.2 및 7.3	x64
NeoKylin 6 Update 1	x64
SLED 11 SP3/SP4, SLED 12 SP1/SP2	x64
SLES 12 SP1/SP2	x64

**참고** Linux Agent에는 일부 Linux 배포에서 종속성 패키지가 있습니다. 자세한 내용은 “Horizon Agent에 대한 종속성 패키지 설치,” (22 페이지)에 나와 있습니다.

### 필수 플랫폼 및 Horizon 7 소프트웨어 버전

Horizon 7 for Linux를 설치하고 사용하려면 배포가 vSphere 플랫폼, Horizon 7 및 Horizon Client 소프트웨어에 대한 특정 요구 사항을 충족해야 합니다.

**표 1-2.** 필수 플랫폼 및 Horizon 7 소프트웨어 버전

플랫폼 및 소프트웨어	지원되는 버전
vSphere 플랫폼 버전	<ul style="list-style-type: none"> <li>■ vSphere 6.0 U2 이상 릴리스</li> <li>■ vSphere 6.5 이상 릴리스</li> </ul>
Horizon 환경	<ul style="list-style-type: none"> <li>■ Horizon 연결 서버 7.2</li> </ul>
Horizon Client 소프트웨어	<ul style="list-style-type: none"> <li>■ Android용 Horizon Client 4.5.0</li> <li>■ Windows용 Horizon Client 4.5.0</li> <li>■ Linux용 Horizon Client 4.5.0</li> <li>■ Mac OS X용 Horizon Client 4.5.0</li> <li>■ iOS용 Horizon Client 4.5.0(iPad Pro)</li> <li>■ Chrome, Firefox 및 Internet Explorer의 HTML Access 4.5.0</li> <li>■ Zero 클라이언트는 지원되지 않습니다.</li> </ul>

## Linux 가상 시스템에서 사용하는 TCP 포트

View Agent 및 Horizon Client에서는 서로, 그리고 다양한 View Server 구성 요소와의 네트워크 액세스에 TCP 포트를 사용합니다.

**표 1-3.** Linux 가상 시스템에서 사용하는 TCP 포트

소스	포트	대상	포트	프로토콜	설명
Horizon Client	*	Linux Agent	22443	TCP	Blast 보안 게이트웨이를 사용하지 않을 경우 Blast
보안 서버, View 연결 서버 또는 Access Point 장치	*	Linux Agent	22443	TCP	Blast 보안 게이트웨이를 사용할 경우 Blast
View Agent	*	View 연결 서버	4001, 4002	TCP	JMS SSL 트래픽입니다.

**참고** 클라이언트에서 사용하는 TCP 및 UDP에 대한 자세한 내용은 View 문서의 Horizon Client 및 View Agent 보안에 있는 "클라이언트 및 View Agent에서 사용하는 TCP 및 UDP 포트"를 참조하십시오.

## Linux 가상 시스템에서 사용하는 Linux 계정 확인

다음 표에서는 Linux 가상 시스템에서 사용하는 계정 이름 및 계정 유형을 나열합니다.

**표 1-4.** 계정 이름 및 계정 유형

계정 이름	계정 유형	사용자
루트	Linux OS 내장	Java Standalone Agent, mksvchanserver, 셸 스크립트
vmwblast	Linux Agent 설치 관리자에서 생성	VMwareBlastServer
<현재 로그인 사용자>	Linux OS 내장 또는 AD 사용자 또는 LDAP 사용자	python 스크립트

## 데스크톱 환경

Horizon 7 for Linux 데스크톱은 여러 다른 Linux 배포의 여러 데스크톱 환경을 지원합니다. 다음 표에는 Horizon 7 for Linux 데스크톱에서 지원되는 각 Linux 배포에 대한 기본 데스크톱 환경이 나와 있습니다.

**표 1-5.** 지원되는 데스크톱 환경

Linux 배포	기본 데스크톱 환경	Horizon 7 for Linux 데스크톱에서 지원되는 데스크톱 환경
Ubuntu 14.04/16.04	Unity	Gnome Fallback/Flashback(Metacity)
Ubuntu 12.04	Unity	Unity
RHEL/CentOS 6	Gnome	Gnome, KDE(K 데스크톱 환경)
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

Ubuntu 14.04/16.04에서 Gnome Fallback/Flashback(Metacity) 데스크톱 환경을 설치하려면 VMware KB 문서 <http://kb.vmware.com/kb/2114809>에서 자세한 내용을 참조하십시오.

RHEL/CentOS 6 배포에서 KDE를 기본 데스크톱 환경으로 설치하려면 다음 정보를 사용하십시오.

**참고** KDE에 대한 SSO(Single Sign-On)는 아래 설명된 단계를 사용하는 경우에만 작동합니다.

- 1 기본 데스크톱 환경 설정을 사용하여 RHEL/CentOS 6 OS를 설치합니다. 그러면 Gnome이 설치됩니다.
- 2 다음 명령을 사용하여 KDE를 설치합니다.

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

여러 데스크톱 환경이 설치된 Linux 배포에서 SSO를 사용하도록 설정한 경우 다음 정보를 사용하여 데스크톱 환경을 선택하십시오.

- Ubuntu 14.04/16.04의 경우 `/etc/vmware/viewagent-custom.conf` 파일에서 `UseGnomeFlashback` 옵션을 사용하도록 설정하여 강제로 최종 사용자가 Gnome Fallback/Flashback(Metacity)을 사용하도록 하십시오.
- RHEL/CentOS 6의 경우 최종 사용자는 `~/.dmrc` 파일에 데스크톱 환경 이름 `kde` 또는 `gnome`을 지정하여 다음 SSO 로그인 세션에 사용될 환경을 결정해야 합니다. 예를 들어 기본 데스크톱 환경으로 KDE를 사용하려면 최종 사용자는 `~/.dmrc` 파일에 다음 설정을 포함해야 합니다.

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

위 예시의 경우, 초기 설정 후에 최종 사용자는 Linux 시스템에서 로그아웃하거나 시스템을 재부팅하여 KDE를 다음 로그인 세션의 기본 데스크톱으로 지정해야 합니다.

여러 데스크톱 환경이 설치된 Linux 배포에서 SSO를 사용하지 않도록 설정한 경우 최종 사용자는 해당 Linux 배포에 로그인할 때 원하는 데스크톱 환경을 선택해야 합니다.

## USB 리디렉션용 VHCI 드라이버

USB 리디렉션 기능은 Horizon 7 for Linux(버전 7.1) 데스크톱부터 지원됩니다. 이 기능은 USB VHCI(Virtual Host Controller Interface) 커널 드라이버에 의존합니다. USB 3.0을 지원하도록 VHCI 드라이버를 패치해야 합니다.

Horizon for Linux 설치 관리자에는 지원되는 Linux 배포의 기본 커널용 VHCI 드라이버 바이너리가 포함되어 있습니다. USB 리디렉션 기능이 선택되면 VHCI 드라이버가 설치됩니다. 다음 표에는 Horizon for Linux 데스크톱 설치 관리자가 설치하는 기본 커널 버전이 나와 있습니다.

**표 1-6.** 기본 커널 버전

Linux 배포	기본 커널 버전
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

Linux 시스템이 Horizon for Linux 설치 관리자에 포함된 기본 버전과는 다른 커널 버전을 사용하는 경우 <https://sourceforge.net/projects/usb-vhci/files/linux%20kernel%20module/>에서 USB VHCI 소스 코드를 다운로드해야 합니다. 그런 후 VHCI 드라이버 소스 코드를 컴파일하고 결과 바이너리를 Linux 시스템에 설치해야 합니다.

**참고** VHCI 드라이버 설치하는 Horizon for Linux 설치 전에 완료해야 합니다.

최신 버전의 USB VHCI 드라이버 소스 코드를 다운로드한 후에 다음 명령을 사용하여 컴파일한 후 Linux 시스템에 설치합니다. 예를 들어 설치 파일 VMware-horizonagent-linux-x86\_64-*<버전>*-*<빌드 번호>*.tar.gz의 압축을 /install\_tmp/ 디렉토리 아래에 풀면 *full-path\_to\_patch-file*은 /install\_tmp/VMware-horizonagent-linux-x86\_64-*<버전>*-*<빌드 번호>*/resources/vhci/patch/vhci.patch이고 사용할 patch 명령은 다음과 같습니다.

```
# patch -p1 < /install_tmp/VMware-horizonagent-linux-x86_64-<버전>-<빌드 번호>
>/resources/vhci/patch/vhci.patch
```

표 1-7. USB VHCI 드라이버 컴파일 및 설치

Linux 배포	USB VHCI 드라이버 컴파일 및 설치 단계
Ubuntu 14.04	VHCI 드라이버를 컴파일하고 설치합니다. # tar -xzf vhci-hcd-1.15.tar.gz
Ubuntu 16.04	# cd vhci-hcd-1.15 # patch -p1 < full-path_to_patch-file # make clean && make && make install
RHEL/CentO S 6.9	1 종속성 패키지를 설치합니다. # yum install gcc-c++
RHEL/CentO S 7.3	# yum install kernel-devel-\$(uname -r) # yum install kernel-headers-\$(uname -r) # yum install patch
	2 VHCI 드라이버를 컴파일하고 설치합니다. # tar -xzf vhci-hcd-1.15.tar.gz # cd vhci-hcd-1.15 # patch -p1 < full-path_to_patch-file # make clean && make && make install
SUSE 11 SP4	1 현재 커널 패키지 버전 확인 # rpm -qa   grep kernel-default-\$(echo \$(uname -r)   cut -d '-' -f 1,2)
SUSE 12 SP2	현재 설치된 커널 패키지 이름이 출력됩니다. 예를 들어 패키지 이름이 kernel-default-3.0.101-63.1이면 현재 커널 패키지 버전은 3.0.101-63.1입니다.
	2 종속성 패키지를 설치합니다. a SUSE 11의 경우 현재 커널 gcc 및 patch 패키지와 일치하는 kernel-source 및 kernel-default-devel 패키지를 설치합니다. #zypper install --oldpackage kernel-source-<커널 패키지 버전> W kernel-default-devel-<커널 패키지 버전> gcc patch 예: #zypper install --oldpackage kernel-source-3.0.101-63.1 kernel-default-devel-3.0.101-63.1 gcc patch
	b SUSE 12의 경우 kernel-devel, kernel-default-devel, kernel-macros 및 patch 패키지를 설치합니다. #zypper install --oldpackage kernel-devel-<커널 패키지 버전> W kernel-default-devel-<커널 패키지 버전> kernel-macros-<커널 패키지 버전> patch 예: #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
	3 VHCI 드라이버를 컴파일하고 설치합니다. # tar -xzf vhci-hcd-1.15.tar.gz # cd vhci-hcd-1.15 # patch -p1 < full-path_to_patch-file # mkdir -p linux/\$(echo \$(uname -r)   cut -d '-' -f 1)/drivers/usb/core # cp /lib/modules/\$(uname -r)/source/include/linux/usb/hcd.h linux/\$(echo \$(uname -r)   cut -d '-' -f 1)/drivers/usb/core # make clean && make && make install

Linux 커널 버전이 변경되면 VHCI 드라이버를 다시 컴파일한 후 다시 설치해야 하지만 Horizon for Linux를 다시 설치할 필요는 없습니다.

## 2D 또는 vSGA 그래픽을 위한 가상 시스템 설정

특정 Horizon 7 for Linux 가상 시스템을 생성할 때 메모리 설정 및 구성 매개 변수를 권장되는 최소값으로 변경해야 합니다.

NVIDIA vDGA를 사용하도록 구성된 가상 시스템은 NVIDIA 물리적 그래픽 카드를 사용합니다. NVIDIA GRID vGPU를 사용하도록 구성된 가상 시스템은 NVIDIA 물리적 그래픽 가속기를 기반으로 하는 NVIDIA 가상 그래픽 카드를 사용합니다. 이러한 가상 시스템에 대해서는 비디오 메모리(vRAM) 설정 및 구성 매개 변수를 변경할 필요가 없습니다.

2D 또는 vSGA 그래픽을 사용하도록 구성된 가상 시스템은 VMware 가상 그래픽 카드를 사용하며 이러한 유형의 가상 시스템에 대해 다음 설정을 변경해야 합니다.

- 비디오 메모리(vRAM) 설정
- 구성 매개 변수
- 3D 메모리 설정
- 성능 요구 사항에 대한 vCPU 및 가상 메모리 설정

### 비디오 메모리(vRAM) 설정

vSphere Client에서 Linux 가상 시스템을 생성할 때는 표 1-8에 나온 것처럼 vRAM 크기를 구성하십시오. 가상 시스템에 대해 구성하는 모니터의 수 및 해상도에 대해 권장되는 vRAM 크기를 설정하십시오.

표 1-8. 2D 또는 vSGA 그래픽을 위한 권장 vRAM 설정

vRAM 크기	모니터 수	최대 해상도
10 MB	1	1600x1200 또는 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

이러한 vRAM 크기는 최소 권장 사항입니다. 가상 시스템에서 더 많은 리소스를 사용할 수 있으면 비디오 성능 향상을 위해 vRAM을 더 큰 값으로 설정할 수 있습니다.

최저 해상도의 단일 모니터로 구성된 시스템에서 권장되는 최소 vRAM 크기는 10MB입니다.

가상 시스템의 전원을 끄고 [“가상 시스템 생성 및 Linux 설치.”](#) (19 페이지)에 설명된 대로 사용할 디스플레이 수 및 비디오 메모리 크기를 설정해야 합니다.

Horizon 연결 서버 7은 Windows 가상 시스템에서처럼 Linux 가상 시스템의 vRAM 설정을 자동으로 구성하지 않습니다. vSphere Client에서 vRAM 설정을 수동으로 구성해야 합니다.

Linux 가상 시스템이 권장되는 크기보다 더 작은 vRAM 크기로 구성되어 있으면 다음 문제가 발생할 수 있습니다.

- 초기 연결이 설정된 직후에 데스크톱 세션의 연결이 끊어질 수 있습니다.
- 자동 맞춤이 작동되지 않을 수 있습니다. 그러면 데스크톱이 화면의 작은 영역에 표시됩니다.

Linux 가상 시스템의 **디스플레이 수** 값이 실제로 필요한 수보다 작으면 데스크톱에 대해 한 대 이상의 모니터가 빈 화면을 표시합니다.



권장 설정으로 자동 맞춤을 수행하는 데 문제가 있는 경우 더 큰 vRAM 크기를 지정합니다. vSphere Client는 최대 128MB의 vRAM 크기를 허용합니다. 지정한 크기가 128MB를 초과하는 경우 vmx 구성 파일을 수동으로 수정해야 합니다. 다음 예에서는 vRAM 크기 256MB를 지정합니다.

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

## 구성 매개 변수

여러 대의 모니터에 Linux 원격 데스크톱을 표시하려면 가상 시스템에 대해 특정 구성 매개 변수를 설정해야 합니다. 가상 시스템에 대해 구성 매개 변수를 설정하는 일반적인 단계는 다음과 같습니다.

- 1 가상 시스템의 전원을 끕니다.
- 2 vSphere Web Client에서 가상 시스템을 마우스 오른쪽 버튼으로 클릭하고 **설정 편집**을 선택합니다.
- 3 **VM 옵션** 탭을 클릭하고 **고급**을 클릭합니다.
- 4 **구성 편집**을 클릭한 다음 **행 추가**를 클릭합니다.
- 5 구성 매개 변수 이름 및 값을 입력합니다.
- 6 변경 사항을 저장하려면 **확인**을 클릭합니다.

다음 구성 매개 변수를 설정해야 합니다.

- svga.autodetect를 [false]로 설정합니다.

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

- 디스플레이 모니터의 수 및 방향(가로 또는 세로)에 따라 svga.maxWidth 및 svga.maxHeight 값을 계산합니다. 일반 규칙은 svga.maxWidth 및 svga.maxHeight 값이 모든 디스플레이를 지원하기에 충분할 정도로 커야 한다는 것입니다. 예를 들어 최대 2560x1600의 해상도에서 4대의 디스플레이를 지원하려면 다음 값을 설정해야 합니다.

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

모니터가 여러 대인 경우에는 이러한 구성 매개 변수를 설정해야 합니다. 설정하지 않을 경우 다음과 같은 문제 하나 이상이 발생할 수 있습니다.

- 데스크톱이 일부 모니터에만 표시되고 다른 모니터에는 빈 화면이 표시됩니다.
- 키입력이 여러 번 표시됩니다.
- 데스크톱이 느려집니다.
- 데스크톱이 화면의 작은 영역에 표시됩니다.

## 4096x4096 모니터의 화면 크기 제한

RHEL 6.8/6.9/7.3, CentOS 6.8/6.9/7.3, Ubuntu 16.04, SLED 12 SP2 및 SLES 12 SP2의 경우 2D 및 vSGA의 최대 화면 크기는 4096x4096입니다. xrandr 명령을 실행할 경우 출력의 첫 번째 줄에는 maximum 4096x4096이 표시됩니다.

Ubuntu 14.04의 경우 공식 Ubuntu 저장소에서 최신 패치를 설치하면 제한 사항도 적용됩니다.

다중 모니터 연결을 위해서는 화면 크기가 4096x4096보다 커야 합니다. 이 제한을 우회하려면 다음 해결 방법 중 하나를 사용하십시오.

- 가상 시스템에 VMware Hardware 버전 11(HWv11) 이상을 사용해야 하는 경우 가상 시스템의 VMX 구성 파일에 다음 줄을 추가합니다.

```
mks.enable3d = TRUE
```

이 해결 방법을 사용하면 Linux 운영 체제는 Chrome과 같은 소프트웨어 애플리케이션에 3D 기능을 보고할 수 있습니다. 그러나 이 경우 Linux 시스템 성능에 영향을 줄 수 있습니다.

- 가상 시스템이 VMware Hardware 버전 10을 사용할 수 있으면 가상 시스템의 VMX 파일에 다음 줄을 사용하십시오.

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

이 해결 방법은 RHEL 6.8 및 CentOS 6.8에는 적용할 수 없습니다.

이 해결 방법이 Ubuntu 16.04 시스템에서 작동되게 하려면 공식 Ubuntu 저장소의 최신 패치를 설치해야 합니다.

### vCPU 및 메모리 설정

2D 또는 vSGA 데스크톱의 성능을 향상하려면 Linux 가상 시스템에 대해 더 많은 vCPU 및 가상 메모리를 설정합니다. 예를 들어 2개의 vCPU 및 2GB의 가상 메모리를 설정합니다.

4대의 모니터를 사용하는 경우처럼 여러 모니터를 위한 더 큰 화면이 필요한 경우 가상 시스템에 대해 4개의 vCPU 및 4GB의 가상 메모리를 설정합니다.

2D 또는 vSGA 데스크톱에서 비디오를 재생하려는 경우에는 가상 시스템에 대해 4개의 vCPU 및 4GB의 가상 메모리를 설정합니다.

### 3D 메모리 설정

vSGA 다중 모니터 환경에서 성능을 향상하려면 가상 시스템에 대한 **3D 메모리** 설정을 1GB 이상으로 설정합니다.

# 데스크톱 배포를 위해 Linux 가상 시스템 준비

# 2

Linux 데스크톱을 설정하려면 Linux 가상 시스템을 만들고 원격 데스크톱 배포를 위해 운영 체제를 준비해야 합니다.

이 장에서는 다음 주제에 대해 설명합니다.

- “가상 시스템 생성 및 Linux 설치,” (19 페이지)
- “원격 데스크톱 배포를 위한 Linux 시스템 준비,” (20 페이지)
- “Horizon Agent에 대한 종속성 패키지 설치,” (22 페이지)

## 가상 시스템 생성 및 Linux 설치

Horizon 7에 배포된 각 원격 데스크톱에 대한 새 가상 시스템을 vCenter Server에서 만듭니다. 가상 시스템에 Linux 배포를 설치해야 합니다.

### 필수 조건

- 배포가 Linux 데스크톱을 지원하기 위한 요구 사항을 충족하는지 확인하십시오. “[Horizon 7 for Linux에 대한 시스템 요구 사항](#),” (11 페이지)를 참조하십시오.
- vCenter Server에서 가상 시스템을 생성하고 게스트 운영 체제를 설치하는 단계를 숙지하십시오. Horizon 7에서 가상 데스크톱 설정 문서의 “가상 시스템 생성 및 준비”를 참조하십시오.
- 가상 시스템에서 사용할 모니터에 대해 권장되는 비디오 메모리(vRAM) 값을 숙지하십시오. “[Horizon 7 for Linux에 대한 시스템 요구 사항](#),” (11 페이지)를 참조하십시오.

### 프로시저

- 1 vSphere Web Client 또는 vSphere Client에서 새 가상 시스템을 만듭니다.

- 2 사용자 지정 구성 옵션을 구성합니다.
  - a 가상 시스템을 마우스 오른쪽 버튼으로 클릭한 후 **설정 편집**을 클릭합니다.
  - b vCPU 수와 vMemory 크기를 지정합니다.
 

권장 값을 보려면 Linux 배포에 대한 설치 설명서의 지침을 따르십시오.

예를 들어 Ubuntu 12.04에서는 vMemory 2048MB 및 vCPU 2개를 구성할 것을 권장합니다.
  - c **비디오 카드**를 선택하고 디스플레이 수 및 총 비디오 메모리(vRAM)를 지정하십시오.
 

VMware 드라이버를 사용하는 2D 또는 vSGA를 사용할 경우 가상 시스템용 vSphere Web Client에서 vRAM 크기를 설정합니다. NVIDIA 드라이버를 사용하는 vDGA 또는 NVIDIA GRID vGPU 시스템에서는 vRAM 크기가 영향을 미치지 않습니다.

권장되는 값을 보려면 Horizon 7 for Linux에 대한 시스템 요구 사항의 지침을 따르십시오. Video Memory Calculator를 사용하지 마십시오.
- 3 가상 시스템의 전원을 켜고 Linux 배포를 설치합니다.
- 4 특정 Linux 배포에 사용할 수 있도록 데스크톱 환경을 구성합니다.
 

자세한 내용은 [“Horizon 7 for Linux에 대한 시스템 요구 사항,”](#) (11 페이지)의 “데스크톱 환경” 섹션을 참조하십시오.
- 5 시스템 호스트 이름을 127.0.0.1로 확인할 수 있어야 합니다.

## 원격 데스크톱 배포를 위한 Linux 시스템 준비

Horizon 7 배포에서 데스크톱으로 사용하도록 Linux 시스템을 준비하려면 특정 작업을 수행해야 합니다.

Horizon 7에서 Linux 시스템을 관리할 수 있으려면 먼저 시스템에서 연결 서버와 통신할 수 있어야 합니다. Linux 시스템에서 FQDN(정규화된 도메인 이름)을 사용하여 연결 서버 인스턴스를 ping할 수 있도록 Linux 시스템에 네트워크를 구성해야 합니다.

OVT(Open VMware Tools)는 RHEL 7, CentOS 7, SLED 12 및 SLES 12 시스템에 미리 설치됩니다. 원격 데스크톱으로 사용하기 위해 이러한 시스템을 준비하는 경우 설치 관리자를 수동으로 실행하여 VMware Tools 설치 방법을 설명하는 다음 절차에서 1~5단계를 건너뛸 수 있습니다.

Ubuntu 16.04 시스템을 사용하는 경우 시스템에 OVT를 설치합니다. 원격 데스크톱으로 사용하기 위해 이 시스템을 준비하는 경우 다음 절차에서 1~5단계를 건너뛰고 다음 명령을 사용하여 Ubuntu 16.04 시스템에 OVT를 수동으로 설치할 수 있습니다.

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

### 필수 조건

- vCenter Server에서 새 VM(가상 시스템)이 생성되었으며 Linux 배포가 시스템에 설치되어 있는지 확인합니다.
- Linux VM에서 VMware Tools를 마운트하고 설치하는 단계를 숙지하십시오. vSphere 가상 시스템 관리 문서의 “Linux 가상 시스템에서 수동으로 VMware Tools 설치 또는 업그레이드”를 참조하십시오.
- DNS를 통해 확인할 수 있도록 Linux 시스템을 구성하는 단계를 숙지하십시오. 이러한 단계는 Linux 배포 및 릴리스마다 다릅니다. 지침을 보려면 사용 중인 Linux 배포 및 릴리스에 대한 설명서를 참조하십시오.

### 프로시저

- 1 vSphere Web Client 또는 vSphere Client에서 VMware Tools 가상 디스크를 VM에 마운트합니다.

- 2 VMware Tools 설치 관리자 파일, VMwareTools.x.x.x-xxxx.tar.gz를 마우스 오른쪽 버튼으로 클릭하고 **압축을 풀 위치**를 클릭한 후 Linux 배포 데스크톱을 선택합니다.

vmware-tools-distrib 폴더가 데스크톱에 압축이 풀립니다.

- 3 VM에서 루트로 로그인하고 터미널 창을 엽니다.
- 4 VMware Tools tar 설치 관리자 파일의 압축을 풉니다.

예:

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

- 5 설치 관리자를 실행하고 VMware Tools를 구성합니다.

명령은 Linux 배포마다 약간씩 다를 수 있습니다. 예:

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

일반적으로 vmware-config-tools.pl 구성 파일은 설치 관리자 파일의 실행이 끝나야 실행됩니다.

- 6 /etc/hosts 파일에서 Linux 시스템의 호스트 이름을 127.0.0.1에 매핑합니다.

RHEL, CentOS, SLES 및 SLED의 경우 호스트 이름이 자동으로 매핑되지 않으므로 이를 127.0.0.1에 수동으로 매핑해야 합니다. Ubuntu의 경우 매핑이 기본적으로 설정되어 있으므로 이 단계가 필요하지 않습니다. 복제 프로세스에서 이 매핑을 추가하므로 데스크톱을 대량 배포하는 경우에도 이 단계가 필요 없습니다.

---

**참고** Horizon Agent를 설치한 후에 Linux 시스템의 호스트 이름을 변경하는 경우 /etc/hosts 파일에서 새 호스트 이름을 127.0.0.1에 매핑해야 합니다. 그러지 않으면 이전 호스트 이름이 계속 사용됩니다.

---

- 7 RHEL 7 및 CentOS 7의 경우 virbr0를 사용하지 않도록 설정했는지 확인합니다.

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

- 8 DNS를 통해 포드의 View 연결 서버 인스턴스를 확인할 수 있어야 합니다.

- 9 기본 실행 수준이 5가 되도록 Linux 시스템을 구성합니다.

Linux 데스크톱이 작동하려면 실행 수준이 5여야 합니다.

- 10 OpenLDAP 서버를 사용하여 인증하도록 구성된 Ubuntu 시스템에서는 시스템에 정규화된 도메인 이름을 설정합니다.

이 단계에서는 View Administrator의 [세션] 페이지에 있는 사용자 필드에 정보가 올바르게 표시될 수 있도록 합니다. /etc/hosts 파일을 다음과 같이 편집합니다.

a # nano /etc/hosts

b 정규화된 도메인 이름을 추가합니다. 예: 127.0.0.1 hostname.domainname hostname.

c 종료하고 파일을 저장합니다.

- 11 SUSE의 경우 [DHCP를 통한 호스트 이름 변경]을 사용하지 않도록 설정합니다. 호스트 이름 또는 도메인 이름을 설정합니다.

a Yast에서 **네트워크 설정**을 클릭합니다.

b **호스트 이름/DNS** 탭을 클릭합니다.

c **DHCP를 통한 호스트 이름 변경**을 선택 취소합니다.

- d 호스트 이름과 도메인 이름을 입력합니다.
- e **확인**을 클릭합니다.

VMware Tools를 설치한 후에 Linux 커널을 업그레이드하면 VMware Tools가 실행되지 않을 수 있습니다. 이 문제를 해결하려면 <http://kb.vmware.com/kb/2050592>을 참조하십시오.

## Horizon Agent에 대한 종속성 패키지 설치

Horizon Agent for Linux에는 Linux 배포에 고유한 일부 종속성 패키지가 있습니다. Horizon Agent for Linux를 설치하기 전에 이러한 패키지를 설치해야 합니다.

### 필수 조건

vCenter Server에서 새 VM(가상 시스템)이 생성되었으며 Linux 배포가 시스템에 설치되어 있는지 확인합니다.

### 프로시저

- 1 기본적으로 설치 또는 업그레이드되지 않은 필수 패키지를 설치합니다. 요구 사항을 충족하지 않는 패키지가 있으면 설치 관리자가 설치를 중단합니다.

**표 2-1.** 필수 종속성 패키지

Linux 배포	패키지
SLED 11 SP3/SP4 xorg-x11-server를 7.4.27.111.1보다 높은 버 전으로 업그레이드	zypper install xorg-x11-server
SLES 12 SP1/SLED 12 SP1 SUSE 저장소에서 xf86- video-vmware를 13.0.2-3.2 보다 높은 버전으로 업그레이 드	<ol style="list-style-type: none"> <li>1 SUSE 저장소를 사용하도록 설정하려면 SUSE 12를 등록합니다. SUSEConnect -r 등록 코드 -e 이메일</li> <li>2 xf86-video-vmware 버전을 업데이트합니다. zypper install xf86-video-vmware</li> </ol>
SLES 12	<p>Horizon Agent를 설치하는 경우 SLES 12 Linux 데스크톱을 사용하려면 python-gobject2를 설치해야 합니다.</p> <ol style="list-style-type: none"> <li>1 SUSE 저장소를 사용하도록 설정하려면 SUSE 12를 등록합니다. SUSEConnect -r 등록 코드 -e 이메일</li> <li>2 python-gobject2를 설치합니다. zypper install python-gobject2</li> </ol>
Ubuntu 14.04 indicator-session을 <a href="https://launchpad.net/ubuntu/wily/amd64/indicator-session/12.10.5+15.04.20150327-0ubuntu1">https://launchpad.net/u buntu/wily/amd64/indic ator- session/12.10.5+15.04. 20150327-0ubuntu1</a> 에서 사용할 수 있는 12.10.5+15.04.201503 27로 업그레이드	<pre>wget http://launchpadlibrarian.net/201393830/indicator- session_12.10.5+15.04.20150327-0ubuntu1_amd64.deb sudo dpkg -i ./indicator-session_12.10.5+15.04.20150327-0ubuntu1_amd64.deb</pre>
Ubuntu 16.04	apt-get install python-dbus python-gobject

## 2 Horizon Agent에 선택 패키지를 설치합니다.

- 기본적으로 RHEL 또는 CentOS 6.7에는 교착 상태 문제를 야기할 수 있는 glibc-2.12-1.166.el6.x86\_64가 설치되어 있습니다. 결과적으로 데스크톱 연결이 끊어집니다. 이 문제를 해결하려면 glibc를 온라인 저장소에서 최신 버전으로 업그레이드해야 합니다.

```
sudo yum install glibc
```

- 다중 모니터가 있는 Ubuntu 14.04 데스크톱에서는 더 나은 성능을 위해 Compiz를 사용하지 않도록 설정하는 데 gnome-session-fallback이 필요합니다.

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

- 다중 모니터가 있는 Ubuntu 16.04 데스크톱에서는 더 나은 성능을 위해 Compiz를 사용하지 않도록 설정하는 데 gnome-session-flashback이 필요합니다.

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





# Linux 데스크톱의 Active Directory 통합 설정

# 3

View에서는 사용자 인증 및 관리를 위해 기존 Microsoft AD(Active Directory) 인프라를 사용합니다. 사용자가 Active Directory 사용자 계정을 사용하여 Linux 데스크톱에 로그인할 수 있도록 Linux 데스크톱을 Active Directory와 통합할 수 있습니다.

이 장에서는 다음 주제에 대해 설명합니다.

- “Linux와 Active Directory 통합,” (25 페이지)
- “Single Sign-On 및 스마트 카드 리디렉션 설정,” (26 페이지)

## Linux와 Active Directory 통합

Linux를 AD(Active Directory)에 통합하기 위한 많은 솔루션이 있으며 Horizon 7 for Linux 데스크톱은 사용되는 솔루션에 좌우되지 않습니다.

다음 솔루션은 Horizon 7 for Linux 데스크톱 환경에서 작동되는 것으로 알려져 있습니다.

- OpenLDAP 서버 패스스루 인증
- Microsoft Active Directory에 대한 SSSD(System Security Services Daemon) LDAP 인증
- Winbind 도메인 가입

개괄적으로 OpenLDAP 패스스루 인증 솔루션은 다음 단계에 따라 진행됩니다.

- 1 Active Directory에서 인증서 서비스를 설치하여 LDAPS(Lightweight Directory Access Protocol over SSL)를 사용하도록 설정합니다.
- 2 OpenLDAP 서버를 설정합니다.
- 3 Active Directory의 사용자 정보(암호 제외)를 OpenLDAP 서버와 동기화합니다.
- 4 Active Directory에 대해 암호 확인을 수행할 수 있는 saslauthd 등의 별도 프로세스에 암호 확인을 위임하도록 OpenLDAP 서버를 구성합니다.
- 5 LDAP 클라이언트를 사용해서 OpenLDAP 서버에서 사용자를 인증하도록 Linux 데스크톱을 구성합니다.

Microsoft Active Directory 솔루션에 대한 SSSD LDAP 인증은 다음 단계에 따라 진행됩니다.

- 1 Active Directory에서 인증서 서비스를 설치하여 LDAPS를 사용하도록 설정합니다.
- 2 Microsoft Active Directory에 대해 LDAP 인증을 직접 사용하도록 Linux 데스크톱의 SSSD를 구성합니다.

Winbind 도메인 가입 솔루션은 다음 단계에 따라 진행됩니다.

- 1 Linux 데스크톱에서 Winbind, Samba 및 Kerberos 패키지를 설치합니다.

## 2 Linux 데스크톱을 Microsoft Active Directory에 가입시킵니다.

LDAP 기반 솔루션을 사용하는 경우 템플릿 가상 시스템에서 구성을 수행해야 하며, 복제된 가상 시스템에서는 추가로 수행할 단계가 없습니다.

Winbind 도메인 가입 솔루션 또는 기타 Kerberos 인증 기반 솔루션을 사용하는 경우 템플릿 가상 시스템을 Active Directory에 가입시키고 복제된 가상 시스템을 Active Directory에 다시 가입시켜야 합니다. 예를 들어 다음 명령을 사용합니다.

```
sudo /usr/bin/net ads join -U <도메인 사용자>%<도메인 암호>
```

Winbind 솔루션의 경우 다음 옵션을 사용하여 복제된 가상 시스템에 대해 도메인 재가입 명령을 실행합니다.

- 각 가상 시스템에 대해 SSH 또는 vSphere PowerCLI와 같은 원격 연결을 수행하고 명령을 실행합니다. 스크립트에 대한 자세한 내용은 8장, “수동 데스크톱 풀의 Horizon 7 대량 배포,” (65 페이지)를 참조하십시오.
- 셸 스크립트에 이 명령을 포함하고 /etc/vmware/viewagent-custom.conf 파일에서 스크립트 경로를 Horizon Agent 옵션 RunOnceScript로 지정합니다. 자세한 내용은 “Linux 데스크톱의 구성 파일에서 옵션 설정,” (47 페이지)의 내용을 참조하십시오.

---

**참고** 간편한 배포를 위해 Microsoft Active Directory 솔루션에 대해 SSSD LDAP 인증을 사용합니다.

---

## Single Sign-On 및 스마트 카드 리디렉션 설정

Single Sign-On(SSO) 및 스마트 카드 리디렉션을 설정하려면 몇 가지 구성 단계를 수행해야 합니다.

### Single Sign-On

Horizon View Single Sign-On 모듈에서는 Linux의 PAM(Pluggable Authentication Modules)에 연결하며, Linux를 AD(Active Directory)에 통합하는 데 사용하는 방법에 종속되지 않습니다. Horizon View SSO는 Linux를 AD와 통합하는 OpenLDAP 및 Winbind 솔루션과 작동하는 것으로 알려져 있습니다.

기본적으로 SSO에서는 AD의 sAMAccountName 특성이 로그인 ID라고 가정합니다. OpenLDAP 또는 Winbind 솔루션을 사용하는 경우 SSO에 올바른 로그인 ID가 사용되도록 하려면 다음과 같은 구성 단계를 수행해야 합니다.

- OpenLDAP의 경우 sAMAccountName을 uid로 설정합니다.
- Winbind의 경우 구성 파일 /etc/samba/smb.conf에 다음 문을 추가합니다.

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

사용자가 로그인할 도메인 이름을 지정해야 하는 경우에는 Linux 데스크톱에서 SSUserFormat 옵션을 설정해야 합니다. 자세한 내용은 “Linux 데스크톱의 구성 파일에서 옵션 설정,” (47 페이지)의 내용을 참조하십시오. SSO는 항상 대문자로 된 짧은 도메인 이름을 사용한다는 점에 주의하십시오. 예를 들어, 도메인이 mydomain.com이면 SSO에서는 MYDOMAIN을 도메인 이름으로 사용합니다. 따라서 SSUserFormat 옵션을 설정할 때 MYDOMAIN을 지정해야 합니다. 짧은 도메인 이름과 긴 도메인 이름에는 다음 규칙이 적용됩니다.

- OpenLDAP의 경우는 대문자로 된 짧은 도메인 이름을 사용해야 합니다.
- Winbind는 긴 도메인 이름과 짧은 도메인 이름을 모두 지원합니다.

AD는 로그인 이름에서 특수 문자를 지원하지 않지만 Linux는 지원하지 않습니다. 그러므로 SSO를 설정할 때 로그인 이름에 특수 문자를 사용하지 마십시오.

AD에서 사용자의 UserPrincipalName(UPN) 특성과 sAMAccount 특성이 일치하지 않고 사용자가 UPN으로 로그인한 경우에는 SSO가 실패합니다. 해결 방법은 사용자가 sAMAccount에 저장된 이름을 사용하여 로그인하는 것입니다.

View에서는 사용자 이름의 대소문자를 구분할 필요가 없습니다. Linux 운영 체제에서 대소문자를 구분하지 않은 사용자 이름을 처리할 수 있는지 확인해야 합니다.

- Winbind의 경우는 기본적으로 사용자 이름의 대소문자를 구분하지 않습니다.
- OpenLDAP의 경우는 Ubuntu에서 NSCD를 사용하여 사용자를 인증하며 기본적으로 대소문자를 구분하지 않습니다. RHEL 및 CentOS에서는 SSSD를 사용하여 사용자를 인증하며 기본적으로 대소문자를 구분합니다. 설정을 변경하려면 `/etc/sss/sss.conf` 파일을 편집하여 `[domain/default]` 섹션에 다음 줄을 추가합니다.

```
case_sensitive = false
```

Ubuntu16.04 또는 14.04의 경우 GNOME Flashback(Metacity) 데스크톱 환경을 사용하도록 `/etc/vmware/viewagent-custom.conf` 파일에서 `UseGnomeFlashback=TRUE`를 구성합니다.

## 스마트 카드 리디렉션

스마트 카드 리디렉션을 설정하려면 먼저 Linux 디스트리뷰터와 스마트 카드 벤더의 지침에 따릅니다. 그런 다음 `pcsc-lite` 패키지를 1.7.4로 업데이트합니다. 예를 들면, 다음 명령을 실행합니다.

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

Winbind의 경우 구성 파일 `/etc/samba/smb.conf`에 다음 문을 추가합니다.

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

Horizon Agent를 설치할 때, 먼저 SELinux를 사용하지 않도록 설정하거나 SELinux에 대해 허용 모드를 사용해야 합니다. 구성 요소가 기본적으로 선택되지 않으므로 스마트 카드 리디렉션 구성 요소도 구체적으로 선택해야 합니다. 자세한 내용은 “[install\\_viewagent.sh 명령줄 옵션](#),” (42 페이지)의 내용을 참조하십시오.

스마트 카드 SSO는 Horizon View 7.0.1 이상에서 사용되도록 설정됩니다. 또한 가상 시스템에 스마트 카드 리디렉션 기능이 설치되어 있는 경우에는 vSphere 클라이언트의 USB 리디렉션이 스마트 카드에서 작동하지 않습니다.

스마트 카드 리디렉션에서는 하나의 스마트 카드 판독기만 지원합니다. 두 개 이상의 판독기가 클라이언트 디바이스에 연결된 경우는 이 기능이 작동하지 않습니다.

스마트 카드 리디렉션은 카드에서 하나의 인증서만 지원합니다. 카드에 두 개 이상의 인증서가 있는 경우에는 첫 번째 슬롯에 있는 인증서가 사용되고 다른 인증서는 무시됩니다. 이는 Linux의 제한 사항입니다.

---

#### 참고

- 스마트 카드는 다음 Winbind 값을 지원합니다. 그렇지 않은 경우 스마트 카드 SSO 및 수동 로그인에 실패합니다.

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

- Linux 데스크톱 스마트 카드 리디렉션에서 지원되는 Linux 클라이언트를 사용하여 PIV 카드로 브로커를 인증할 경우 SSL 오류를 방지하기 위해 `~/.vmware/view-preferences`에서 Linux 클라이언트에 대한 `view.sslProtocolString = "TLSv1.1"` 구성을 추가해야 합니다.
-

## Linux 데스크톱의 그래픽 설정

ESXi 호스트 또는 게스트 운영 체제의 NVIDIA 기능을 활용하도록 현재 지원되는 RHEL 배포를 구성할 수 있습니다.

### 3D 그래픽 설정을 위한 VM 클론 요구 사항

3D 그래픽을 설정하기 전에 다음과 같은 VM 클론 요구 사항을 고려해야 합니다.

- vGPU 및 vSGA의 경우 기본 VM에서 그래픽 설정을 완료합니다. VM을 복제합니다. 복제된 VM에 그래픽 설정이 적용되며 추가 설정은 필요하지 않습니다.
- vDGA의 경우 기본 VM에서 그래픽 설정을 완료합니다. VM을 복제합니다. 그러나 복제된 VM의 전원을 켜기 전에 복제된 VM에서 기존 NVIDIA 패스스루 PCI 디바이스를 제거하고 복제된 VM에 새 NVIDIA 패스스루 PCI 디바이스를 추가해야 합니다. NVIDIA 패스스루 PCI 디바이스는 VM 간에 공유할 수 없습니다. 각 VM에서는 전용 NVIDIA 패스스루 PCI 디바이스를 사용합니다.

이 장에서는 다음 주제에 대해 설명합니다.

- [“vGPU에 대해 지원되는 RHEL 배포 구성,”](#) (29 페이지)
- [“vDGA용 RHEL 6 구성,”](#) (34 페이지)
- [“vSGA용 RHEL 7 구성,”](#) (38 페이지)

## vGPU에 대해 지원되는 RHEL 배포 구성

지원되는 RHEL 배포가 ESXi 호스트에서 NVIDIA vGPU(공유 GPU 하드웨어 가속화) 기능을 활용하도록 설정할 수 있습니다.

ESXi 호스트 GPU 드라이버(.vib)와 일치하는 NVIDIA Linux VM 디스플레이 드라이버를 사용해야 합니다. 드라이버 패키지에 대한 내용은 NVIDIA 웹 사이트를 참조하십시오.

**중요** NVIDIA vGPU는 NVIDIA Maxwell M60 그래픽 카드 및 NVIDIA M6 그래픽 카드에서 지원됩니다. GRID K1이나 K2 등의 다른 NVIDIA 그래픽 카드에서는 이 기능이 작동하지 않습니다.



**주의** 시작하기 전에 Horizon Agent가 Linux 가상 시스템에 설치되어 있지 않은지 확인합니다. NVIDIA vGPU를 사용하도록 시스템을 구성하기 전에 Horizon Agent를 설치하면 xorg.conf 파일에서 필요한 구성 매개 변수를 덮어쓰며 NVIDIA vGPU가 작동하지 않습니다. NVIDIA vGPU 구성이 완료된 후에 Horizon Agent를 설치해야 합니다.

## ESXi 호스트에서 NVIDIA GRID vGPU 그래픽 카드용 VIB 설치

NVIDIA GRID 그래픽 카드용 VIB를 ESXi 6.0 U1 이상 호스트에 다운로드하고 설치해야 합니다.

NVIDIA에서는 이 절차에서 ESXi 호스트에 설치하는 vGPU Manager가 포함된 vGPU 소프트웨어 패키지와 이후 절차에서 Linux 가상 시스템에 설치하는 Linux 디스플레이 드라이버를 제공합니다.

**필수 조건**

- 작업 환경에 vSphere 6.0 U1 이상 릴리스가 설치되어 있는지 확인하십시오.
- ESXi 호스트에 NVIDIA Maxwell M60 GPU 또는 M6 GPU가 설치되어 있는지 확인하십시오.

**프로시저**

- 1 [NVIDIA 드라이버 다운로드](#) 사이트에서 NVIDIA GRID vGPU 그래픽 카드용 VIB를 다운로드합니다.

드롭다운 메뉴에서 적절한 VIB 버전을 선택합니다.

옵션	설명
제품 유형	<b>GRID</b>
제품 시리즈	<b>NVIDIA GRID vGPU</b> 를 선택합니다.
제품	ESXi 호스트에 설치되어 있는 버전(예: <b>GRID K2</b> )을 선택합니다.
운영 체제	VMware vSphere ESXi 버전을 선택합니다.

- 2 vGPU 소프트웨어 패키지 .zip 파일을 압축 해제합니다.
- 3 vGPU Manager 폴더를 ESXi 6.0 U1 호스트에 업로드합니다.

**참고** 이후 절차에서 Linux 가상 시스템에 Linux 디스플레이 드라이버를 설치합니다.

- 4 ESXi 호스트에 있는 모든 가상 시스템을 일시 중단하거나 전원을 끕니다.
- 5 SSH를 사용하여 ESXi 호스트에 연결합니다.
- 6 xorg 서비스를 중지합니다.

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

- 7 NVIDIA VIB를 설치합니다.

예:

```
# 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 ESXi 호스트를 재부팅하거나 업데이트합니다.

- ◆ 설치된 ESXi 호스트의 경우는 호스트를 재부팅합니다.
- ◆ 상태 비저장 ESXi 호스트의 경우, 다음 단계를 수행하여 호스트를 업데이트합니다. (설치된 호스트에서도 이 단계를 사용할 수 있습니다.)

```
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 호스트가 다시 시작된 후에 xorg 서비스가 실행 중인지 확인합니다.

## Linux 가상 시스템에서 vGPU용으로 공유 PCI 디바이스 구성

NVIDIA vGPU를 사용하려면 Linux 가상 시스템에 대한 공유 PCI 디바이스를 구성해야 합니다.

### 필수 조건

- Linux 가상 시스템을 데스크톱으로 사용할 준비가 되어 있는지 확인합니다. 자세한 내용은 “[가상 시스템 생성 및 Linux 설치](#),” (19 페이지) 및 “[원격 데스크톱 배포를 위한 Linux 시스템 준비](#),” (20 페이지)에 나와 있습니다.
- Linux 가상 시스템에 Horizon Agent가 설치되어 있지 않은지 확인합니다.
- NVIDIA VIB가 ESXi 호스트에 설치되어 있는지 확인합니다. “[ESXi 호스트에서 NVIDIA GRID vGPU 그래픽 카드용 VIB 설치](#),” (29 페이지)의 내용을 참조하십시오.
- **GPU 프로파일** 설정에서 선택되어 있으며 NVIDIA vGPU에서 사용할 수 있는 가상 GPU 유형을 숙지합니다. 가상 GPU 유형에서는 ESXi 호스트에 설치된 물리적 GPU에 다양한 기능을 제공합니다. “[NVIDIA 가상 GPU 유형](#),” (31 페이지)의 내용을 참조하십시오.

### 프로시저

- 1 가상 시스템의 전원을 끕니다.
- 2 vSphere Web Client에서 가상 시스템을 선택하고 **VM 하드웨어** 탭에서 **설정 편집**을 클릭합니다.
- 3 **새 디바이스** 메뉴에서 **공유 PCI 디바이스**를 선택합니다.
- 4 **추가**를 클릭하고 드롭다운 메뉴에서 **NVIDIA GRID vGPU**를 선택합니다.
- 5 **GPU 프로파일** 설정의 드롭다운 메뉴에서 가상 GPU 유형을 선택합니다.
- 6 **모든 메모리 예약**을 클릭하고 **확인**을 클릭합니다.

GPU에서 NVIDIA GRID vGPU를 지원할 수 있도록 하려면 모든 가상 시스템 메모리를 예약해야 합니다.

- 7 가상 시스템의 전원을 켭니다.

## NVIDIA 가상 GPU 유형

vSphere Web Client의 [가상 하드웨어] 페이지에 있는 **GPU 프로파일** 설정을 사용하면 ESXi 호스트의 물리적 NVIDIA GPU에 특정 기능을 제공하는 가상 GPU 유형을 선택할 수 있습니다.

Linux 가상 시스템에서는 NVIDIA GRID vGPU가 NVIDIA Maxwell M60 GPU 또는 NVIDIA M6 GPU에서 지원됩니다.

**표 4-1.** Linux 가상 시스템에서 NVIDIA GRID M60 vGPU에 사용할 수 있는 가상 GPU 유형

가상 GPU 유형	물리적 보드	물리적 GPU	가상 GPU 당 FB	디스플레이 헤드	최대 해상도	물리적 GPU당 최대 가상 GPU 수	물리적 보드 당 최대 가상 GPU 수
GRID M60-0q	GRID M60	2	512M	2	2560x1600	16	32
GRID M60-1q	GRID M60	2	1G	2	2560x1600	8	16
GRID M60-2q	GRID M60	2	2G	4	2560x1600	4	8

**표 4-1.** Linux 가상 시스템에서 NVIDIA GRID M60 vGPU에 사용할 수 있는 가상 GPU 유형 (계속)

가상 GPU 유형	물리적 보드	물리적 GPU	가상 GPU 당 FB	디스플레이 헤드	최대 해상도	물리적 GPU당 최대 가상 GPU 수	물리적 보드 당 최대 가상 GPU 수
GRID M60-4q	GRID M60	2	4G	4	3840x2160	2	4
GRID M60-8q	GRID M60	2	8G	4	3840x2160	1	2

**표 4-2.** Linux 가상 시스템에서 NVIDIA GRID M6 vGPU에 사용할 수 있는 가상 GPU 유형

가상 GPU 유형	물리적 보드	물리적 GPU	가상 GPU 당 FB	디스플레이 헤드	최대 해상도	물리적 GPU당 최대 가상 GPU 수	물리적 보드 당 최대 가상 GPU 수
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

## NVIDIA GRID vGPU 디스플레이 드라이버 설치

NVIDIA GRID vGPU 디스플레이 드라이버를 설치하려면 기본 NVIDIA 드라이버를 사용하지 않도록 설정하고, NVIDIA 디스플레이 드라이버를 다운로드하고, 가상 시스템에서 PCI 디바이스를 구성해야 합니다.

### 필수 조건

- NVIDIA 다운로드 사이트에서 vGPU 소프트웨어 패키지를 다운로드하고, 패키지를 압축 해제하고, Linux 디스플레이 드라이버(패키지 구성 요소)가 준비되었는지 확인합니다. [“ESXi 호스트에서 NVIDIA GRID vGPU 그래픽 카드용 VIB 설치,”](#) (29 페이지)를 참조하십시오.

공유 PCI 디바이스가 가상 시스템에 추가되었는지 확인합니다. [“Linux 가상 시스템에서 vGPU용으로 공유 PCI 디바이스 구성,”](#) (31 페이지)의 내용을 참조하십시오.

### 프로시저

1 기본 NVIDIA Nouveau 드라이버를 비활성화하고 블랙리스트에 추가합니다.

a grub.conf 또는 grub 파일을 편집합니다.

RHEL 6의 경우 이 파일은 /boot/grub/grub.conf입니다. RHEL 7의 경우 이 파일은 /etc/default/grub입니다.

RHEL 버전	명령
6	sudo vi /boot/grub/grub.conf
7	sudo vi /etc/default/grub

b 커널 옵션 끝에 rdblacklist=nouveau 줄을 추가합니다.



- c blacklist.conf 파일을 편집합니다.
 

```
sudo vi /etc/modprobe.d/blacklist.conf
```
- d blacklist.conf 파일의 아무 위치에나 다음 줄을 추가합니다.
 

```
blacklist nouveau
```
- 2 가상 시스템을 다시 시작합니다.
 

디스플레이의 형태 및 느낌이 바뀌었습니다.
- 3 (선택 사항) Nouveau 드라이버가 비활성화되었는지 확인하십시오.
 

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

grep 검색을 수행해도 결과가 반환되지 않으면 Nouveau 드라이버가 비활성화된 것입니다.
- 4 NVIDIA Linux 디스플레이 드라이버를 가상 시스템으로 복사합니다.
- 5 가상 시스템에 대한 원격 터미널을 열거나 Ctrl+Alt+F2를 입력하여 텍스트 콘솔로 전환하고 루트 권한으로 로그인한 후 init 3 명령을 실행하여 X Windows를 비활성화합니다.
- 6 NVIDIA 드라이버에 필요한 추가 구성 요소를 설치합니다.
 

```
sudo yum install gcc-c++
sudo yum install kernel-devel-$(uname -r)
sudo yum install kernel-headers-$(uname -r)
```
- 7 NVIDIA GRID vGPU 드라이버 패키지에 실행 가능한 플래그를 추가합니다.
 

```
chmod +x NVIDIA-Linux-x86_64-버전-grid.run
```
- 8 NVIDIA GRID vGPU 설치 관리자를 시작합니다.
 

```
sudo ./NVIDIA-Linux-x86_64-버전-grid.run
```
- 9 NVIDIA 소프트웨어 라이선스 계약에 동의하고 예를 선택하여 X 구성 설정을 자동으로 업데이트합니다.

### 후속 작업

Linux 가상 시스템에 Horizon Agent를 설치합니다. [“Linux 가상 시스템에 Horizon Agent 설치,”](#) (41 페이지)를 참조하십시오.

구성된 Linux 가상 시스템을 포함하는 데스크톱 풀을 생성합니다. [“Linux용 수동 데스크톱 풀 생성,”](#) (57 페이지)를 참조하십시오.

## NVIDIA 디스플레이 드라이버의 설치 여부 확인

View 데스크톱 세션에 NVIDIA 드라이버 출력을 표시하여 RHEL 6 가상 시스템에 NVIDIA 디스플레이 드라이버가 설치되어 있는지 확인할 수 있습니다.

### 필수 조건

- NVIDIA 디스플레이 드라이버를 설치했는지 확인합니다.
- Linux 가상 시스템에 Horizon Agent가 설치되어 있는지 확인합니다. [“Linux 가상 시스템에 Horizon Agent 설치,”](#) (41 페이지)를 참조하십시오.
- 데스크톱 풀에 Linux 가상 시스템이 배포되어 있는지 확인합니다. [“Linux용 수동 데스크톱 풀 생성,”](#) (57 페이지)를 참조하십시오.

**프로시저**

- 1 Linux 가상 시스템을 다시 시작합니다.  
Horizon Agent 시작 스크립트는 X 서버 및 디스플레이 토폴로지를 초기화합니다.  
vSphere 콘솔에서 가상 시스템 디스플레이를 더 이상 볼 수 없습니다.
- 2 Horizon Client에서 Linux 데스크톱에 연결합니다.
- 3 Linux 데스크톱 세션에서 NVIDIA 디스플레이 드라이버가 설치되어 있는지 확인합니다.  
터미널 창을 열고 `glxinfo | grep NVIDIA` 명령을 실행합니다.  
NVIDIA 드라이버 출력이 표시됩니다. 예:

```
[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
```

사용자는 원격 데스크톱의 NVIDIA 그래픽 기능에 액세스할 수 있습니다.

NVIDIA 디스플레이 드라이버의 설치를 확인한 후에 설치 작업이 올바르게 실행되도록 다음 작업을 수행합니다.

- Linux 커널을 업그레이드하면 Horizon Agent가 View 연결 서버와 통신하지 못할 수도 있습니다. 이 문제를 해결하려면 NVIDIA 드라이버를 다시 설치하십시오.
- Linux VM에서 NVIDIA GRID 라이선싱을 설정합니다. 자세한 내용은 NVIDIA 설명서를 참조하십시오. 라이선싱을 설정하지 않으면 Linux 데스크톱이 올바르게 작동하지 않습니다. 예를 들어, 자동 맞춤이 작동하지 않습니다.

**vDGA용 RHEL 6 구성**

Horizon 7 for Linux 데스크톱이 ESXi 호스트에서 vDGA 기능을 활용할 수 있도록 RHEL 6 게스트 운영 체제를 설정할 수 있습니다.



**주의** 시작하기 전에 Horizon Agent가 Linux 가상 시스템에 설치되어 있지 않은지 확인합니다. vDGA를 사용하도록 시스템을 구성하기 전에 Horizon Agent를 설치하면 `xorg.conf` 파일에서 필요한 구성 매개 변수를 덮어쓰며 vDGA가 작동하지 않습니다. vDGA 구성이 완료된 후에 Horizon Agent를 설치해야 합니다.

**호스트에서 NVIDIA GRID용 DirectPath I/O 사용**

vDGA를 사용하도록 Linux 가상 시스템을 구성하기 전에 ESXi 호스트의 DirectPath I/O 패스스루를 위해 NVIDIA GRID GPU PCI 디바이스를 사용할 수 있도록 해야 합니다.

**필수 조건**

- 작업 환경에 vSphere 6.0 이상 릴리스가 설치되어 있는지 확인하십시오.
- ESXi 호스트에 NVIDIA GRID K1 또는 K2 그래픽 카드가 설치되어 있는지 확인하십시오.

**프로시저**

- 1 vSphere Web Client에서 ESXi 호스트로 이동합니다.
- 2 **관리** 탭을 클릭하고 **설정**을 클릭합니다.
- 3 하드웨어 섹션에서 **PCI 디바이스**를 클릭합니다.

- 4 NVIDIA GRID GPU에 대해 DirectPath I/O 패스스루를 활성화하려면 **편집**을 클릭하십시오.

아이콘	설명
녹색 아이콘	PCI 디바이스가 활성 상태이며 사용하도록 설정할 수 있습니다.
주황색 아이콘	디바이스 상태가 변경되었으며, 디바이스를 사용하려면 먼저 호스트를 재부팅해야 합니다.

- 5 NVIDIA GRID GPU를 선택하고 **확인**을 클릭합니다.  
PCI 디바이스가 VM에서 사용할 수 있는 DirectPath I/O PCI 디바이스 테이블에 추가됩니다.
- 6 PCI 디바이스를 Linux 가상 시스템에서 사용할 수 있도록 하려면 호스트를 재부팅하십시오.

## RHEL 6 가상 시스템에 vDGA 패스스루 디바이스 추가

vDGA를 사용하도록 RHEL 6 가상 시스템을 구성하려면 해당 가상 시스템에 PCI 디바이스를 추가해야 합니다. 이 단계를 수행하면 가상 시스템에서의 사용을 위해 ESXi 호스트의 물리적 디바이스를 패스스루할 수 있습니다.

### 필수 조건

- Linux 가상 시스템을 데스크톱으로 사용할 준비가 되어 있는지 확인합니다. 자세한 내용은 “[가상 시스템 생성 및 Linux 설치](#),” (19 페이지) 및 “[원격 데스크톱 배포를 위한 Linux 시스템 준비](#),” (20 페이지)의 내용을 참조하십시오.
- Linux 가상 시스템에 Horizon Agent가 설치되어 있지 않은지 확인합니다.
- NVIDIA GRID GPU PCI 디바이스를 호스트의 DirectPath I/O 패스스루에 사용할 수 있는지 확인합니다. “[호스트에서 NVIDIA GRID용 DirectPath I/O 사용](#),” (34 페이지)를 참조하십시오.

### 프로시저

- 1 RHEL 6 게스트 운영 체제에 sudo 권한을 가지도록 구성된 로컬 사용자로 로그인합니다.
  - 2 vSphere Web Client에서 가상 시스템을 선택하고 **VM 하드웨어** 탭에서 **설정 편집**을 클릭합니다.
  - 3 **새 디바이스** 메뉴에서 **PCI 디바이스**를 선택합니다.
  - 4 **추가**를 클릭하고 드롭다운 메뉴에서 PCI 디바이스를 선택합니다.
  - 5 **모든 메모리 예약**을 클릭하고 **확인**을 클릭합니다.
- GPU에서 vDGA를 지원할 수 있도록 하려면 모든 가상 시스템 메모리를 예약해야 합니다.
- 6 가상 시스템 전원을 켜고 vSphere 콘솔을 열어 시스템에 연결합니다.
  - 7 NVIDIA GRID 디바이스가 가상 시스템으로 패스스루되는지 확인합니다.

터미널 창을 열고 다음 명령을 실행합니다.

```
lspci | grep NVIDIA
```

XX:00.0 VGA 호환 컨트롤러가 표시됩니다. 예:

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

## vDGA용 NVIDIA 디스플레이 드라이버 설치

vDGA용 NVIDIA 디스플레이 드라이버를 설치하려면 기본 NVIDIA 드라이버를 사용하지 않도록 설정하고, NVIDIA 디스플레이 드라이버를 다운로드하고, 가상 시스템에서 PCI 디바이스를 구성해야 합니다.

### 필수 조건

- PCI 디바이스가 RHEL 6 가상 시스템에 추가되었는지 확인하십시오. “[RHEL 6 가상 시스템에 vDGA 패스스루 디바이스 추가](#),” (35 페이지)를 참조하십시오.

**프로시저**

- 1 기본 NVIDIA Nouveau 드라이버를 비활성화하고 블랙리스트에 추가합니다.
  - a grub.conf 파일을 편집합니다.  
RHEL 6의 경우 이 파일은 /boot/grub/grub.conf입니다.

RHEL 버전	명령
6	sudo vi /boot/grub/grub.conf

- b 커널 옵션 끝에 rdblacklist=nouveau 줄을 추가합니다.
  - c blacklist.conf 파일을 편집합니다.  
sudo vi /etc/modprobe.d/blacklist.conf
  - d blacklist.conf 파일의 아무 위치에나 다음 줄을 추가합니다.  
blacklist nouveau

- 2 가상 시스템을 다시 시작합니다.  
디스플레이의 형태 및 느낌이 바뀌었습니다.
- 3 (선택 사항) Nouveau 드라이버가 비활성화되었는지 확인하십시오.  
/sbin/lsmmod | grep nouveau  
grep 검색을 수행해도 결과가 반환되지 않으면 Nouveau 드라이버가 비활성화된 것입니다.
- 4 [NVIDIA 드라이버 다운로드](#) 사이트에서 NVIDIA 드라이버를 다운로드합니다.  
NVIDIA 드롭다운 메뉴에서 적절한 드라이버 버전을 선택합니다.

옵션	설명
제품 유형	GRID
제품 시리즈	GRID 시리즈
제품	ESXi 호스트에 설치되어 있는 버전(예: <b>GRID K2</b> )을 선택합니다.
운영 체제	Linux 64비트 또는 Linux 32비트

- 5 가상 시스템에 대한 원격 터미널을 열거나 Ctrl+Alt+F2를 입력하여 텍스트 콘솔로 전환하고 루트 권한으로 로그인한 후 init 3 명령을 실행하여 X Windows를 비활성화합니다.
- 6 NVIDIA 드라이버에 필요한 추가 구성 요소를 설치합니다.  
sudo yum install gcc-c++  
sudo yum install kernel-devel-\$(uname -r)  
sudo yum install kernel-headers-\$(uname -r)
- 7 vDGA용 NVIDIA 드라이버 패키지에 실행 가능한 플래그를 추가합니다.  
chmod +x NVIDIA-Linux-x86\_64-버전.run
- 8 NVIDIA 설치 관리자를 시작합니다.  
sudo ./NVIDIA-Linux-x86\_64-버전.run
- 9 NVIDIA 소프트웨어 라이선스 계약에 동의하고 **예**를 선택하여 X 구성 설정을 자동으로 업데이트합니다.

## 후속 작업

Linux 가상 시스템에 Horizon Agent를 설치합니다. “Linux 가상 시스템에 Horizon Agent 설치,” (41 페이지)를 참조하십시오.

구성된 Linux 가상 시스템을 포함하는 데스크톱 풀을 생성합니다. “Linux용 수동 데스크톱 풀 생성,” (57 페이지)를 참조하십시오.

## NVIDIA 디스플레이 드라이버의 설치 여부 확인

View 데스크톱 세션에 NVIDIA 드라이버 출력을 표시하여 RHEL 6 가상 시스템에 NVIDIA 디스플레이 드라이버가 설치되어 있는지 확인할 수 있습니다.

### 필수 조건

- NVIDIA 디스플레이 드라이버를 설치했는지 확인합니다.
- Linux 가상 시스템에 Horizon Agent가 설치되어 있는지 확인합니다. “Linux 가상 시스템에 Horizon Agent 설치,” (41 페이지)를 참조하십시오.
- 데스크톱 풀에 Linux 가상 시스템이 배포되어 있는지 확인합니다. “Linux용 수동 데스크톱 풀 생성,” (57 페이지)를 참조하십시오.

### 프로시저

- 1 Linux 가상 시스템을 다시 시작합니다.

Horizon Agent 시작 스크립트는 X 서버 및 디스플레이 토폴로지를 초기화합니다.

vSphere 콘솔에서 가상 시스템 디스플레이를 더 이상 볼 수 없습니다.

- 2 Horizon Client에서 Linux 데스크톱에 연결합니다.

- 3 Linux 데스크톱 세션에서 NVIDIA 디스플레이 드라이버가 설치되어 있는지 확인합니다.

터미널 창을 열고 `glxinfo | grep NVIDIA` 명령을 실행합니다.

NVIDIA 드라이버 출력이 표시됩니다. 예:

```
[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
```

사용자는 원격 데스크톱의 NVIDIA 그래픽 기능에 액세스할 수 있습니다.

NVIDIA 디스플레이 드라이버의 설치를 확인한 후에 설치 작업이 올바르게 실행되도록 다음 작업을 수행합니다.

- Linux 커널을 업그레이드하면 Horizon Agent가 View 연결 서버와 통신하지 못할 수도 있습니다. 이 문제를 해결하려면 NVIDIA 드라이버를 다시 설치하십시오.
- Linux VM에서 NVIDIA GRID 라이선싱을 설정합니다. 자세한 내용은 NVIDIA 설명서를 참조하십시오. 라이선싱을 설정하지 않으면 Linux 데스크톱이 올바르게 작동하지 않습니다. 예를 들어, 자동 맞춤이 작동하지 않습니다.

## vSGA용 RHEL 7 구성

Horizon 7 for Linux 데스크톱에서 vSGA 기능을 활용할 수 있도록 RHEL 7 게스트 운영 체제를 설정할 수 있습니다.

### ESXi 호스트에서 vSGA에 대한 NVIDIA 그래픽 카드용 VIB 설치

NVIDIA GRID 그래픽 카드용 VIB를 ESXi 6.0 U1 이상 호스트에 다운로드하고 설치해야 합니다.

NVIDIA는 vSGA에 대한 VMware vSphere ESXi 드라이버를 제공합니다. vSGA의 경우, Linux 가상 시스템에 NVIDIA 디스플레이 드라이버가 설치되어 있지 않습니다.

#### 필수 조건

- 작업 환경에 vSphere 6.0 U1 이상 릴리스가 설치되어 있는지 확인하십시오.
- 작업 환경에 NVIDIA 드라이버가 설치되어 있는지 확인하십시오.
- ESXi 호스트에 NVIDIA GRID K1 또는 K2 GPU가 설치되어 있는지 확인하십시오.

#### 프로시저

- 1 [NVIDIA 드라이버 다운로드](#) 사이트에서 NVIDIA GRID vGPU 그래픽 카드용 VIB를 다운로드합니다.

드롭다운 메뉴에서 적절한 VIB 버전을 선택합니다.

옵션	설명
제품 유형	GRID
제품 시리즈	GRID 시리즈를 선택합니다.
제품	ESXi 호스트에 설치되어 있는 버전(예: <b>GRID K2</b> )을 선택합니다.
운영 체제	VMware vSphere ESXi 버전을 선택합니다.

- 2 vSGA용 VMware vSphere ESXi 드라이버를 ESXi 6.0 U1 호스트에 업로드합니다.
- 3 ESXi 호스트에 있는 모든 가상 시스템을 일시 중단하거나 전원을 끕니다.
- 4 SSH를 사용하여 ESXi 호스트에 연결합니다.
- 5 xorg 서비스를 중지합니다.
 

```
# /etc/init.d/xorg stop
```
- 6 NVIDIA VIB를 설치합니다.
 

예:

```
# 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 GPU 할당에 사용되는 xorg를 다시 시작합니다.
 

```
# /etc/init.d/xorg start
```
- 8 ESXi 호스트를 재부팅합니다.
- 9 호스트가 다시 시작된 후에 xorg 서비스가 실행 중인지 확인합니다.

## Linux 가상 시스템에서 vSGA에 대한 3D 기능 구성

vSGA를 사용하도록 RHEL 7 가상 시스템을 구성하려면 vSphere Web Client에서 해당 가상 시스템의 비디오 카드에 대한 3D 설정을 구성해야 합니다.

### 필수 조건

- Linux 가상 시스템을 데스크톱으로 사용할 준비가 되었는지, Horizon Agent가 설치되어 있는지, 시스템이 데스크톱 풀에 배포되었는지 확인합니다.
- NVIDIA VIB가 ESXi 호스트에 설치되어 있는지 확인합니다. [“ESXi 호스트에서 vSGA에 대한 NVIDIA 그래픽 카드용 VIB 설치,”](#) (38 페이지)를 참조하십시오.

### 프로시저

- 1 가상 시스템의 전원을 끕니다.
- 2 vSphere Web Client에서 가상 시스템을 선택하고 **VM 하드웨어** 탭에서 **설정 편집**을 클릭합니다.
- 3 [가상 하드웨어] 탭에서 **비디오 카드**를 클릭하여 메뉴 설정을 확장합니다.
- 4 **총 비디오 메모리**를 128MB로 설정합니다.
- 5 **3D 그래픽**에서 **3D 지원 사용**을 선택합니다.
- 6 **3D 렌더러**의 드롭다운 메뉴에서 **하드웨어**를 선택합니다.
- 7 **3D 메모리**의 경우 애플리케이션 요구 사항에 따라 적절한 값을 선택합니다.  
사용자가 3개를 넘는 모니터에 연결할 경우에는 이 값을 1024MB 이상으로 설정합니다.
- 8 **확인**을 클릭합니다.
- 9 가상 시스템의 전원을 켭니다.

### 후속 작업

Linux 가상 시스템에서 vSGA가 실행 중인지 확인합니다.

그런 다음 Linux 가상 시스템에 Horizon Agent를 설치합니다. [“Linux 가상 시스템에 Horizon Agent 설치,”](#) (41 페이지)를 참조하십시오.

## Linux 가상 시스템에서 vSGA가 실행 중인지 확인

가상 시스템 로그 파일과 게스트 운영 체제를 확인하면 vSGA가 RHEL 7 가상 시스템에서 실행 중인지 확인할 수 있습니다.

### 프로시저

- 1 가상 시스템의 vmware.log 파일을 엽니다.

지원되는 GPU와 NVIDIA VIB가 올바르게 설치되어 있으면 로그 파일에 다음 예와 같은 줄이 표시 됩니다.

```
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"
```

지원되는 GPU 및 NVIDIA VIB가 올바르게 설치되어 있지 않은 경우에는 가상 시스템에서 소프트웨어 렌더러를 사용합니다. 다음 예에서는 vmware.log 파일에 이러한 줄이 표시됩니다.

```
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 가상 시스템에 있는 게스트 운영 체제에서 다음 명령을 입력합니다.

```
glxinfo|grep Gallium
```

vSGA가 작동하는 경우에는 명령에서 다음 텍스트를 반환합니다.

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

vSGA가 올바르게 작동하지 않는 경우에는 명령에서 다음 텍스트를 반환합니다.

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

### 후속 작업

Linux 가상 시스템에 Horizon Agent를 설치합니다. [“Linux 가상 시스템에 Horizon Agent 설치,”](#) (41 페이지)를 참조하십시오.



## Horizon Agent 설치

View 연결 서버에서 데스크톱과 통신하고 데스크톱을 관리할 수 있도록 Horizon Agent를 Linux 데스크톱에 설치해야 합니다.

이 장에서는 다음 주제에 대해 설명합니다.

- “Linux 가상 시스템에 Horizon Agent 설치,” (41 페이지)
- “Linux Agent용 인증서 구성,” (43 페이지)
- “Linux 가상 시스템에서 Horizon Agent 업그레이드,” (44 페이지)
- “Horizon 7 for Linux 시스템 제거,” (46 페이지)

### Linux 가상 시스템에 Horizon Agent 설치

시스템을 원격 데스크톱으로 배포하려면 먼저 Linux 가상 시스템에 Horizon Agent를 설치해야 합니다.

Horizon 7.0.1 릴리스부터 Horizon Agent for Linux는 vCenter에서 관리하는 가상 시스템을 사용합니다. 관리되는 가상 시스템은 다음과 같은 향상된 기능을 제공합니다.

- vCenter는 Linux 데스크톱 배포에 대한 필수 요구 사항입니다.
- Linux에 Horizon Agent를 설치하는 데에는 등록이 필요하지 않습니다.
- 많은 수의 Linux 데스크톱 배포를 수행하는 경우 기본 가상 시스템에서 Horizon Agent를 설치할 수 있습니다.



**주의** NVIDIA GRID vGPU, vDGA 또는 vSGA를 사용하려는 경우에는 Horizon Agent를 설치하기 전에 Linux 가상 시스템에서 이러한 3D 기능을 구성해야 합니다. Horizon Agent를 먼저 설치하면 xorg.conf 파일에서 필수 매개 변수를 덮어쓰며, 3D 그래픽 기능이 작동하지 않습니다.

“vGPU에 대해 지원되는 RHEL 배포 구성,” (29 페이지), “vDGA용 RHEL 6 구성,” (34 페이지) 또는 “vSGA용 RHEL 7 구성,” (38 페이지)를 참조하십시오. 3D 그래픽 구성이 완료된 후에 Horizon Agent를 설치합니다.

2D 그래픽 구성의 경우, “원격 데스크톱 배포를 위한 Linux 시스템 준비,” (20 페이지)에서 단계를 완료한 후에 Horizon Agent를 설치할 수 있습니다.

#### 필수 조건

- Linux 게스트 운영 체제를 데스크톱에서 사용할 준비가 되었는지 확인하십시오. “원격 데스크톱 배포를 위한 Linux 시스템 준비,” (20 페이지)를 참조하십시오.
- Linux용 Horizon Agent 설치 관리자 스크립트를 숙지합니다. “install\_viewagent.sh 명령줄 옵션,” (42 페이지)를 참조하십시오.

## 프로시저

- 1 VMware 다운로드 사이트(<https://my.vmware.com/web/vmware/downloads>)에서 Linux용 Horizon Agent 설치 관리자 파일을 다운로드합니다.  
 [Desktop & End-User Computing]에서 Linux용 Horizon Agent 설치 관리자가 포함되어 있는 VMware Horizon 7 다운로드를 선택합니다.  
 설치 관리자 파일 이름은 64비트 Linux의 경우 VMware-horizonagent-linux-x86\_64-y.y.y-xxxxxxx.tar.gz입니다. 여기서 y.y.y는 버전 번호이고 xxxxxxx는 빌드 번호입니다.
- 2 게스트 운영 체제에서 Linux 배포용 tarball의 압축을 풉니다.  
 예:  

```
tar -xzf <View Agent tar ball>
```
- 3 tar ball 폴더로 이동합니다.
- 4 고급 사용자로 install\_viewagent.sh 스크립트를 실행합니다.  
 명령줄 옵션의 목록은 “[install\\_viewagent.sh 명령줄 옵션](#),” (42 페이지)을 참조하십시오.  
 예:  

```
sudo ./install_viewagent.sh
```
- 5 -A 옵션을 지정하지 않고 install\_viewagent.sh를 실행한 경우는 **Yes**를 입력하여 EULA를 수락합니다.  
 설치 관리자는 사용자가 EULA에 동의해야만 계속 실행됩니다.
- 6 변경 내용을 적용하려면 Linux를 재부팅합니다.  
 설치 후에 viewagent 서비스가 시작됩니다. sudo service viewagent status를 사용하여 서비스가 시작되었는지 확인합니다.

## 후속 작업

데스크톱 풀에서 가상 시스템을 배포합니다. “[Linux용 수동 데스크톱 풀 생성](#),” (57 페이지)를 참조하십시오.

## install\_viewagent.sh 명령줄 옵션

install\_viewagent.sh 스크립트는 Linux 게스트 운영 체제에 Horizon Agent를 설치합니다.

gnome 데스크톱 환경의 명령 창에서 다음 형식의 install\_viewagent.sh 스크립트를 사용합니다.

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

install\_viewagent.sh 스크립트에는 필수 및 옵션 매개 변수가 포함되어 있습니다.

**표 5-1.** install\_viewagent.sh 옵션이지만 필수인 매개 변수

옵션 매개 변수 (필수 정보)	설명
-A yes	EULA 및 FIPS 설명 동의. 설치를 계속하려면 <b>yes</b> 를 지정해야 합니다. 이 매개 변수를 지정하지 않으면 설치 중에 값을 묻는 메시지가 표시됩니다.

**표 5-2.** install\_viewagent.sh 옵션 매개 변수

옵션 매개 변수	설명
-a yes no	오디오 입력 리디렉션 지원을 설치하거나 우회합니다. 기본값은 <b>yes</b> 입니다.
-f yes no	FIPS(Federal Information Processing Standards) 140-2용으로 고안된 암호화 모듈의 지원을 설치하거나 우회합니다. 기본값은 <b>no</b> 입니다. 자세한 내용은 “ <a href="#">Horizon Linux 데스크톱의 기능</a> ,” (7 페이지)의 FIPS 140-2 모드 설명을 참조하십시오.

표 5-2. install\_viewagent.sh 옵션 매개 변수 (계속)

옵션 매개 변수	설명
-j	JMS SSL 키 저장소 암호입니다. 기본적으로 설치 관리자는 임의 문자열을 생성합니다.
-m yes no	스마트 카드 리디렉션 지원을 설치하거나 우회합니다. 기본값은 <b>no</b> 입니다.
-r yes no	설치 후에 시스템을 자동으로 다시 시작합니다. 기본값은 <b>no</b> 입니다.
-s	자체 서명된 인증서 주체 DN입니다. 기본적으로 설치 관리자는 Blast를 사용합니다.
-C yes no	클립보드 리디렉션 지원을 설치하거나 우회합니다. 기본값은 <b>yes</b> 입니다.
-F yes no	CDR 지원을 설치하거나 우회합니다. 기본값은 <b>yes</b> 입니다.
-M yes no	Linux Agent를 관리되는 에이전트나 관리되지 않는 에이전트로 업그레이드합니다. 기본값은 <b>yes</b> 입니다.
-S yes no	SSO(Single Sign-On) 지원을 설치하거나 우회합니다. 기본값은 <b>yes</b> 입니다.
-U yes no	USB 지원을 설치하거나 우회합니다. 기본값은 <b>no</b> 입니다.

표 5-3. install\_viewagent.sh 매개 변수의 예

조건	예제
새로 설치	<code>sudo ./install_viewagent.sh -A yes</code> 새로 설치를 하려면 항상 데스크톱 풀을 새로 생성해야 합니다.
관리되지 않는 가상 시스템에서 업그레이드하고 관리되지 않는 가상 시스템 스타일 유지	<code>sudo ./install_viewagent.sh -A yes -M no</code> 이 유형의 업그레이드에서는 데스크톱 풀을 새로 생성할 필요가 없습니다. 기존 데스크톱 풀을 재사용할 수 있습니다. <b>참고</b> 가능한 최고의 성능을 유지하려면 관리되지 않는 가상 시스템을 사용하지 마십시오.
관리되지 않는 가상 시스템 배포에서 업그레이드하고 관리되는 가상 시스템 스타일로 변환합니다. 업그레이드를 위해 브로커에서 새 데스크톱 풀 생성 필요	<code>sudo ./install_viewagent.sh -A yes</code> 이 유형의 업그레이드에서는 데스크톱 풀을 새로 생성해야 합니다. 기존 데스크톱 풀은 삭제해야 합니다.

## Linux Agent용 인증서 구성

Linux Agent를 설치할 때, 설치 관리자에서는 VMwareBlastServer용으로 자체 서명된 인증서를 생성합니다.

- 브로커에서 Blast 보안 게이트웨이를 사용하지 않도록 설정한 경우에는 VMwareBlastServer에서 HTML Access를 사용하여 Linux 데스크톱에 연결하는 브라우저에 이 인증서를 제시합니다.
- 브로커에서 Blast 보안 게이트웨이를 사용하도록 설정한 경우에는 Blast 보안 게이트웨이 인증서에서 브라우저에 인증서를 제시합니다.

산업 또는 보안 규정을 준수하기 위해 자체 서명된 인증서를 CA(인증 기관)에서 서명된 인증서로 교체할 수 있습니다.

### 프로시저

- 1 개인 키와 인증서를 VMwareBlastServer에 설치합니다.
  - a 개인 키의 이름을 rui.key로 변경하고 인증서의 이름을 rui.crt로 변경합니다.
  - b `sudo chmod 550 /etc/vmware/ssl`을 실행합니다.

- c rui.crt와 rui.key를 /etc/vmware/ssl에 복사합니다.
  - d chmod 440 /etc/vmware/ssl을 실행합니다.
- 2 Linux OS CA(인증 기관) 저장소에 루트 및 중간 CA(인증 기관)를 설치합니다.

---

**참고** Linux 배포 설명서에서 Linux 시스템 설정의 변경 사항을 확인하십시오.

---

## Linux 가상 시스템에서 Horizon Agent 업그레이드

Linux 가상 시스템에서 최신 버전의 Horizon Agent를 설치하여 Horizon Agent를 업그레이드할 수 있습니다.

관리되지 않는 가상 시스템: 에이전트 설치 관리자는 브로커 관리자 정보가 필요한 가상 시스템을 브로커에 등록합니다. 데스크톱 풀 생성 마법사는 [시스템 소스] 페이지의 **기타 소스**를 사용하여 등록된 가상 시스템을 선택합니다.

관리되는 가상 시스템: 설치 관리자는 브로커와 통신하지 않습니다. 데스크톱 풀 생성 마법사는 [시스템 소스] 페이지의 **vCenter 가상 시스템**을 사용하여 vCenter를 통해 가상 시스템을 선택합니다. 관리되는 가상 시스템 배포는 다음 기능을 지원합니다.

- 원격 시스템 전원 정책
- 사용자가 시스템을 재설정할 수 있도록 허용

---

**참고** Horizon Agent for Linux 7.0.0 이하 버전은 관리되지 않는 가상 시스템으로 작동했습니다. Horizon Agent for Linux 7.0.1은 관리되는 가상 시스템 지원으로 작동합니다.

---

다음 방법을 사용하여 관리되지 않는 가상 시스템 배포에서 관리되는 가상 시스템 배포로 업그레이드할 수 있습니다.

- 관리되지 않는 가상 시스템 배포를 유지하고 필요한 버전으로 업그레이드합니다. 이러한 유형의 업그레이드에서는 View 연결 서버에서 구성을 수정할 필요가 없습니다.
- 관리되지 않는 가상 시스템 배포에서 관리되는 가상 시스템 배포의 임의 버전으로 업그레이드합니다. 이러한 유형의 업그레이드에서는 View 연결 서버에서 새 데스크톱 풀을 생성해야 합니다.

---

**참고** 관리되는 가상 시스템 배포에서 업그레이드하는 경우 관리되는 가상 시스템 배포를 유지하고 필요한 버전으로 업그레이드할 수 있습니다. 하지만 업그레이드 중에 관리되는 가상 시스템 배포를 관리되지 않는 가상 시스템 배포로 변환하는 것은 지원되지 않습니다.

---

업그레이드에 대해 다음 매개 변수를 사용할 수 있습니다.

**표 5-4.** Horizon Agent를 업그레이드하기 위한 옵션 매개 변수

매개 변수	설명
-A yes	EULA 및 FIPS 설명 동의. 설치를 계속하려면 <b>yes</b> 를 지정해야 합니다. 이 매개 변수를 지정하지 않으면 설치 중에 값을 묻는 메시지가 표시됩니다.
-a yes no	오디오 입력 리디렉션 지원을 설치하거나 우회합니다. 기본값은 <b>yes</b> 입니다.
-f yes no	FIPS(Federal Information Processing Standards) 140-2용으로 고안된 암호화 모듈의 지원을 설치하거나 우회합니다. 기본값은 <b>no</b> 입니다. 자세한 내용은 " <b>Horizon Linux 데스크톱의 기능</b> ," (7 페이지)의 FIPS 140-2 모드 설명을 참조하십시오.
-m yes no	스마트 카드 리디렉션 지원을 설치하거나 우회합니다. 기본값은 <b>no</b> 입니다.
-r yes no	설치 후에 운영 체제를 재부팅합니다. 기본값은 <b>no</b> 입니다.
-C yes no	클립보드 리디렉션 지원을 설치하거나 우회합니다. 기본값은 <b>yes</b> 입니다.
-F yes no	CDR 지원을 설치하거나 우회합니다. 기본값은 <b>yes</b> 입니다.

표 5-4. Horizon Agent를 업그레이드하기 위한 옵션 매개 변수 (계속)

매개 변수	설명
-M yes no	Linux Agent를 관리되는 관리되지 않는 에이전트로 업그레이드합니다. 기본값은 <b>yes</b> 입니다.
-S yes no	SSO(Single Sign-On) 지원을 설치하거나 우회합니다. 기본값은 <b>yes</b> 입니다.
-U yes no	USB 지원을 설치하거나 우회합니다. 기본값은 <b>no</b> 입니다.

## Linux 가상 시스템에서 Horizon Agent 업그레이드

Linux 시스템에서 최신 버전의 Horizon Agent를 설치하여 Horizon Agent를 업그레이드할 수 있습니다.

### 필수 조건

- VMwareBlastServer 프로세스를 실행하고 있지 않은지 확인합니다.  
이 프로세스를 중지하려면 사용자가 시스템에서 로그오프하고 활성 상태인 데스크톱 세션이 없도록 하거나 시스템을 다시 부팅합니다.

### 프로시저

- VMware 다운로드 사이트(<https://my.vmware.com/web/vmware/downloads>)에서 최신 Linux용 Horizon Agent 설치 관리자 파일을 다운로드합니다.

[Desktop & End-User Computing]에서 Linux용 Horizon Agent 설치 관리자가 포함되어 있는 VMware Horizon 7 다운로드를 선택합니다.

설치 관리자 파일 이름은 64비트 Linux의 경우 VMware-viewagent-linux-x86\_64-y.y.y-xxxxxxx.tar.gz입니다. 여기서 y.y.y는 버전 번호이고 xxxxxxx는 빌드 번호입니다.

- 게스트 운영 체제에서 Linux 배포용 tarball의 압축을 풉니다.

예:

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

- tar ball 폴더로 이동합니다.
- install\_viewagent.sh 스크립트를 실행하여 다음 배포 시나리오 중 하나로 관리되지 않는 가상 시스템을 업그레이드합니다.

옵션	설명
관리되지 않는 가상 시스템 배포 업그레이드 및 관리되지 않는 가상 시스템 배포 유지	sudo ./install_viewagent.sh -A yes -M no <b>참고</b> 가능한 최고의 성능을 유지하려면 관리되지 않는 가상 시스템을 사용하지 마십시오.
관리되지 않는 가상 시스템 배포 업그레이드 및 이를 관리되는 가상 시스템 배포로 변경	sudo ./install_viewagent.sh -A yes -M yes <b>참고</b> View Administrator에서 관리되지 않는 가상 시스템 배포에 대한 기존 데스크톱 풀을 삭제하고 관리되는 가상 시스템 배포에 대한 새 데스크톱 풀을 생성합니다. 자세한 내용은 “Linux용 수동 데스크톱 풀 생성,” (57 페이지)을 참조하십시오.
관리되는 가상 시스템 배포 업그레이드	sudo ./install_viewagent.sh -A yes -M yes <b>참고</b> 업그레이드 후에 기존 데스크톱 풀을 재사용할 수 있습니다.

## Horizon 7 for Linux 시스템 제거

가상 시스템에서 Horizon 7 for Linux를 제거하려면 Horizon Agent를 제거하고 구성 파일을 제거해야 합니다.

### 필수 조건

VMwareBlastServer 프로세스를 실행하고 있지 않은지 확인합니다. 이 프로세스를 중지하려면 사용자가 가상 시스템에서 로그오프하고 활성 상태인 데스크톱 세션이 없도록 하거나 시스템을 다시 부팅합니다.

### 프로시저

- 1 가상 시스템에서 터미널 창을 열고 Horizon Agent 제거 스크립트를 실행합니다.

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

이 스크립트는 Horizon Agent 프로세스를 중지하고 설치 디렉토리 `/usr/lib/vmware/viewagent`에서 Horizon Agent 서비스 및 소프트웨어를 삭제합니다.

- 2 `/etc/vmware` 디렉토리에서 Horizon 7 for Linux 구성 파일을 수동으로 삭제합니다.

## Linux 데스크톱용 구성 옵션

구성 파일을 사용하여 다양한 옵션을 구성하고 사용자 경험을 사용자 지정할 수 있습니다.

이 장에서는 다음 주제에 대해 설명합니다.

- “Linux 데스크톱의 구성 파일에서 옵션 설정,” (47 페이지)
- “Linux 데스크톱용 Blast 설정 예,” (55 페이지)
- “Linux 데스크톱에 대한 클라이언트 드라이브 리디렉션 옵션의 예,” (55 페이지)
- “Linux 데스크톱의 vSphere 콘솔 디스플레이 표시하지 않음,” (56 페이지)

### Linux 데스크톱의 구성 파일에서 옵션 설정

`/etc/vmware/config` 또는 `/etc/vmware/viewagent-custom.conf` 파일에 항목을 추가하여 특정 옵션을 구성할 수 있습니다.

View Agent 또는 Horizon Agent를 설치하는 동안 설치 관리자가 두 개의 구성 템플릿 파일 `config.template` 및 `viewagent-custom.conf.template`을 `/etc/vmware`에 복사합니다. 또한, 파일 `/etc/vmware/config` 및 `/etc/vmware/viewagent-custom.conf`가 없는 경우에는 설치 관리자가 `config.template`을 `config`에 복사하고 `viewagent-custom.conf.template`을 `viewagent-custom.conf`에 복사합니다. 템플릿 파일에 모든 구성 옵션이 나열되고 기록됩니다. 옵션을 설정하려면 설명을 제거하고 값을 적절하게 변경합니다.

예를 들어, `/etc/vmware/config`에서 다음 줄은 무손실 PNG 모드를 사용하도록 설정합니다.

```
RemoteDisplay.alwaysLossless=TRUE
```

구성을 변경하고 나면 변경이 적용되도록 Linux를 재부팅합니다.

#### `/etc/vmware/config`의 구성 옵션

VMwareBlastServer 및 관련 플러그인에서는 구성 파일 `/etc/vmware/config`를 사용합니다.

**참고** 다음 표에는 Horizon Agent 구성 파일의 USB에 대한 각 에이전트 적용 정책 설정의 설명이 포함되어 있습니다. Horizon Agent는 설정을 사용하여 USB가 호스트 시스템으로 전달될 수 있는지 여부를 결정합니다. 또한 Horizon Agent는 `merge(m)` 수정자를 지정하여 Horizon Client 필터 정책 설정 외에 Horizon Agent 필터 정책 설정을 적용할지 또는 `override(o)` 수정자를 사용하여 Horizon Client 필터 정책 설정 대신 Horizon Agent 필터 정책 설정을 사용할지 여부에 따라 해석 및 적용을 위해 Horizon Client에 설정을 전달합니다.

**표 6-1.** /etc/vmware/config 의 구성 옵션

옵션	값/형식	기본값	설명
RemoteDisplay.alwaysLossless	true 또는 false	false	그래픽 애플리케이션, 특히 그래픽 디자인 애플리케이션에서는 Linux 데스크톱의 클라이언트 디스플레이에서 이미지의 정확한 픽셀 렌더링이 수행되어야 합니다. Linux 데스크톱에서 생성되고 클라이언트 디바이스에서 렌더링되는 이미지 및 비디오 재생을 위해 무손실 PNG 모드를 구성할 수 있습니다. 이 기능에서는 클라이언트와 ESXi 호스트 사이에 추가 대역폭을 사용합니다.
mksVNCServer.useInputButtonMapping	true 또는 false	false	Ubuntu 또는 RHEL 7에서 왼손 마우스 지원을 사용하도록 설정하려면 이 옵션을 설정합니다. CentOS 및 RHEL 6에서는 왼손 마우스를 지원하며 이 옵션을 설정할 필요가 없습니다.
RemoteDisplay.allowAudio	true 또는 false	true	오디오 출력을 사용/사용하지 않으려면 이 옵션을 설정합니다.
VVC.ScRedir.Enable	true 또는 false	true	스마트 카드 리디렉션을 사용/사용하지 않으려면 이 옵션을 설정합니다.
VVC.logLevel	fatal error, warn, info, debug 또는 trace	info	VVC 프로세스 노드의 로그 수준을 설정하려면 이 옵션을 사용합니다.
VVC.RTAV.Enable	true 또는 false	true	오디오 입력을 사용/사용하지 않으려면 이 옵션을 설정합니다.
Clipboard.Direction	0, 1, 2, 또는 3	2	이 옵션에 따라 클립보드 리디렉션 정책이 결정됩니다. <ul style="list-style-type: none"> <li>■ 0 - 클립보드 리디렉션을 사용하지 않도록 설정합니다.</li> <li>■ 1 - 클립보드 리디렉션을 양방향 모두에서 사용하도록 설정합니다.</li> <li>■ 2 - 클립보드 리디렉션을 클라이언트에서 원격 데스크톱 방향으로만 사용하도록 설정합니다.</li> <li>■ 3 - 클립보드 리디렉션을 원격 데스크톱에서 클라이언트 방향으로만 사용하도록 설정합니다.</li> </ul>
cdrserver.logLevel	error, warn, info, debug trace 또는 verbose	info	vmware-CDRserver.log의 로그 수준을 설정하려면 이 옵션을 사용합니다.
cdrserver.forcedByAdmin	true 또는 false	false	클라이언트가 cdrserver.shareFolders 옵션에 지정되지 않은 추가 폴더를 공유하도록 허용하거나 금지하려면 이 옵션을 설정합니다.
cdrserver.sharedFolders	<i>file_path1,R; file_path2,; file_path3,R; ..</i>	정의되지 않음	클라이언트가 Linux 데스크톱과 공유할 수 있는 폴더에 대한 하나 이상의 파일 경로를 지정합니다. 예: <ul style="list-style-type: none"> <li>■ Windows 클라이언트: C:\Spreadsheets, ;D:\Webbooks,R</li> <li>■ 비 Windows 클라이언트: /tmp/spreadsheets;/tmp/ebooks, ;/home/finance,R</li> </ul>



표 6-1. /etc/vmware/config 의 구성 옵션 (계속)

옵션	값/형식	기본값	설명
cdrserver.permissions	R	RW	Horizon Client가 공유하는 폴더에 대해 Horizon Agent가 갖는 추가 읽기/쓰기 사용 권한을 적용하려면 이 옵션을 사용합니다. 예: <ul style="list-style-type: none"> <li>Horizon Client가 공유하는 폴더에 read 및 write 사용 권한이 있으며 <b>cdrserver.permissions=R</b>을 설정했으면 Horizon Agent는 read 액세스 사용 권한만 갖습니다.</li> <li>Horizon Client가 공유하는 폴더에 read 사용 권한만 있고 <b>cdrserver.permissions=RW</b>를 설정했으면 Horizon Agent는 여전히 read 액세스 사용 권한을 갖게 됩니다. Horizon Agent는 Horizon Client가 설정한 read 전용 특성을 변경할 수 없습니다. Horizon Agent가 수행할 수 있는 유일한 작업은 쓰기 액세스 사용 권한을 제거하는 것입니다.</li> </ul> 일반적인 사용법: <ul style="list-style-type: none"> <li><b>cdrserver.permissions=R</b></li> <li><b>#cdrserver.permissions=R</b>(항목의 주석을 해제하거나 삭제)</li> </ul>
cdrserver.cacheEnable	true 또는 false	true	에이전트에서 클라이언트 측으로의 쓰기 캐싱 기능을 사용하거나 사용하지 않도록 설정하려면 이 옵션을 설정합니다.
UsbRedirPlugin.log.logLevel	error, warn, info, debug, trace 또는 verbose	info	USB 리디렉션 플러그인의 로그 수준을 설정하려면 이 옵션을 사용합니다.
UsbRedirServer.log.logLevel	error, warn, info, debug, trace 또는 verbose	info	USB 리디렉션 서버의 로그 수준을 설정하려면 이 옵션을 사용합니다.
viewusb.AllowAutoDeviceSplitting	{m o}: {true false}	정의되지 않음, false와 동일	복합 USB 디바이스의 자동 분할을 허용하거나 허용하지 않으려면 이 옵션을 설정합니다. 예: <b>m:true</b>
viewusb.SplitExcludeVidPid	{m o}:vid-xxx1_pid-yyy1[;vid-xxx2_pid-yyy2;...]	정의되지 않음	벤더 및 제품 ID로 분할하여 지정된 복합 USB 디바이스를 제외하거나 포함하려면 이 옵션을 사용합니다. 이 설정의 형식은 <b>vid-xxx1_pid-yyy1[;vid-xxx2_pid-yyy2;...]</b> 입니다. ID 번호는 16진수로 지정해야 합니다. ID에서 개별 자릿수 대신 와일드카드 문자(*)를 사용할 수 있습니다. 예: <b>m:vid-0f0f_pid-55**</b>

표 6-1. /etc/vmware/config 의 구성 옵션 (계속)

옵션	값/형식	기본값	설명
viewusb.SplitVidPid	<code>{m o}: vid-xxxx_pid-yyyy([exintf:zz[:exintf:ww]])[:...]</code>	정의되지 않음	벤더 및 제품 ID별로 지정된 복합 USB 디바이스의 구성 요소를 개별 디바이스로 처리하려면 이 옵션을 설정합니다. 이 설정의 형식은 <code>vid-xxxx_pid-yyyy(exintf:zz[:exintf:ww])</code> 입니다.  <code>exintf</code> 키워드를 사용하면 인터페이스 번호를 지정하여 구성 요소를 리디렉션에서 제외할 수 있습니다. ID 번호는 16진수로, 인터페이스 번호는 앞에 0이 표시되는 10진수로 지정해야 합니다. ID에서 개별 자릿수 대신 와일드카드 문자(*)를 사용할 수 있습니다.  예: <code>o:vid-0f0f_pid-***(exintf-01);vid-0781_pid-554c(exintf:01;exintf:02)</code> <b>참고</b> Horizon은 명시적으로 제외하지 않은 구성 요소를 자동으로 포함시키지 않습니다. <b>VidPid 디바이스 포함</b> 과 같은 필터 정책을 지정하여 해당 구성 요소를 포함시켜야 합니다.
viewusb.AllowAudioIn	<code>{m o}: {true false}</code>	정의되지 않음, true와 동일	오디오 입력 디바이스가 리디렉션되도록 허용하거나 허용하지 않으려면 이 옵션을 사용합니다. 예: <code>o:false</code>
viewusb.AllowAudioOut	<code>{m o}: {true false}</code>	정의되지 않음, false와 동일	오디오 출력 디바이스의 리디렉션을 허용하거나 허용하지 않으려면 이 옵션을 설정합니다.
viewusb.AllowHIDBootable	<code>{m o}: {true false}</code>	정의되지 않음, true와 동일	HID 부팅 가능 디바이스로도 알려져 있는, 부팅 시에 사용 가능한 키보드 또는 마우스 이외의 입력 디바이스의 리디렉션을 허용하거나 허용하지 않으려면 이 옵션을 사용합니다.
viewusb.AllowDevDescFailsafe	<code>{m o}: {true false}</code>	정의되지 않음, false와 동일	Horizon Client가 구성 또는 디바이스 설명자를 가져오지 못할 경우에도 디바이스가 리디렉션되도록 허용하거나 허용하지 않으려면 이 옵션을 설정합니다. 구성 또는 디바이스 설명자를 가져오지 못할 경우에도 디바이스를 허용하려면 <b>IncludeVidPid</b> 또는 <b>IncludePath</b> 와 같은 Include 필터에 포함하십시오.
viewusb.AllowKeyboardMouse	<code>{m o}: {true false}</code>	정의되지 않음, false와 동일	통합형 포인팅 디바이스(예: 마우스, 트랙볼 또는 터치패드)를 사용하여 키보드의 리디렉션을 허용하거나 허용하지 않으려면 이 옵션을 설정합니다.
viewusb.AllowSmartcard	<code>{m o}: {true false}</code>	정의되지 않음, false와 동일	스마트 카드 디바이스가 리디렉션되도록 허용하거나 허용하지 않으려면 이 옵션을 설정합니다.
viewusb.AllowVideo	<code>{m o}: {true false}</code>	정의되지 않음, true와 동일	비디오 디바이스가 리디렉션되도록 허용하거나 허용하지 않으려면 이 옵션을 설정합니다.
viewusb.DisableRemoteConfig	<code>{m o}: {true false}</code>	정의되지 않음, false와 동일	USB 디바이스 필터링을 수행할 때 Horizon Agent 설정을 사용하거나 사용하지 않도록 설정하려면 이 옵션을 설정합니다.

표 6-1. /etc/vmware/config 의 구성 옵션 (계속)

옵션	값/형식	기본값	설명
viewusb.ExcludeAllDevices	{true false}	정의되지 않음, false와 동일	모든 USB 디바이스를 리디렉션에 포함하거나 리디렉션에서 제외하려면 이 옵션을 사용합니다. <b>true</b> 로 설정되면 기타 정책 설정을 사용하여 특정 디바이스 또는 디바이스 제품군을 리디렉션할 수 있습니다. <b>false</b> 로 설정되면 기타 정책 설정을 사용하여 특정 디바이스 또는 디바이스 제품군이 리디렉션되지 않도록 방지할 수 있습니다. Horizon Agent에서 <b>ExcludeAllDevices</b> 값을 <b>true</b> 로 설정하고 이 설정이 Horizon Client로 전달될 경우, Horizon Agent 설정이 Horizon Client 설정을 재정의합니다.
viewusb.ExcludeFamily	{m o}:family_name_1[;family_name_2;...]	정의되지 않음	디바이스 제품군을 리디렉션에서 제외하려면 이 옵션을 사용합니다. 예: <b>m:bluetooth;smart-card</b> 자동 디바이스 분할을 사용하도록 설정한 경우 Horizon은 복합 USB 디바이스 각 인터페이스의 디바이스 제품군을 검토하여 제외해야 할 인터페이스를 결정합니다. 자동 디바이스 분할을 사용하지 않도록 설정한 경우, Horizon은 전체 복합 USB 디바이스의 디바이스 제품군을 검토합니다. <b>참고</b> 그러나 기본적으로 마우스 및 키보드는 리디렉션에서 제외되므로 이 설정을 사용하여 제외할 필요가 없습니다.
viewusb.ExcludeVidPid	{m o}:vid-xxx1_pid-yyy1[;vid-xxx2_pid-yyy2;...]	정의되지 않음	지정된 벤더 및 제품 ID를 가진 디바이스를 리디렉션에서 제외하려면 이 옵션을 설정합니다. ID 번호를 16진수로 지정해야 합니다. ID에서 개별 자릿수 대신 와일드카드 문자(*)를 사용할 수 있습니다. 예: <b>o:vid-0781_pid-****;vid-0561_pid-554c</b>
viewusb.ExcludePath	{m o}:bus-x1[/y1].../port-z1[;bus-x2[/y2].../port-z2;...]	정의되지 않음	지정된 허브 또는 포트 경로의 디바이스를 리디렉션에서 제외하려면 이 옵션을 사용합니다. 버스 및 포트 번호를 16진수로 지정해야 합니다. 와일드카드 문자는 경로에 사용할 수 없습니다. 예: <b>m:bus-1/2/3_port-02;bus-1/1/1/4_port-ff</b>
viewusb.IncludeFamily	{m o}:family_name_1[;family_name_2]...	정의되지 않음	리디렉션될 수 있는 디바이스 제품군을 포함하려면 이 옵션을 설정합니다. 예: <b>o:storage;smart-card</b>
viewusb.IncludePath	{m o}:bus-x1[/y1].../port-z1[;bus-x2[/y2].../port-z2;...]	정의되지 않음	리디렉션될 수 있는 지정된 허브 또는 포트 경로의 디바이스를 포함하려면 이 옵션을 사용합니다. 버스 및 포트 번호를 16진수로 지정해야 합니다. 와일드카드 문자는 경로에 사용할 수 없습니다. 예: <b>m:bus-1/2_port-02;bus-1/7/1/4_port-0f</b>
viewusb.IncludeVidPid	{m o}:vid-xxx1_pid-yyy1[;vid-xxx2_pid-yyy2;...]	정의되지 않음	리디렉션될 수 있는 지정된 벤더 및 제품 ID를 가진 디바이스를 포함하려면 이 옵션을 설정합니다. ID 번호를 16진수로 지정해야 합니다. ID에서 개별 자릿수 대신 와일드카드 문자(*)를 사용할 수 있습니다. 예: <b>o:vid-***_pid-0001;vid-0561_pid-554c</b>
mksVNCServer.useXExtButtonMapping	true 또는 false	false	이 옵션을 통해 SLED 11 SP3에서 왼쪽 마우스 지원을 사용하도록 또는 사용하지 않도록 설정합니다.
mksvhan.clipboardSize	정수	1024	이 옵션을 사용하여 복사하여 붙여넣을 클립보드 최대 크기를 지정합니다.

**표 6-1.** /etc/vmware/config 의 구성 옵션 (계속)

옵션	값/형식	기본값	설명
RemoteDisplay.maxBandwidthKbps	정수	4096000	VMware Blast 세션에 대해 초당 킬로비트 (kbps)로 최대 대역폭을 지정합니다. 대역폭은 모든 이미징, 오디오, 가상 채널 및 VMware Blast 제어 트래픽을 포함합니다. 최댓값은 4Gbps(4096000)입니다.
RemoteDisplay.maxFPS	정수	60	화면 업데이트의 최대 속도를 지정합니다. 사용자가 소비하는 평균 대역폭을 관리하려면 이 설정을 사용합니다. 값은 3에서 60 사이여야 합니다. 기본값은 초당 60회 업데이트입니다.
RemoteDisplay.enableStats	true 또는 false	false	mks 로그에서 대역폭, FPS, RTT 등과 같은 Blast 프로토콜 통계를 사용하거나 사용하지 않도록 설정합니다.
RemoteDisplay.allowH264	true 또는 false	true	이 옵션을 설정하여 H.264 인코딩을 사용하거나 사용하지 않도록 설정합니다.
vdpservice.log.logLevel	fatal error, warn, info, debug 또는 trace	info	vdpservice의 로그 수준을 설정하려면 이 옵션을 사용합니다.
RemoteDisplay.qpmaxH264	사용할 수 있는 값 범위: 0-51	36	이 옵션을 사용하여 H.264 인코딩을 사용하도록 구성된 원격 디스플레이에서 최상의 이미지 품질을 지정하는 H264minQP 양자화 매개 변수를 설정합니다. RemoteDisplay.qpminH264에 설정된 값보다 큰 값을 설정합니다.
RemoteDisplay.qpminH264	사용할 수 있는 값 범위: 0-51	10	이 옵션을 사용하여 H.264 인코딩을 사용하도록 구성된 원격 디스플레이에서 최저 이미지 품질을 지정하는 H264maxQP 양자화 매개 변수를 설정합니다. RemoteDisplay.qpmaxH264에 설정된 값보다 작은 값을 설정합니다.
RemoteDisplay.minQualityJPEG	사용할 수 있는 값 범위: 1-100	25	JPEG/PNG 인코딩의 데스크톱 디스플레이 이미지 품질을 지정합니다. 낮은 품질 설정은 스크롤이 발생하는 경우와 같이 자주 변경되는 화면 영역에 사용됩니다.
RemoteDisplay.midQualityJPEG	사용할 수 있는 값 범위: 1-100	35	JPEG/PNG 인코딩의 데스크톱 디스플레이 이미지 품질을 지정합니다. 데스크톱 디스플레이의 보통 품질 설정에 사용됩니다.
RemoteDisplay.maxQualityJPEG	사용할 수 있는 값 범위: 1-100	90	JPEG/PNG 인코딩의 데스크톱 디스플레이 이미지 품질을 지정합니다. 높은 품질 설정은 더 정적이어서 이미지 품질이 더 좋은 화면 영역에 사용됩니다.

**/etc/vmware/viewagent-custom.conf의 구성 옵션**

Java Standalone Agent에서는 구성 파일 /etc/vmware/viewagent-custom.conf를 사용합니다.

표 6-2. /etc/vmware/viewagent-custom.conf 의 구성 옵션

옵션	값	기본값	설명
서브넷	IP 주소/CIDR 형식의 NULL 또는 네트워크 주소 및 마스크	NULL	<p>다른 서브넷을 갖는 여러 로컬 IP 주소가 있는 경우 이 옵션을 사용하여 Linux Agent가 View 연결 서버에 제공하는 서브넷을 설정합니다.</p> <p>Linux Agent 시스템에서 여러 서브넷 구성이 감지되면 Linux Agent에서 사용되어야 하는 올바른 서브넷을 지정하는 데 이 옵션이 필요합니다. 예를 들어 Linux 시스템에 Docker를 설치한 경우 가상 네트워크 어댑터로 사용됩니다. Linux Agent가 Docker를 가상 네트워크 어댑터로 사용하지 않도록 하려면 실제 물리적 네트워크 어댑터를 사용하도록 이 옵션을 설정해야 합니다.</p> <p>IP 주소/CIDR 형식으로 값을 지정해야 합니다. 예를 들어 Subnet=192.168.1.0/24와 같이 지정합니다. NULL은 Linux Agent가 IP 주소를 임의로 선택한다는 뜻입니다.</p>
SSOEnable	true 또는 false	true	SSO(Single Sign On)를 사용하거나 사용하지 않도록 설정하려면 이 옵션을 설정합니다.
SSOUserFormat	텍스트 문자열	[username]	<p>Single Sign-On에 로그인 이름 형식을 지정하려면 이 옵션을 사용합니다. 기본값은 사용자 이름만입니다. 도메인 이름도 필요한 경우 이 옵션을 설정합니다. 일반적으로 로그인 이름은 도메인 이름과 특수 문자 뒤에 사용자 이름을 추가한 것입니다. 특수 문자가 백슬래시인 경우에는 백슬래시를 하나 더 사용해서 이스케이프 처리해야 합니다. 로그인 이름 형식의 예:</p> <ul style="list-style-type: none"> <li>■ SSOUserFormat=[domain]\\[username]</li> <li>■ SSOUserFormat=[domain]+[username]</li> <li>■ SSOUserFormat=[username]@[domain]</li> </ul>
CDREnable	true 또는 false	true	CDR(클라이언트 드라이브 리디렉션) 기능을 사용하거나 사용하지 않도록 설정하려면 이 옵션을 설정합니다.
USBEnable	true 또는 false	true	USB 리디렉션 기능을 사용하거나 사용하지 않도록 설정하려면 이 옵션을 설정합니다.
KeyboardLayoutSync	true 또는 false	true	<p>클라이언트의 시스템 로캘 목록 및 현재 키보드 레이아웃을 Horizon Agent for Linux 데스크톱과 동기화할지 여부를 지정하려면 이 옵션을 사용합니다.</p> <p>이 설정이 사용되도록 설정되거나 구성되지 않은 경우, 동기화가 허용됩니다. 이 설정이 사용되지 않도록 설정되면 동기화가 허용되지 않습니다.</p> <p>이 기능은 Windows용 Horizon Client에서만 지원되고 영어, 프랑스어, 독일어, 일본어, 한국어, 스페인어, 중국어 간체 및 중국어 번체 로캘에서만 지원됩니다.</p>
StartBlastServerTimeout	정수	20	이 옵션은 VMwareBlastServer 프로세스에서 초기화에 사용할 수 있는 시간을 초 단위로 결정합니다. 이 시간 초과 값 이내에 프로세스가 준비되지 않으면 사용자의 로그인이 실패합니다.
SSLCiphers	텍스트 문자열	!aNULL:kECDH+AESGCM:ECDH+AESGCM:RSA+AESGCM:kECDH+AES:ECDH+AES:RSA+AES	암호 목록을 지정하려면 이 옵션을 사용합니다. <a href="https://www.openssl.org/docs/manmaster/man1/ciphers.html">https://www.openssl.org/docs/manmaster/man1/ciphers.html</a> 에 정의된 형식을 사용해야 합니다.
SSLProtocols	텍스트 문자열	TLSv1_1:TLSv1_2	보안 프로토콜을 지정하려면 이 옵션을 사용합니다. 지원되는 프로토콜은 TLSv1.0, TLSv1.1 및 TLSv1.2입니다.

**표 6-2.** /etc/vmware/viewagent-custom.conf 의 구성 옵션 (계속)

옵션	값	기본값	설명
SSLCipherServerPreference	true 또는 false	true	이 옵션을 사용하여 SSL_OP_CIPHER_SERVER_PREFERENCE 옵션을 사용하거나 사용하지 않도록 설정합니다. 자세한 내용은 <a href="https://www.openssl.org/docs/manmaster/ssl/SSL_CTX_set_options.html">https://www.openssl.org/docs/manmaster/ssl/SSL_CTX_set_options.html</a> 의 내용을 참조하십시오.
UseGnomeFlashback	true 또는 false	false	이 옵션은 Ubuntu 14.04 또는 Ubuntu 16.04 시스템에 설치된 GNOME Flashback(Metacity) 데스크톱 환경을 사용할지 여부를 결정합니다. 이 옵션은 SSO 기능이 사용되도록 설정되었는지 여부에 관계없이 적용됩니다. 이 옵션을 <b>TRUE</b> 로 설정하면 항상 기본 데스크톱 환경 대신 GNOME Flashback(Metacity) 데스크톱 환경이 사용됩니다. <b>팁</b> 시스템 성능을 향상시키려면 Ubuntu 14.04 또는 Ubuntu 16.04 시스템에 GNOME Flashback(Metacity) 데스크톱을 설치한 후에 <b>UseGnomeFlashback=TRUE</b> 를 구성하십시오.
LogCnt	정수	-1	이 옵션을 사용하여 /tmp/vmware-root에서 예약된 로그 파일 수를 설정합니다. <ul style="list-style-type: none"> <li>■ -1 - 모두 유지</li> <li>■ 0 - 모두 삭제</li> <li>■ &gt; 0 - 예약된 로그 수</li> </ul>
RunOnceScript			이 옵션을 사용하여 복제된 VM을 AD에 다시 가입시킵니다. 호스트 이름이 변경된 후에 run once script를 설정합니다. 지정된 스크립트는 첫 번째 호스트 이름이 변경된 후에만 실행됩니다. 스크립트는 에이전트 설치 후에 에이전트 서비스가 시작되고 호스트 이름이 변경되면 루트 권한으로 실행됩니다. 예를 들어 Winbind 솔루션의 경우 Winbind를 사용하여 기본 VM을 AD에 가입시키고 이 옵션을 스크립트 경로로 설정해야 합니다. 여기에는 도메인 재가입 명령 /usr/bin/net ads join -U <ADUserName> %<ADUserPassword>가 포함되어야 합니다. VM 복제 후에 운영 체제 사용자 지정에 따라 호스트 이름이 변경됩니다. 에이전트 서비스가 시작되면 복제된 VM을 AD에 가입시키기 위해 이 스크립트가 실행됩니다.
RunOnceScriptTimeout		120	RunOnceScript 옵션의 시간 초과 값을 초 단위로 설정하려면 이 옵션을 사용합니다. 예를 들어 RunOnceScriptTimeout=120을 설정합니다.

**참고** 세 개의 보안 옵션 SSLCiphers, SSLProtocols 및 SSLCipherServerPreference는 VMwareBlastServer 프로세스에 사용됩니다. VMwareBlastServer 프로세스를 시작할 때 Java Standalone Agent는 다음과 같은 옵션을 매개 변수로 전달합니다. BSG(Blast 보안 게이트웨이)가 사용하도록 설정된 경우 이러한 옵션은 BSG와 Linux 데스크톱 사이의 연결에 영향을 줍니다. BSG가 사용하지 않도록 설정된 경우 이러한 옵션은 클라이언트와 Linux 데스크톱 사이의 연결에 영향을 줍니다.

## Linux 데스크톱용 Blast 설정 예

원격 데스크톱 디스플레이의 이미지 품질을 조정하여 사용자 환경을 개선할 수 있습니다. 이미지 품질을 개선하면 네트워크 연결이 불량할 때 일관성 있는 사용자 환경을 유지하는 데 유용합니다.

### VMware Blast Extreme 프로토콜 설정 예

VMwareBlastServer 및 관련 플러그인에서는 구성 파일 `/etc/vmware/config`를 사용합니다.

표 6-3. `/etc/vmware/config`의 Blast 구성 옵션 예

옵션 이름	매개 변수	고속 LAN	LAN	전용 WAN	광대역 WAN	저속 WAN	초저속
대역폭 설정	RemoteDisplay.maxBandwidthKbps	1000000(1Gbps)	100000(1Gbps)	1000000(1Gbps)	5000(5Mbps)	2000(2Mbps)	1000(1Mbps)
최대 FPS	RemoteDisplay.maxFPS	60	30	30	20	15	5
오디오 재생	RemoteDisplay.allowAudio	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE
디스플레이 품질 (JPEG/PNG)	RemoteDisplay.maxQualityJPEG	90	90	90	70	60	50
디스플레이 품질 (JPEG/PNG)	RemoteDisplay.midQualityJPEG	35	35	35	35	35	35
디스플레이 품질 (JPEG/PNG)	RemoteDisplay.minQualityJPEG	25	25	25	20	20	20
디스플레이 품질 (H.264)	RemoteDisplay.qpmaxH264	28	36	36	36	36	42
디스플레이 품질 (H.264)	RemoteDisplay.qpminH264	10	10	10	10	10	10

## Linux 데스크톱에 대한 클라이언트 드라이브 리디렉션 옵션의 예

CDR(클라이언트 드라이브 리디렉션) 옵션을 구성하여 원격 Linux 데스크톱에서 로컬 시스템의 공유 폴더 및 드라이브에 액세스할 수 있는지 여부를 결정합니다.

`/etc/vmware/config` 파일에 항목을 추가하여 CDR 설정을 구성합니다.

다음 구성 예에서는 `C:\webooks` 및 `C:\spreadsheets` 폴더를 공유하고, 두 폴더 모두를 읽기 전용으로 만들고, 클라이언트가 추가 폴더를 공유하지 못하도록 합니다.

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

이전 예에서 `ebooks` 및 `spreadsheets` 뒤에 붙는 쉼표(",")는 올바른 옵션 구문 분석을 위해 반드시 필요합니다.

`cdserver.sharedFolders` 옵션에 포함된 모든 "R"은 해당 설정에 나열되는 모든 폴더에 영향을 미칩니다. 다음 예에서 `/home/jsmith` 폴더 경로 다음에만 R 값이 붙더라도 `ebooks` 및 `spreadsheets` 폴더는 둘 다 읽기 전용입니다.

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

## Linux 데스크톱의 vSphere 콘솔 디스플레이 표시하지 않음

사용자가 Linux 데스크톱에 연결하면 Linux 가상 시스템에 대한 vSphere 콘솔에도 데스크톱이 표시될 수 있습니다. 사용자가 데스크톱에 연결할 때 vSphere 콘솔이 비어 있도록 Linux 가상 시스템을 구성할 수 있습니다.

### 프로시저

- ◆ ESXi 호스트에서 Linux 가상 시스템의 vmx 파일에 다음 줄을 추가합니다.

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

사용자가 데스크톱에서 로그아웃하면 가상 시스템에 연결되어 있어도 vSphere 콘솔 디스플레이는 빈 상태를 유지합니다.



## Linux 데스크톱 풀 생성 및 관리

원격 데스크톱으로 사용할 Linux 가상 시스템을 구성하려면 Linux 가상 시스템으로 데스크톱 풀을 생성해야 합니다.

Horizon for Linux에서는 다음과 같은 데스크톱 풀 유형을 지원합니다.

- vCenter 가상 시스템이 있는 수동 데스크톱 풀
- 자동화된 전체 클론 데스크톱 풀

vCenter 가상 시스템으로 수동 데스크톱 풀을 생성하려면 모든 가상 시스템에 Horizon 에이전트를 설치해야 합니다. 그런 다음 연결 서버 데스크톱 풀 생성 마법사를 사용하여 가상 시스템을 데스크톱 풀에 추가합니다. 많은 수의 가상 시스템을 복제하려면 [“Linux 데스크톱의 대량 배포 개요,”](#) (65 페이지)를 참조하십시오.

자동화된 전체 클론 데스크톱 풀을 생성하려면 Linux 가상 시스템 템플릿에 Horizon 에이전트를 설치해야 합니다. 그런 다음 연결 서버 데스크톱 풀 생성 마법사를 사용하여 전체 가상 시스템을 복제합니다.

이 장에서는 다음 주제에 대해 설명합니다.

- [“Linux용 수동 데스크톱 풀 생성,”](#) (57 페이지)
- [“Linux용 데스크톱 풀 관리,”](#) (58 페이지)
- [“Linux용 자동화된 전체 클론 데스크톱 풀 생성,”](#) (59 페이지)
- [“브로커 PowerCLI 명령,”](#) (61 페이지)

### Linux용 수동 데스크톱 풀 생성

Linux 가상 시스템용으로 수동 데스크톱 풀을 생성할 수 있습니다.

#### 필수 조건

- Linux 게스트 운영 체제에 Horizon Agent가 설치되어 있는지 확인합니다. [“Linux 가상 시스템에 Horizon Agent 설치,”](#) (41 페이지)의 내용을 참조하십시오.
- vCenter Server가 Horizon 연결 서버에 추가되었는지 확인합니다.

#### 프로시저

- 1 View Administrator에서 수동 데스크톱 풀을 추가합니다.  
**카탈로그 > 데스크톱 풀 > 추가**를 선택합니다.
- 2 **수동 데스크톱 풀**을 선택합니다.
- 3 [사용자 할당] 페이지에서 데스크톱 풀의 시스템에 대해 전용 또는 부동 사용자 할당을 선택하고 **다음**을 클릭합니다.

- 4 [시스템 소스] 페이지에서 **vCenter 가상 시스템**을 선택하고 **다음**을 클릭합니다.
- 5 [vCenter Server] 페이지에서 해당 vCenter Server를 선택하고 **다음**을 클릭합니다.
- 6 **데스크톱 풀 ID**를 입력합니다.
- 7 [데스크톱 풀 설정] 페이지에서 다음 옵션을 설정합니다.

옵션	설명
기본 디스플레이 프로토콜	VMware Blast
사용자가 프로토콜을 선택할 수 있도록 허용함	아니오
3D 렌더러	2D, vSGA 또는 vDGA 데스크톱용 vSphere Client 및 vGPU 데스크톱용 NVIDIA GRID vGPU를 사용하여 관리

**참고** 풀 설정은 필수입니다. 풀을 설정하지 않으면 데스크톱 연결에 실패할 수 있고 프로토콜 오류가 발생하거나 검은색 화면이 표시됩니다.

- 8 [vCenter 가상 시스템 추가] 페이지에서 Linux 가상 시스템을 선택합니다. **추가**를 클릭하고 **다음**을 클릭합니다.

**참고** Windows 및 Linux 가상 시스템을 동일한 데스크톱 풀에 생성하지 마십시오.

- 9 메시지를 따라 절차를 완료합니다.

사용자에게 데스크톱 풀의 시스템에 대한 권한을 부여합니다. View Administrator에서 데스크톱 풀을 선택하고 **사용 권한 > 권한 추가**를 선택하고 사용자나 그룹을 추가합니다.

Linux 가상 시스템은 Horizon 7 배포에서 원격 데스크톱으로 사용될 준비가 되었습니다.

## Linux용 데스크톱 풀 관리

수동 데스크톱 풀을 만들고 Linux 시스템을 풀에 추가하면 설정을 구성하여 수동 데스크톱 풀을 관리할 수 있습니다. 수동 데스크톱 풀에는 Linux 게스트 운영 체제만 추가해야 합니다. 풀에 Windows 및 Linux 게스트 운영 체제가 둘 다 포함되어 있으면 풀은 Windows 풀로 취급되고 Linux 데스크톱에 연결할 수 없습니다.

### 관리 작업 지원

- 데스크톱 풀 사용 또는 사용 안 함
- 자동화된 데스크톱 풀 복제
- 데스크톱 풀 삭제

View Manager에서 가상 시스템을 제거하거나 디스크에서 가상 시스템을 삭제할 수 있습니다.

### 원격 설정 지원

**표 7-1.** 원격 설정

원격 설정	옵션
원격 시스템 전원 정책	<ul style="list-style-type: none"> <li>■ 전원 작업 수행 안 함</li> <li>■ 시스템 전원이 항상 켜져 있는지 확인</li> <li>■ 일시 중단</li> <li>■ 전원 끄기</li> </ul>
연결 해제 후 자동 로그오프	<ul style="list-style-type: none"> <li>■ 즉시</li> <li>■ 안 함</li> <li>■ n분 후</li> </ul>

표 7-1. 원격 설정 (계속)

원격 설정	옵션
사용자가 시스템을 재설정/다시 시작할 수 있도록 허용	<ul style="list-style-type: none"> <li>■ 예</li> <li>■ 아니요</li> </ul>
사용자가 여러 클라이언트 디바이스에서 별도의 세션을 시작할 수 있도록 허용	<ul style="list-style-type: none"> <li>■ 예</li> <li>■ 아니요</li> </ul>
전체 클론 및 부동을 사용하는 자동화된 데스크톱 풀에서 [로그오프 후 시스템 삭제]	<ul style="list-style-type: none"> <li>■ 예</li> <li>■ 아니요</li> </ul>

## View Administrator 작업 지원

- 세션 연결 해제
- 세션 로그오프
- 데스크톱 재설정/다시 시작
- 메시지 보내기

전용 데스크톱 풀에 대해 각 가상 시스템에 대한 사용자 할당을 추가하거나 제거할 수 있습니다. 작업이 많은 경우 View PowerCLI Cmdlet을 사용해야 합니다.

- Update-UserOwnership
- Remove-UserOwnership

**참고** 원격 디스플레이 프로토콜 설정은 변경하지 마십시오. 이 설정은 항상 데스크톱 풀 생성의 경우와 같습니다.

설정	옵션
기본 디스플레이 프로토콜	VMware Blast
사용자가 프로토콜을 선택할 수 있도록 허용	아니요
3D 렌더러	<ul style="list-style-type: none"> <li>■ 2D, vSGA 또는 vDGA용 vSphere Client를 사용하여 관리</li> <li>■ NVIDIA GRID vGPU</li> </ul>

자세한 내용은 View 관리 가이드를 참조하십시오.

## Linux용 자동화된 전체 클론 데스크톱 풀 생성

Linux 가상 시스템용으로 자동화된 전체 클론 데스크톱 풀을 생성할 수 있습니다. 자동화된 전체 클론 데스크톱 풀을 생성하고 나면 Linux 가상 시스템을 Horizon 7 배포에서 원격 데스크톱으로 사용할 수 있습니다.

### 필수 조건

- Linux 게스트 운영 체제에 Horizon Agent가 설치되어 있는지 확인합니다. “Linux 가상 시스템에 Horizon Agent 설치,” (41 페이지)의 내용을 참조하십시오.
- Winbind 솔루션을 사용하여 Linux 가상 시스템을 Active Directory에 연결할 경우, 가상 시스템 템플릿에서 Winbind 솔루션의 구성을 마쳐야 합니다.
- Winbind 솔루션을 사용할 경우에는 가상 시스템에서 도메인 가입 명령을 실행해야 합니다. 셸 스크립트에 명령을 포함하고 /etc/vmware/viewagent-custom.conf에서 Horizon Agent 옵션 RunOnceScript의 스크립트 경로를 지정합니다. 자세한 내용은 “Linux 데스크톱의 구성 파일에서 옵션 설정,” (47 페이지)의 내용을 참조하십시오.

- vCenter Server가 Horizon 연결 서버에 추가되었는지 확인합니다.

**프로시저**

- 1 게스트 사용자 지정 규격을 생성합니다.

vSphere 가상 시스템 관리 문서의 “vSphere Web Client에서 Linux용 사용자 지정 규격 생성”을 참조하십시오. 규격을 생성할 때는 다음 설정을 올바르게 지정해야 합니다.

설정	값
대상 가상 시스템 OS	Linux
컴퓨터 이름	가상 시스템 이름을 사용합니다.
도메인	View 환경의 도메인을 지정합니다.
네트워크 설정	표준 네트워크 설정을 사용합니다.
기본 DNS	유효한 주소를 지정합니다.

**참고** 게스트 OS 사용자 지정 지원 매트릭스에 대한 자세한 내용은 <http://partnerweb.vmware.com/programs/guestOS/guest-os-customization-matrix.pdf>를 참조하십시오.

- 2 Horizon Administrator에서 **카탈로그 > 데스크톱 풀 > 추가**를 선택합니다.
- 3 **자동화된 데스크톱 풀**을 선택하고 **다음**을 클릭합니다.
- 4 데스크톱 풀의 시스템에 대해 **전용 또는 부동 사용자 할당**을 선택하고 **다음**을 클릭합니다.
- 5 [vCenter Server] 페이지에서 **전체 가상 시스템**을 선택하고 적절한 vCenter Server를 선택한 후 **다음**을 클릭합니다.
- 6 [데스크톱 풀 ID] 페이지에서 데스크톱 풀 ID를 입력하고 **다음**을 클릭합니다.
- 7 [데스크톱 풀 설정] 페이지에서 다음 옵션을 설정하고 **다음**을 클릭합니다.

옵션	설명
기본 디스플레이 프로토콜	VMware Blast
사용자가 프로토콜을 선택할 수 있도록 허용함	아니요
3D 렌더러	2D, vSGA 또는 vDGA 데스크톱용 vSphere Client 및 vGPU 데스크톱용 NVIDIA GRID vGPU를 사용하여 관리

**참고** 풀 설정은 필수입니다. 풀을 설정하지 않으면 데스크톱 연결에 실패할 수 있고 프로토콜 오류가 발생하거나 검은색 화면이 표시됩니다.

- 8 [프로비저닝 설정] 페이지에서 **가상 시스템 이름 지정** 옵션을 설정하고 **다음**을 클릭합니다.

옵션	설명
수동으로 이름 지정	이름을 수동으로 입력합니다.
이름 지정 패턴 사용	예를 들어 LinuxVM-{n}을 지정합니다. 다음과 같은 데스크톱 풀 크기 지정 옵션도 지정해야 합니다. <ul style="list-style-type: none"> <li>■ 최대 시스템 수</li> <li>■ 전원이 켜진 예비 시스템 수</li> </ul>

- 9 [스토리지 최적화] 페이지에서 스토리지 관리 정책을 선택하고 **다음**을 클릭합니다.

- 10 [vCenter 설정] 페이지에서 [찾아보기]를 클릭하고 vCenter Server 설정을 순서대로 선택한 후 다음을 클릭합니다.  
다음 vCenter Server 설정을 건너 뛸 수 없습니다.
  - a 템플릿
  - b VM 폴더 위치
  - c 호스트 또는 클러스터
  - d 리소스 풀
  - e 데이터스토어
- 11 [고급 스토리지 옵션] 페이지에서 적절한 스토리지 옵션을 선택하고 다음을 클릭합니다.
- 12 [게스트 사용자 지정] 페이지에서 Linux의 게스트 사용자 지정을 선택하고 다음을 클릭합니다.
- 13 [완료 준비] 페이지에서 세부 정보를 검토하고 이 마법사가 완료된 후 사용자에게 권한 부여를 선택합니다.
- 14 마침을 클릭합니다.
- 15 사용자에게 데스크톱 풀에 있는 시스템의 사용 권한을 부여하려면 데스크톱 풀을 선택하고 사용 권한 > 권한 추가를 클릭한 후 사용자와 그룹을 추가합니다.
- 16 데스크톱 풀에 있는 모든 Linux 가상 시스템을 사용할 수 있게 될 때까지 기다립니다.

## 브로커 PowerCLI 명령

연결 서버 및 Windows 데스크톱에서 다양한 관리 작업을 수행하기 위한 View PowerCLI cmdlet은 Linux 데스크톱에서도 작동됩니다.

### 수동 데스크톱 풀 생성

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

Linux 데스크톱의 경우 다음 옵션 및 값은 필수입니다.

- DefaultProtocol Blast
- AllowProtocolOverride \$false
- threadRender usevc|vgpu. vGPU 데스크톱의 경우에는 -threadRender vgpu를 사용하고 2D/vSGA/vDGA 데스크톱의 경우에는 -threadRender usevc를 사용합니다.

#### 예제

- 가상 시스템 LinuxVM-01을 사용하여 LinuxDesktop이라는 이름의 부동 Linux 데스크톱 풀을 생성합니다.

```
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
```

- VM 이름이 LinuxVM-으로 시작하는 모든 VM을 사용하여 LinuxDesktop이라는 이름의 전용 Linux vGPU 데스크톱 풀을 생성합니다.

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

- 첫 번째 RHEL 6 x64 VM을 사용하여 부동 Linux 데스크톱 풀을 생성합니다.

```
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
```

## 전체 클론 자동화된 데스크톱 풀 생성

```
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]
```

Linux 데스크톱의 경우 다음 옵션 및 값은 필수입니다.

- DefaultProtocol Blast
- AllowProtocolOverride \$false
- threadRender usevc|vgpu. vGPU 데스크톱의 경우에는 -threadRender vgpu를 사용하고 2D/vSGA 데스크톱의 경우에는 -threadRender usevc를 사용합니다.

### 예

```
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
```

## 데스크톱 풀 사용 권한 추가 또는 제거

- 도메인 mydomain.org의 도메인 사용자 그룹에 LinuxDesktop에 대한 사용 권한을 부여합니다.

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

- LinuxDesktop에서 mydomain.org 도메인의 도메인 사용자 그룹에 대한 사용 권한을 제거합니다.

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

## 전용 데스크톱 풀의 VM에 사용자 할당 또는 VM에서 사용자 제거

- 전용 데스크톱 풀에 있는 LinuxVM-01 VM에 myuser 사용자를 할당합니다.

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

- 전용 데스크톱 풀에 있는 LinuxVM-01 VM에서 myuser 사용자를 제거합니다.

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

## 데스크톱 연결 로그오프

- myuser의 데스크톱 세션에서 로그오프합니다.

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

브로커 PowerCLI cmdlet에 대한 자세한 내용은 View 통합에서 View PowerCLI 사용을 참조하십시오.





# 수동 데스크톱 풀의 Horizon 7 대량 배포

# 8

View Administrator를 사용하여 Linux가 아닌 Windows 데스크톱 시스템 풀을 자동으로 만들 수 있습니다. 그러나 Linux 데스크톱 시스템 풀의 배포를 자동화하는 스크립트를 개발할 수 있습니다.

제공되는 샘플 스크립트는 설명을 위해서만 제공됩니다. VMware는 사용자가 샘플 스크립트를 사용할 때 발생할 수 있는 어떠한 문제에 대해서도 책임지지 않습니다.

이 장에서는 다음 주제에 대해 설명합니다.

- “Linux 데스크톱의 대량 배포 개요,” (65 페이지)
- “Linux 데스크톱의 대량 업그레이드 개요,” (67 페이지)
- “Linux 데스크톱 시스템의 복제를 위한 가상 시스템 템플릿 만들기,” (67 페이지)
- “Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일,” (69 페이지)
- “Linux 가상 시스템을 복제하기 위한 샘플 스크립트,” (69 페이지)
- “복제된 가상 시스템을 AD 도메인에 연결하기 위한 샘플 스크립트,” (73 페이지)
- “SSH를 사용하여 복제된 가상 시스템을 AD 도메인에 연결하기 위한 샘플 스크립트,” (76 페이지)
- “구성 파일을 Linux 가상 시스템에 업로드하기 위한 샘플 스크립트,” (79 페이지)
- “SSH를 사용하여 구성 파일을 Linux 가상 시스템에 업로드하기 위한 샘플 스크립트,” (83 페이지)
- “Linux 데스크톱 시스템에서 Horizon Agent를 업그레이드하기 위한 샘플 스크립트,” (87 페이지)
- “SSH를 사용하여 Linux 가상 시스템에서 Horizon Agent를 업그레이드하기 위한 샘플 스크립트,” (91 페이지)
- “Linux 가상 시스템에서 작업을 수행하기 위한 샘플 스크립트,” (97 페이지)

## Linux 데스크톱의 대량 배포 개요

Linux용 수동 데스크톱의 배포는 몇 가지 단계로 진행됩니다. 데스크톱을 여러 대 배포하는 경우 PowerCLI 스크립트를 사용하면 일부 단계를 자동화할 수 있습니다.

일부 운영 체제에서는 PowerCLI 또는 SSH가 Linux 시스템에서 명령을 실행하도록 할 수 있습니다. 다음 표에서는 두 접근 방식의 차이점을 설명합니다.

PowerCLI	SSH
추가 도구를 설치할 필요가 없습니다.	<ul style="list-style-type: none"> <li>■ Ubuntu의 경우는 <code>sudo apt-get install openssh-server</code> 명령으로 SSH 서버를 설치해야 합니다. RHEL 및 CentOS의 경우는 <code>openssh-server</code>가 기본적으로 설치되지만 방화벽 설정에서 <code>ssh</code>가 허용되는지 확인해야 합니다.</li> <li>■ SSH 클라이언트 애플리케이션 <code>pscp.exe</code> 및 <code>plink.exe</code>를 다운로드하고 PowerCLI 스크립트와 같은 폴더에 넣어야 합니다.</li> </ul>
파일 업로드와 명령 실행이 느립니다.	파일 업로드와 명령 실행이 빠릅니다.
ESXi 호스트의 관리자 자격 증명을 제공해야 합니다.	ESXi 호스트의 관리자 자격 증명을 제공할 필요가 없습니다.
스크립트를 실행하여 Horizon Agent를 설치하는 경우의 관리자 암호나 스크립트를 실행하여 도메인에 연결하는 경우의 AD 사용자 암호에서 특수 문자를 처리할 수 없습니다.	스크립트를 실행하여 Horizon Agent를 설치하는 경우의 관리자 암호나 스크립트를 실행하여 도메인에 연결하는 경우의 AD 사용자 암호에서 특수 문자를 처리할 수 있습니다.

**참고** PowerCLI 기반 및 SSH 기반 스크립트 모두에서 vCenter Server 관리자 및 Linux 관리자의 암호에 사용된 특수 문자를 처리할 수 있습니다. PowerCLI 기반 스크립트에서는 ESXi 호스트 관리자의 암호에 있는 특수 문자도 처리할 수 있습니다. 이 모든 경우에 이스케이프 문자가 필요하지 않습니다.

vSphere PowerCLI에 대한 자세한 내용은

<https://www.vmware.com/support/developer/PowerCLI>를 참조하십시오.

Linux 데스크톱 풀을 대량으로 배포하는 프로세스는 다음 단계를 통해 수행됩니다.

- 1 가상 시스템 템플릿을 생성하고 가상 시스템에 Horizon Agent를 설치합니다.

“Linux 데스크톱 시스템의 복제를 위한 가상 시스템 템플릿 만들기,” (67 페이지)의 내용을 참조하십시오.

- 2 게스트 사용자 지정 규칙을 생성합니다.

vSphere 가상 시스템 관리 문서의 “vSphere Web Client에서 Linux용 사용자 지정 규칙 생성”을 참조하십시오. 규칙을 생성할 때는 다음 설정을 올바르게 지정해야 합니다.

설정	값
대상 가상 시스템 OS	Linux
컴퓨터 이름	가상 시스템 이름을 사용합니다.
도메인	View 환경의 도메인을 지정합니다.
네트워크 설정	표준 네트워크 설정을 사용합니다.
기본 DNS	유효한 주소를 지정합니다.

**참고** 게스트 OS 사용자 지정 지원 매트릭스에 대한 자세한 내용은

<http://partnerweb.vmware.com/programs/guestOS/guest-os-customization-matrix.pdf>를 참조하십시오.

- 3 가상 시스템을 복제합니다.

“Linux 가상 시스템을 복제하기 위한 샘플 스크립트,” (69 페이지)의 내용을 참조하십시오.

- 4 Winbind 솔루션을 사용하고 있는 경우 복제된 VM을 AD(Active Directory) 도메인에 가입시킵니다. 아래의 예제 스크립트를 사용하여 도메인 가입 명령을 실행하거나 템플릿 가상 시스템에 구성된 `/etc/vmware/viewagent-custom.conf`에서 `RunOnceScript` 옵션을 사용할 수 있습니다.

자세한 내용은 “복제된 가상 시스템을 AD 도메인에 연결하기 위한 샘플 스크립트,” (73 페이지) 또는 “SSH를 사용하여 복제된 가상 시스템을 AD 도메인에 연결하기 위한 샘플 스크립트,” (76 페이지)에 나와 있습니다.

- 5 가상 시스템에서 구성 옵션을 업데이트합니다.

자세한 내용은 “구성 파일을 Linux 가상 시스템에 업로드하기 위한 샘플 스크립트,” (79 페이지) 또는 “SSH를 사용하여 구성 파일을 Linux 가상 시스템에 업로드하기 위한 샘플 스크립트,” (83 페이지)에 나와 있습니다.

- 6 데스크톱 풀을 만듭니다.

“Linux용 수동 데스크톱 풀 생성,” (57 페이지)의 내용을 참조하십시오.

전원 켜기, 종료, 다시 시작 또는 가상 시스템 삭제와 같은 작업을 수행하는 샘플 스크립트는 “Linux 가상 시스템에서 작업을 수행하기 위한 샘플 스크립트,” (97 페이지)를 참조하십시오. 이 스크립트는 vCenter Server에서 가상 시스템을 삭제할 수 있습니다.

## Linux 데스크톱의 대량 업그레이드 개요

Linux용 수동 데스크톱의 대량 업그레이드는 몇 가지 단계로 진행됩니다. PowerCLI 스크립트를 사용하여 일부 단계를 자동화할 수 있습니다.

### 관리되지 않는 데스크톱 대량 업그레이드

관리되지 않는 가상 시스템을 관리되거나 관리되지 않는 가상 시스템으로 대량으로 업그레이드하려면 샘플 업그레이드 스크립트를 사용하여 새 Horizon Agent를 기존 가상 시스템에 업로드하고 업그레이드 명령을 실행해야 합니다.

- 관리되지 않는 가상 시스템을 유지하는 경우 기존 데스크톱 풀을 재사용할 수 있습니다.
- 관리되지 않는 가상 시스템에서 관리되는 가상 시스템으로 업그레이드하는 경우 기존 데스크톱 풀을 삭제하고 새 데스크톱 풀을 생성해야 합니다. 자세한 내용은 “Linux 가상 시스템에서 Horizon Agent 업그레이드,” (45 페이지)의 내용을 참조하십시오.

### 관리되는 데스크톱 대량 업그레이드

관리되는 가상 시스템을 대량으로 업그레이드하려면 다음 방법 중 하나를 사용합니다.

방법	설명
템플릿 가상 시스템에서 새 Horizon Agent를 설치하거나 업그레이드하고 스냅샷을 생성합니다.	<ul style="list-style-type: none"> <li>■ 사용자 데이터 및 프로파일이 NFS 서버와 같은 공유 서버에 있지 않는 한, 기존 가상 시스템이 삭제되었으므로 사용자 데이터 및 프로파일은 손실됩니다.</li> <li>■ 가상 시스템 교체 후에 View Administrator의 가상 시스템 상태가 누락될 수 있습니다. 이 문제를 해결하려면 브로커 서비스를 다시 시작해야 합니다.</li> <li>■ 연결된 복제를 사용하는 경우 이 방법을 사용하면 각 가상 시스템에서 데이터가 중복되는 것을 방지할 수 있습니다.</li> </ul>
업그레이드 샘플 스크립트를 사용하여 기존 가상 시스템에 새 Horizon Agent를 업로드하고 업그레이드 명령을 실행합니다.	<ul style="list-style-type: none"> <li>■ 사용자 데이터 및 프로파일은 보존됩니다.</li> <li>■ 연결된 복제를 사용하는 경우 이 방법을 사용하면 각 가상 시스템에서 데이터가 중복됩니다.</li> </ul>

## Linux 데스크톱 시스템의 복제를 위한 가상 시스템 템플릿 만들기

가상 시스템 복제를 수행하기 전에 복제의 기반이 되는 가상 시스템 템플릿을 생성해야 합니다.

### 필수 조건

- 배포가 Linux 데스크톱을 지원하기 위한 요구 사항을 충족하는지 확인하십시오. “Horizon 7 for Linux에 대한 시스템 요구 사항,” (11 페이지)를 참조하십시오.
- vCenter Server에서 가상 시스템을 생성하고 게스트 운영 체제를 설치하는 단계를 숙지하십시오. Horizon 7에서 가상 데스크톱 설정 문서의 “가상 시스템 생성 및 준비”를 참조하십시오.

- 가상 시스템에서 사용할 모니터에 대해 권장되는 비디오 메모리(vRAM) 값을 숙지하십시오. “Horizon 7 for Linux에 대한 시스템 요구 사항,” (11 페이지)를 참조하십시오.
- AD 통합 단계를 숙지합니다. 3장, “Linux 데스크톱의 Active Directory 통합 설정,” (25 페이지)를 참조하십시오.
- Linux에 Horizon Agent를 설치하는 단계를 숙지하십시오. “Horizon Agent 설치 및 Linux 데스크톱 관리” 장을 참조하십시오.
- 필요한 경우 View 구성 파일을 사용하여 옵션을 구성하는 단계를 숙지합니다. 6장, “Linux 데스크톱용 구성 옵션,” (47 페이지)를 참조하십시오.
- 그래픽을 설정하려는 경우에는 관련 단계를 숙지합니다. 4장, “Linux 데스크톱의 그래픽 설정,” (29 페이지)를 참조하십시오.

### 프로시저

- 1 vSphere Web Client 또는 vSphere Client에서 새 가상 시스템을 만듭니다.
- 2 사용자 지정 구성 옵션을 구성합니다.
  - a 가상 시스템을 마우스 오른쪽 버튼으로 클릭한 후 **설정 편집**을 클릭합니다.
  - b vCPU 수와 vMemory 크기를 지정합니다.  
권장 값을 보려면 Linux 배포에 대한 설치 설명서의 지침을 따르십시오.  
예를 들어 Ubuntu 12.04에서는 vMemory 2048MB 및 vCPU 2개를 구성할 것을 권장합니다.
  - c **비디오 카드**를 선택하고 디스플레이 수 및 총 비디오 메모리(vRAM)를 지정하십시오.  
VMware 드라이버를 사용하는 2D 또는 vSGA를 사용할 경우 가상 시스템용 vSphere Web Client에서 vRAM 크기를 설정합니다. NVIDIA 드라이버를 사용하는 vDGA 또는 NVIDIA GRID vGPU 시스템에서는 vRAM 크기가 영향을 미치지 않습니다.  
권장되는 값을 보려면 Horizon 7 for Linux에 대한 시스템 요구 사항의 지침을 따르십시오. Video Memory Calculator를 사용하지 마십시오.
- 3 가상 시스템의 전원을 켜고 Linux 배포를 설치합니다.
- 4 루트 권한이 있는 사용자(예: ViewUser)를 생성합니다. 이 사용자는 Horizon Agent를 설치 및 제거하는 데만 사용됩니다.
- 5 /etc/sudoers를 편집하고 ViewUser ALL=(ALL) NOPASSWD:ALL 줄을 추가합니다.  
/etc/sudoers에 이 줄이 포함되어 있으면 sudo를 ViewUser로 실행하는 데 암호가 필요하지 않습니다. 이 장에 제공된 샘플 스크립트를 실행하여 Horizon Agent를 설치할 경우 ViewUser를 입력으로 지정합니다.
- 6 Linux 배포가 RHEL, CentOS 또는 NeoKylin인 경우 /etc/sudoers를 편집하고 다음 줄을 주석 처리합니다.  
Defaults requiretty  
Defaults !visiblepw
- 7 Linux 배포가 RHEL 7, CentOS 7, SLED 12 또는 SLES 12가 아닌 경우 VMware Tools를 설치합니다.  
RHEL 7, CentOS 7, SLED 12 및 SLES 12에는 기본적으로 Open VM Tools가 설치되어 있습니다.
- 8 Linux 배포가 RHEL 7, CentOS 7 또는 SLES 12인 경우 deployPkg 플러그인을 설치합니다.  
지침은 <http://kb.vmware.com/kb/2075048>에서 확인할 수 있습니다.
- 9 RHEL 및 CentOS의 경우 네트워크 연결 설정 **자동으로 연결**을 사용하도록 설정합니다.

- 10 AD 통합 작업을 수행합니다.
- 11 그래픽 설정 단계를 수행합니다.
- 12 Horizon Agent 설치
 

```
sudo ./install_viewagent.sh -A yes
```
- 13 View 구성 파일을 사용하여 추가 구성을 수행합니다.
- 14 가상 시스템을 종료하고 스냅샷을 생성합니다.

## Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일

Linux 데스크톱을 배포하기 위한 샘플 PowerCLI 스크립트는 데스크톱 시스템에 대한 정보를 포함하는 하나의 입력 파일을 읽습니다.

입력 파일은 csv 형식이며 다음 정보를 포함합니다.

- 데스크톱 가상 시스템 이름
- 상위 가상 시스템 이름
- 게스트 사용자 지정 규칙
- 복제된 데스크톱 시스템이 있는 데이터 저장소
- 데스크톱 시스템을 호스팅하는 ESXi 서버
- 복제에 사용되는 상위 가상 시스템 스냅샷
- 데스크톱 가상 시스템을 삭제할지 여부를 나타내는 플래그(있는 경우)

다음 예에서는 입력 파일에 포함될 수 있는 항목이 표시됩니다.

```
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
```

샘플 스크립트는 이 입력 파일의 이름이 CloneVMs.csv이고 이 파일이 스크립트와 동일한 폴더에 있다고 간주합니다.

## Linux 가상 시스템을 복제하기 위한 샘플 스크립트

다음 샘플 스크립트를 사용자 지정 후 사용하여 VM(가상 시스템)을 개수에 제한 없이 복제할 수 있습니다.

페이지 구분 없이 스크립트 내용을 복사한 후 붙여넣으려면

[https://www.vmware.com/support/pubs/view\\_pubs.html](https://www.vmware.com/support/pubs/view_pubs.html)의 Horizon 7 설명서 페이지에서 이 항목의 HTML 버전을 사용하십시오.

### 스크립트 입력

이 스크립트는 “Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일,” (69 페이지)에 설명된 단일 입력 파일을 읽습니다. 이 스크립트는 또한 다음 정보를 대화형으로 요청합니다.

- vCenter Server의 IP 주소
- vCenter Server의 관리자 로그인 이름
- vCenter Server의 관리자 암호
- 복제 유형(링크 가능 또는 전체)

- vSphere VM 콘솔을 비활성화할지 여부

## 스크립트 내용

```
<#
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 -----

$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
$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 = '.WCloneVMs.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 $csvFile -
header("VMName","Parentvm","CustomSpec","Datastore","Host","FromSnapshot","DeletelfPresent")
foreach ($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 = $line.DeletelfPresent
    if (IsVMExists ($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 tasks 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
```

## 스크립트 실행

스크립트를 실행하면 다음 메시지가 표시됩니다.

```
PowerCLI C:\Wscripts> .WCloneVMs.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
```

복제 프로세스에 소요되는 시간은 데스크톱 시스템 수에 따라 좌우되며 몇 분에서 몇 시간까지 걸릴 수 있습니다. 프로세스가 완료되었는지 확인하려면 vSphere Client에서 마지막 가상 시스템의 전원이 켜져 있고, 고유한 호스트 이름이 있으며, VMware Tools가 실행되고 있는지 확인하십시오.

## 복제된 가상 시스템을 AD 도메인에 연결하기 위한 샘플 스크립트

다음과 같은 샘플 스크립트를 사용자 지정하고 사용하여 복제된 VM(가상 시스템)을 AD(Active Directory) 도메인에 연결할 수 있습니다.

AD 통합에 Winbind 솔루션을 사용하는 경우 복제된 VM에서는 도메인 연결 단계가 실패하기 때문에 이 스크립트를 실행해야 합니다. 이 스크립트에서는 각 VM의 도메인에 연결하는 명령을 실행합니다. OpenLDAP 솔루션을 사용하는 경우에는 이 스크립트를 실행할 필요가 없습니다.

페이지 구분 없이 스크립트 내용을 복사한 후 붙여넣으려면

[https://www.vmware.com/support/pubs/view\\_pubs.html](https://www.vmware.com/support/pubs/view_pubs.html)의 Horizon 7 설명서 페이지에서 이 항목의 HTML 버전을 사용하십시오.

## 스크립트 입력

이 스크립트는 “Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일,” (69 페이지)에 설명된 단일 입력 파일을 읽습니다. 이 스크립트는 또한 다음 정보를 대화형으로 요청합니다.

- vCenter Server의 IP 주소
- vCenter Server의 관리자 로그인 이름
- vCenter Server의 관리자 암호
- ESXi 호스트의 관리자 로그인 이름
- ESXi 호스트의 관리자 암호
- Linux VM의 사용자 로그인 이름

- Linux VM의 사용자 암호
- 시스템을 도메인에 연결할 권한이 있는 AD 사용자의 로그인 이름
- 권한이 있는 AD 사용자의 암호

## 스크립트 내용

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

.DESCRPTION
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
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBSTR($input))
    }
    else
    {
        $input = Read-Host
    }

    [Console]::ResetColor()
    return $input
}
#----- 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
"-----"
$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."
"Please 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 = '.WCloneVMs.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

```

## 스크립트 실행

스크립트를 실행하면 다음 메시지가 표시됩니다.

```

PowerCLI C:\Scripts> .WClonedVMs_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: *****
```

## SSH를 사용하여 복제된 가상 시스템을 AD 도메인에 연결하기 위한 샘플 스크립트

다음과 같은 샘플 스크립트를 사용자 지정하고 사용하여 복제된 VM(가상 시스템)을 AD(Active Directory) 도메인에 연결할 수 있습니다. 이 스크립트에서는 SSH를 사용하여 Linux VM에서 명령을 실행할 수 있습니다.

AD 통합에 Winbind 솔루션을 사용하는 경우 복제된 VM에서는 도메인 연결 단계가 실패하기 때문에 이 스크립트를 실행해야 합니다. 이 스크립트에서는 각 VM의 도메인에 연결하는 명령을 실행합니다. OpenLDAP 솔루션을 사용하는 경우에는 이 스크립트를 실행할 필요가 없습니다.

페이지 구분 없이 스크립트 내용을 복사한 후 붙여넣으려면 [https://www.vmware.com/support/pubs/view\\_pubs.html](https://www.vmware.com/support/pubs/view_pubs.html)의 Horizon 7 설명서 페이지에서 이 항목의 HTML 버전을 사용하십시오.

### 스크립트 입력

이 스크립트는 “Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일,” (69 페이지)에 설명된 단일 입력 파일을 읽습니다. 이 스크립트는 또한 다음 정보를 대화형으로 요청합니다.

- vCenter Server의 IP 주소
- vCenter Server의 관리자 로그인 이름
- vCenter Server의 관리자 암호
- Linux VM의 사용자 로그인 이름
- Linux VM의 사용자 암호
- 시스템을 도메인에 연결할 권한이 있는 AD 사용자의 로그인 이름
- 권한이 있는 AD 사용자의 암호

### 스크립트 내용

```
<#
.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
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBSTR($input))
    }
    else
    {
        $input = Read-Host
    }

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

function Check_SSH_Client
{
    Param($IsPlink, $IsPSCP)
    if ($IsPlink)
    {
        if (Test-Path ".\Wplink.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 ".\Wpscp.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 | .\Wplink.exe -ssh -l $user -pw $password $IP " + "'" + $cmd + "'"
    }
}

```

```

        $output = Invoke-Expression $command
        return $output
    }
    else
    {
        echo yes | .WpLink.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 | .Wpscp.exe -l $User -pw $Password $LocalPath $IP" + ":" + "$DestPath"
    write-host "Upload file: $command"
    Invoke-Expression $command
}

#----- Handle input -----
"-----"
Check_SSH_Client -IsPlink $true -IsPSCP $false
"-----"
$vcAddress = GetInput -prompt "Your vCenter address" -IsPassword $false
$vcAdmin = GetInput -prompt "Your vCenter admin user name" -IsPassword $false
$vcPassword = GetInput -prompt "Your vCenter admin user password" -IsPassword $true
"-----"
$guestUser = GetInput -prompt 'Your VM guest OS user name' -IsPassword $false
$guestPassword = GetInput -prompt 'Your VM guest OS user password' -IsPassword $true
"-----"
$adUser = GetInput -prompt 'Type the AD user name to join the AD' -IsPassword $false
""
"
Please type the AD user password."
[Console]::ForegroundColor = "Yellow"
"Please note that special character should be escaped. For example, $ should be W$"
[Console]::ResetColor()
$adUserPassword = GetInput -prompt 'Your AD user password' -IsPassword $true
"-----"

#$csvFile = Read-Host 'Csv File '
$csvFile = '.WCloneVMs.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

```

## 스크립트 실행

스크립트를 실행하면 다음 메시지가 표시됩니다.

```

PowerCLI C:\wscripts> .\WClonedVMs_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 W$
Your AD user password: *****

```

## 구성 파일을 Linux 가상 시스템에 업로드하기 위한 샘플 스크립트

다음 샘플 스크립트를 사용자 지정 및 사용하여 구성 파일 config 및 viewagent-custom.conf를 여러 Linux VM(가상 시스템)에 업로드할 수 있습니다.

페이지 구분 없이 스크립트 내용을 복사한 후 붙여넣으려면

[https://www.vmware.com/support/pubs/view\\_pubs.html](https://www.vmware.com/support/pubs/view_pubs.html)의 Horizon 7 설명서 페이지에서 이 항목의 HTML 버전을 사용하십시오.

## 스크립트 입력

이 스크립트는 “Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일,” (69 페이지)에 설명된 단일 입력 파일을 읽습니다. 이 스크립트는 또한 다음 정보를 대화형으로 요청합니다.

- vCenter Server의 IP 주소

- vCenter Server의 관리자 로그인 이름
- vCenter Server의 관리자 암호
- ESXi 호스트의 관리자 로그인 이름
- ESXi 호스트의 관리자 암호
- Linux VM의 사용자 로그인 이름
- Linux VM의 사용자 암호

## 스크립트 내용

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

```

## 스크립트 실행

스크립트를 실행하면 다음 메시지가 표시됩니다.

```
PowerCLI C:\wscripts> .\WUpdateOptionFile.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: *****
```

## SSH를 사용하여 구성 파일을 Linux 가상 시스템에 업로드하기 위한 샘플 스크립트

다음 샘플 스크립트를 사용자 지정 및 사용하여 구성 파일 config 및 viewagent-custom.conf를 여러 Linux VM(가상 시스템)에 업로드할 수 있습니다. 이 스크립트에서는 SSH를 사용하여 Linux VM에서 명령을 실행할 수 있습니다.

페이지 구분 없이 스크립트 내용을 복사한 후 붙여넣으려면

[https://www.vmware.com/support/pubs/view\\_pubs.html](https://www.vmware.com/support/pubs/view_pubs.html)의 Horizon 7 설명서 페이지에서 이 항목의 HTML 버전을 사용하십시오.

### 스크립트 입력

이 스크립트는 “Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일,” (69 페이지)에 설명된 단일 입력 파일을 읽습니다. 이 스크립트는 또한 다음 정보를 대화형으로 요청합니다.

- vCenter Server의 IP 주소
- vCenter Server의 관리자 로그인 이름
- vCenter Server의 관리자 암호
- Linux VM의 사용자 로그인 이름
- Linux VM의 사용자 암호

### 스크립트 내용

```
<#
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
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBSTR($input))
    }
    else
    {
        $input = Read-Host
    }

    [Console]::ResetColor()
    return $input
}
function Check_SSH_Client
{
    Param($IsPlink, $IsPSCP)
    if ($IsPlink)
```

```

    {
        if (Test-Path ".Wplink.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 ".Wpscp.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 | .Wplink.exe -ssh -l $user -pw $password $IP " + "'" + $cmd + "'"
        $output = Invoke-Expression $command
        return $output
    }
    else
    {
        echo yes | .Wplink.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 | .Wpscp.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 = '.WCloneVMs.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

```

## 스크립트 실행

스크립트를 실행하면 다음 메시지가 표시됩니다.

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

## Linux 데스크톱 시스템에서 Horizon Agent를 업그레이드하기 위한 샘플 스크립트

다음 샘플 스크립트를 사용자 지정한 후 사용하여 여러 Linux VM(가상 시스템)에서 Horizon Agent를 업그레이드할 수 있습니다.

이 스크립트에서는 Horizon Agent를 설치하기 전에 각 VM에 설치 관리자 tar ball을 업로드합니다. 업로드 작업은 시간이 오래 걸릴 수 있으며, 관련된 VM의 수가 많고 네트워크 속도가 느린 경우에는 특히 더 그렇습니다. 시간을 절약하려는 경우에는 SSH를 사용하는 스크립트를 실행하거나 파일을 업로드할 필요가 없도록 각 VM에서 사용할 수 있는 공유 위치에 설치 관리자 tar ball을 넣을 수 있습니다.

페이지 구분 없이 스크립트 내용을 복사한 후 붙여넣으려면

[https://www.vmware.com/support/pubs/view\\_pubs.html](https://www.vmware.com/support/pubs/view_pubs.html)의 Horizon 7 설명서 페이지에서 이 항목의 HTML 버전을 사용하십시오.

## 스크립트 입력

이 스크립트는 “Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일,” (69 페이지)에 설명된 단일 입력 파일을 읽습니다. 이 스크립트는 또한 다음 정보를 대화형으로 요청합니다.

- Horizon Agent EULA(최종 사용자 라이선스 계약) 수락
- vCenter Server의 IP 주소
- vCenter Server의 관리자 로그인 이름
- vCenter Server의 관리자 암호
- ESXi 호스트의 관리자 로그인 이름
- ESXi 호스트의 관리자 암호
- Linux 게스트 운영 체제의 사용자 로그인 이름
- Linux 게스트 운영 체제의 사용자 암호
- Horizon Agent tar ball 경로
- 관리되는 VM으로 업그레이드
- 스마트 카드 리디렉션 기능 설치

## 스크립트 내용

```

<#
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
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBSTR($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
}
$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
"-----"
$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 = '.WCloneVMs.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

```

## 스크립트 실행

스크립트를 실행하면 다음 메시지가 표시됩니다.

```

PowerCLI C:\wscripts> .\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

```

## SSH를 사용하여 Linux 가상 시스템에서 Horizon Agent를 업그레이드하기 위한 샘플 스크립트

다음 샘플 스크립트를 사용자 지정 한 후 사용하여 여러 Linux VM(가상 시스템)에서 Horizon Agent를 업그레이드할 수 있습니다. 이 스크립트에서는 SSH를 사용하여 Linux VM에서 명령을 실행할 수 있습니다.

페이지 구분 없이 스크립트 내용을 복사한 후 붙여넣으려면

[https://www.vmware.com/support/pubs/view\\_pubs.html](https://www.vmware.com/support/pubs/view_pubs.html)의 Horizon 7 설명서 페이지에서 이 항목의 HTML 버전을 사용하십시오.

## 스크립트 입력

이 스크립트는 “Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일,” (69 페이지)에 설명된 단일 입력 파일을 읽습니다. 이 스크립트는 또한 다음 정보를 대화형으로 요청합니다.

- Horizon Agent EULA(최종 사용자 라이선스 계약) 수락

- vCenter Server의 IP 주소
- vCenter Server의 관리자 로그인 이름
- vCenter Server의 관리자 암호
- ESXi 호스트의 관리자 로그인 이름
- ESXi 호스트의 관리자 암호
- Linux 게스트 운영 체제의 사용자 로그인 이름
- Linux 게스트 운영 체제의 사용자 암호
- Horizon Agent tar ball 경로
- 관리되는 VM으로 업그레이드
- 스마트 카드 리디렉션 기능 설치

## 스크립트 내용

```
<#
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
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBSTR($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 ".Wpscp.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 | .Wplink.exe -ssh -l $user -pw $password $IP " + "'" + $cmd + "'"
        $output = Invoke-Expression $command
        return $output
    }
    else
    {
        echo yes | .Wplink.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 | .Wpscp.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
$false
if ($acceptEULA -ne "yes")
{
    write-host -ForegroundColor Red "You need accept the EULA with 'yes'(case sensitive)"
    exit
}
$vcAddress = GetInput -prompt "Your vCenter address" -IsPassword $false
$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
"-----"
$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 = '.WCloneVMs.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

```

## 스크립트 실행

스크립트를 실행하면 다음 메시지가 표시됩니다.

```

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

```



## Linux 가상 시스템에서 작업을 수행하기 위한 샘플 스크립트

다음 샘플 스크립트를 사용자 지정 후 사용하여 여러 Linux VM(가상 시스템)에서 작업을 수행할 수 있습니다. 작업에는 VM의 전원 켜기, 전원 끄기, 종료, 다시 시작 및 삭제가 포함됩니다.

이 스크립트는 vCenter Server에서는 가상 시스템을 삭제할 수 있지만 View에서는 삭제할 수 없습니다.

페이지 구분 없이 스크립트 내용을 복사한 후 붙여넣으려면

[https://www.vmware.com/support/pubs/view\\_pubs.html](https://www.vmware.com/support/pubs/view_pubs.html)의 Horizon 7 설명서 페이지에서 이 항목의 HTML 버전을 사용하십시오.

### 스크립트 입력

이 스크립트는 “Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일,” (69 페이지)에 설명된 단일 입력 파일을 읽습니다. 이 스크립트는 또한 다음 정보를 대화형으로 요청합니다.

- vCenter Server의 IP 주소
- vCenter Server의 관리자 로그인 이름
- vCenter Server의 관리자 암호
- 전원 켜기, 전원 끄기, 게스트 종료, VM 다시 시작, VM 게스트 다시 시작 또는 VM 삭제와 같은 수행할 작업입니다.
- VM에서 작업 사이에 기다리는 시간(초)입니다.

### 스크립트 내용

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

```

## 스크립트 실행

스크립트를 실행하면 다음 메시지가 표시됩니다.

```

PowerCLI C:\wscripts> .\WVMOperations.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

VM 전원을 켜고, 다시 시작하고, VM 게스트를 다시 시작하는 작업의 경우, 가상 시스템 사이에 20초
이상의 대기 시간을 지정하여 일부 작업의 실패를 초래할 수 있는 부트 스톱 상황을 피하십시오.

```

## Linux 데스크톱 문제 해결

Linux 데스크톱을 관리할 때 문제가 발생할 수 있습니다. 다양한 절차에 따라 문제를 진단하고 수정할 수 있습니다.

이 장에서는 다음 주제에 대해 설명합니다.

- “Horizon 7 for Linux 시스템에 대한 진단 정보 수집,” (101 페이지)
- “원격 데스크톱과 클라이언트 호스트 간의 복사 및 붙여넣기 문제 해결,” (102 페이지)
- “들어오는 TCP 연결을 허용하도록 Linux 방화벽 구성,” (102 페이지)
- “View Agent에서 iPad Pro Horizon Client에 연결하지 못함,” (102 페이지)
- “SLES 12 SP1 데스크톱은 드래그 앤 드롭 이후 자동 새로 고침을 수행하지 않음,” (103 페이지)
- “SSO에서 PowerOff 에이전트에 연결하지 못함,” (103 페이지)
- “Linux용 수동 데스크톱 풀을 생성한 후에 VM에 연결할 수 없음,” (103 페이지)

### Horizon 7 for Linux 시스템에 대한 진단 정보 수집

진단 정보를 수집하여 VMware 기술 지원에서 Horizon 7 for Linux 시스템의 문제를 진단하고 해결하도록 도움을 줄 수 있습니다. 시스템의 구성 정보 및 로그를 압축된 tarball에 수집하는 DCT(데이터 수집 도구) 번들을 만듭니다.

#### 프로시저

- 1 필요한 권한이 있는 사용자로 Linux 가상 시스템에 로그인합니다.
- 2 명령 프롬프트를 열고 `dct-debug.sh` 스크립트를 실행합니다.

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

이 스크립트는 DCT 번들이 포함된 tarball을 생성합니다. 예:

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

tarball은 스크립트가 실행된 디렉토리(현재 작업 디렉토리)에 생성됩니다.

## 원격 데스크톱과 클라이언트 호스트 간의 복사 및 붙여넣기 문제 해결

원격 데스크톱과 클라이언트 호스트 사이에서 복사 및 붙여넣기를 수행하려면 최대 지원 데이터 1MB에서 3초 이상이 걸립니다. 작은 크기의 데이터를 복사 및 붙여넣기할 때에는 이 문제가 발생하지 않습니다.

### 문제점

SLED 11 SP3/SP4 데스크톱에 1개의 vCPU와 1GB의 메모리를 구성할 경우 원격 데스크톱과 로컬 클라이언트 호스트 사이에서의 복사 및 붙여넣기 시간이 3초 이상 걸릴 수 있습니다.

### 원인

복사 및 붙여넣기가 지연되는 것은 SLED 11 SP3/SP4의 오래된 운영 체제 API 때문일 수 있습니다.

### 해결 방법

- ◆ SLED 11 SP3/SP4에 2개의 vCPU와 2GB의 메모리를 구성합니다.

## 들어오는 TCP 연결을 허용하도록 Linux 방화벽 구성

사용자가 Linux 데스크톱에 연결할 수 있도록 하려면 데스크톱은 Horizon Client 디바이스, 보안 서버 및 View 연결 서버에서 들어오는 TCP 연결을 수락할 수 있어야 합니다.

Ubuntu 및 Kylin 배포에서 iptables 방화벽은 기본적으로 입력 정책 ACCEPT를 사용하여 구성됩니다.

RHEL 및 CentOS 배포에서는 가능한 경우 Horizon Agent 설치 관리자 스크립트가 입력 정책 ACCEPT를 사용하여 iptables 방화벽을 구성합니다.

RHEL 또는 CentOS 게스트 운영 체제의 iptables에는 Blast 포트 22443에서 들어오는 새 연결에 대해 입력 정책 ACCEPT가 지정되어 있습니다.

BSG가 활성화되면 보안 서버 또는 View 연결 서버의 BSG를 통해 Horizon Client 디바이스에서 Linux 데스크톱으로의 클라이언트 연결이 설정됩니다. BSG가 활성화되지 않으면 Horizon Client 디바이스에서 Linux 데스크톱으로 직접 연결이 설정됩니다.

## View Agent에서 iPad Pro Horizon Client에 연결하지 못함

iPad Pro Horizon Client에서 다시 시작 또는 종료를 수행한 후에 SUSE View Agent에 다시 연결하지 못합니다.

### 문제점

iPad Pro Horizon Client에서 SUSE 가상 시스템을 다시 시작하거나 종료하면 데스크톱이 응답하지 않습니다. View Agent의 연결을 해제하지 못합니다.

### 원인

SUSE 시스템은 다시 시작 또는 종료 작업 후에 Horizon Client에 메시지를 제대로 전송하지 못할 수 있습니다.

### 해결 방법

- ◆ iPad Pro Horizon Client에서 데스크톱 연결을 수동으로 해제합니다.

## SLES 12 SP1 데스크톱은 드래그 앤 드롭 이후 자동 새로 고침을 수행하지 않음

SLES 12 SP1은 gnome 터미널을 드래그 앤 드롭할 때 다중 모니터 모드에서 자동 새로 고침을 수행하지 않습니다.

### 문제점

다중 모니터 모드에서 SLES 12 SP1을 실행한 후 창 모드로 돌아갈 경우 gnome 터미널을 드래그 앤 드롭하면 데스크톱은 자동으로 새로 고쳐지지 않습니다.

### 원인

gnome 터미널은 드래그 앤 드롭 작업에 응답하지 않습니다.

### 해결 방법

- 1 gnome-shell을 종료합니다.
  - ◆ `kill -9 `pidof gnome-shell``
- 2 gnome-shell을 다시 시작합니다.

## SSO에서 PowerOff 에이전트에 연결하지 못함

SSO에서 PowerOff 에이전트에 연결하지 못합니다.

### 문제점

브로커로 로그인하고 에이전트에 연결할 경우 SSO는 poweroff 에이전트에 연결하지 못합니다.

### 해결 방법

- ◆ 데스크톱에 수동으로 로그인하거나 에이전트 연결을 끊은 후 다시 연결합니다.

## Linux용 수동 데스크톱 풀을 생성한 후에 VM에 연결할 수 없음

가상 시스템 상태가 응답하지 않습니다.

### 문제점

수동 데스크톱 풀을 생성한 후에 가상 시스템 상태가 [에이전트 대기]이거나 [연결할 수 없음]일 수 있습니다.

### 원인

가상 시스템 상태를 [연결할 수 없음] 또는 [에이전트 대기]로 만드는 몇 가지 사용자 오류 구성 또는 설정이 있을 수 있습니다.

- `machine.id` 옵션이 가상 시스템 `vmx` 구성 파일에 있는지 확인합니다.
  - 이 옵션이 없는 경우 가상 시스템이 데스크톱 풀에 올바르게 추가되었는지 확인합니다. 데스크톱 풀을 다시 생성하여 브로커가 `vmx` 구성 파일에 이 옵션을 다시 쓸 수 있도록 합니다.
- VMware Tool 또는 Open VM Tool이 올바르게 설치되어 있는지 확인합니다.

VMware Tool 또는 Open VM Tool 설치 단계가 올바르게 수행되지 않으면 `vmware-rpctool` 명령이 Linux 가상 시스템의 PATH에 없는 것일 수 있습니다. 지침에 따라 VMware Tool 또는 Open VM Tool을 설치합니다.

설치를 마친 후에 이 명령을 실행합니다.

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

machine.id 값은 가상 시스템 vmx 구성 파일에서 나열됩니다.

- 브로커의 FQDN을 에이전트 Linux 가상 시스템의 IP 주소로 확인할 수 있는지 검토합니다.



# 색인

## A

Active Directory, Linux와 통합 25

## D

DIGEST-MD5 41

## H

Horizon 7 for Linux, 대량 배포 65

Horizon Agent

Linux 가상 시스템에 설치 41

Linux 가상 시스템에서 업그레이드 45

Linux 설치 명령 옵션 42

## I

install\_viewagent.sh, 명령 사용법 및 옵션 42

## K

Kerberos 41

## L

Linux 가상 시스템, 데스크톱 배포 준비 19

Linux 데스크톱

AD 통합 설정 25

Horizon Agent 설치 41

Linux 데스크톱 풀 생성 및 관리 57

구성 47

기능 7

기능 및 요구 사항 7

Linux 시스템

vSphere 콘솔 디스플레이 56

데스크톱 5

데스크톱 사용 준비 20

데스크톱 설정 단계 개요 10

데스크톱 풀에 추가 57

망화벽 구성 102

자동화된 데스크톱 풀에 추가 59

진단 정보 수집 101

## N

NVIDIA GRID vGPU

Linux 데스크톱용 VIB 설치 29

가상 GPU 유형 31

NVIDIA GRID vGPU 디스플레이 드라이버, 설치 32

NVIDIA vGPU, 공유 PCI 디바이스 구성 31

NVIDIA 디스플레이 드라이버, 설치 확인 33, 37

## O

OpenLDAP 25

## R

RHEL

vDGA에 대해 구성 34

vGPU용으로 구성 29

vSGA에 대한 구성 38

## S

sso 실패 103

SSSD LDAP 인증 25

## V

vDGA

Linux 시스템에 PCI 디바이스 추가 35

호스트에서 DirectPath I/O 활성화 34

vDGA(가상 전용 그래픽 가속화), RHEL 6 용 34

vDGA용 NVIDIA 디스플레이 드라이버, 설치 35

vGPU, 지원되는 RHEL 배포 구성 29

View Agent, Linux 운영 체제 요구 사항 11

VMware Tools, Linux 게스트에서 구성 20

vSGA

Linux 데스크톱용 VIB 설치 38

Linux 비디오 카드에서 3D 설정 구성 39

Linux 시스템에서 확인 39

RHEL 7용 38

vSphere 콘솔, 표시 안 함 56

## W

Winbind 25

## 가

가상 시스템, Linux 데스크톱용 생성 19

구성 43

구성 옵션

Linux 데스크톱에 대한 클라이언트 드라이브 리디렉션의 예 55

Linux 데스크톱용 Blast 설정 예) 55

단일 로그인(SSO) 47

무손실 PNG 모드 47

오디오 출력 47

원손 사용자용 마우스 **47**  
클립보드 리디렉션 **47**  
그래픽 **29**

**ㄷ**

단일 로그인 **26**  
대량 배포  
개요 **65**  
복제용 가상 시스템 템플릿 생성 **67**  
샘플 스크립트에 대한 입력 파일 **69**  
대량 업그레이드 **67**  
대상 **5**  
데스크톱 풀 생성, Linux 가상 시스템 **57**

**ㄹ**

문제 해결 **101 - 103**  
문제 해결, 연결 해제 **102**

**ㅂ**

방화벽, Linux 데스크톱 **102**  
브로커 powercli **61**

**ㅅ**

샘플 스크립트  
Horizon Agent 업그레이드 **87**  
SSH를 사용하여 Horizon Agent 업그레이드 **91**  
SSH를 사용하여 구성 파일 업로드 **83**  
SSH를 사용하여 데스크톱 시스템 복제 **76**  
VM에서 작업 수행 **97**  
구성 파일 업로드 **79**  
데스크톱 시스템 복제 **69, 73**  
입력 파일 **69**  
설치 **22**  
수동 데스크톱 관리 **58**  
스마트 카드 리디렉션 **26**

**ㅇ**

연결할 수 없는 VM **103**  
용어집 **5**

**ㅈ**

자동화된 전체 클론 데스크톱 풀 생성, Linux 가상 시스템 **59**  
제거 **46**