

Horizon 7 for Linux 데스크 톱 설정

2020년 3월

VMware Horizon 7 7.12



vmware®

다음 VMware 웹 사이트에서 최신 기술 문서를 확인할 수 있습니다.

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

본 문서에 대한 의견이 있으시면 다음 주소로 피드백을 보내주십시오.

docfeedback@vmware.com

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

VMware 코리아
서울시 강남구
영동대로 517
아셈타워 13층
(우) 06164
전화: +82 2 3016 6500
팩스: +82 2 3016 6501
www.vmware.com/kr

Copyright © 2016-2020 VMware, Inc. All rights reserved. [저작권 및 상표 정보](#)

목차

Linux 데스크톱용 Horizon 7 설정 6

1 기능 및 시스템 요구 사항 7

- Horizon Linux 데스크톱의 기능 7
- Horizon 7 for Linux 데스크톱의 구성 단계 개요 13
- Horizon 7 for Linux에 대한 시스템 요구 사항 14
 - 2D 그래픽을 위한 가상 시스템 설정 24
 - Linux 데스크톱에서 세션 공동 작업 구성 24

2 데스크톱 배포를 위해 Linux 가상 시스템 준비 28

- 가상 시스템 생성 및 Linux 설치 28
- 원격 데스크톱 배포를 위한 Linux 시스템 준비 29
- Horizon Agent에 대한 종속성 패키지 설치 31

3 Linux 데스크톱에 대한 Active Directory 통합 및 사용자 인증 기능 설정 33

- Linux와 Active Directory 통합 33
 - OpenLDAP 서버 패스스루 인증 사용 34
 - Microsoft Active Directory에 대해 SSSD LDAP 인증 설정 34
 - Winbind 도메인 가입 솔루션 사용 34
 - PBISO(PowerBroker Identity Services Open) 인증 구성 35
 - Samba 오프라인 도메인 가입 구성 36
 - RHEL/CentOS 8.x용 Realmd 가입 솔루션 사용 38
- Single Sign-On 설정 39
- 스마트 카드 리더렉션 설정 40
 - RHEL 8.x 데스크톱에 대한 스마트 카드 리더렉션 구성 42
 - RHEL 7.x/6.x 데스크톱에 대한 스마트 카드 리더렉션 구성 47
 - Ubuntu 데스크톱에 대한 스마트 카드 리더렉션 구성 53
 - SLED/SLES 데스크톱에 대한 스마트 카드 리더렉션 구성 63
- Linux 데스크톱용 True SSO 설정 69
 - RHEL/CentOS 8.x 데스크톱에서 True SSO 구성 70
 - RHEL/CentOS 7.x 데스크톱에 대한 True SSO 구성 72
 - Ubuntu 데스크톱에 대한 True SSO 구성 76
 - SLED/SLES 데스크톱에 대한 True SSO 구성 81

4 Linux 데스크톱의 그래픽 설정 85

- vGPU에 대해 지원되는 Linux 배포 구성 85

ESXi 호스트에서 NVIDIA GRID vGPU 그래픽 카드용 VIB 설치	86
Linux 가상 시스템에서 vGPU용으로 공유 PCI 디바이스 구성	87
NVIDIA GRID vGPU 디스플레이 드라이버 설치	88
NVIDIA 디스플레이 드라이버의 설치 여부 확인	89
vDGA용 RHEL 6.x 구성	90
호스트에서 NVIDIA GRID용 DirectPath I/O 사용	90
RHEL 6.x 가상 시스템에 vDGA 패스스루 디바이스 추가	90
vDGA용 NVIDIA 디스플레이 드라이버 설치	91
NVIDIA 디스플레이 드라이버의 설치 여부 확인	93
5 Horizon Agent 설치	95
Linux 가상 시스템에 Horizon Agent 설치	95
install_viewagent.sh 명령줄 옵션	97
Linux Agent용 인증서 구성	98
Linux 가상 시스템에서 Horizon Agent 업그레이드	99
Linux 가상 시스템에서 Horizon Agent 업그레이드	100
Horizon 7 for Linux 시스템 제거	101
6 Linux 데스크톱용 구성 옵션	102
Linux 데스크톱의 구성 파일에서 옵션 설정	102
스마트 정책 사용	112
스마트 정책 요구 사항	112
Dynamic Environment Manager 설치	113
Dynamic Environment Manager 구성	113
Horizon 스마트 정책 설정	113
Horizon 스마트 정책 정의에 조건 추가	114
Dynamic Environment Manager에 Horizon 스마트 정책 생성	114
Linux 데스크톱용 Blast 설정 예	116
Linux 데스크톱에 대한 클라이언트 드라이브 리디렉션 옵션의 예	117
7 Linux 데스크톱 풀 생성 및 관리	118
Linux용 수동 데스크톱 풀 생성	118
Linux 데스크톱 풀 관리	119
Linux용 자동화된 전체 클론 데스크톱 풀 생성	121
Linux용 인스턴트 클론 플로팅 데스크톱 풀 생성	123
브로커 PowerCLI 명령	127
8 수동 데스크톱 풀의 Horizon 7 대량 배포	130
Linux 데스크톱의 대량 배포 개요	130
Linux 데스크톱의 대량 업그레이드 개요	132

Linux 데스크톱 시스템의 복제를 위한 가상 시스템 템플릿 만들기	133
Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일	135
Linux 가상 시스템을 복제하기 위한 샘플 스크립트	136
복제된 가상 시스템을 AD 도메인에 연결하기 위한 샘플 스크립트	140
SSH를 사용하여 복제된 가상 시스템을 AD 도메인에 연결하기 위한 샘플 스크립트	143
구성 파일을 Linux 가상 시스템에 업로드하기 위한 샘플 스크립트	146
SSH를 사용하여 구성 파일을 Linux 가상 시스템에 업로드하기 위한 샘플 스크립트	150
Linux 데스크톱 시스템에서 Horizon Agent를 업그레이드하기 위한 샘플 PowerCLI 스크립트	154
SSH를 사용하여 Linux 가상 시스템에서 Horizon Agent를 업그레이드하기 위한 샘플 스크립트	159
Linux 가상 시스템에서 작업을 수행하기 위한 샘플 스크립트	164

9 Linux 데스크톱 문제 해결 169

Horizon Console에서 Horizon Help Desk Tool 사용	169
Horizon Console에서 Horizon Help Desk Tool 시작	170
Horizon Help Desk Tool에서 사용자 문제 해결	170
Horizon Help Desk Tool에 대한 세션 세부 정보	173
Horizon Help Desk Tool에 대한 세션 프로세스	176
Horizon Help Desk Tool에서 Linux 데스크톱 세션 문제 해결	177
Horizon 7 for Linux 시스템에 대한 진단 정보 수집	178
Horizon Agent에서 iPad Pro Horizon Client에 연결 해제하지 못함	179
SLES 12 SP1 데스크톱이 자동으로 새로 고쳐지지 않음	179
SSO에서 PowerOff 에이전트에 연결하지 못함	179
Linux용 수동 데스크톱 풀을 생성한 후에 VM에 연결할 수 없음	180

Linux 데스크톱용 Horizon 7 설정

"Horizon 7 for Linux 데스크톱 설정" 문서는 Linux 데스크톱용 VMware Horizon[®] 7으로 사용하도록 Linux 가상 시스템을 설정하는 방법에 대한 정보를 제공합니다. 이러한 정보에는 Linux 게스트 운영 체제 준비, 가상 시스템에 Horizon Agent 설치 및 Horizon 7 배포에서 사용하기 위해 Horizon Console에서 시스템 구성 등에 대한 내용이 포함됩니다.

대상

이 정보는 Linux 게스트 운영 체제에서 실행되는 원격 데스크톱을 구성 및 사용하려는 모든 사용자를 위해 작성되었습니다. 이 정보는 가상 시스템 기술과 데이터 센터 운영을 잘 아는 숙련된 Linux 시스템 관리자를 대상으로 작성되었습니다.

기능 및 시스템 요구 사항

1

Horizon 6.2.x 이상에서는 사용자가 Linux 운영 체제를 실행하는 원격 데스크톱에 연결할 수 있습니다.

본 장은 다음 항목을 포함합니다.

- [Horizon Linux 데스크톱의 기능](#)
- [Horizon 7 for Linux 데스크톱의 구성 단계 개요](#)
- [Horizon 7 for Linux에 대한 시스템 요구 사항](#)

Horizon Linux 데스크톱의 기능

다음 목록에서는 Horizon Linux 데스크톱에 지원되는 핵심 기능을 보여줍니다.

Linux 데스크톱에서 지원되는 기능

Active Directory 통합

다음 Linux 배포를 실행하는 인스턴트 클론 데스크톱은

PBISO(PowerBroker Identity Services Open)를 사용하여 Active Directory와의 오프라인 도메인 가입을 수행할 수 있습니다.

- Ubuntu 16.04 및 18.04
- SLED/SLES 12.x

자세한 내용은 [Linux와 Active Directory 통합](#)에서 "PBISO(PowerBroker Identity Services Open) 인증" 섹션을 참조하십시오.

다음 Linux 배포를 실행하는 인스턴트 클론 데스크톱은 Samba를 사용하여 Active Directory와의 오프라인 도메인 가입을 수행할 수 있습니다.

- Ubuntu 16.04 및 18.04
- RHEL 7.3, 8.0 및 8.1

오디오 입력

클라이언트 호스트에서 원격 Linux 데스크톱으로의 오디오 입력 리디렉션이 지원됩니다. 이 기능은 USB 리디렉션 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 다음과 같은 **Linux** 배포에서 지원됩니다.

- **Ubuntu 16.04 x64(MATE 또는 Gnome Flashback(Metacity) 데스크톱 환경 포함)**
- **Ubuntu 18.04 x64(MATE 또는 Gnome Ubuntu 데스크톱 환경 포함)**
- **RHEL 7.x Workstation x64(KDE 또는 Gnome 데스크톱 환경 포함)**
- **RHEL 8.x Workstation x64(Gnome 데스크톱 환경 포함)**
- **SLED/SLES 12.x SP3 x64**

오디오 출력

오디오 출력 리디렉션이 지원됩니다. 이 기능은 기본적으로 사용하도록 설정됩니다. 이 기능을 사용하지 않도록 설정하려면

`RemoteDisplay.allowAudio` 옵션을 **false**로 설정해야 합니다. **Chrome** 및 **Firefox** 브라우저를 사용하여 액세스할 경우 **VMware Horizon HTML Access**는 **Linux** 데스크톱에 대한 오디오 출력 지원을 제공합니다.

자동화된 전체 클론 데스크톱 풀

Linux 데스크톱에 대한 자동화된 전체 클론 데스크톱 풀을 생성할 수 있습니다.

클라이언트 드라이브 리디렉션

CDR(클라이언트 드라이브 리디렉션) 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.

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.

FIPS 140-2 모드

FIPS(Federal Information Processing Standard) 140-2 모드 지원은 아직 **NIST CMVP(Cryptographic Module Validation Program)**를 통해 유효성이 검사되지 않았지만 **Linux** 데스크톱에 대해 사용할 수 있습니다.

Linux용 **Horizon 7 Agent**는 **FIPS 140-2** 규정 준수를 위해 설계된 암호화 모듈을 구현합니다. 이러한 모듈은 **CMVP** 인증서 **#2839** 및 **#2866**에 나열된 운영 환경에서 유효성이 검사된 후 이 플랫폼에 이식되었습니다. 그렇지만 **VMware**의 **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 **1.2**가 필요합니다.

헬프 데스크 도구

Horizon Help Desk Tool은 Linux 데스크톱 세션 문제를 해결하는 데 사용할 수 있는 웹 애플리케이션입니다. Horizon Help Desk Tool을 사용하여 Horizon 7 사용자 세션의 상태를 가져오고 문제 해결 및 유지 보수 작업을 수행할 수 있습니다. [Horizon Console](#)에서 [Horizon Help Desk Tool 사용](#)의 내용을 참조하십시오.

Horizon 스마트 정책

VMware Dynamic Environment Manager™ 9.4 이상을 사용하여 특정 원격 Linux 데스크톱에서 USB 리디렉션, 클립보드 리디렉션 및 클라이언트 드라이브 리디렉션 기능의 동작을 제어하는 Horizon 스마트 정책을 생성할 수 있습니다. [스마트 정책 사용](#)의 내용을 참조하십시오.

H.264 인코더

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

H.264 인코더는 하드웨어 H.264 지원 및 소프트웨어 인코더 지원을 모두 포함합니다. 하드웨어 H.264가 지원되려면 다음 요구 사항이 충족되어야 합니다.

- vGPU가 NVIDIA 그래픽 카드로 구성되어 있습니다.
- NVIDIA 드라이버 384 시리즈 이상이 NVIDIA 그래픽 카드에 설치되어 있습니다.

시스템이 위의 요구 사항을 충족할 경우 Horizon 7 for Linux는 하드웨어 H.264 인코더를 사용합니다. 그렇지 않으면 소프트웨어 H.264 인코더가 사용됩니다.

인스턴트 클론 부동 데스크톱 풀

Linux 데스크톱에 대한 인스턴트 클론 부동 데스크톱 풀을 생성할 수 있습니다. 이 기능은 다음과 같은 Linux 배포가 설치되어 있는 시스템에서만 지원됩니다.

- Ubuntu 16.04 및 18.04
- RHEL 7.1 이상
- RHEL 8.x
- SLED/SLES 12.x

자세한 내용은 [Linux용 인스턴트 클론 플로팅 데스크톱 풀 생성](#)의 내용을 참조하십시오.

K 데스크톱 환경

KDE(K 데스크톱 환경)는 다음 Linux 배포에서 지원됩니다.

- CentOS 6.x 및 7.x
- RHEL 6.x 및 7.x

키보드 레이아웃 및 로캘 동기화

■ Ubuntu 16.04 및 18.04

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 VMware Horizon**에서만 지원되고 영어, 프랑스어, 독일어, 일본어, 한국어, 스페인어, 중국어 간체 및 중국어 번체 로캘에서만 지원됩니다.

무손실 PNG

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

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.

MATE 데스크톱 환경

MATE 데스크톱 환경은 다음 **Linux** 배포에서 지원됩니다.

- Ubuntu 16.04
- Ubuntu 18.04

다중 모니터

- **vDGA/vGPU** 데스크톱은 4대의 모니터에서 최대 **2560x1600**의 해상도를 지원합니다.
- **VMware vSphere® 6.0** 이상에서의 **2D** 데스크톱은 4대의 모니터에서 최대 **2048x1536**의 해상도를 지원하고, 3대의 모니터에서 최대 **2560x1600**의 해상도를 지원합니다.

Ubuntu 16.04 및 18.04의 경우 다중 모니터 기능을 사용하려면 **Gnome**, **KDE** 또는 **MATE** 데스크톱 환경을 사용해야 합니다. 자세한 내용은 <http://kb.vmware.com/kb/2151294>에 나와 있습니다.

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

VMware Horizon HTML Access™ 버전 5.0부터 **Horizon 7 for Linux** 데스크톱에서 다중 모니터 기능이 지원됩니다.

VMware Blast에 대한 네트워크 인텔리전스 지원

네트워크 인텔리전스 전송은 VMware Blast에서 지원됩니다. 이 기능은 기본적으로 사용하도록 설정됩니다.

UDP(User Datagram Protocol)를 사용하도록 설정하면 Blast는 TCP(Transmission Control Protocol) 및 UDP 연결을 모두 설정합니다. 현재 네트워크 조건에 따라, Blast에서는 데이터 전송 방법 중 하나를 동적으로 선택하여 최상의 사용자 환경을 제공합니다. 예를 들어, 로컬 네트워크에서 TCP는 UDP보다 더 나은 성능을 제공하므로 Blast는 데이터 전송을 위해 TCP를 선택합니다. 마찬가지로 WAN(wide area network)에서는 UDP 성능이 TCP보다 나으므로 Blast는 해당 환경에서 UDP 전송을 선택합니다.

사용된 인라인 구성 요소 중 하나가 UDP를 지원하지 않는 경우 Blast는 TCP 연결만 설정합니다. 예를 들어, 연결이 Horizon 연결 서버 또는 보안 서버의 Blast 보안 게이트웨이 구성 요소를 사용하는 경우 TCP 연결만 설정됩니다. 클라이언트와 에이전트 둘 다 UDP가 사용되도록 설정되었다고 Blast 보안 게이트웨이는 UDP를 지원하지 않으므로 연결에는 TCP가 사용됩니다. 사용자가 회사 네트워크 외부에서 연결하는 경우 UDP 구성 요소에는 UDP를 지원하는 VMware Unified Access Gateway(이전의 Access Point)가 필요합니다.

다음 정보를 사용하여 UDP 기반 Blast 연결을 설정합니다.

- 클라이언트가 Linux 데스크톱에 직접 연결되는 경우 클라이언트 및 에이전트 둘 다에서 UDP를 사용하도록 설정합니다. UDP는 클라이언트 및 에이전트에서 기본적으로 사용되도록 설정되어 있습니다.
- 클라이언트가 Unified Access Gateway를 사용하여 Linux 데스크톱에 연결되는 경우 클라이언트, 에이전트 및 Unified Access Gateway에서 UDP를 사용하도록 설정합니다.

세션 공동 작업

사용자는 세션 공동 작업 기능을 사용하여 기존 원격 Linux 데스크톱 세션에 가입하도록 다른 사용자를 초대할 수 있고, 다른 사용자의 초대를 받은 경우 공동 작업 세션에 가입할 수 있습니다. 이 기능은 다음과 같은 Linux 배포가 설치되어 있는 원격 Linux 데스크톱에서만 지원됩니다.

- Gnome 데스크톱 환경의 Ubuntu 18.04
- Gnome Classic 또는 KDE 데스크톱 환경의 RHEL 7.5
- Gnome Classic 데스크톱 환경의 RHEL 7.6 이상
- Gnome Classic 데스크톱 환경의 RHEL 8.x

Single Sign-On

SSO(Single Sign-On)는 다음 Linux 배포에서 지원됩니다.

- RHEL 8.x/7.x/6.x Workstation x64
- CentOS 8.x/7.x/6.x x64

- SLED/SLES 12.x SP3/SP2/SP1
- Ubuntu 18.04/16.04 x64

스마트 카드 리더렉션

스마트 카드 리더렉션은 다음과 같은 Linux 배포에서 지원됩니다.

- RHEL 8.x
- RHEL 7.1 이상
- RHEL 6.6 이상
- Ubuntu 18.04/16.04
- SLED/SLES 12.x SP3

이 기능은 PIV(개인 ID 확인) 카드 및 CAC(Common Access Card)를 지원합니다. 자세한 내용은 [스마트 카드 리더렉션 설정](#)의 내용을 참조하십시오.

True SSO 지원

True SSO는 다음과 같은 Linux 배포에서 지원됩니다.

- RHEL 7.x/8.x
- CentOS 7.x/8.x
- SLED/SLES 12.x SP3
- Ubuntu 18.04/16.04

자세한 내용은 [Linux 데스크톱용 True SSO 설정](#)의 내용을 참조하십시오.

USB 리더렉션

USB 리더렉션 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.

3Dconnexion 마우스

3Dconnexion 마우스 사용을 시작하려면 적절한 디바이스 드라이버를 설치하고 Linux 데스크톱에서 [USB 디바이스 연결] 메뉴를 사용하여 마우스를 연결해야 합니다.

3D 그래픽

3D 그래픽 기능은 다음과 같은 Linux 버전 및 그래픽 카드 조합을 지원합니다.

- vDGA는 NVIDIA GRID K1 또는 K2 그래픽 카드가 장착된 RHEL 6.x Workstation x64에서 지원됩니다.
- vGPU는 <https://docs.nvidia.com/grid/9.0/product-support-matrix/index.html>에 나열된 Linux 배포 및 NVIDIA 그래픽 카드에서 지원됩니다.

Linux 데스크톱 및 데스크톱 풀 제한

Linux 데스크톱 및 데스크톱 풀에는 다음과 같은 제한이 있습니다.

- 가상 인쇄, 위치 기반 인쇄 및 실시간 비디오는 지원되지 않습니다.
- VMware HTML Access 파일 전송 기능은 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](#)의 내용을 참조하십시오.
- 2 vSphere에서 가상 시스템을 생성하고 Linux 운영 체제를 설치합니다. [가상 시스템 생성 및 Linux 설치](#)의 내용을 참조하십시오.
- 3 Horizon 7 환경에서 데스크톱으로 배포할 게스트 운영 체제를 준비합니다. [원격 데스크톱 배포를 위한 Linux 시스템 준비](#)의 내용을 참조하십시오.
- 4 Linux 게스트 운영 체제를 Active Directory로 인증하도록 구성합니다. 이 단계는 환경의 요구 사항에 따라 타사 소프트웨어로 구현합니다. 자세한 내용은 [Linux와 Active Directory 통합](#)에 나와 있습니다.
- 5 Linux 가상 시스템에 Horizon Agent를 설치합니다. [Linux 가상 시스템에 Horizon Agent 설치](#)의 내용을 참조하십시오.
- 6 구성된 Linux 가상 시스템을 포함하는 데스크톱 풀을 생성합니다. [Linux용 수동 데스크톱 풀 생성](#)의 내용을 참조하십시오.

3D 그래픽 - 구성 단계 개요

시스템에서 Horizon Agent를 설치하고 Horizon Console에 데스크톱 풀을 배포하려면 먼저 Linux 가상 시스템에서 NVIDIA GRID vGPU 또는 vDGA 구성을 완료해야 합니다.

- 1 Horizon 7 for Linux 배포를 설정하기 위한 시스템 요구 사항을 검토합니다. [Horizon 7 for Linux에 대한 시스템 요구 사항](#)의 내용을 참조하십시오.
- 2 vSphere에서 가상 시스템을 생성하고 Linux 운영 체제를 설치합니다. [가상 시스템 생성 및 Linux 설치](#)의 내용을 참조하십시오.

- 3 Horizon 7 환경에서 데스크톱으로 배포할 게스트 운영 체제를 준비합니다. [원격 데스크톱 배포를 위한 Linux 시스템 준비](#)의 내용을 참조하십시오.
- 4 Linux 게스트 운영 체제를 **Active Directory**로 인증하도록 구성합니다. 이 단계는 환경의 요구 사항에 따라 타사 소프트웨어로 구현합니다. 자세한 내용은 [Linux와 Active Directory 통합](#)에 나와 있습니다.
- 5 ESXi 호스트와 Linux 가상 시스템에 3D 기능을 구성합니다. 설치하려는 3D 기능의 절차를 수행합니다.
 - **vGPU**에 대해 지원되는 **Linux 배포 구성**의 내용을 참조하십시오.
 - **vDGA**용 **RHEL 6.x 구성**의 내용을 참조하십시오.
- 6 Linux 가상 시스템에 **Horizon Agent**를 설치합니다. [Linux 가상 시스템에 Horizon Agent 설치](#)의 내용을 참조하십시오.
- 7 구성된 Linux 가상 시스템을 포함하는 데스크톱 풀을 생성합니다. [Linux용 수동 데스크톱 풀 생성](#)의 내용을 참조하십시오.

대량 배포

Horizon Console에서는 수동 데스크톱 풀에만 Linux 가상 시스템을 배포할 수 있습니다. vSphere PowerCLI를 사용하여 Linux 데스크톱 시스템 풀의 배포를 자동화하는 스크립트를 개발할 수 있습니다. [장 8 수동 데스크톱 풀의 Horizon 7 대량 배포](#)의 내용을 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 Horizon Agent에 대해 지원되는 Linux 운영 체제가 나와 있습니다.

표 1-1. Horizon Agent에 대한 지원되는 Linux 운영 체제

Linux 배포	아키텍처
Ubuntu 16.04 및 18.04	x64
참고 VMware KB 문서 http://kb.vmware.com/kb/2151294 에 설명된 솔루션 중 하나를 적용해야 합니다.	
RHEL 6.6, 6.7, 6.8, 6.9, 6.10, 7.2, 7.3, 7.4, 7.5, 7.6, 7.7, 8.0, 8.1	x64
CentOS 6.6, 6.7, 6.8, 6.9, 6.10, 7.2, 7.3, 7.4, 7.5, 7.6, 7.7, 8.0, 8.1	x64
NeoKylin 6 Update 1	x64

표 1-1. Horizon Agent에 대한 지원되는 Linux 운영 체제 (계속)

Linux 배포	아키텍처
SLED 12.x SP1/SP2/SP3	x64
SLES 12.x SP1/SP2/SP3	x64

참고 일부 Linux 배포의 경우 Linux Agent에 종속성 패키지가 포함되어 있습니다. 자세한 내용은 [Horizon Agent에 대한 종속성 패키지 설치](#)의 내용을 참조하십시오.

참고 RHEL/CentOS 8.x 시스템의 경우 Horizon Agent는 X11 디스플레이 서버 프로토콜만 지원합니다. Wayland 프로토콜은 지원되지 않습니다.

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"> ■ vSphere 6.0 U2 이상 릴리스 ■ vSphere 6.5 U1 이상 릴리스 ■ vSphere 6.7 이상 릴리스
Horizon 환경	<ul style="list-style-type: none"> ■ Horizon 연결 서버 7.11
Horizon Client 소프트웨어	<ul style="list-style-type: none"> ■ Android용 Horizon Client 5.3.0 ■ Windows용 Horizon Client 5.3.0 ■ Linux용 Horizon Client 5.3.0 ■ Mac OS X용 Horizon Client 5.3.0 ■ iOS(iPad Pro)용 Horizon Client 5.3.0 ■ Chrome, Firefox 및 Internet Explorer의 HTML Access 5.3.0 ■ Zero 클라이언트는 지원되지 않습니다.

Linux 가상 시스템에서 사용하는 TCP/UDP 포트

Horizon Agent 및 Horizon Client에서는 상호 네트워크 액세스 그리고 다양한 Horizon Server 구성 요소와의 네트워크 액세스에 TCP 또는 UDP 포트를 사용합니다.

표 1-3. Linux 가상 시스템에서 사용하는 TCP/UDP 포트

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

참고 클라이언트에서 사용하는 TCP 및 UDP 포트에 대한 자세한 내용은 "Horizon Client 및 Agent 보안" 문서 및 [VMware Horizon 7의 네트워크 포트 가이드](#)를 참조하십시오.

사용자가 Linux 데스크톱에 연결할 수 있도록 하려면 데스크톱은 Horizon Client 디바이스, 보안 서버 및 Horizon Connection Server에서 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 사용되도록 설정되면 보안 서버 또는 Horizon Connection Server의 BSG를 통해 Horizon Client 디바이스에서 Linux 데스크톱으로의 클라이언트 연결이 설정됩니다. BSG가 활성화되지 않으면 Horizon Client 디바이스에서 Linux 데스크톱으로 직접 연결이 설정됩니다.

Linux 가상 시스템에서 사용하는 Linux 계정 확인

표 1-4. 계정 이름 및 계정 유형에는 Linux 가상 시스템에서 사용되는 계정 이름 및 계정 유형이 표시됩니다.

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 Linux는 여러 Linux 배포에서 다중 데스크톱 환경을 지원합니다. 표 1-5. 지원되는 데스크톱 환경에는 각 Linux 배포에 대한 기본 데스크톱 환경 및 Horizon 7 for Linux에서 지원되는 추가 데스크톱 환경이 표시됩니다.

표 1-5. 지원되는 데스크톱 환경

Linux 배포	기본 데스크톱 환경	Horizon 7 for Linux 데스크톱에서 지원되는 데스크톱 환경
Ubuntu 18.04	Gnome	Gnome Ubuntu, KDE(K Desktop Environment), MATE
Ubuntu 16.04	Unity	Gnome Flashback(Metacity), KDE, MATE
RHEL/CentOS 6.x	Gnome	Gnome, KDE
RHEL/CentOS 7.x	Gnome	Gnome, KDE
RHEL/CentOS 8.x	Gnome	Gnome
SLED 12 SP1/SP2/SP3	Gnome	Gnome
SLES 12 SP1/SP2/SP3	Gnome	Gnome
NeoKylin 6 Update 1	Mate	Mate

지원되는 Linux 배포 중 하나에 사용되는 기본 데스크톱 환경을 변경하려면 다음 단계와 Linux 데스크톱에 대한 적절한 명령을 사용해야 합니다.

참고 KDE 및 MATE 데스크톱 환경용 SSO(Single Sign-On)는 Linux 데스크톱에서 GDM3 초기 화면(로그인 화면)을 사용 중일 경우에만 작동합니다. 표 1-6. 데스크톱 환경을 설치하는 명령에 나열된 명령을 사용하여 KDE 및 MATE를 설치해야 합니다.

RHEL/CentOS 7.x 및 Ubuntu 18.04/16.04 배포를 사용할 경우 SSO는 잠긴 KDE 세션의 잠금을 해제하지 못합니다. 잠긴 세션의 잠금을 해제하려면 암호를 직접 입력해야 합니다.

- 1 기본 데스크톱 환경 설정을 사용하여 지원되는 Linux 배포의 운영 체제를 설치합니다.
- 2 특정 Linux 배포를 위해 표 1-6. 데스크톱 환경을 설치하는 명령의 해당 명령을 실행합니다.

표 1-6. 데스크톱 환경을 설치하는 명령

Linux 배포	새 기본 데스크톱 환경	기본 데스크톱 환경을 변경하는 명령
RHEL/CentOS 6.x	KDE	# yum groupinstall "X Window System" "KDE Desktop"
RHEL/CentOS 7.x	KDE	# yum groupinstall "KDE Plasma Workspaces"
Ubuntu 18.04/16.04	KDE	# apt install plasma-desktop
Ubuntu 18.04	MATE 1.225	# apt install ubuntu-mate-desktop

표 1-6. 데스크톱 환경을 설치하는 명령 (계속)

Linux 배포	새 기본 데스크톱 환경	기본 데스크톱 환경을 변경하는 명령
Ubuntu 16.04	MATE 1.16	<pre># apt-add-repository ppa:ubuntu-mate-dev/xenial-mate # apt update # apt upgrade # apt install mate # apt install ubuntu-mate-themes</pre>
Ubuntu 16.04	Gnome Flashback(Metacity)	<pre># apt install gnome-session-flashback</pre>

3 새 기본 데스크톱 환경을 사용하려면 데스크톱을 다시 시작합니다.

여러 데스크톱 환경이 설치된 Linux 데스크톱에서 SSO를 사용하도록 설정한 경우 다음 정보를 사용하여 SSO 세션에서 사용할 데스크톱 환경을 선택하십시오.

- Ubuntu 18.04/16.04 및 RHEL/CentOS 7.x의 경우 표 1-7. **SSODesktopType** 옵션의 정보를 사용하여 SSO와 함께 사용할 데스크톱 환경을 지정하도록 `/etc/vmware/viewagent-custom.conf` 파일의 **SSODesktopType** 옵션을 설정합니다.

표 1-7. SSODesktopType 옵션

데스크톱 유형	SSODesktopType 옵션 설정
MATE	SSODesktopType=UseMATE
GnomeUbuntu	SSODesktopType=UseGnomeUbuntu
GnomeFlashback	SSODesktopType=UseGnomeFlashback
KDE	SSODesktopType=UseKdePlasma
GnomeClassic	SSODesktopType=UseGnomeClassic

- RHEL/CentOS 6.x의 경우 SSO 로그인 세션에서 KDE를 사용하도록 하려면 `/usr/share/xsession` 디렉토리에서 KDE 시작 파일을 제외한 모든 데스크톱 시작 파일을 제거합니다. 다음 명령 집합을 예로 사용합니다.

```
# cd /usr/share/xsessions
# mkdir backup
# mv *.desktop backup
# mv backup/kde*.desktop ./
```

초기 설정 후에 최종 사용자는 Linux 데스크톱에서 로그아웃하거나 데스크톱을 재부팅하여 KDE를 다음 SSO 세션의 기본 데스크톱으로 지정해야 합니다.

- RHEL/CentOS 8.x의 경우 SSO 로그인 세션에서 Gnome Classic을 사용하도록 하려면 `/usr/share/xsession` 디렉토리에서 Gnome Classic 시작 파일을 제외한 모든 데스크톱 시작 파일을 제거합니다. 다음 명령 집합을 예로 사용합니다.

```
# cd /usr/share/xsessions
# mkdir backup
# mv *.desktop backup
# mv backup/gnome-classic.desktop ./
```

초기 설정 후에 최종 사용자는 Linux 데스크톱에서 로그아웃하거나 데스크톱을 재부팅하여 Gnome Classic을 다음 SSO 세션의 기본 데스크톱으로 지정해야 합니다.

여러 데스크톱 환경이 설치되어 있는 Linux 데스크톱에서 SSO를 사용하지 않도록 설정한 경우 이전에 설명된 단계를 수행할 필요가 없습니다. 최종 사용자는 해당 Linux 데스크톱에 로그인할 때 원하는 데스크톱 환경을 선택해야 합니다.

네트워크 요구 사항

VMware Blast Extreme은 UDP(User Datagram Protocol) 및 TCP(Transmission Control Protocol) 모두 지원됩니다. 네트워크 상태가 TCP 및 UDP 성능에 영향을 줍니다. 최상의 사용자 환경을 위해서는 네트워크 상태에 따라 UDP 또는 TCP를 선택합니다.

- LAN(Local Area Network) 환경에서와 같이 네트워크 상태가 양호한 경우에는 TCP를 선택합니다.
- 패킷 손실 및 시간 지연이 발생하는 WAN(Wide Area Network) 환경에서와 같이 네트워크 상태가 좋지 않은 경우에는 UDP를 선택합니다.

Wireshark와 같은 네트워크 분석기 도구를 사용하여 VMware Blast Extreme이 TCP를 사용하는지 또는 UDP를 사용하는지 확인합니다. Wireshark를 사용하는 다음 단계를 참조 예제로 사용하십시오.

- 1 Linux VM에 Wireshark를 다운로드하여 설치합니다.

RHEL/CentOS 6:

```
sudo yum install wireshark
```

Ubuntu 18.04/16.04:

```
sudo apt install tshark
```

SLED/SLES 12:

```
sudo zypper install wireshark
```

- 2 VMware Horizon Client를 사용하여 Linux 데스크톱에 연결합니다.
- 3 터미널 창을 열고 다음 명령을 실행합니다. 그러면 VMware Blast Extreme에서 사용되는 TCP 패키지 또는 UDP 패키지가 표시됩니다.

```
sudo tshark -i any | grep 22443
```

USB 리디렉션 및 CDR(클라이언트 드라이브 리디렉션) 기능은 네트워크 상태에 민감합니다. 시간 지연 및 패킷 손실이 발생하는 제한된 대역폭과 같이 네트워크 상태가 나쁜 경우에는 사용자 환경도 좋지 않습니다. 이러한 상태에서 최종 사용자에게는 다음 중 하나가 발생할 수 있습니다.

- 원격 파일 복사가 느려질 수 있습니다. 이 상황에서는 대신 더 작은 크기의 파일을 전송합니다.
- USB 디바이스가 원격 Linux 데스크톱에 나타나지 않습니다.
- USB 데이터가 완전히 전송되지 않습니다. 예를 들어 큰 파일을 복사하는 경우 파일 크기가 원본 파일보다 더 작아질 수 있습니다.

USB 리디렉션용 VHCI 드라이버

USB 리디렉션 기능은 USB VHCI(Virtual Host Controller Interface) 커널 드라이버에 따라 달라집니다.

USB 3.0 및 USB 리디렉션 기능을 지원하려면 다음 단계를 수행해야 합니다.

- 1 <https://sourceforge.net/projects/usb-vhci/files/linux%20kernel%20module/>에서 USB VHCI 소스 코드를 다운로드합니다.
- 2 VHCI 드라이버 소스 코드를 컴파일하고 결과 바이너리를 Linux 시스템에 설치하려면 [표 1-8. USB VHCI 드라이버 컴파일 및 설치](#)의 명령을 사용하십시오.

예를 들어 설치 파일 VMware-horizonagent-linux-x86_64-*<version>*-*<build-number>*.tar.gz의 압축을 /install_tmp/ 디렉토리 아래에 풀면 *full-path_to_patch-file*은 /install_tmp/VMware-horizonagent-linux-x86_64-*<version>*-*<buildnumber>*/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
```

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

표 1-8. USB VHCI 드라이버 컴파일 및 설치

Linux 배포	USB VHCI 드라이버 컴파일 및 설치 단계
Ubuntu 18.04	<p>1 종속성 패키지를 설치합니다.</p> <pre># apt-get install make # apt-get install gcc # apt-get install libelf-dev</pre> <p>2 VHCI 드라이버를 컴파일하고 설치합니다.</p> <pre># tar -xzvf vhci-hcd-1.15.tar.gz # cd vhci-hcd-1.15 # patch -p1 < "full-path_to_patch-file" # make clean && make && make install</pre>
Ubuntu 16.04	<p>VHCI 드라이버를 컴파일하고 설치합니다.</p> <pre># tar -xzvf vhci-hcd-1.15.tar.gz # cd vhci-hcd-1.15 # patch -p1 < "full-path_to_patch-file" # make clean && make && make install</pre>

표 1-8. USB VHCI 드라이버 컴파일 및 설치 (계속)

Linux 배포	USB VHCI 드라이버 컴파일 및 설치 단계
RHEL/CentOS 6.9/6.10	1 종속성 패키지를 설치합니다.
RHEL/CentOS 7.x	<pre># yum install gcc-c++ # yum install kernel-devel-\$(uname -r) # yum install kernel-headers-\$(uname -r) # yum install patch # yum install elfutils-libelf-devel</pre>
RHEL/CentOS 8.x	<pre># yum install elfutils-libelf-devel</pre>
	2 VHCI 드라이버를 컴파일하고 설치합니다.
	<pre># 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</pre>
	3 (RHEL/CentOS 8.x) VHCI 드라이버가 USB 리디렉션을 사용하여 제대로 작동하도록 하려면 USB 드라이버에 대한 서명 설정을 구성합니다.
	a USB 드라이버에 대한 SSL 키 쌍을 생성합니다.
	<pre>openssl req -new -x509 -newkey rsa:2048 -keyout MOK.priv -outform DER -out MOK.der -nodes -days 36500 -subj "/CN=Descriptive name/"</pre>
	b USB 드라이버에 서명합니다.
	<pre>sudo /usr/src/kernels/\$(uname -r)/scripts/sign-file sha256 ./MOK.priv ./MOK.der /lib/modules/\$(uname -r)/kernel/drivers/usb/host/usb-vhci-iocifc.ko sudo /usr/src/kernels/\$(uname -r)/scripts/sign-file sha256 ./MOK.priv ./MOK.der /lib/modules/\$(uname -r)/kernel/drivers/usb/host/usb-vhci-hcd.ko</pre>
	c UEFI 보안 부팅을 위해 키를 등록합니다.
	<pre>sudo mokutil --import MOK.der</pre>
	참고 이 명령은 UEFI 보안 부팅을 위해 MOK(시스템 소유자 키) 암호를 설정하는 요청을 실행합니다.
	d vSphere 콘솔에서 UEFI 보안 부팅을 설정하려면 시스템을 재부팅합니다. 자세한 내용은 https://sourceware.org/systemtap/wiki/SecureBoot 를 참조하십시오.
SLED/SLES 12 SP2	1 현재 커널 패키지 버전을 확인합니다.
	<pre># rpm -qa grep kernel-default-\$(echo \$(uname -r) cut -d '-' -f 1,2)</pre>
	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 kernel-devel, kernel-default-devel, kernel-macros 및 patch 패키지를 설치합니다.
	<pre># zypper install --oldpackage kernel-devel- "<커널 패키지 버전>" \ kernel-default-devel- "<커널 패키지 버전>" kernel-macros- "<커널 패키지 버전>" patch</pre>
	예:
	<pre># zypper install --oldpackage kernel-devel-4.4.21-90.1 kernel-default-devel-4.4.21-90.1 kernel-macros-4.4.21-90.1 patch</pre>

표 1-8. USB VHCI 드라이버 컴파일 및 설치 (계속)

Linux 배포	USB VHCI 드라이버 컴파일 및 설치 단계
	<p>3 VHCI 드라이버를 컴파일하고 설치합니다.</p> <pre># 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</pre>

또한 다음 지침을 준수하십시오.

- Linux 커널이 새 버전으로 변경되면 VHCI 드라이버를 다시 컴파일한 후 다시 설치해야 하지만 Horizon for Linux를 다시 설치할 필요는 없습니다.
- 또한 Ubuntu 18.04/16.04 시스템에 대한 다음 예제와 비슷한 단계를 사용하여 VHCI 드라이버에 DKMS(Dynamic Kernel Module Support)를 추가할 수 있습니다.
 - a 커널 헤더를 설치합니다.

```
# apt install linux-headers-`uname -r`
```

- b 다음 명령을 사용하여 dkms를 설치합니다.

```
# apt install dkms
```

- c VHCI TAR 파일의 압축을 풀고 패치합니다.

```
# tar xzvf vhci-hcd-1.15.tar.gz
# cd vhci-hcd-1.15
# patch -p1 "<full-path_to_patch-file>"
# cd ..
```

- d /usr/src 디렉토리에 압축을 푼 VHCI 소스 파일을 복사합니다.

```
# cp -r vhci-hcd-1.15 /usr/src/usb-vhci-hcd-1.15
```

- e dkms.conf라는 파일을 생성하고 /usr/src/usb-vhci-hcd-1.15 디렉토리에 배치합니다.

```
# touch /usr/src/usb-vhci-hcd-1.15/dkms.conf
```

- f 다음 콘텐츠를 dkms.conf 파일에 추가합니다.

```
PACKAGE_NAME="usb-vhci-hcd"
PACKAGE_VERSION=1.15
MAKE_CMD_TMPL="make KVERSION=$kernelver"

CLEAN="$MAKE_CMD_TMPL clean"

BUILT_MODULE_NAME[0]="usb-vhci-iocifc"
```

```
DEST_MODULE_LOCATION[0]="/kernel/drivers/usb/host"
MAKE[0]="$MAKE_CMD_TMPL"

BUILT_MODULE_NAME[1]="usb-vhci-hcd"
DEST_MODULE_LOCATION[1]="/kernel/drivers/usb/host"
MAKE[1]="$MAKE_CMD_TMPL"

AUTOINSTALL="YES"
```

- g dkms에서 이 VHCI 드라이버를 추가합니다.

```
# dkms add usb-vhci-hcd/1.15
```

- h VHCI 드라이버를 빌드합니다.

```
# dkms build usb-vhci-hcd/1.15
```

- i VHCI 드라이버를 설치합니다.

```
# dkms install usb-vhci-hcd/1.15
```

2D 그래픽을 위한 가상 시스템 설정

Linux 가상 시스템에 대해 특정 Horizon 7을 생성하는 경우 성능 요구 사항에 대해 vCPU 및 가상 메모리 설정을 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 시스템의 경우 vCPU 및 가상 메모리 설정을 변경할 필요가 없습니다.

2D 그래픽을 사용하도록 구성된 가상 시스템은 VMware 가상 그래픽 카드를 사용하며, vCPU 및 가상 메모리 설정을 변경하여 데스크톱 성능을 개선해야 합니다. 다음 지침을 사용하십시오.

- 2D 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 화면 디스플레이가 필요한 경우 가상 시스템에 대해 4개의 vCPU 및 4GB의 가상 메모리를 설정합니다.
- 2D 데스크톱에서 비디오 재생을 개선하려면 가상 시스템에 대해 4개의 vCPU 및 4GB의 가상 메모리를 설정합니다.

Linux 데스크톱에서 세션 공동 작업 구성

사용자는 세션 공동 작업 기능을 사용하여 기존의 Linux 원격 데스크톱 세션에 가입하도록 다른 사용자를 초대할 수 있습니다.

세션 공동 작업에 대한 시스템 요구 사항

세션 공동 작업 기능을 지원하려면 Horizon 배포가 특정 요구 사항을 충족해야 합니다.

표 1-9. 세션 공동 작업에 대한 시스템 요구 사항

구성 요소	요구 사항
클라이언트 시스템	세션 소유자 및 공동 작업자는 클라이언트 시스템에 Windows, Mac 또는 Linux용 Horizon Client 4.7 이상을 설치하거나 HTML Access 4.7 이상을 사용해야 합니다.
Linux 원격 데스크톱	<p>세션 공동 작업 기능은 다음과 같은 Linux 배포 및 데스크톱 환경을 실행하는 원격 데스크톱에서 지원됩니다.</p> <ul style="list-style-type: none"> ■ Gnome 데스크톱 환경의 Ubuntu 18.04 ■ Gnome Classic 또는 KDE 데스크톱 환경의 RHEL 7.5 ■ Gnome Classic 데스크톱 환경의 RHEL 7.6 이상 ■ Gnome Classic 데스크톱 환경의 RHEL 8.x <p>또한 원격 데스크톱은 다음 요구 사항을 충족해야 합니다.</p> <ul style="list-style-type: none"> ■ Gnome 데스크톱 환경에서 세션 공동 작업을 지원하려면 Horizon Agent 7.7 이상을 설치해야 합니다. ■ KDE 데스크톱이 있는 RHEL 7.5 데스크톱 환경에서 세션 공동 작업을 지원하려면 Horizon Agent 7.12 이상을 설치해야 합니다. ■ 데스크톱 풀 또는 원격 데스크톱 수준에서 세션 공동 작업 기능을 사용하도록 설정해야 합니다.
연결 서버	연결 서버 인스턴스는 Enterprise 라이선스를 사용합니다.
디스플레이 프로토콜	VMware Blast

참고 RHEL 8.x 데스크톱은 세션 공동 작업을 지원하기 위해 추가적인 시스템 구성이 필요합니다. [세션 공동 작업을 위해 RHEL 8.x 데스크톱 구성](#)의 내용을 참조하십시오.

세션 공동 작업 기능을 사용하는 방법에 대한 내용은 [Horizon Client](#) 설명서를 참조하십시오.

구성 파일에서 세션 공동 작업 옵션 설정

/Etc/vmware/viewagent-custom.conf 파일에서 다음 옵션을 설정하여 세션 공동 작업 기능을 사용하거나 사용하지 않도록 설정합니다.

- CollaborationEnable

/Etc/vmware/config 파일에서 다음 옵션을 설정하여 공동 작업 세션 동안 사용하는 설정을 구성합니다.

- collaboration.logLevel
- collaboration.maxCollabors
- collaboration.enableEmail
- collaboration.serverUrl
- collaboration.enableControlPassing

자세한 내용은 [Linux 데스크톱의 구성 파일에서 옵션 설정](#)에 나와 있습니다.

세션 공동 작업 기능 제한 사항

세션 공동 작업 기능에는 다음과 같은 일반 제한 사항이 적용됩니다.

- 공동 작업 세션에서는 다음 원격 데스크톱 기능을 사용할 수 없습니다.
 - USB 리디렉션
 - 오디오 입력 리디렉션
 - 클라이언트 드라이브 리디렉션
 - 스마트 카드 리디렉션
 - 클립보드 리디렉션
- 공동 작업 세션에서는 원격 데스크톱 해상도를 변경할 수 없습니다.
- 사용자가 동일한 클라이언트 시스템에서 여러 공동 작업 세션을 사용할 수 없습니다.

KDE 데스크톱이 있는 RHEL 7.5 데스크톱 환경에서 공동 작업 세션에는 다음과 같은 제한 사항이 적용됩니다.

- 세션 공동 작업 메뉴를 표시하려면 사용자가 시스템 트레이에서 세션 공동 작업 아이콘을 마우스 오른쪽 버튼으로 클릭해야 합니다. 아이콘을 왼쪽 버튼으로 클릭해도 효과가 없습니다.
- 사용자에게 공동 작업자에게 초대 이메일을 보내는 데 사용되는 **이메일** 버튼이 표시되지만 처음에는 비활성 상태입니다. 버튼을 활성화하려면 먼저 사용자가 KDE 데스크톱 환경에 대한 기본 이메일 애플리케이션을 구성해야 합니다.

참고 세션 공동 작업과 관련된 문제를 해결하려면 다음 해결 방법을 사용하십시오.

- 사용자가 원격 데스크톱에 처음으로 로그인한 후에 세션 공동 작업 아이콘이 시스템 트레이에 표시되지 않으면 사용자에게 데스크톱의 연결을 끊었다가 다시 연결하도록 지시합니다. 세션 공동 작업 아이콘은 일반적으로 데스크톱에 다시 연결한 후에 나타납니다.
 - 사용자가 원격 데스크톱에 처음으로 로그인한 후 시스템 트레이의 세션 공동 작업 아이콘이 응답하지 않는 경우 사용자에게 원격 데스크톱 창의 크기를 조정하도록 지시합니다. 세션 공동 작업 아이콘은 데스크톱 창의 크기가 조정되면 응답합니다.
-

세션 공동 작업을 위해 RHEL 8.x 데스크톱 구성

RHEL 8.x 데스크톱에서 세션 공동 작업 기능을 사용하려면 먼저 GNOME 3.28.26 셸 확장을 다운로드한 후 설치해야 합니다.

절차

- 1 <https://extensions.gnome.org/extension/615/appindicator-support/>에서 RHEL 8.x 시스템에 필요한 GNOME 셸 확장을 다운로드합니다. 셸 버전으로 **3.28**을 선택합니다. 확장 버전으로 **26**을 선택합니다.
- 2 다운로드한 패키지의 압축을 풀고 디렉토리의 이름을 `appindicator-support@rgcjonas.gmail.com`으로 바꿉니다(패키지의 `metadata.json` 파일에 있는 “uuid” 값).

- 3 `mv` 명령을 사용하여 `appindicatorsupport@rgcjonas.gmail.com` 디렉토리를 위치 `/usr/share/gnome-shell/extensions`로 이동합니다.

기본적으로 `appindicatorsupport@rgcjonas.gmail.com` 디렉토리의 `metadata.json` 파일은 루트 사용자만 읽을 수 있습니다. 세션 공동 작업을 지원하려면 이 파일을 다른 사용자도 읽을 수 있도록 만들어야 합니다.

- 4 다음 예제와 같이 해당 명령을 실행하여 `metadata.json`을 다른 사용자가 읽을 수 있도록 합니다.

```
chmod a+r metadata.json
```

- 5 `gnome-tweaks`를 설치합니다.

- 6 데스크톱 환경의 키보드에서 다음 키 순서를 눌러 GNOME 셸을 다시 시작합니다.

```
Alt+F2  
r  
Enter
```

- 7 데스크톱 환경에서 `gnome-tweaks`를 실행한 후 **KStatusNotifierItem/AppIndicator Support**를 사용하도록 설정합니다.

데스크톱 배포를 위해 Linux 가상 시스템 준비

2

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

본 장은 다음 항목을 포함합니다.

- 가상 시스템 생성 및 Linux 설치
- 원격 데스크톱 배포를 위한 Linux 시스템 준비
- Horizon Agent에 대한 종속성 패키지 설치

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.

사전 요구 사항

- 배포가 Linux 데스크톱을 지원하기 위한 요구 사항을 충족하는지 확인하십시오. [Horizon 7 for Linux에 대한 시스템 요구 사항](#)의 내용을 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 대한 비디오 메모리(vRAM) 설정 요구 사항을 숙지하십시오. [Horizon 7 for Linux에 대한 시스템 요구 사항](#)의 내용을 참조하십시오.

절차

- 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.

필요한 설정을 보려면 **Linux** 배포에 대한 설치 설명서의 지침을 따르십시오.

예를 들어 **Ubuntu 18.04**에서는 vMemory 2048MB 및 vCPU 2개를 구성하도록 지정합니다.

- c **비디오 카드**를 선택하고 디스플레이 수 및 총 비디오 메모리(vRAM)를 지정하십시오.

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

필수 설정에 대해서는 **2D 그래픽을 위한 가상 시스템 설정**의 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](#)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 관리를 위해 Linux 시스템을 준비하려면 시스템과 연결 서버 간의 통신을 사용하도록 설정해야 합니다. Linux 시스템에서 FQDN(정규화된 도메인 이름)을 사용하여 연결 서버 인스턴스를 ping할 수 있도록 Linux 시스템에 네트워킹을 구성해야 합니다.

OVT(Open VMware Tools)는 RHEL 8.x/7.x, CentOS 8.x/7.x 및 SLED/SLES 12.x 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.

Ubuntu16.04/18.04 시스템을 사용하는 경우 시스템에 OVT를 설치합니다. 원격 데스크톱으로 사용하기 위해 이 시스템을 준비하는 경우 다음 절차에서 1~5단계를 건너뛰고 다음 명령을 사용하여 Ubuntu 16.04/18.04 시스템에 OVT를 수동으로 설치할 수 있습니다.

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

사전 요구 사항

- 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.

절차

- 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.gz`를 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 `127.0.0.1`에 매핑합니다.

RHEL, CentOS, SLES 및 SLED의 경우 호스트 이름이 자동으로 매핑되지 않으므로 이를 `127.0.0.1`에 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 `127.0.0.1`에 매핑해야 합니다. 그렇지 않으면 이전 호스트 이름이 계속 사용됩니다.

- 7 RHEL 및 CentOS의 경우 `virbr0`를 사용하지 않도록 설정했는지 확인합니다.

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

- 8 DNS를 통해 포드의 Horizon Connection Server 인스턴스를 확인할 수 있어야 합니다.

- 9 기본적으로 그래픽 모드로 실행되도록 Linux 시스템을 구성합니다.

예를 들어, 다음 명령은 CentOS 시스템을 그래픽 모드로 실행하도록 구성합니다.

```
systemctl set-default graphical.target
```

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

이 단계에서는 Horizon Console의 [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.

결과

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.

사전 요구 사항

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

절차

- 1 기본적으로 설치 또는 업그레이드되지 않은 필수 패키지를 설치합니다. 패키지 요구 사항을 충족하지 않으면 설치 관리자는 설치를 중단합니다.

표 2-1. 필수 종속성 패키지

Linux 배포	패키지
RHEL 7.5	<code>yum install libappindicator-gtk3</code>
SLES 12.x SP1/SLED 12.x SP1 SUSE 저장소에서 <code>xf86-video-vmware</code> 를 13.0.2-3.2보다 높은 버전으로 업그레이드	<ol style="list-style-type: none"> 1 SUSE 저장소를 사용하도록 설정하려면 SUSE 12.x를 등록합니다. <code>SUSEConnect -r 등록 코드 -e 이메일</code> 2 <code>xf86-video-vmware</code> 버전을 업데이트합니다. <code>zypper update xf86-video-vmware</code>
SLES 12.x	<p>Horizon Agent를 설치하는 경우 SLES 12.x Linux 데스크톱을 사용하려면 <code>python-gobject2</code>를 설치해야 합니다.</p> <ol style="list-style-type: none"> 1 SUSE 저장소를 사용하도록 설정하려면 SUSE 12.x를 등록합니다. <code>SUSEConnect -r 등록 코드 -e 이메일</code> 2 <code>python-gobject2</code>를 설치합니다. <code>zypper install python-gobject2</code>
Ubuntu 16.04	<code>apt-get install python-dbus python-gobject</code>
Ubuntu 18.04	<code>apt-get install python python-dbus python-gobject</code>

2 Horizon Agent에 선택 패키지를 설치합니다.

- 기본적으로 RHEL 또는 CentOS 6.7에는 `glibc-2.12-1.166.el6.x86_64`가 설치되어 있어서 교착 상태를 야기할 수 있습니다. 결과적으로 데스크톱 연결이 끊어집니다. 이 문제를 해결하려면 `glibc`를 온라인 저장소에서 최신 버전으로 업그레이드해야 합니다.

```
sudo yum install glibc
```


Linux 데스크톱에 대한 Active Directory 통합 및 사용자 인증 기능 설정

3

Horizon 7에서는 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. SSO(Single Sign-On), 스마트 카드 리디렉션 및 True SSO와 같은 사용자 인증 기능을 구성할 수도 있습니다.

참고 Horizon Agent에서는 Linux 데스크톱 및 클라이언트 사용자가 동일한 Active Directory 도메인에 상주해야 합니다. 데스크톱 및 사용자가 서로 다른 도메인에 상주하는 경우 Horizon Agent가 데스크톱 도메인을 사용자 도메인으로 잘못 인식할 수 있습니다.

본 장은 다음 항목을 포함합니다.

- [Linux와 Active Directory 통합](#)
- [Single Sign-On 설정](#)
- [스마트 카드 리디렉션 설정](#)
- [Linux 데스크톱용 True SSO 설정](#)

Linux와 Active Directory 통합

Linux를 Microsoft 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
- PBISO(PowerBroker Identity Services Open) 인증
- Samba 오프라인 도메인 가입

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

참고 간편한 배포를 위해 Microsoft Active Directory에 대해 SSSD LDAP 인증을 사용하는 솔루션을 사용합니다.

OpenLDAP 서버 패스스루 인증 사용

OpenLDAP 서버를 설정하고 PTA(패스스루 인증) 메커니즘을 사용하여 Active Directory에 대해 사용자 자격 증명을 확인할 수 있습니다.

개괄적으로 OpenLDAP 패스스루 인증 솔루션은 다음 단계에 따라 진행됩니다.

절차

- 1 LDAPS(Lightweight Directory Access Protocol over SSL)를 사용하도록 설정하려면 Active Directory에서 인증서 서비스를 설치합니다.
- 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 Directory에 대해 SSSD LDAP 인증 설정

Linux 데스크톱에서 SSSD(System Security Services Daemon)를 구성하여 Windows Active Directory에 대해 LDAP 인증을 사용할 수 있습니다.

SSSD LDAP 인증 솔루션에 대해 다음과 같은 개괄적인 단계를 사용합니다.

절차

- 1 LDAPS(Lightweight Directory Access Protocol Over Secure Socket Layer)를 사용하도록 설정하려면 Active Directory 서버에 인증서 서비스를 설치합니다.
- 2 Microsoft Active Directory에 대해 직접 LDAP 인증을 사용하려면 Linux 데스크톱에서 SSSD를 구성합니다.

Winbind 도메인 가입 솔루션 사용

Kerberos 기반 인증 솔루션인 Winbind 도메인 가입 솔루션은 Active Directory에서 인증을 받는 또 다른 방법입니다.

Winbind 도메인 가입 솔루션을 설정하려면 다음과 같은 개괄적인 단계를 사용합니다.

절차

- 1 Linux 데스크톱에서 winbind, samba 및 Kerberos 패키지를 설치합니다.
- 2 Linux 데스크톱을 Microsoft Active Directory에 가입시킵니다.

다음에 수행할 작업

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

```
sudo /usr/bin/net ads join -U "<domain_user>%<domain_password>"
```

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 [장 8 수동 데스크톱 풀의 Horizon 7 대량 배포](#)를 참조하십시오.
- 셸 스크립트에 이 명령을 포함하고 `/etc/vmware/viewagent-custom.conf` 파일에서 스크립트 경로를 Horizon Agent RunOnceScript 옵션으로 설정합니다. 자세한 내용은 [Linux 데스크톱의 구성 파일에서 옵션 설정](#)의 내용을 참조하십시오.

PBISO(PowerBroker Identity Services Open) 인증 구성

PBISO(PowerBroker Identity Services Open) 인증 방법은 오프라인 도메인 가입을 수행하기 위해 지원되는 솔루션 중 하나입니다.

다음 단계에 따라 PBISO를 사용하여 Linux 데스크톱을 Active Directory에 가입합니다.

절차

- 1 <https://www.beyondtrust.com/products/powerbroker-identity-services-open/>에서 PBISO 8.5.6 이상을 다운로드합니다.
- 2 Linux VM에 PBISO를 설치합니다.

```
sudo ./pbis-open-8.5.6.2029.linux.x86_64.deb.sh
```

- 3 Linux용 Horizon 7 Agent를 설치합니다.
- 4 PBISO를 사용하여 Linux 데스크톱을 AD 도메인에 가입합니다.

다음 예에서 **lxd.vdi**는 도메인 이름이고 **administrator**는 도메인 사용자 이름입니다.

```
sudo domainjoin-cli join lxd.vdi administrator
```

5 도메인 사용자에게 대해 기본 구성을 설정합니다.

```
sudo /opt/pbis/bin/config UserDomainPrefix lxdc
sudo /opt/pbis/bin/config AssumeDefaultDomain true
sudo /opt/pbis/bin/config LoginShellTemplate /bin/bash
sudo /opt/pbis/bin/config HomeDirTemplate %H/%U
```

6 /Etc/pamd.d/common-session 파일을 편집합니다.

- a **session sufficient pam_lsass.so**라는 줄을 찾습니다.
- b 이 줄을 **session [success=ok default=ignore] pam_lsass.so**로 바꿉니다.

참고 Linux 용 Horizon Agent를 설치하거나 업데이트한 후에는 이 단계를 반복해야 합니다.

7 Ubuntu 16.04의 경우 /usr/share/lightdm/lightdm.conf.d/50-unity-greeter.conf 구성 파일에 다음 줄을 추가합니다.

```
allow-guest=false
greeter-show-manual-login=true
```

참고 Ubuntu 18.04를 사용하는 경우 **lightdm** 구성 파일을 변경할 필요가 없습니다.

8 시스템을 재부팅하고 로그인합니다.

다음에 수행할 작업

참고

- /opt/pbis/bin/config AssumeDefaultDomain 옵션을 **false**로 설정하는 경우 /etc/vmware/viewagent-custom.conf 파일에서 SSOUserFormat=<username>@<domain> 설정을 업데이트해야 합니다.
- Horizon 인스턴트 클론 플로팅 데스크톱 풀 기능을 사용하는 경우 복제된 VM에 새 네트워크 어댑터를 추가할 때 DNS 서버 설정이 손실되지 않도록 하려면 Linux 시스템에 대한 **resolv.conf** 파일을 수정합니다. Ubuntu 16.04 시스템의 경우 다음 예를 참조하여 /etc/resolvconf/resolv.conf.d/head 파일에 필요한 줄을 추가하십시오.

```
nameserver 10.10.10.10
search mydomain.org
```

Samba 오프라인 도메인 가입 구성

Horizon 7 Linux 데스크톱 환경에서 인스턴트 클론 VM에 대해 SSO를 지원하려면 마스터 Linux VM에서 Samba를 구성합니다.

다음 절차를 참조하여 **Samba**를 사용하여 오프라인 도메인 가입 방식으로 인스턴트 클론 Linux 데스크톱을 Active Directory에 가입시킵니다. 이 절차에서는 Ubuntu 시스템에 대한 단계를 제공합니다.

절차

- 1 마스터 Linux VM에서 기타 종속 라이브러리(예: `smbfs` 및 `smbclient`)를 포함하여 `winbind` 및 `samba` 패키지를 설치합니다.
- 2 다음 명령을 사용하여 Samba `tdb-tools` 패키지를 설치합니다.

```
sudo apt-get install tdb-tools
```

- 3 Linux용 Horizon 7 Agent를 설치합니다.
- 4 다음 예제와 비슷한 콘텐츠를 포함하도록 `/etc/samba/smb.conf` 구성 파일을 편집합니다.

```
[global]
security = ads
realm = LAB.EXAMPLE.COM
workgroup = LAB
idmap uid = 10000-20000
idmap gid = 10000-20000
winbind enum users = yes
winbind enum group = yes
template homedir = /home/%D/%U
template shell = /bin/bash
client use spnego = yes
client ntlmv2 auth = yes
encrypt passwords = yes
winbind use default domain = yes
restrict anonymous = 2
```

- 5 다음 예제와 비슷한 콘텐츠를 포함하도록 `/etc/krb5.conf` 구성 파일을 편집합니다.

```
[libdefaults]
default_realm = EXAMPLE.COM

krb4_config = /etc/krb.conf
krb4_realms = /etc/krb.realms

kdc_timesync = 1
ccache_type = 4
forwardable = true
proxiable = true

[realms]
YOUR-DOMAIN = {
kdc = 10.111.222.33
}

[domain_realm]
your-domain = EXAMPLE.COM
.your-domain = EXAMPLE.COM
```

- 6 다음 예제에 표시된 것처럼 `/etc/nsswitch.conf` 구성 파일을 편집합니다.

```
passwd: files winbind
group: files winbind
shadow: files winbind
gshadow: files
```

- 7 호스트 이름이 올바른지와 시스템 날짜 및 시간이 DNS 시스템과 동기화되어 있는지 확인합니다.
- 8 Linux VM이 Samba 방법을 사용하여 도메인에 가입되었음을 Horizon Agent에 알려려면 `/etc/vmware/viewagent-custom.conf` 파일에서 다음 옵션을 설정합니다.

```
OfflineJoinDomain=samba
```

- 9 시스템을 재부팅하고 다시 로그인합니다.

RHEL/CentOS 8.x용 Realmd 가입 솔루션 사용

RHEL/CentOS 8.x 데스크톱에 대해 Single Sign-On과 같은 기능의 작동을 보장하려면 Realmd 솔루션을 사용하여 데스크톱을 AD(Active Directory) 도메인에 가입시킵니다.

절차

- 1 RHEL/CentOS 8.x 시스템에 대해 정규화된 호스트 이름을 구성합니다.

예를 들어, **rhel8**이 시스템의 정규화되지 않은 호스트 이름이고 **LXD.VDI**가 AD 도메인인 경우 다음 명령을 실행합니다.

```
# hostnamectl set-hostname rhel8.lxd.vdi
```

- 2 다음 예와 같이 AD 도메인과의 네트워크 연결을 확인합니다.

```
# realm discover -vvv LXD.VDI
```

- 3 다음 예와 같이 필요한 종속성 패키지를 설치합니다.

```
# dnf install -y sssd adcli samba-common-tools oddjob oddjob-mkhomedir
```

- 4 다음 예제와 같이 AD 도메인에 가입합니다.

```
# realm join -U Administrator LXD.VDI
```

- 5 다음 예와 유사하게 `/etc/sss/sss.conf`를 편집합니다. `[domain/domain name]` 섹션 아래에 `ad_gpo_map_interactive = +gdm-vmwcred`를 추가합니다.

```
[sss]
domains = LXD.VDI
config_file_version = 2
services = nss, pam

[domain/LXD.VDI]
```

```
ad_domain = LXD.VDI
krb5_realm = LXD.VDI
realmd_tags = manages-system joined-with-adcli
cache_credentials = True
id_provider = ad
krb5_store_password_if_offline = True
default_shell = /bin/bash
ldap_id_mapping = True
use_fully_qualified_names = False
fallback_homedir = /home/%u
access_provider = ad
ad_gpo_map_interactive = +gdm-vmwcred
```

- 6 도메인 가입이 적용되도록 하려면 시스템을 재부팅한 후 다시 로그인합니다.
- 7 도메인 사용자가 올바르게 구성되었는지 확인합니다. 다음 예에서는 **id** 명령을 사용하여 도메인 사용자 **zyc1**의 구성 출력을 반환하는 방법을 보여 줍니다.

```
# id zyc1

uid=1084401403(zyc1) gid=1084400513(domain users) groups=1084400513(domain users)
```

- 8 도메인 사용자의 자격 증명을 사용하여 데스크톱에 성공적으로 로그인할 수 있는지 확인합니다.

참고 Horizon Agent는 RHEL/CentOS 8.x 데스크톱에 대해 X11 디스플레이 서버 프로토콜만 지원합니다.

Single Sign-On 설정

SSO(Single Sign-On)를 설정하려면 몇 가지 구성 단계를 수행해야 합니다.

Horizon Single Sign-On 모듈에서는 Linux의 PAM(착탈형 인증 모듈)과 통신하며, Linux를 AD(Active Directory)에 통합하는 데 사용하는 방법에 종속되지 않습니다. Horizon SSO는 Linux를 AD와 통합하는 OpenLDAP 및 Winbind 솔루션에서 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.conf`에 다음 문을 추가합니다.

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

사용자가 로그인할 도메인 이름을 지정해야 하는 경우에는 Linux 데스크톱에서 **SSOUserFormat** 옵션을 설정해야 합니다. 자세한 내용은 [Linux 데스크톱의 구성 파일에서 옵션 설정](#)의 내용을 참조하십시오. SSO는 항상 대문자로 된 짧은 도메인 이름을 사용합니다. 예를 들어, 도메인이 **mydomain.com**이면 SSO에서는 **MYDOMAIN**을 도메인 이름으로 사용합니다. 따라서 **SSOUserFormat** 옵션을 설정할 때 **MYDOMAIN**을 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 지원하지는 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. 예를 들어 AD **mycompany.com**에 **juser** 사용자가 있지만 해당 사용자의 UPN이 **juser@mycompany.com** 대신 **juser123@mycompany.com**으로 설정되어 있으면 SSO는 실패합니다. 해결 방법은 사용자가 **sAMAccount**에 저장된 이름을 사용하여 로그인하는 것입니다. 예: **juser**.

Horizon 7에서는 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
```

Linux 데스크톱에 여러 데스크톱 환경이 설치되어 있는 경우 [데스크톱 환경](#)을 참조하여 SSO와 함께 사용할 데스크톱 환경을 선택합니다.

스마트 카드 리더렉션 설정

원격 데스크톱에서 스마트 카드 리더렉션을 사용하도록 설정하면 사용자가 클라이언트 시스템에 연결된 스마트 카드 판독기를 사용하여 데스크톱에서 인증을 받을 수 있습니다. 스마트 카드 리더렉션을 설정하려면 몇 가지 구성 단계를 수행해야 합니다.

스마트 카드 리더렉션 개요

스마트 카드 리더렉션은 특정 Horizon Agent 버전이 설치된 다음과 같은 Linux 배포를 실행하는 데스크톱에서 지원됩니다.

표 3-1. 스마트 카드 리더렉션에 대한 시스템 요구 사항

Linux 배포	Horizon Agent
RHEL 8.1	Horizon Agent 7.12 이상
RHEL 8.0	Horizon Agent 7.10 이상
RHEL 7.1 이상	Horizon Agent 7.8 이상
RHEL 6.6 이상	Horizon Agent 6.2.1 이상
Ubuntu 18.04/16.04	Horizon Agent 7.9 이상
SLED/SLES 12.x SP3	Horizon Agent 7.9 이상

Horizon Agent를 설치할 경우 먼저 SELinux를 사용하지 않도록 설정해야 합니다. 구성 요소가 기본적으로 선택되지 않으므로 스마트 카드 리더렉션 구성 요소도 구체적으로 선택해야 합니다. 자세한 내용은 [install_viewagent.sh 명령줄 옵션](#)의 내용을 참조하십시오.

가상 시스템에서 스마트 카드 리더렉션 기능을 사용하도록 설정한 경우에는 vSphere 클라이언트의 USB 리더렉션이 스마트 카드에서 작동하지 않습니다.

스마트카드 리더렉션은 한 번에 하나의 스마트카드 판독기만 지원합니다. 두 개 이상의 판독기가 클라이언트 시스템에 연결된 경우는 이 기능이 작동하지 않습니다.

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

참고 스마트 카드 리더렉션은 Linux 데스크톱에서 PIV 카드를 지원합니다. Linux용 Horizon Client를 사용하여 PIV 카드로 브로커를 인증할 경우 SSL 오류가 나타나지 않도록 하기 위해 TLSv1.2 지원을 사용해서 PIV 스마트 카드를 구성해야 합니다. VMware 기술 자료 문서 <http://kb.vmware.com/kb/2150470>에 설명된 해결 방법을 사용하십시오.

참고 스마트 카드 SSO는 Horizon 7 버전 7.0.1 이상에서 사용하도록 설정됩니다. RHEL 6.x 데스크톱은 스마트 카드 SSO를 지원하지 않지만 RHEL 7.x 및 RHEL 8.x 데스크톱은 이 기능을 지원하지 않습니다.

스마트 카드 리더렉션 구성

스마트 카드 리더렉션을 구성하려면 다음 작업을 수행합니다.

- 1 Linux 배포자 및 스마트 카드 벤더의 지침에 따라 데스크톱에 대한 스마트 카드를 설정합니다.
- 2 Linux 배포에 해당하는 절차에 따라 데스크톱을 Active Directory 도메인과 통합합니다.
- 3 Linux 배포에 해당하는 절차에 따라 데스크톱에서 스마트 카드 리더렉션을 구성합니다.

RHEL 8.x 데스크톱에 대한 스마트 카드 리더렉션 구성

RHEL 8.x 데스크톱에 대한 스마트 카드 방향을 설정하려면 먼저 **Active Directory** 도메인과 데스크톱을 통합합니다. 그런 다음, **Horizon Agent**를 설치하기 전에 필요한 라이브러리 및 루트 **CA** 인증서를 설치합니다.

스마트 카드 리더렉션을 위해 **RHEL 8.x** 데스크톱을 **Active Directory**와 통합

스마트 카드 리더렉션을 위해 RHEL 8.x 데스크톱을 AD(Active Directory) 도메인에 통합하려면 다음 절차를 사용합니다.

이 절차의 일부 예제는 **AD** 도메인의 **DNS** 이름과 같은 네트워크 구성의 엔티티를 나타내기 위해 자리 표시자 값을 사용합니다. 다음 표에 설명된 대로 자리 표시자 값을 구성과 관련된 정보로 바꿉니다.

자리 표시자 값	설명
<code>dns_IP_ADDRESS</code>	DNS 이름 서버의 IP 주소
<code>rhel8sc.rzview2.com</code>	RHEL 8.0 시스템의 정규화된 호스트 이름
<code>rhel8sc</code>	RHEL 8.0 시스템의 정규화되지 않은 호스트 이름
<code>rzview2.com</code>	AD 도메인의 DNS 이름
<code>RZVIEW2.COM</code>	AD 도메인의 DNS 이름(모두 대문자)
<code>RZVIEW2</code>	Samba 서버를 포함하는 작업 그룹 또는 NT 도메인의 DNS 이름(모두 대문자)
<code>rzviewdns.rzview2.com</code>	AD 서버의 호스트 이름

절차

- 1 RHEL 8.x 시스템에서 다음을 수행합니다.
 - a 조직에 필요한 네트워크 및 DNS 설정을 구성합니다.
 - b **IPv6**를 사용하지 않도록 설정합니다.
 - c **자동 DNS**를 사용하지 않도록 설정합니다.
- 2 다음 예와 유사하게 `/etc/hosts` 구성 파일을 구성합니다.

```
127.0.0.1      rhel8sc.rzview2.com rhel8sc localhost localhost.localdomain localhost4
localhost4.localdomain4
::1           localhost localhost.localdomain localhost6 localhost6.localdomain6

dns_IP_ADDRESS rzviewdns.rzview2.com
```

- 3 다음 예와 유사하게 `/etc/resolv.conf` 구성 파일을 구성합니다.

```
# Generated by NetworkManager
search rzview2.com
nameserver dns_IP_ADDRESS
```

- 4 AD 통합에 필요한 패키지를 설치합니다.

```
# yum install -y samba-common-tools oddjob-mkhomedir
```

- 5 oddjobd 서비스를 사용하도록 설정합니다.

```
# systemctl enable oddjobd.service
# systemctl start oddjobd.service
```

- 6 시스템 ID 및 인증 소스를 지정합니다.

```
# authselect select sssd with-smartcard with-mkhomedir
```

- 7 oddjobd 서비스를 시작합니다.

```
# systemctl enable oddjobd.service
# systemctl start oddjobd.service
```

- 8 스마트 카드 인증을 지원하려면 /etc/sss/sssd.conf 파일을 생성합니다.

```
# touch /etc/sss/sssd.conf
# chmod 600 touch /etc/sss/sssd.conf
# chown root:root /etc/sss/sssd.conf
```

- 9 다음 예와 같이 필요한 콘텐츠를 /etc/sss/sssd.conf에 추가합니다. **[pam]** 섹션에서 **pam_cert_auth = True**를 지정합니다.

```
[sss]
config_file_version = 2
domains = rzview2.com
services = nss, pam, pac

[domain/RZVIEW2.COM]
id_provider = ad
auth_provider = ad
chpass_provider = ad
access_provider = ad
cache_credentials = true

[pam]
pam_cert_auth = True
```

- 10 sssd 서비스를 사용하도록 설정합니다.

```
# systemctl enable sssd.service
# systemctl start sssd.service
```

- 11 다음 예와 유사하게 /etc/krb5.conf 구성 파일을 편집합니다.

```
# To opt out of the system crypto-policies configuration of krb5, remove the
# symlink at /etc/krb5.conf.d/crypto-policies which will not be recreated.
```

```

includedir /etc/krb5.conf.d/

[logging]
    default = FILE:/var/log/krb5libs.log
    kdc = FILE:/var/log/krb5kdc.log
    admin_server = FILE:/var/log/kadmind.log

[libdefaults]
    dns_lookup_realm = false
    ticket_lifetime = 24h
    renew_lifetime = 7d
    forwardable = true
    rdns = false
    pkinit_anchors = /etc/pki/tls/certs/ca-bundle.crt
    spake_preauth_groups = edwards25519
    default_realm = RZVIEW2.COM
    default_ccache_name = KEYRING:persistent:%{uid}

[realms]
    RZVIEW2.COM = {
        kdc = rzviewdns.rzview2.com
        admin_server = rzviewdns.rzview2.com
        default_domain = rzviewdns.rzview2.com
        pkinit_anchors = FILE:/etc/pki/nssdb/certificate.pem
        pkinit_cert_match = <KU>digitalSignature
        pkinit_kdc_hostname = rzviewdns.rzview2.com
    }

[domain_realm]
    .rzview2.com = RZVIEW2.COM
    rzview2.com = RZVIEW2.COM

```

12 다음 예와 유사하게 `/etc/samba/smb.conf` 구성 파일을 편집합니다.

```

[global]
    workgroup = RZVIEW2
    security = ads
    passdb backend = tdbsam
    printing = cups
    printcap name = cups
    load printers = yes
    cups options = raw
    password server = rzviewdns.rzview2.com
    realm = RZVIEW2.COM
    idmap config * : range = 16777216-33554431
    template homedir = /home/RZVIEW2/%U
    template shell = /bin/bash
    kerberos method = secrets and keytab

[homes]
    comment = Home Directories
    valid users = %S, %D%w%S
    browseable = No
    read only = No
    inherit acls = Yes

```

```
[printers]
    comment = All Printers
    path = /var/tmp
    printable = Yes
    create mask = 0600
    browseable = No

[print$]
    comment = Printer Drivers
    path = /var/lib/samba/drivers
    write list = @printadmin root
    force group = @printadmin
    create mask = 0664
    directory mask = 0775
```

13 다음 예제와 같이 AD 도메인에 가입합니다.

```
# net ads join -U AdminUser
```

join 명령을 실행하면 다음 예와 유사한 출력이 반환됩니다.

```
Enter AdminUser's password:
Using short domain name -- RZVIEW2
Joined 'RHEL8SC' to dns domain 'rzview2.com'
```

14 RHEL 8.x 데스크톱이 AD 도메인에 성공적으로 가입되었는지 확인합니다.

```
# net ads testjoin

Join is OK
```

다음에 수행할 작업

RHEL 8.x 데스크톱에 스마트 카드 리더렉션 구성

RHEL 8.x 데스크톱에 스마트 카드 리더렉션 구성

RHEL 8.x 데스크톱에서 스마트 카드 리더렉션을 구성하려면 기능이 의존하는 라이브러리, 신뢰할 수 있는 스마트 카드 인증을 지원하기 위한 루트 CA 인증서 및 필요한 PC/SC Lite 라이브러리를 설치합니다.

사전 요구 사항

스마트 카드 리더렉션을 위해 RHEL 8.x 데스크톱을 Active Directory와 통합

절차

1 필수 라이브러리를 설치합니다.

```
# yum install -y opencsc pcsc-lite pcsc-lite-libs pcsc-lite-ccid nss-tools
```

2 pcsd 서비스를 사용하도록 설정합니다.

```
# systemctl enable pcsd
# systemctl start pcsd
```

3 /etc/sss/sss.conf 구성 파일에 스마트 카드 인증을 사용하도록 설정하는 다음 줄이 포함되어 있는지 확인합니다.

```
[pam]
pam_cert_auth = True
```

4 필요한 CA 인증서를 /etc/sss/pki/sss_auth_ca_db.pem에 복사합니다.

```
# openssl x509 -inform der -in certificate.cer -out certificate.pem
# cp certificate.pem /etc/sss/pki/sss_auth_ca_db.pem
```

5 스마트 카드의 상태를 확인하려면 다음 pkcs11-tool 명령을 실행하고 올바른 출력을 반환하는지 확인합니다.

```
# pkcs11-tool -L

# pkcs11-tool --login -0

# pkcs11-tool --test --login
```

6 PKCS11 모듈을 설정합니다.

```
cp libcmP11.so /usr/lib64/
```

7 /usr/share/p11-kit/modules/libcmP11.module 파일을 생성합니다. 파일에 다음 콘텐츠를 추가합니다.

```
# This file describes how to load the opensc module
# See: http://p11-glue.freedesktop.org/doc/p11-kit/config.html

# This is a relative path, which means it will be loaded from
# the p11-kit default path which is usually $(libdir)/pkcs11.
# Doing it this way allows for packagers to package opensc for
# 32-bit and 64-bit and make them parallel installable
module: /usr/lib64/libcmP11.so
priority: 99
```

8 PC/SC Lite를 버전 1.8.8로 업데이트합니다.

```
# yum install -y git flex autoconf automake libtool libudev-devel flex
# git clone https://salsa.debian.org/rousseau/PCSC.git
# cd PCSC
# git checkout -b 1.8.8 pcsc-1.8.8
# ./bootstrap
# ./configure --build=x86_64-redhat-linux-gnu --host=x86_64-redhat-linux-gnu
--program-prefix= --disable-dependency-tracking --prefix=/usr --exec-prefix=/usr
```

```
--bindir=/usr/bin --sbindir=/usr/sbin --sysconfdir=/etc --datadir=/usr/share
--includedir=/usr/include --libdir=/usr/lib64 --libexecdir=/usr/libexec
--localstatedir=/var --sharedstatedir=/var/lib --mandir=/usr/share/man
--infodir=/usr/share/info --disable-static --enable-usbdropdir=/usr/lib64/pcsc/drivers
# make
# make install
```

- 9 스마트 카드 리더렉션을 사용하도록 설정하고 Horizon Agent 패키지를 설치합니다.

```
# sudo ./install_viewagent.sh -M yes
```

참고 스마트 카드 리더렉션을 사용하려면 다음 표에 설명된 대로 Linux 배포에 필요한 Horizon Agent 버전을 설치해야 합니다.

Linux 배포	Horizon Agent
RHEL 8.1	Horizon Agent 7.12 이상
RHEL 8.0	Horizon Agent 7.10 이상

- 10 시스템을 재부팅하고 다시 로그인합니다.

RHEL 7.x/6.x 데스크톱에 대한 스마트 카드 리더렉션 구성

RHEL 7.x/6.x 데스크톱에 대한 스마트 카드 방향을 설정하려면 먼저 **Active Directory** 도메인과 데스크톱을 통합합니다. 그런 다음, Horizon Agent를 설치하기 전에 필요한 라이브러리 및 루트 CA 인증서를 설치합니다.

스마트 카드 리더렉션을 위해 **RHEL 7.x/6.x** 데스크톱을 **Active Directory**와 통합

RHEL 7.x/6.x 데스크톱에서 스마트 카드 리더렉션을 지원하려면 **Samba** 및 **Winbind** 솔루션을 사용하여 **AD(Active Directory)** 도메인에 데스크톱을 통합합니다.

스마트 카드 리더렉션을 위해 **RHEL 7.x/6.x** 데스크톱을 **AD** 도메인에 통합하려면 다음 절차를 사용합니다.

이 절차의 일부 예제는 **AD** 도메인의 **DNS** 이름과 같은 네트워크 구성의 엔티티를 나타내기 위해 자리 표시자 값을 사용합니다. 다음 표에 설명된 대로 자리 표시자 값을 구성과 관련된 정보로 바꿉니다.

자리 표시자 값	설명
dns_IP_ADDRESS	DNS 이름 서버의 IP 주소
mydomain.com	AD 도메인의 DNS 이름
MYDOMAIN.COM	AD 도메인의 DNS 이름(모두 대문자)
MYDOMAIN	Samba 서버를 포함하는 작업 그룹 또는 NT 도메인의 DNS 이름(모두 대문자)
ads-hostname	AD 서버의 호스트 이름

참고 스마트 카드 리더렉션은 RHEL 6.0 이상 또는 RHEL 7.1 이상을 실행하는 데스크톱에서 지원됩니다.

절차

- 1 RHEL 7.x/6.x 데스크톱에 필요한 패키지를 설치합니다.

```
# yum install nscd samba-winbind krb5-workstation pam_krb5 samba-winbind-clients authconfig-gtk
```

- 2 시스템 연결에 대한 네트워크 설정을 편집합니다. **NetworkManager** 제어판을 열고 시스템 연결에 대한 **IPv4 설정**으로 이동합니다. IPv4 방법으로 **자동(DHCP)**을 선택합니다. **DNS** 텍스트 상자에 DNS 이름 서버의 IP 주소를 입력합니다. 그런 후 **적용**을 클릭합니다.
- 3 다음 명령을 실행하고 RHEL 데스크톱의 FQDN(정규화된 도메인 이름)을 반환하는지 확인합니다.

```
# hostname -f
```

- 4 다음 예제에 표시된 것처럼 `/etc/resolve.conf` 구성 파일을 편집합니다.

```
search mydomain.com
nameserver dns_IP_ADDRESS
```

- 5 RHEL 데스크톱에서 SELinux(Security-Enhanced Linux)를 사용하지 않도록 설정합니다. 다음 예제에 표시된 것처럼 `/etc/selinux/config` 구성 파일을 편집합니다.

```
SELINUX=disabled
```

- 6 다음 예제와 같이 `/etc/krb5.conf` 구성 파일을 편집합니다.

```
[libdefaults]
    dns_lookup_realm = false
    ticket_lifetime = 24h
    renew_lifetime = 7d
    forwardable = true
    rdns = false
    default_realm = MYDOMAIN.COM
    default_ccache_name = KEYRING:persistent:%{uid}

[realms]
    MYDOMAIN.COM = {
        kdc = ads-hostname
        admin_server = ads-hostname
        default_domain = ads-hostname
    }

[domain_realm]
    .mydomain.com = MYDOMAIN.COM
    mydomain.com = MYDOMAIN.COM
```

- 7 다음 예제와 같이 `/etc/samba/smb.conf` 구성 파일을 편집합니다.

```
[global]
    workgroup = MYDOMAIN
    password server = ads-hostname
    realm = MYDOMAIN.COM
```



```
security = ads
idmap config * : range = 16777216-33554431
template homedir = /home/MYDOMAIN/%U
template shell = /bin/bash
kerberos method = secrets and keytab
winbind use default domain = true
winbind offline logon = false
winbind refresh tickets = true

passdb backend = tdbsam
```

8 `authconfig-gtk` 도구를 열고 다음과 같이 설정을 구성합니다.

- a **ID 및 인증** 탭을 선택합니다. 사용자 계정 데이터베이스로 **Winbind**를 선택합니다.
- b **고급 옵션** 탭을 선택하고 **첫 번째 로그인 시 홈 디렉토리 생성** 확인란을 선택합니다.
- c **ID 및 인증** 탭을 선택하고 **도메인 가입**을 클릭합니다. 변경 내용을 저장할지 묻는 경고가 표시되면 **저장**을 클릭합니다.
- d 도메인 관리자의 사용자 이름 및 암호를 입력하라는 메시지가 표시되면 입력한 후 **확인**을 클릭합니다.

RHEL 데스크톱이 AD 도메인에 가입됩니다.

9 PAM Winbind에서 티켓 캐시를 설정합니다. 다음 예제에 표시된 줄을 포함하도록 `/etc/security/pam_winbind.conf` 구성 파일을 편집합니다.

```
[global]

# authenticate using kerberos
;krb5_auth = yes

# create homedirectory on the fly
;mkhomedir = yes
```

10 Winbind 서비스를 다시 시작합니다.

```
# sudo service winbind restart
```

11 AD 가입을 확인하려면 다음 명령을 실행하고 올바른 출력을 반환하는지 확인합니다.

- `net ads testjoin`
- `net ads info`

12 시스템을 재부팅하고 다시 로그인합니다.

다음에 수행할 작업

[RHEL 7.x/6.x 데스크톱에 대한 스마트 카드 리디렉션 설정](#)

RHEL 7.x/6.x 데스크톱에 대한 스마트 카드 리더렉션 설정

RHEL 7.x/6.x 데스크톱에서 스마트 카드 리더렉션을 구성하려면 기능이 의존하는 라이브러리, 인증에 필요한 루트 CA 인증서 및 필요한 PC/SC Lite 라이브러리를 설치합니다. 또한 인증 설정을 완료하려면 일부 구성 파일을 편집해야 합니다.

RHEL 7.x/6.x 데스크톱에 대한 스마트 카드 리더렉션을 설정하려면 다음 절차를 사용하십시오.

이 절차의 일부 예제는 AD 도메인의 DNS 이름과 같은 네트워크 구성의 엔티티를 나타내기 위해 자리 표시자 값을 사용합니다. 다음 표에 설명된 대로 자리 표시자 값을 구성과 관련된 정보로 바꿉니다.

자리 표시자 값	설명
dns_IP_ADDRESS	DNS 이름 서버의 IP 주소
mydomain.com	AD 도메인의 DNS 이름
MYDOMAIN.COM	AD 도메인의 DNS 이름(모두 대문자)
MYDOMAIN	Samba 서버를 포함하는 작업 그룹 또는 NT 도메인의 DNS 이름(모두 대문자)
ads-hostname	AD 서버의 호스트 이름

스마트 카드 리더렉션은 RHEL 6.0 이상 또는 RHEL 7.1 이상을 실행하는 데스크톱에서 지원됩니다.

참고 vSphere 콘솔을 사용하여 Horizon Agent가 설치되고 스마트 카드 리더렉션을 사용하도록 설정한 RHEL 7.x에 로그인하는 경우 로그아웃 시간이 2분 이상 지연될 수 있습니다. 이러한 지연된 로그아웃은 vSphere 콘솔에서만 발생합니다. Horizon Client의 RHEL 7.x 로그아웃 환경은 영향을 받지 않습니다.

사전 요구 사항

스마트 카드 리더렉션을 위해 RHEL 7.x/6.x 데스크톱을 Active Directory와 통합

절차

1 필수 라이브러리를 설치합니다.

```
yum install nss-tools nss-pam-ldapd esc pam_pkcs11 pam_krb5 opensc pcsc-lite-ccid authconfig
authconfig-gtk krb5-libs krb5-workstation krb5-pkinit pcsc-lite pcsc-lite-libs
```

2 루트 CA(인증 기관) 인증서를 설치합니다.

- a 루트 CA 인증서를 다운로드하고 데스크톱의 /tmp/certificate.cer에 저장합니다. 루트 인증 기관 인증서를 내보내는 방법을 참조하십시오.
- b 다운로드한 루트 CA 인증서를 찾은 후 .pem 파일에 전송합니다.

```
openssl x509 -inform der -in /tmp/certificate.cer -out /tmp/certificate.pem
```

- c `certutil` 명령을 사용하여 시스템 데이터베이스 `/etc/pki/nssdb`에 루트 CA 인증서를 설치합니다.

```
certutil -A -d /etc/pki/nssdb -n "root CA cert" -t "CT,C,C" -i /tmp/certificate.pem
```

- d 루트 CA 인증서를 `/etc/pam_pkcs11/cacerts` 디렉토리에 복사합니다.

```
mkdir -p /etc/pam_pkcs11/cacerts

cp /tmp/certificate.pem /etc/pam_pkcs11/cacerts
```

- 3 애플리케이션 > Sundry > 인증으로 이동한 후 스마트 카드 지원 사용 확인란을 선택하고 적용을 클릭합니다.
- 4 스마트카드 드라이버를 복사하고 드라이버 라이브러리를 시스템 데이터베이스 `/etc/pki/nssdb`에 추가합니다.

```
cp libcmP11.so /usr/lib64/
modutil -add "piv card 2.0" -libfile /usr/lib64/libcmP11.so -dbdir /etc/pki/nssdb/
```

- 5 다음 예제와 같이 `/etc/pam_pkcs11/pam_pkcs11.conf` 구성 파일에서 모듈 설정을 편집합니다.

```
pkcs11_module coolkey {
    module = libcmP11.so;
    description = "Cool Key";
    slot_num = 0;
    ca_dir = /etc/pam_pkcs11/cacerts;
    nss_dir = /etc/pki/nssdb;
    cert_policy = ca, signature;
}
```

- 6 다음 예제와 비슷한 콘텐츠를 포함하도록 `/etc/pam_pkcs11/cn_map` 파일을 편집합니다. 포함할 특정 콘텐츠에 대해서는 스마트 카드 인증서에 나열된 사용자 정보를 참조하십시오.

```
user sc -> user-sc
```

- 7 다음 예제에 표시된 것처럼 `/etc/krb5.conf/` 구성 파일을 편집합니다.

```
[libdefaults]
    dns_lookup_realm = false
    ticket_lifetime = 24h
    renew_lifetime = 7d
    forwardable = true
    rdns = false
    default_realm = MYDOMAIN.COM
    default_ccache_name = KEYRING:persistent:%{uid}

[realms]
    MYDOMAIN.COM = {
        kdc = ads-hostname
        admin_server = ads-hostname
        default_domain = ads-hostname
```

```

    pkinit_anchors = FILE:/etc/pki/nssdb/certificate.pem
    pkinit_cert_match = <KU>digitalSignature
    pkinit_kdc_hostname = ads-hostname
}

[domain_realm]
    .mydomain.com = MYDOMAIN.COM
    mydomain.com = MYDOMAIN.COM

```

- 8 다음 예제에 표시된 줄을 포함하도록 `/etc/pam.d/system-auth` 구성 파일을 편집합니다.

```

auth optional pam_krb5.so use_first_pass no_subsequent_prompt
preauth_options=X509_user_identity=PKCS11:/usr/lib64/libcnp11.so

```

- 9 PC/SC 데몬을 다시 시작합니다.

```

chkconfig pcscd on
service pcscd start

```

- 10 RHEL 배포를 위한 필수 PC/SC Lite 버전을 설치합니다.

- RHEL 7.x의 경우 PC/SC Lite 버전 1.8.8을 설치합니다.

```

yum install git flex autoconf automake libtool libudev-devel flex
git clone https://salsa.debian.org/rousseau/PCSC.git
cd PCSC
git checkout -b 1.8.8 pcsc-1.8.8
./bootstrap
./configure --build=x86_64-redhat-linux-gnu --host=x86_64-redhat-linux-gnu --program-prefix=
--disable-dependency-tracking --prefix=/usr --exec-prefix=/usr --bindir=/usr/bin --
sbindir=/usr/sbin
--sysconfdir=/etc --datadir=/usr/share --includedir=/usr/include --libdir=/usr/lib64
--libexecdir=/usr/libexec --localstatedir=/var --sharedstatedir=/var/lib --mandir=/usr/
share/man
--infodir=/usr/share/info --disable-static --enable-usbdropdir=/usr/lib64/pcsc/drivers
make
make install

```

- RHEL 6.x의 경우 PC/SC Lite 버전 1.7.4을 설치합니다.

```

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-conffdir=/etc --enable-ipcdir=/var/run --disable-libusb --disable-serial --disable-
usb
--disable-libudev
make
make install
service pcscd start

```

11 스마트 카드 리디렉션을 사용하도록 설정하고 Horizon Agent 패키지를 설치합니다.

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

RHEL 배포를 위한 필수 패키지를 설치합니다.

- RHEL 7.x의 경우 Horizon Agent 7.8 이상을 설치합니다.
- RHEL 6.x에서는 View Agent 6.2.1 이상을 설치합니다.

12 시스템을 재부팅하고 다시 로그인합니다.

Ubuntu 데스크톱에 대한 스마트 카드 리디렉션 구성

Ubuntu 데스크톱에 대한 스마트 카드 방향을 설정하려면 먼저 Active Directory 도메인과 데스크톱을 통합합니다. 그런 다음, Horizon Agent를 설치하기 전에 필요한 라이브러리 및 루트 CA 인증서를 설치합니다.

스마트 카드 리디렉션을 위해 Ubuntu 데스크톱을 Active Directory와 통합

Ubuntu 데스크톱에서 스마트 카드 리디렉션을 지원하려면 Samba 및 Winbind 솔루션을 사용하여 AD(Active Directory) 도메인에 데스크톱을 통합합니다.

스마트 카드 리디렉션을 위해 Ubuntu 데스크톱을 AD 도메인에 통합하려면 다음 절차를 사용합니다.

이 절차의 일부 예제는 AD 도메인의 DNS 이름과 같은 네트워크 구성의 엔티티를 나타내기 위해 자리 표시자 값을 사용합니다. 다음 표에 설명된 대로 자리 표시자 값을 구성과 관련된 정보로 바꿉니다.

자리 표시자 값	설명
dns_IP_ADDRESS	DNS 이름 서버의 IP 주소
mydomain.com	AD 도메인의 DNS 이름
MYDOMAIN.COM	AD 도메인의 DNS 이름(모두 대문자)
MYDOMAIN	Samba 서버를 포함하는 작업 그룹 또는 NT 도메인의 DNS 이름(모두 대문자)
ads-hostname	AD 서버의 호스트 이름
ads-hostname.mydomain.com	AD 서버의 FQDN(정규화된 도메인 이름)
mytimeserver.mycompany.com	NTP 시간 서버의 DNS 이름
AdminUser	Linux 데스크톱 관리자의 사용자 이름

절차

1 Ubuntu 데스크톱에서 /etc/hostname 구성 파일을 편집하여 데스크톱의 호스트 이름을 정의합니다.

2 DNS를 구성합니다.

- a DNS 서버 이름 및 IP 주소를 `/etc/hosts` 구성 파일에 추가합니다.
- b 다음 예와 같이 DNS 이름 서버의 IP 주소와 AD 도메인의 DNS 이름을 `/etc/network/interfaces` 구성 파일에 추가합니다.

```
dns-nameservers dns_IP_ADDRESS
dns-search mydomain.com
```

3 resolvconf 패키지를 설치합니다.

- a 설치 명령을 실행합니다.

```
# apt-get install -y resolvconf
```

시스템이 패키지를 설치하고 재부팅하도록 허용합니다.

- b 다음 예제와 같이 `/etc/resolv.conf` 파일에서 DNS 구성을 확인합니다.

```
# cat /etc/resolv.conf
...
nameserver dns_IP_ADDRESS
search mydomain.com
```

4 네트워크 시간 동기화를 구성합니다.

- a ntpdate 패키지를 설치합니다.

```
# apt-get install -y ntpdate
```

- b 다음 예제와 같이 NTP 서버 정보를 `/etc/systemd/timesyncd.conf` 구성 파일에 추가합니다.

```
[Time]
NTP=mytimeserver.mycompany.com
```

5 NTP 서비스를 다시 시작합니다.

```
sudo service ntpdate restart
```

6 필요한 AD 가입 패키지를 설치합니다.

- a 설치 명령을 실행합니다.

```
# apt-get install -y samba krb5-config krb5-user winbind libpam-winbind
libnss-winbind
```

- b 기본 Kerberos 영역을 묻는 설치 메시지가 표시되면 AD 도메인의 DNS 이름을 대문자로 입력합니다(예: MYDOMAIN.COM). 그런 다음, **확인**을 선택합니다.

7 다음 예제와 같이 /etc/krb5.conf 구성 파일을 편집합니다.

```
[libdefaults]
    dns_lookup_realm = false
    ticket_lifetime = 24h
    renew_lifetime = 7d
    forwardable = true
    rdns = false
    default_realm = MYDOMAIN.COM
    default_ccache_name = KEYRING:persistent:%{uid}

[realms]
    MYDOMAIN.COM = {
        kdc = ads-hostname.mydomain.com
        admin_server = ads-hostname.mydomain.com
        default_domain = ads-hostname.mydomain.com
        pkinit_anchors = FILE:/etc/pki/nssdb/certificate.pem
        pkinit_cert_match = <KU>digitalSignature
        pkinit_kdc_hostname = ads-hostname.mydomain.com
    }

[domain_realm]
    .mydomain.com = MYDOMAIN.COM
    mydomain.com = MYDOMAIN.COM
```

8 Kerberos 인증을 확인하려면 다음 명령을 실행합니다.

```
# kinit Administrator@MYDOMAIN.COM

# klist
```

명령이 다음 예제와 유사한 출력을 반환하는지 확인합니다.

```
Ticket cache: FILE:/tmp/krb5cc_0
Default principal: Administrator@MYDOMAIN.COM
principal
2019-05-27T17:12:03    2019-05-28T03:12:03    krbtgt/MYDOMAIN.COM@MYDOMAIN.COM
    renew until 2019-05-28T17:12:03
```

9 다음 예제와 같이 /etc/samba/smb.conf 구성 파일을 편집합니다.

```
[global]
    workgroup = MYDOMAIN
    realm = MYDOMAIN.COM
    password server = ads-hostname.mydomain.com
    security = ads
    kerberos method = secrets only
    winbind use default domain = true
    winbind offline logon = false
    template homedir = /home/%D/%U
    template shell = /bin/bash
    client use spnego = yes
    client ntlmv2 auth = yes
    encrypt passwords = yes
```

```
passdb backend = tdbsam
winbind enum users = yes
winbind enum groups = yes
idmap uid = 10000-20000
idmap gid = 10000-20000
```

10 AD 도메인에 가입하고 통합을 확인합니다.

a AD 가입 명령을 실행합니다.

```
# net ads join -U AdminUser@mydomain.com
# systemctl stop samba-ad-dc
# systemctl enable smbdc nmbd winbind
# systemctl restart smbdc nmbd winbind
```

b 다음 예제와 같이 /etc/nsswitch.conf 구성 파일을 수정합니다.

```
passwd:      compat systemd winbind
group:       compat systemd winbind
shadow:      compat
gshadow:     files
```

c AD 가입 결과를 확인하려면 다음 명령을 실행하고 올바른 출력을 반환하는지 확인합니다.

```
# wbinform -u
# wbinform -g
```

d Winbind 이름 서비스 스위치를 확인하려면 다음 명령을 실행하고 올바른 출력을 반환하는지 확인합니다.

```
# getent group | grep 'domain admins'
# getent passwd | grep 'ads-hostname'
```

11 모든 PAM 프로파일을 사용하도록 설정합니다.

```
# pam-auth-update
```

PAM 구성 화면에서 **로그인 시 홈 디렉토리 생성**을 포함하는 모든 PAM 프로파일을 선택한 다음, **확인**을 선택합니다.

12 Ubuntu 16.04에서 로그인 화면에 있는 사용자 스위치를 사용하도록 설정합니다. 다음 예제와 같이 /usr/share/lightdm/lightdm.conf.d/50-ubuntu.conf 파일을 수정합니다.

```
user-session=ubuntu
greeter-show-manual-login=true
```

다음에 수행할 작업

[Ubuntu 데스크톱에 대한 스마트 카드 리디렉션 설정](#)

Ubuntu 데스크톱에 대한 스마트 카드 리더렉션 설정

Ubuntu 데스크톱에서 스마트 카드 리더렉션을 구성하려면 해당 기능이 의존하는 라이브러리와 스마트 카드의 신뢰할 수 있는 인증을 지원하기 위한 루트 CA 인증서를 설치합니다. 또한 인증 설정을 완료하려면 일부 구성 파일을 편집해야 합니다.

이 절차의 일부 예제는 AD 도메인의 DNS 이름과 같은 네트워크 구성의 엔티티를 나타내기 위해 자리 표시자 값을 사용합니다. 다음 표에 설명된 대로 자리 표시자 값을 구성과 관련된 정보로 바꿉니다.

자리 표시자 값	설명
dns_IP_ADDRESS	DNS 이름 서버의 IP 주소
mydomain.com	AD 도메인의 DNS 이름
MYDOMAIN.COM	AD 도메인의 DNS 이름(모두 대문자)
MYDOMAIN	Samba 서버를 포함하는 작업 그룹 또는 NT 도메인의 DNS 이름(모두 대문자)
ads-hostname	AD 서버의 호스트 이름
ads-hostname.mydomain.com	AD 서버의 FQDN(정규화된 도메인 이름)
mytimeserver.mycompany.com	NTP 시간 서버의 DNS 이름
AdminUser	Linux 데스크톱 관리자의 사용자 이름

사전 요구 사항

스마트 카드 리더렉션을 위해 [Ubuntu 데스크톱을 Active Directory와 통합](#)

절차

1 필요한 라이브러리를 설치합니다.

```
# apt-get install -y pcscd pcsc-tools pkg-config libpam-pkcs11 opensc
libengine-pkcs11-openssl libnss3-tools
```

2 루트 CA(인증 기관) 인증서를 설치합니다.

- 루트 CA 인증서를 다운로드하고 데스크톱의 /tmp/certificate.cer에 저장합니다. [루트 인증 기관 인증서를 내보내는 방법](#)을 참조하십시오.
- 다운로드한 루트 CA 인증서를 찾은 후 .pem 파일에 전송합니다.

```
# openssl x509 -inform der -in /tmp/certificate.cer -out /tmp/certificate.pem
```

- c `certutil` 명령을 사용하여 시스템 데이터베이스 `/etc/pki/nssdb`에 루트 CA 인증서를 설치합니다.

```
# certutil -A -d /etc/pki/nssdb -n "root CA cert" -t "CT,C,C" -i /tmp/certificate.pem
```

- d 루트 CA 인증서를 `/etc/pam_pkcs11/cacerts` 디렉토리에 복사합니다.

```
# mkdir -p /etc/pam_pkcs11/cacerts  
  
# cp /tmp/certificate.pem /etc/pam_pkcs11/cacerts
```

- 3 `pkcs11` 해시 파일을 생성합니다.

```
# chmod a+r certificate.pem  
# pkcs11_make_hash_link
```

4 필요한 드라이버를 복사하고 필수 라이브러리 파일을 **nssdb** 디렉토리에 추가합니다.

a 다음 명령을 실행합니다.

```
# cp libcmP11.so /usr/lib/
# mkdir -p /etc/pki/nssdb
# certutil -N -d /etc/pki/nssdb
# certutil -A -n rootca -i certificate.pem -t "CT,CT,CT" -d /etc/pki/nssdb
# modutil -dbdir /etc/pki/nssdb/ -add "piv card 2.0" -libfile /usr/lib/libcmP11.so
```

b 필요한 인증서가 성공적으로 로드되었는지 확인합니다.

```
# certutil -L -d /etc/pki/nssdb

Certificate Nickname

rootca
```

c 필요한 라이브러리가 성공적으로 추가되었는지 확인합니다.

```
modutil -dbdir /etc/pki/nssdb -list

Listing of PKCS #11 Modules
-----
1. NSS Internal PKCS #11 Module
   slots: 2 slots attached
   status: loaded

   slot: NSS Internal Cryptographic Services
   token: NSS Generic Crypto Services

   slot: NSS User Private Key and Certificate Services
   token: NSS Certificate DB

2. piv card 2.0
   library name: /usr/lib/libcmP11.so
   slots: There are no slots attached to this module
   status: loaded
-----
```

5 pam_pkcs11 라이브러리를 구성합니다.

- a 기본 예제 콘텐츠를 사용하여 `pam_pkcs11.conf` 파일을 생성합니다.

```
# mkdir /etc/pam_pkcs11
# zcat /usr/share/doc/libpam-pkcs11/examples/pam_pkcs11.conf.example.gz |
tee /etc/pam_pkcs11/pam_pkcs11.conf
```

- b 다음 예제와 같이 `/etc/pam_pkcs11/pam_pkcs11.conf` 파일을 편집합니다.

```
use_pkcs11_module = mysc;

pkcs11_module mysc {
    module = /usr/lib/libcmP11.so;
    description = "LIBCMP11";
    slot_num = 0;
    ca_dir = /etc/pki/cacerts;
    nss_dir = /etc/pki/nssdb;
    cert_policy = ca;
}
...
use_mappers = cn, null;
...
mapper cn {
    debug = false;
    module = internal;
    # module = /lib/pam_pkcs11/cn_mapper.so;
    ignorecase = true;
    mapfile = file:///etc/pam_pkcs11/cn_map;
    # mapfile = "none";
}
```

- c 다음 줄을 포함하도록 `/etc/pam_pkcs11/cn_map` 파일을 편집합니다.

```
ads-hostname -> ads-hostname
```

6 PAM 인증을 구성합니다.

- a /etc/pam.d/gdm-password 구성 파일을 편집합니다. 다음 예제와 같이 **common-auth** 줄 앞에 **pam_pkcs11.so** 인증 줄을 넣습니다.

```
#%PAM-1.0
auth    requisite      pam_nologin.so
auth    required       pam_succeed_if.so user != root quiet_success
auth    sufficient
pam_pkcs11.so
@include common-auth
auth    optional       pam_gnome_keyring.so
@include common-account
```

- b Ubuntu 16.04의 경우 /etc/pam.d/lightdm 구성 파일을 편집합니다. 다음 예제와 같이 **common-auth** 줄 앞에 **pam_pkcs11.so** 인증 줄을 넣습니다.

```
#%PAM-1.0
auth    requisite      pam_nologin.so debug
auth    sufficient     pam_succeed_if.so user ingroup nopasswdlogin debug
auth    [success=3 default=ignore]    pam_pkcs11.so
@include common-auth
auth    optional       pam_gnome_keyring.so
auth    optional       pam_kwallet.so
```

- c Ubuntu 16.04의 경우 /etc/pam.d/unity 구성 파일을 편집합니다. 다음 예제와 같이 **common-auth** 줄 앞에 **pam_pkcs11.so** 인증 줄을 넣습니다.

```
auth    [success=3 default=ignore]    pam_pkcs11.so
@include common-auth
auth    optional       pam_gnome_keyring.so
```

- 7 스마트 카드에 설치된 스마트 카드 하드웨어와 인증서를 확인하려면 다음 명령을 실행합니다.

```
# pcsc_scan

# pkcs11_listcerts

# pkcs11_inspect
```

- 8 시스템이 재부팅된 후 자동으로 시작되도록 **pcscd** 서비스를 구성합니다.

참고 시스템이 재부팅된 후에 **pcscd** 서비스가 시작되지 않으면 **pam_pkcs11**를 통한 첫 번째 로그인 이 실패합니다.

- a [Install] 섹션에 **WantedBy=multi-user.target** 줄을 추가하여 **/lib/systemd/system/pcscd.service** 파일을 편집합니다.

편집한 파일이 다음 예와 유사한지 확인합니다.

```
[Unit]
Description=PC/SC Smart Card Daemon
Requires=pcscd.socket

[Service]
ExecStart=/usr/sbin/pcscd --foreground --auto-exit
ExecReload=/usr/sbin/pcscd --hotplug

[Install]
WantedBy=multi-user.target
Also=pcscd.socket
```

- b **pcscd** 서비스를 사용하도록 설정합니다.

```
# systemctl enable pcscd.service
```

- 9 다음 명령 시퀀스를 사용하여 **PC/SC Lite** 라이브러리를 버전 **1.8.8**로 업데이트합니다.

```
# apt-get install -y git autoconf automake libtool flex libudev-dev
# git clone https://salsa.debian.org/rousseau/PCSC.git
# cd PCSC/
# git checkout -b 1.8.8 pcsc-1.8.8
# ./bootstrap
# ./configure --prefix=/usr --sysconfdir=/etc --libdir=/lib/x86_64-linux-gnu/
  CFLAGS="-g -O2 -fstack-protector-strong -Wformat -Werror=format-security"
  LIBS="-ldl" LDFLAGS="-Wl,-Bsymbolic-functions -Wl,-z,relro" CPPFLAGS="-Wdate-time -
D_FORTIFY_SOURCE=2"
# make
# make install
```

- 10 스마트 카드 리더렉션을 사용하도록 설정하고 **Horizon Agent** 패키지를 설치합니다.

```
# sudo ./install_viewagent.sh -m yes
```

참고 Horizon Agent 7.9 이상을 설치해야 합니다.

- 11 시스템을 재부팅하고 다시 로그인합니다.

SLED/SLES 데스크톱에 대한 스마트 카드 리더렉션 구성

SLED/SLES 데스크톱에 대한 스마트 카드 방향을 설정하려면 먼저 **Active Directory** 도메인과 데스크톱을 통합합니다. 그런 다음, **Horizon Agent**를 설치하기 전에 필요한 라이브러리 및 루트 **CA** 인증서를 설치합니다.

스마트 카드 리더렉션을 위해 **SLED/SLES** 데스크톱을 **Active Directory**와 통합

SLED/SLES 데스크톱에서 스마트 카드 리더렉션을 지원하려면 **Samba** 및 **Winbind** 솔루션을 사용하여 **AD(Active Directory)** 도메인에 데스크톱을 통합합니다.

스마트 카드 리더렉션을 위해 **SLED/SLES** 데스크톱을 **AD** 도메인에 통합하려면 다음 절차를 사용합니다.

이 절차의 일부 예제는 **AD** 도메인의 **DNS** 이름과 같은 네트워크 구성의 엔티티를 나타내기 위해 자리 표시자 값을 사용합니다. 다음 표에 설명된 대로 자리 표시자 값을 구성과 관련된 정보로 바꿉니다.

자리 표시자 값	설명
dns_IP_ADDRESS	DNS 이름 서버의 IP 주소
mydomain.com	AD 도메인의 DNS 이름
MYDOMAIN.COM	AD 도메인의 DNS 이름(모두 대문자)
MYDOMAIN	Samba 서버를 포함하는 작업 그룹 또는 NT 도메인의 DNS 이름(모두 대문자)
ads-hostname	AD 서버의 호스트 이름
ads-hostname.mydomain.com	AD 서버의 FQDN(정규화된 도메인 이름)
mytimeserver.mycompany.com	NTP 시간 서버의 DNS 이름
AdminUser	Linux 데스크톱 관리자의 사용자 이름

절차

1 SLED/SLES 데스크톱에 대한 네트워크 설정을 구성합니다.

- a /etc/hostname 및 /etc/hosts 구성 파일을 편집하여 데스크톱의 호스트 이름을 정의합니다.
- b DNS 서버 IP 주소를 구성하고 **자동 DNS**를 사용하지 않도록 설정합니다. SLES 12 SP3의 경우 **DHCP를 통해 호스트 이름 변경**도 사용하지 않도록 설정합니다.
- c 네트워크 시간 동기화를 구성하려면 다음 예제와 같이 **NTP** 서버 정보를 /etc/ntp.conf 파일에 추가합니다.

```
server mytimeserver.mycompany.com
```

2 필요한 AD 가입 패키지를 설치합니다.

```
# zypper in krb5-client samba-winbind
```

3 필요한 구성 파일을 편집합니다.

- a 다음 예제와 같이 `/etc/samba/smb.conf` 파일을 편집합니다.

```
[global]
    workgroup = MYDOMAIN
    usershare allow guests = NO
    idmap gid = 10000-20000
    idmap uid = 10000-20000
    kerberos method = secrets and keytab
    realm = MYDOMAIN.COM
    security = ADS
    template homedir = /home/%D/%U
    template shell = /bin/bash
    winbind use default domain=true
    winbind offline logon = yes
    winbind refresh tickets = yes

[homes]
    ...
```

- b 다음 예제와 같이 `/etc/krb5.conf` 파일을 편집합니다.

```
[libdefaults]
    default_realm = MYDOMAIN.COM
    clocks skew = 300

[realms]
    MYDOMAIN.COM = {
        kdc = ads-hostname.mydomain.com
        default_domain = mydomain.com
        admin_server = ads-hostname.mydomain.com
    }

[logging]
    kdc = FILE:/var/log/krb5/krb5kdc.log
    admin_server = FILE:/var/log/krb5/kadmind.log
    default = SYSLOG:NOTICE:DAEMON

[domain_realm]
    .mydomain.com = MYDOMAIN.COM
    mydomain.com = MYDOMAIN.COM

[appdefaults]
    pam = {
        ticket_lifetime = 1d
        renew_lifetime = 1d
        forwardable = true
        proxiable = false
        minimum_uid = 1
    }
```


- c 다음 예제와 같이 `/etc/security/pam_winbind.conf` 파일을 편집합니다.

```
cached_login = yes
krb5_auth = yes
krb5_ccache_type = FILE
```

- d 다음 예제와 같이 `/etc/nsswitch.conf` 파일을 편집합니다.

```
passwd: compat winbind
group: compat winbind
```

- 4 다음 예제와 같이 AD 도메인에 가입합니다.

```
# net ads join -U AdminUser
```

- 5 Winbind 서비스를 사용하도록 설정합니다.

- a Winbind를 사용하도록 설정하고 시작하려면 다음 명령 순서를 실행합니다.

```
# pam-config --add --winbind
# pam-config -a --mkhomedir
# systemctl enable winbind
# systemctl start winbind
```

- b AD 사용자가 Linux 서버를 다시 시작하지 않고 데스크톱에 로그인할 수 있도록 하려면 다음 명령 순서를 실행합니다.

```
# systemctl stop nscd
# nscd -i passwd
# nscd -i group
# systemctl start nscd
```

- 6 AD 가입을 확인하려면 다음 명령을 실행한 후 올바른 출력을 반환하는지 확인합니다.

```
# wbinfo -u

# wbinfo -g
```

다음에 수행할 작업

SLED/SLES 데스크톱에 대한 스마트 카드 리더렉션 설정

SLED/SLES 데스크톱에 대한 스마트 카드 리더렉션 설정

SLED/SLES 데스크톱에서 스마트 카드 리더렉션을 구성하려면 해당 기능이 의존하는 라이브러리와 스마트 카드의 신뢰할 수 있는 인증을 지원하기 위한 루트 CA 인증서를 설치합니다. 또한 인증 설정을 완료하려면 일부 구성 파일을 편집해야 합니다.

이 절차의 일부 예제는 AD 도메인의 DNS 이름과 같은 네트워크 구성의 엔티티를 나타내기 위해 자리 표시자 값을 사용합니다. 다음 표에 설명된 대로 자리 표시자 값을 구성과 관련된 정보로 바꿉니다.

자리 표시자 값	설명
dns_IP_ADDRESS	DNS 이름 서버의 IP 주소
mydomain.com	AD 도메인의 DNS 이름
MYDOMAIN.COM	AD 도메인의 DNS 이름(모두 대문자)
MYDOMAIN	Samba 서버를 포함하는 작업 그룹 또는 NT 도메인의 DNS 이름(모두 대문자)
ads-hostname	AD 서버의 호스트 이름
ads-hostname.mydomain.com	AD 서버의 FQDN(정규화된 도메인 이름)
mytimeserver.mycompany.com	NTP 시간 서버의 DNS 이름
AdminUser	Linux 데스크톱 관리자의 사용자 이름

사전 요구 사항

스마트 카드 리더렉션을 위해 [SLED/SLES 데스크톱을 Active Directory와 통합](#)

절차

1 필요한 라이브러리 패키지를 설치합니다.

a PAM 라이브러리 및 기타 패키지를 설치합니다.

```
# zypper install pam_pkcs11 mozilla-nss mozilla-nss-tools
pcsc-lite pcsc-ccid opensc coolkey pcsc-tools
```

b PC/SC 도구를 설치하려면 다음 명령 시리즈를 실행합니다.

```
# SUSEConnect --list-extensions
# SUSEConnect -p PackageHub/12.3/x86_64
# zypper in pcsc-tools
```

2 루트 CA(인증 기관) 인증서를 설치합니다.

a 루트 CA 인증서를 다운로드하고 데스크톱의 /tmp/certificate.cer에 저장합니다. [루트 인증 기관 인증서를 내보내는 방법](#)을 참조하십시오.

b 다운로드한 루트 CA 인증서를 찾은 후 .pem 파일에 전송하고 해시 파일을 생성합니다.

```
# openssl x509 -inform der -in /tmp/certificate.cer -out /tmp/certificate.pem
# cp /tmp/certificate.pem /etc/pam_pkcs11/cacerts
# chmod a+r /etc/pam_pkcs11/cacerts/certificate.pem
# cd /etc/pam_pkcs11/cacerts
# pkcs11_make_hash_link
```

- c NSS 데이터베이스에 신뢰 앵커를 설치합니다.

```
# mkdir /etc/pam_pkcs11/nssdb
# certutil -N -d /etc/pam_pkcs11/nssdb
# certutil -L -d /etc/pam_pkcs11/nssdb
# certutil -A -n rootca -i certificate.pem -t "CT,CT,CT" -d /etc/pam_pkcs11/nssdb
```

- d 필요한 드라이버를 설치합니다.

```
# cp libcmP11.so /usr/lib64/
# modutil -add "piv card 2.0" -libfile /usr/lib64/libcmP11.so -dbdir /etc/pam_pkcs11/nssdb/
```

3 /etc/pam_pkcs11/pam_pkcs11.conf 파일을 편집합니다.

- a use_pkcs11_module = nss 줄을 삭제합니다. 해당 위치에 use_pkcs11_module = mysc 줄을 추가합니다.
- b 다음 예제와 같이 mysc 모듈을 추가합니다.

```
pkcs11_module mysc {
    module = /usr/lib64/libcmP11.so;
    description = "MY Smartcard";
    slot_num = 0;
    nss_dir = /etc/pam_pkcs11/nssdb;
    cert_policy = ca, ocsp_on, signature, crl_auto;
}
```

- c 다음 예제와 같이 일반 이름 매핑 구성을 업데이트합니다.

```
# Assume common name (CN) to be the login
mapper cn {
    debug = false;
    module = internal;
    # module = /usr/lib64/pam_pkcs11/cn_mapper.so;
    ignorecase = true;
    mapfile = file:///etc/pam_pkcs11/cn_map;}
```

- d use_mappers = ms 줄을 삭제합니다. 해당 위치에 use_mappers = cn, null 줄을 추가합니다.

4 다음 줄을 포함하도록 /etc/pam_pkcs11/cn_map 구성 파일을 편집합니다.

```
ads-hostname -> ads-hostname
```

5 PAM 구성을 수정합니다.

- a 스마트 카드 인증을 구성할 수 있도록 하려면 먼저 **pam_config** 도구를 사용하지 않도록 설정합니다.

```
# find /etc/pam.d/ -type l -iname "common-*" -delete
# for X in /etc/pam.d/common-*-pc; do cp -ivp $X ${X:0:-3}; done
```

- b `/etc/pam.d/` 디렉토리에 `common-auth-smartcard`라는 파일을 생성합니다. 파일에 다음 콘텐츠를 추가합니다.

```
auth    required      pam_env.so
auth    sufficient    pam_pkcs11.so
auth    optional      pam_gnome_keyring.so
auth    [success=1 default=ignore] pam_unix.so nullok_secure try_first_pass
auth    required      pam_winbind.so use_first_pass
```

- c SLED/SLES 12 SP3의 경우 두 파일 `/etc/pam.d/gdm` 및 `/etc/pam.d/xscreensaver`에서 `auth include common-auth` 줄을 `auth include common-auth-smartcard` 줄로 바꿉니다.

- 6 시스템이 재부팅된 후에 **pcscd** 서비스가 자동으로 시작되도록 구성하려면 `/etc/init.d/after.local` 파일을 편집합니다. 파일이 다음 예와 유사해지도록 `rcpcscd start` 줄을 추가합니다.

```
#!/bin/sh
#
# Copyright (c) 2010 SuSE LINUX Products GmbH, Germany. All rights reserved.
#
# Author: Werner Fink, 2010
#
# /etc/init.d/after.local
#
# script with local commands to be executed from init after all scripts
# of a runlevel have been executed.
#
# Here you should add things, that should happen directly after
# runlevel has been reached.
#
rcpcscd start
```

참고 시스템이 재부팅된 후에 **pcscd** 서비스가 시작되지 않으면 **pam_pkcs11**를 통한 첫 번째 로그인 이 실패합니다.

- 7 방화벽을 사용하지 않도록 설정합니다.

```
# rcSuSEfirewall2 stop
# chkconfig SuSEfirewall2_setup off
# chkconfig SuSEfirewall2_init off
```

참고 방화벽이 사용하도록 설정되어 있으면 스마트 카드 리더렉션이 실패할 수 있습니다.

8 스마트 카드 리더렉션에 필요한 라이브러리 패키지를 설치합니다.

a SLED/SLES 12 SP3의 경우 다음 설치 명령을 실행합니다.

```
# SUSEConnect -p sle-sdk/12.3/x86_64
# zypper in git autoconf automake libtool flex libudev-devel gcc
```

b SLES 12 SP3의 경우 `systemd-devel`을 설치합니다.

```
# zypper in systemd-devel
```

9 다음 명령 시퀀스를 사용하여 PC/SC Lite 라이브러리를 버전 1.8.8로 업데이트합니다.

```
# SUSEConnect -p sle-sdk/12.3/x86_64
# zypper in git autoconf automake libtool flex libudev-devel gcc
# git clone https://salsa.debian.org/rousseau/PCSC.git
# cd PCSC/
# git checkout -b 1.8.8 pcsc-1.8.8
# ./bootstrap
# ./configure --program-prefix= --disable-dependency-tracking --prefix=/usr --exec-prefix=/usr
--bindir=/usr/bin --sbindir=/usr/sbin >--sysconfdir=/etc --datadir=/usr/share --includedir=/usr/
include
--libdir=/usr/lib64 --libexecdir=/usr/libexec --localstatedir=/var >--sharedstatedir=/var/lib64
--mandir=/usr/share/man --infodir=/usr/share/info --disable-static --enable->usbdropdir=/usr/
lib64/pcsc/drivers
# make
# make install
```

10 스마트 카드 리더렉션을 사용하도록 설정하고 Horizon Agent 패키지를 설치합니다.

```
# sudo ./install_viewagent.sh -m yes
```

참고 Horizon Agent 7.9 이상을 설치해야 합니다.

11 시스템을 재부팅하고 다시 로그인합니다.

Linux 데스크톱용 True SSO 설정

True SSO(True Single Sign-on) 기능은 사용자가 VMware Workspace ONE에 처음 로그인한 후 Linux 가상 데스크톱이나 게시된 데스크톱 또는 애플리케이션에 액세스할 수 있도록 허용합니다. 사용자는 스마트 카드나 RSA SecurID 또는 RADIUS 인증을 사용하여 VMware Workspace ONE에 로그인한 다음, Active Directory 자격 증명을 입력하지 않고 원격 Linux 리소스에 액세스할 수 있습니다.

True SSO 개요

사용자가 AD(Active Directory) 자격 증명을 사용하여 인증을 받는 경우 True SSO 기능이 필요하지 않습니다. 그러나 이 경우에도 사용될 수 있게 True SSO를 구성하여 데스크톱이 AD 자격 증명 및 True SSO를 모두 지원하도록 할 수 있습니다.

Linux 가상 데스크톱이나 게시된 데스크톱 또는 애플리케이션에 연결할 때 사용자는 기본 Horizon Client 또는 HTML Access를 사용하도록 선택할 수 있습니다.

True SSO에 대한 시스템 요구 사항

True SSO는 특정 Horizon Agent 버전이 설치된 다음과 같은 Linux 배포를 실행하는 데스크톱에서 지원됩니다.

Linux 배포	Horizon Agent
RHEL/CentOS 8.1	Horizon Agent 7.12 이상
RHEL/CentOS 8.0	Horizon Agent 7.11 이상
RHEL/CentOS 7.x	Horizon Agent 7.6 이상 참고 True SSO는 기본 가입 도메인 도구, Samba, SSSD(시스템 보안 서비스 데몬) 및 Kerberos 네트워크 인증 프로토콜 가입 방법의 RHEL/CentOS 7.x 데스크톱에서만 지원됩니다.
Ubuntu 18.04/16.04	Horizon Agent 7.8 이상
SLED/SLES 12.x SP3	Horizon Agent 7.8 이상

True SSO 구성

Linux 데스크톱에서 True SSO를 설정하려면 다음 작업을 수행합니다.

- 1 Horizon 7 환경에서 True SSO를 설정하고 구성합니다. "Horizon 7 관리" 문서에서 "True SSO 설정"을 참조하십시오.
- 2 Linux 배포에 해당하는 절차에 따라 데스크톱을 AD 도메인과 통합합니다.
- 3 Linux 배포에 해당하는 절차에 따라 데스크톱에서 True SSO를 구성합니다.

RHEL/CentOS 8.x 데스크톱에서 True SSO 구성

RHEL/CentOS 8.x 데스크톱에서 True SSO를 지원하려면 먼저 시스템을 AD(Active Directory) 도메인과 통합해야 합니다. 그런 다음, True SSO 기능을 지원하도록 시스템의 특정 구성을 수정해야 합니다.

RHEL/CentOS 8.1 데스크톱에서 True SSO를 지원하려면 Horizon Agent 7.12 이상을 설치해야 합니다.

RHEL/CentOS 8.0 데스크톱에서 True SSO를 지원하려면 Horizon Agent 7.11 이상을 설치해야 합니다.

참고 인스턴트 클론 RHEL 8.x 데스크톱에서는 True SSO가 지원되지 않습니다.

이 절차의 일부 예제는 AD 도메인의 DNS 이름과 같은 네트워크 구성의 엔티티를 나타내기 위해 자리 표시자 값을 사용합니다. 다음 표에 설명된 대로 자리 표시자 값을 구성과 관련된 정보로 바꿉니다.

자리 표시자 값	설명
mydomain.com	AD 도메인의 DNS 이름
MYDOMAIN.COM	AD 도메인의 DNS 이름(모두 대문자)
MYDOMAIN	NetBIOS 도메인의 이름

사전 요구 사항

- RHEL/CentOS 8.x 시스템의 DNS에서 AD(Active Directory) 서버를 확인할 수 있는지 확인합니다.
- 시스템의 호스트 이름을 구성합니다.
- 시스템에서 NTP(Network Time Protocol)를 구성합니다.

절차

- 1 RHEL/CentOS 8.x 시스템에서 Active Directory에 대한 네트워크 연결을 확인합니다.

```
# realm discover "mydomain.com"
```

- 2 필요한 종속성 패키지를 설치합니다.

```
# yum install oddjob oddjob-mkhomedir sssd adcli samba-common-tools
```

- 3 AD 도메인에 가입합니다.

```
# realm join --verbose "mydomain.com" -U administrator
```

- 4 루트 CA 인증서를 다운로드하고 필요한 디렉토리에 .pem 파일로 복사합니다.

```
# openssl x509 -inform der -in /tmp/certificate.cer -out /tmp/certificate.pem
# cp /tmp/certificate.pem /etc/sssdpki/sssdpki_auth_ca_db.pem
```

- 5 다음 예제와 같이 /etc/sssdpki/sssdpki.conf 구성 파일을 수정합니다.

```
[sssdpki]
domains = "mydomain.com"
config_file_version = 2
services = nss, pam

[domain/mydomain.com]
ad_domain = "mydomain.com"
krb5_realm = I "MYDOMAIN.COM"
realmd_tags = manages-system joined-with-adcli
cache_credentials = True
id_provider = ad
krb5_store_password_if_offline = True
default_shell = /bin/bash
ldap_id_mapping = True
use_fully_qualified_names = False <----- Use short name for user
fallback_homedir = /home/%u@%d
```

```

access_provider = ad
ad_gpo_map_interactive = +gdm-vmwcred <----- Add this line for SSO

[pam] <----- Add pam section for certificate logon
pam_cert_auth = True <----- Add this line to enable certificate
logon for system
pam_p11_allowed_services = +gdm-vmwcred <----- Add this line to enable certificate
logon for VMware Horizon Agent

[certmap/ "mydomain.com" /truesso] <----- Add this section and following
lines to set match and map rule for certificate user
matchrule = <EKU>msScLogin
maprule = (&(|(userPrincipal={subject_principal})(samAccountName={subject_principal.short_name}))
domains = "mydomain.com"
priority = 10

```

6 True SSO를 사용하도록 설정하고 Horizon Agent 패키지를 설치합니다.

```
# sudo ./install_viewagent.sh -T yes
```

참고 True SSO 기능을 사용하려면 다음 표에 설명된 대로 Linux 배포에 필요한 Horizon Agent 버전을 설치해야 합니다.

Linux 배포	Horizon Agent
RHEL/CentOS 8.1	Horizon Agent 7.12 이상
RHEL/CentOS 8.0	Horizon Agent 7.11 이상

7 다음 줄을 포함하도록 /etc/vmware/viewagent-custom.conf 구성 파일을 수정합니다.

```
NetbiosDomain = "MYDOMAIN"
```

8 시스템을 재부팅하고 다시 로그인합니다.

RHEL/CentOS 7.x 데스크톱에 대한 True SSO 구성

RHEL/CentOS 7.x 데스크톱에 대한 True SSO를 설정하려면 먼저 Active Directory 도메인과 데스크톱을 통합합니다. 그런 다음, Horizon Agent를 설치하기 전에 필요한 라이브러리 및 루트 CA 인증서를 설치합니다.

True SSO를 위해 RHEL/CentOS 7.x 데스크톱을 Active Directory와 통합

RHEL/CentOS 7.x 시스템의 Horizon 7 Linux 데스크톱 환경에서 인스턴트 클론 VM에 대해 True SSO를 지원하려면 마스터 Linux VM에서 Samba를 구성해야 합니다.

RHEL/CentOS 7.x realmd 기능은 ID 도메인을 검색하고 가입하는 간단한 방법을 제공합니다. 시스템을 도메인 자체에 연결하는 대신, realmd는 SSSD 또는 Winbind와 같은 기본 Linux 시스템 서비스가 도메인에 연결하도록 구성합니다. 다음 단계에서는 realmd 및 Samba를 사용하여 Active Directory에 대해 RHEL/CentOS 7.x 데스크톱의 오프라인 도메인 가입을 수행하는 방법을 설명합니다.

사전 요구 사항

- RHEL(RedHat Enterprise Linux) 시스템이 RHN(Red Hat Network)을 구독하거나 yum 도구가 로컬로 설치되어 있습니다.
- AD(Active Directory) 서버는 Linux 시스템에서 DNS로 확인할 수 있습니다.
- NTP(Network Time Protocol)는 Linux 시스템에서 구성됩니다.

절차

- 1 RHEL/CentOS 시스템이 AD 서버를 검색할 수 있는지 확인합니다. 다음 예제를 사용합니다. 여기서 *ADdomain.example.com*을 사용자의 AD 서버 정보로 바꾸십시오.

```
sudo realm discover ADdomain.example.com
```

- 2 Samba tdb-tools 패키지를 설치합니다.

Samba tdb-tools 패키지는 공식 Red Hat 저장소에서 다운로드할 수 없습니다. 수동으로 다운로드해야 합니다. 예를 들어 다음 명령을 사용하여 CentOS 7.5 시스템에서 다운로드한 후 다운로드한 패키지를 RHEL 시스템에 설치합니다.

```
yumdownloader tdb-tools
```

CentOS 시스템이 없는 경우 <https://rpmfind.net/linux/rpm2html/search.php?query=tdb-tools&submit=Search+...&system=&arch=>로 이동하여 tdb-tools-1.3.15-1.el7.x86_64.rpm 패키지를 다운로드한 후 이를 RHEL 시스템에 설치합니다.

- 3 Samba 및 종속성 패키지를 설치합니다.

```
sudo yum install sssd-tools sssd adcli samba-common pam_ldap pam_krb5 samba samba-client krb5-workstation
```

- 4 다음 예제를 사용하여 join 명령을 실행합니다. 여기서 *DNSdomain.example.com*을 사용자 환경에 해당하는 특정 DNS 도메인 경로로 바꾸어야 합니다.

```
sudo realm join DNSdomain.example.com -U administrator
```

join 명령이 성공적으로 수행되면 다음 메시지가 표시됩니다.

```
시스템이 영역에 등록되었습니다.
```

- 5 시스템을 재부팅하고 다시 로그인합니다.

다음에 수행할 작업

[RHEL/CentOS 7.x 데스크톱에서 True SSO 구성](#)

RHEL/CentOS 7.x 데스크톱에서 True SSO 구성

RHEL/CentOS 7.x 데스크톱에서 True SSO 기능을 사용하도록 설정하려면 True SSO 기능이 의존하는 라이브러리, 신뢰할 수 있는 인증을 지원하기 위한 루트 CA 인증서 및 Horizon Agent를 설치합니다. 또한 인증 설정을 완료하려면 일부 구성 파일을 편집해야 합니다.

RHEL 7.x 및 CentOS 7.x 데스크톱에서 True SSO를 사용하도록 설정하려면 다음 절차를 사용하십시오. 이러한 데스크톱에서 True SSO를 지원하려면 Horizon Agent 7.6 이상을 설치해야 합니다.

이 절차의 일부 예제는 AD 도메인의 DNS 이름과 같은 네트워크 구성의 엔티티를 나타내기 위해 자리 표시자 값을 사용합니다. 다음 표에 설명된 대로 자리 표시자 값을 구성과 관련된 정보로 바꿉니다.

자리 표시자 값	설명
dns_server	DNS 이름 서버의 경로
mydomain.com	AD 도메인의 DNS 이름
MYDOMAIN.COM	AD 도메인의 DNS 이름(모두 대문자)

사전 요구 사항

- VMware Identity Manager 및 Horizon Connection Server에 대해 True SSO를 구성합니다.
- True SSO를 위해 RHEL/CentOS 7.x 데스크톱을 Active Directory와 통합
- 루트 인증 기관 인증서를 가져오고 RHEL/CentOS 7.x 데스크톱의 /tmp/certificate.cer에 저장합니다. 루트 인증 기관 인증서를 내보내는 방법을 참조하십시오.

절차

- 1 PKCS11 지원 패키지 그룹을 설치합니다.

```
yum install -y nss-tools nss-pam-ldapd pam_krb5 krb5-libs krb5-workstation krb5-pkinit
```

- 2 루트 CA(인증 기관) 인증서를 설치합니다.

- a 다운로드한 루트 CA 인증서를 찾은 후 .pem 파일에 전송합니다.

```
openssl x509 -inform der -in /tmp/certificate.cer -out /tmp/certificate.pem
```

- b certutil 명령을 사용하여 시스템 데이터베이스 /etc/pki/nssdb에 루트 CA 인증서를 설치합니다.

```
certutil -A -d /etc/pki/nssdb -n "root CA cert" -t "CT,C,C" -i /tmp/certificate.pem
```

- c RHEL/CentOS 7.x 시스템에서 신뢰할 수 있는 CA 인증서 목록에 루트 CA 인증서를 추가하고 update-ca-trust 명령을 사용하여 시스템 전체 신뢰 저장소 구성을 업데이트합니다.

```
cp /tmp/certificate.pem /etc/pki/ca-trust/source/anchors/ca_cert.pem
update-ca-trust
```

- 3 다음 예제에 표시된 대로 도메인에 대한 시스템의 SSSD 구성 파일에서 해당 섹션을 수정합니다.

```
[domain/mydomain.com]
ad_domain = mydomain.com
krb5_realm = MYDOMAIN.COM
realmd_tags = manages-system joined-with-samba
cache_credentials = True
id_provider = ad
krb5_store_password_if_offline = True
default_shell = /bin/bash
ldap_id_mapping = True
#set the next line to false, so you can use the short name instead of the full domain name.
use_fully_qualified_names = False
fallback_homedir = /home/%u@%d
access_provider = ad
```

- 4 다음 예제에 표시된 것처럼 Kerberos 구성 파일 /etc/krb5.conf를 수정합니다.

```
[libdefaults]
dns_lookup_realm = false
ticket_lifetime = 24h
renew_lifetime = 7d
forwardable = true
rdns = false
default_ccache_name = KEYRING:persistent:%{uid}
# Add following line, if the system doesn't add it automatically
default_realm = MYDOMAIN.COM

[realms]
MYDOMAIN.COM = {
    kdc = dns_server
    admin_server = dns_server
    # Add the following three lines for pkinit_*
    pkinit_anchors = DIR:/etc/pki/ca-trust/source/anchors
    pkinit_kdc_hostname = your_org_DNS_server
    pkinit_eku_checking = kpServerAuth
}
[domain_realm]
mydomain.com = MYDOMAIN.COM
.mydomain.com = MYDOMAIN.COM
```

- 5 True SSO를 사용하도록 설정하고 Horizon Agent 패키지를 설치합니다.

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

참고 Horizon Agent 7.6 이상을 설치해야 합니다.

- 6 다음 매개 변수를 Horizon Agent 사용자 지정 구성 파일 /etc/vmware/viewagent-custom.conf에 추가합니다. 다음 예를 사용하십시오. 여기서 **NETBIOS_NAME_OF_DOMAIN**은 조직 도메인의 NetBIOS 이름입니다.

```
NetbiosDomain=NETBIOS_NAME_OF_DOMAIN
```

7 시스템을 재부팅하고 다시 로그인합니다.

Ubuntu 데스크톱에 대한 True SSO 구성

Ubuntu 데스크톱에 대한 True SSO를 설정하려면 먼저 Active Directory 도메인과 데스크톱을 통합합니다. 그런 다음, Horizon Agent를 설치하기 전에 필요한 라이브러리 및 루트 CA 인증서를 설치합니다.

True SSO를 위해 Ubuntu 데스크톱을 Active Directory와 통합

Ubuntu 16.04 또는 18.04 데스크톱에서 True SSO를 지원하려면 Samba 및 Winbind 솔루션을 사용하여 데스크톱을 Active Directory 도메인과 통합합니다.

Ubuntu 16.04 또는 18.04 데스크톱을 AD 도메인과 통합하려면 다음 절차를 사용합니다.

이 절차의 일부 예제는 Ubuntu 데스크톱의 호스트 이름과 같은 네트워크 구성의 엔티티를 나타내기 위해 자리 표시자 값을 사용합니다. 다음 표에 설명된 대로 자리 표시자 값을 구성과 관련된 정보로 바꿉니다.

자리 표시자 값	설명
dns_IP_ADDRESS	DNS 이름 서버의 IP 주소
mydomain.com	AD 도메인의 DNS 이름
MYDOMAIN.COM	AD 도메인의 DNS 이름(모두 대문자)
myhost	Ubuntu 데스크톱의 호스트 이름
MYDOMAIN	Samba 서버를 포함하는 작업 그룹 또는 NT 도메인의 DNS 이름(모두 대문자)
ads-hostname	AD 서버의 호스트 이름
admin-user	AD 도메인 관리자의 사용자 이름

사전 요구 사항

- AD(Active Directory) 서버는 Linux 시스템에서 DNS로 확인할 수 있습니다.
- NTP(Network Time Protocol)는 Linux 시스템에서 구성됩니다.

절차

- 1 Ubuntu 16.04 또는 18.04 데스크톱에서 samba 및 winbind 패키지를 설치합니다.

```
sudo apt install samba krb5-config krb5-user winbind libpam-winbind libnss-winbind
```

- 2 메시지가 표시되면 다음과 같이 Kerberos 인증 설정을 구성합니다.

- a 기본 Kerberos 버전 5 영역으로 AD 도메인의 DNS 이름을 모두 대문자로 입력합니다.

예를 들어, AD 도메인 이름이 mydomain.com이면 MYDOMAIN.COM을 입력합니다.

- b 사용자 영역에 대한 Kerberos 서버에 AD 서버의 호스트 이름(이 절차 전체의 예제에서 ads_hostname으로 표시)을 입력합니다.

- c Kerberos 영역에 대한 관리 서버에 AD 서버의 호스트 이름을 다시 입력합니다.

3 PAM 구성을 업데이트합니다.

- a PAM 구성 페이지를 엽니다.

```
pam-auth-update
```

- b 로그인 시 홈 디렉토리 생성을 선택하고 확인을 선택합니다.

4 다음 예제에 표시된 것처럼 /etc/nsswitch.conf 구성 파일을 편집합니다.

```
passwd: compat winbind
group: compat winbind
shadow: compat
gshadow: files
```

5 자동 생성된 resolv.conf 파일이 AD 도메인을 검색 도메인으로 참조하도록 하려면 시스템 연결에 대한 NetworkManager 설정을 편집합니다.

- a NetworkManager 제어판을 열고 시스템 연결에 대한 **IPv4 설정**으로 이동합니다. 방법으로 **자동 (DHCP) 주소만**을 선택합니다. **DNS 서버** 텍스트 상자에 DNS 이름 서버의 IP 주소(이 절차 전체의 예제에서 **dns_IP_ADDRESS**로 표시)를 입력합니다. 그런 후 **저장**을 클릭합니다.
- b /etc/NetworkManager/system-connections에 있는 시스템 연결에 대한 구성 파일을 편집합니다. 다음 예제를 사용하십시오.

```
[ipv4]
dns=dns_IP_ADDRESS
dns-search=mydomain.com
ignore-auto-dns=true
method=auto
```

참고 새 인스턴트 클론 가상 데스크톱이 생성될 때 새 가상 네트워크 어댑터가 추가됩니다. 인스턴트 클론 가상 데스크톱에 새 네트워크 어댑터가 추가되면 가상 데스크톱 템플릿에서 DNS 서버와 같은 네트워크 어댑터의 모든 설정이 손실됩니다. 복제된 가상 데스크톱에 새 네트워크 어댑터를 추가할 때 DNS 서버 설정이 손실되지 않도록 하려면 Linux 시스템에서 DNS 서버를 지정해야 합니다.

- c 다음 예제에 표시된 것처럼 /etc/resolv.conf 구성 파일을 편집하여 DNS 서버를 지정합니다.

```
nameserver dns_IP_ADDRESS

search mydomain.com
```

- d 시스템을 재부팅하고 다시 로그인합니다.

6 다음 예제에 표시된 것처럼 /etc/hosts 구성 파일을 편집합니다.

```
127.0.0.1    localhost
127.0.1.1    myhost.mydomain.com myhost
```

- 7 다음 예제에 표시된 것처럼 `/etc/samba/smb.conf` 구성 파일을 편집합니다.

```
[global]
security = ads
realm = MYDOMAIN.COM
workgroup = MYDOMAIN
idmap uid = 10000-20000
idmap gid = 10000-20000
winbind enum users = yes
winbind enum group = yes
template homedir = /home/%D/%U
template shell = /bin/bash
client use spnego = yes
client ntlmv2 auth = yes
encrypt passwords = yes
winbind use default domain = yes
restrict anonymous = 2
kerberos method = secrets and keytab
winbind refresh tickets = true
```

- 8 `smbd` 서비스를 다시 시작합니다.

```
sudo systemctl restart smbd.service
```

- 9 다음 예제와 비슷한 콘텐츠를 포함하도록 `/etc/krb5.conf` 구성 파일을 편집합니다.

```
[libdefaults]
    default_realm = MYDOMAIN.COM
    dns_lookup_realm = true
    dns_lookup_kdc = true

[realms]
    MYDOMAIN.COM = {
        kdc = ads-hostname
        admin_server = ads-hostname
    }

[domain_realm]
    .mydomain.com = MYDOMAIN.COMmydomain.com = MYDOMAIN.COM
```

- 10 Ubuntu 데스크톱을 AD 도메인에 가입합니다.

- a Kerberos 티켓을 시작합니다.

```
sudo kinit admin-user
```

메시지가 표시되면 관리자 암호를 입력합니다.

- b 티켓이 성공적으로 생성되었는지 확인합니다.

```
sudo klist
```

이 명령은 해당하는 유효 시작 시간 및 만료 시간을 포함하여 티켓에 대한 정보를 반환합니다.

- c Kerberos keytab 파일을 생성합니다.

```
sudo net ads keytab create -U admin-user
```

- d AD 도메인에 가입합니다.

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

11 Winbind 서비스를 다시 시작하고 확인합니다.

- a Winbind 서비스를 다시 시작합니다.

```
sudo systemctl restart winbind.service
```

- b Winbind 서비스를 확인하려면 다음 명령을 실행하고 올바른 출력을 반환하는지 확인합니다.

- `wbinfo -u`
- `wbinfo -g`
- `getend passwd`
- `getend group`

12 시스템을 재부팅하고 다시 로그인합니다.

다음에 수행할 작업

[Ubuntu 데스크톱에서 True SSO 구성](#)

Ubuntu 데스크톱에서 True SSO 구성

Ubuntu 16.04 또는 18.04 데스크톱에서 True SSO 기능을 사용하도록 설정하려면 True SSO 기능이 의존하는 라이브러리, 신뢰할 수 있는 인증을 지원하기 위한 루트 CA 인증서 및 Horizon Agent를 설치합니다. 또한 인증 설정을 완료하려면 일부 구성 파일을 편집해야 합니다.

Ubuntu 16.04 및 18.04 데스크톱에서 True SSO를 사용하도록 설정하려면 다음 절차를 사용하십시오. 이러한 데스크톱에서 True SSO를 지원하려면 Horizon Agent 7.8 이상을 설치해야 합니다.

사전 요구 사항

- VMware Identity Manager 및 Horizon Connection Server에 대해 True SSO를 구성합니다.
- True SSO를 위해 [Ubuntu 데스크톱을 Active Directory와 통합](#)
- 루트 인증 기관 인증서를 가져오고 데스크톱의 `/tmp/certificate.cer`에 저장합니다. [루트 인증 기관 인증서를 내보내는 방법](#)을 참조하십시오.

절차

- 1 Ubuntu 16.04 또는 18.04 데스크톱에서 `pkcs11` 지원 패키지를 설치합니다.

```
sudo apt install libpam-pkcs11
```

2 libnss3-tools 패키지를 설치합니다.

```
sudo apt install libnss3-tools
```

3 루트 CA(인증 기관) 인증서를 설치합니다.

- a 다운로드한 루트 CA 인증서를 찾은 후 .pem 파일에 전송합니다.

```
openssl x509 -inform der -in /tmp/certificate.cer -out /tmp/certificate.pem
```

- b certutil 명령을 사용하여 시스템 데이터베이스 /etc/pki/nssdb에 루트 CA 인증서를 설치합니다.

```
certutil -A -d /etc/pki/nssdb -n "root CA cert" -t "CT,C,C" -i /tmp/certificate.pem
```

- c 루트 CA 인증서를 /etc/pam_pkcs11/cacerts 디렉토리에 복사합니다.

```
mkdir -p /etc/pam_pkcs11/cacerts  
cp /tmp/certificate.pem /etc/pam_pkcs11/cacerts
```

- d 루트 CA 인증서에 대한 해시 링크를 생성합니다. /etc/pam_pkcs11/cacerts 디렉토리에서 다음 명령을 실행합니다.

```
pkcs11_make_hash_link
```

4 True SSO를 사용하도록 설정하고 Horizon Agent 패키지를 설치합니다.

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

참고 True SSO 기능을 사용하려면 Horizon Agent 7.8 이상을 설치해야 합니다.

- 5 다음 매개 변수를 Horizon Agent 사용자 지정 구성 파일 /etc/vmware/viewagent-custom.conf에 추가합니다. 다음 예를 사용하십시오. 여기서 **NETBIOS_NAME_OF_DOMAIN**은 조직 도메인의 NetBIOS 이름입니다.

```
NetbiosDomain=NETBIOS_NAME_OF_DOMAIN
```

6 /etc/pam_pkcs11/pam_pkcs11.conf 구성 파일을 편집합니다.

- a 필요한 경우 /etc/pam_pkcs11/pam_pkcs11.conf 구성 파일을 생성합니다. /usr/share/doc/libpam-pkcs11/examples에서 예제 파일을 찾아 /etc/pam_pkcs11 디렉토리로 복사한 후 파일 이름을 pam_pkcs11.conf로 바꿉니다. 필요에 따라 파일의 내용에 시스템 정보를 추가합니다.
- b 다음 예제와 비슷한 콘텐츠를 포함하도록 /etc/pam_pkcs11/pam_pkcs11.conf 구성 파일을 수정합니다.

```
use_pkcs11_module = coolkey;  
pkcs11_module coolkey {  
    module = /usr/lib/vmware/viewagent/sso/libvmwpkcs11.so;
```



```
slot_num = 0;
ca_dir = /etc/pam_pkcs11/cacerts;
nss_dir = /etc/pki/nssdb;
}
```

7 PAM 구성 파일에서 **auth** 매개 변수를 수정합니다.

a PAM 구성 파일을 엽니다.

- Ubuntu 16.04에 대해 `/etc/pam.d/lightdm`을 엽니다.
- Ubuntu 18.04에 대해 `/etc/pam.d/gdm-vmwcred`를 엽니다.

b 다음 예제에 표시된 것처럼 PAM 구성 파일을 편집합니다.

```
auth requisite pam_vmw_cred.so
auth sufficient pam_pkcs11.so try_first_pass
```

8 시스템을 재부팅하고 다시 로그인합니다.

SLED/SLES 데스크톱에 대한 True SSO 구성

SLED/SLES 데스크톱에 대한 True SSO를 설정하려면 먼저 Active Directory 도메인과 데스크톱을 통합합니다. 그런 다음, Horizon Agent를 설치하기 전에 필요한 라이브러리 및 루트 CA 인증서를 설치합니다.

True SSO를 위해 SLED/SLES 데스크톱을 Active Directory와 통합

SLED 12.x SP3 또는 SLES 12.x SP3 데스크톱에서 True SSO를 지원하려면 Samba 및 Winbind 솔루션을 사용하여 데스크톱을 Active Directory 도메인과 통합합니다.

SLED/SLES 데스크톱을 AD 도메인과 통합하려면 다음 절차를 사용합니다.

사전 요구 사항

- AD(Active Directory) 서버는 Linux 시스템에서 DNS로 확인할 수 있습니다.
- NTP(Network Time Protocol)는 Linux 시스템에서 구성됩니다.

절차

1 SLED/SLES 데스크톱에서 **samba** 및 **winbind** 패키지를 설치합니다.

```
zypper install samba-winbind krb5-client samba-winbind-32bit
```

2 YaST 설정 도구를 열고 **네트워크 서비스 > Windows 도메인 멤버 자격**으로 이동합니다.

3 [Windows 도메인 멤버 자격] 화면에서 다음과 같이 설정을 구성합니다.

- a **도메인 또는 작업 그룹**에 Samba 서버를 포함하는 작업 그룹 또는 NT 도메인의 DNS 이름을 모두 대문자로 입력합니다. 예를 들어, 작업 그룹 이름이 **mydomain**이면 **MYDOMAIN**을 입력합니다.
- b **Linux 인증을 위해 SMB 정보도 사용**을 선택합니다.
- c **로그인 시 홈 디렉토리 생성**을 선택합니다.

- d **오프라인 인증**을 선택합니다.
 - e **SSH에 대해 Single Sign-On**을 선택합니다.
- 4 도메인에 가입할지 묻는 메시지에서 **예**를 선택합니다.
- 5 지정된 작업 그룹의 관리자 이름 및 암호를 입력하고 **확인**을 선택합니다.
- SLED/SLES 데스크톱이 도메인에 성공적으로 가입되었다는 메시지가 나타납니다. **확인**을 선택합니다.
- 6 다음 매개 변수를 포함하도록 `/etc/samba/smb.conf` 구성 파일을 편집합니다.

```
[global]
...
winbind use default domain = yes
```

- 7 시스템을 재부팅하고 다시 로그인합니다.
- 8 SLED/SLES 데스크톱 통합을 테스트 및 확인합니다.

다음 테스트 명령을 실행하고 올바른 출력을 반환하는지 확인합니다. `mydomain`을 Samba 서버 작업 그룹 또는 NT 도메인의 이름으로 바꿉니다.

- `net ads testjoin`
- `net ads info`
- `wbinfo --krb5auth=mydomain\\open%open`
- `ssh localhost -l mydomain\\open`

다음에 수행할 작업

SLED/SLES 데스크톱에서 True SSO 구성

SLED/SLES 데스크톱에서 True SSO 구성

SLED/SLES 12.x SP3 데스크톱에서 True SSO 기능을 사용하도록 설정하려면 True SSO 기능이 의존하는 라이브러리, 신뢰할 수 있는 인증을 지원하기 위한 루트 CA 인증서 및 Horizon Agent를 설치합니다. 또한 인증 설정을 완료하려면 일부 구성 파일을 편집해야 합니다.

SLED 12.x SP3 및 SLES 12.x SP3 데스크톱에서 True SSO를 사용하도록 설정하려면 다음 절차를 사용하십시오. 이러한 데스크톱에서 True SSO를 지원하려면 Horizon Agent 7.8 이상을 설치해야 합니다.

사전 요구 사항

- VMware Identity Manager 및 Horizon Connection Server에 대해 True SSO를 구성합니다.
- True SSO를 위해 SLED/SLES 데스크톱을 Active Directory와 통합
- 루트 인증 기관 인증서를 가져오고 SLED/SLES 12.x SP3 데스크톱의 `/tmp/certificate.cer`에 저장합니다. 루트 인증 기관 인증서를 내보내는 방법을 참조하십시오.

절차

- 1 SLES 12.x SP3 데스크톱의 경우 다음 명령을 실행하여 필요한 패키지를 설치합니다.

```
zypper install mozilla-nss-tools pam_krb5 krb5-client krb5-plugin-preauth-pkinit
```

- 2 SLED 12.x SP3 데스크톱의 경우 다음 단계를 수행하여 필요한 패키지를 설치합니다.

- a SLED 데스크톱의 로컬 디스크에 SLES .iso 파일을 다운로드합니다(예: /tmp/SLE-12-SP3-Server-DVD-x86_64-GM-DVD1.iso).

필요한 krb5-plugin-preauth-pkinit 패키지는 SLES 시스템에 대해서만 사용할 수 있으므로 SLES .iso 파일을 SLED 데스크톱에 대한 패키지 소스로 추가해야 합니다.

- b SLED 데스크톱에 SLES .iso 파일을 마운트하고 필요한 패키지를 설치합니다.

```
sudo mkdir -p /mnt/sles
sudo mount -t iso9660 /tmp/SLE-12-SP3-Server-DVD-x86_64-GM-DVD1.iso /mnt/sles
sudo zypper ar -f /mnt/sles sles
zypper install mozilla-nss-tools pam_krb5 krb5-client krb5-plugin-preauth-pkinit
```

- c 설치가 완료되면 SLES .iso 파일을 마운트 해제합니다.

```
sudo umount /mnt/sles
```

- 3 루트 CA(인증 기관) 인증서를 설치합니다.

- a 다운로드한 루트 CA 인증서를 찾은 후 .pem 파일에 전송합니다.

```
openssl x509 -inform der -in /tmp/certificate.cer -out /tmp/certificate.pem
```

- b certutil 명령을 사용하여 시스템 데이터베이스 /etc/pki/nssdb에 루트 CA 인증서를 설치합니다.

```
certutil -A -d /etc/pki/nssdb -n "root CA cert" -t "CT,C,C" -i /tmp/certificate.pem
```

- c 루트 CA 인증서를 pam_pkcs11에 추가합니다.

```
cp /tmp/certificate.pem /etc/pki/ca-trust/source/anchors/ca_cert.pem
```

- 4 다음 예제와 비슷한 콘텐츠를 포함하도록 /etc/krb5.conf 구성 파일을 편집합니다.

```
[libdefaults]
    default_realm = MYDOMAIN.COM
    dns_lookup_realm = false
    ticket_lifetime = 24h
    renew_lifetime = 7d
    forwardable = true
    rdns = false
    default_ccache_name = KEYRING:persistent:%{uid}

[realms]
    MYDOMAIN.COM = {
```

```

    kdc = ads-hostname
    admin_server = ads-hostname
    pkinit_anchors = DIR:/etc/pki/ca-trust/source/anchors
    pkinit_kdc_hostname = ads-hostname
    pkinit_eku_checking = kpServerAuth
}

[domain_realm]
    .mydomain.com = MYDOMAIN.COMmydomain.com = MYDOMAIN.COM

```

다음 표에 설명된 대로 이 예제의 자리 표시자 값을 네트워크 구성과 관련된 정보로 바꿉니다.

자리 표시자 값	설명
<i>mydomain.com</i>	AD 도메인의 DNS 이름
<i>MYDOMAIN.COM</i>	AD 도메인의 DNS 이름(모두 대문자)
<i>ads-hostname</i>	AD 서버의 호스트 이름(대소문자 구분)

5 True SSO를 사용하도록 설정하고 Horizon Agent 패키지를 설치합니다.

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

참고 True SSO 기능을 사용하려면 Horizon Agent 7.8 이상을 설치해야 합니다.

6 다음 매개 변수를 Horizon Agent 사용자 지정 구성 파일 `/etc/vmware/viewagent-custom.conf`에 추가합니다. 다음 예를 사용하십시오. 여기서 `NETBIOS_NAME_OF_DOMAIN`은 조직 도메인의 NetBIOS 이름입니다.

```
NetbiosDomain=NETBIOS_NAME_OF_DOMAIN
```

7 시스템을 재부팅하고 다시 로그인합니다.

Linux 데스크톱의 그래픽 설정

4

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

3D 그래픽 설정을 위한 VM 클론 요구 사항

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

- vGPU의 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.

본 장은 다음 항목을 포함합니다.

- vGPU에 대해 지원되는 Linux 배포 구성
- vDGA용 RHEL 6.x 구성

vGPU에 대해 지원되는 Linux 배포 구성

지원되는 Linux 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.

참고 vGPU를 지원하는 NVIDIA 그래픽 카드 및 Linux 배포판에 대한 내용은 <https://docs.nvidia.com/grid/9.0/product-support-matrix/index.html>을 참조하십시오.

경고 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 패키지
와 이후 절차에서 Linux 가상 시스템에 설치하는 Linux 디스플레이 드라이버를 제공합니다.

사전 요구 사항

- 작업 환경에 vSphere 6.0 U1 이상 릴리스가 설치되어 있는지 확인하십시오.
- 필수 vGPU 그래픽 카드가 ESXi 호스트에 설치되어 있는지 확인합니다.

참고 vGPU를 지원하는 NVIDIA 그래픽 카드 및 Linux 배포판에 대한 내용은 <https://docs.nvidia.com/grid/9.0/product-support-matrix/index.html>을 참조하십시오.

절차

- 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	NVIDIA GRID vGPU를 선택합니다.
제품	ESXi 호스트에 설치되어 있는 버전(예: GRID K2)을 선택합니다.
운영 체제	VMware vSphere ESXi 버전을 선택합니다.

- 2 vGPU 소프트웨어 패키지 .zip 파일을 압축 해제합니다.
- 3 vGPU Manager 폴더를 ESXi 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.

사전 요구 사항

- Linux 가상 시스템을 데스크톱으로 사용할 준비가 되어 있는지 확인합니다. 자세한 내용은 [가상 시스템 생성 및 Linux 설치](#) 및 [원격 데스크톱 배포를 위한 Linux 시스템 준비](#)의 내용을 참조하십시오.
- 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](#)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.

참고 vGPU를 지원하는 NVIDIA 그래픽 카드 및 Linux 배포판에 대한 내용은 <https://docs.nvidia.com/grid/9.0/product-support-matrix/index.html>을 참조하십시오.

절차

- 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 예약을 클릭하고 확인을 클릭합니다.

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

7 가상 시스템의 전원을 켭니다.

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.

사전 요구 사항

- 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](#)의 내용을 참조하십시오.

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

절차

- 1 NVIDIA Linux 디스플레이 드라이버를 가상 시스템으로 복사합니다.
- 2 가상 시스템에 대한 원격 터미널을 열거나 **Ctrl+Alt+F2**를 입력하여 텍스트 콘솔로 전환하고 루트 권한으로 로그인한 후 **init 3** 명령을 실행하여 X Windows를 비활성화합니다.
- 3 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)
```

- 4 NVIDIA GRID vGPU 드라이버 패키지에 실행 가능한 플래그를 추가합니다.

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

- 5 NVIDIA GRID vGPU 설치 관리자를 시작합니다.

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

- 6 NVIDIA 소프트웨어 라이선스 계약에 동의하고 예를 선택하여 X 구성 설정을 자동으로 업데이트합니다.

다음에 수행할 작업

Linux 가상 시스템에 Horizon Agent를 설치합니다. [Linux 가상 시스템에 Horizon Agent 설치](#)의 내용을 참조하십시오.

구성된 Linux 가상 시스템을 포함하는 데스크톱 풀을 생성합니다. [Linux용 수동 데스크톱 풀 생성](#)의 내용을 참조하십시오.

NVIDIA 디스플레이 드라이버의 설치 여부 확인

Horizon 데스크톱 세션에 NVIDIA 드라이버 출력을 표시하여 Linux 가상 시스템에 NVIDIA 디스플레이 드라이버가 설치되어 있는지 확인할 수 있습니다.

사전 요구 사항

- NVIDIA 디스플레이 드라이버를 설치했는지 확인합니다.
- Linux 가상 시스템에 Horizon Agent가 설치되어 있는지 확인합니다. [Linux 가상 시스템에 Horizon Agent 설치](#)의 내용을 참조하십시오.
- 데스크톱 풀에 Linux 가상 시스템이 배포되어 있는지 확인합니다. [Linux용 수동 데스크톱 풀 생성](#)의 내용을 참조하십시오.

절차

- 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
```

결과

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 Horizon 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.x 구성

Horizon Linux용 데스크톱 7이 ESXi 호스트에서 vDGA 기능을 활용할 수 있도록 RHEL 6.x 게스트 운영 체제를 설정할 수 있습니다.

경고 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.

사전 요구 사항

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

절차

- 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.x 가상 시스템에 vDGA 패스스루 디바이스 추가

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

사전 요구 사항

- Linux 가상 시스템을 데스크톱으로 사용할 준비가 되어 있는지 확인합니다. 자세한 내용은 [가상 시스템 생성 및 Linux 설치](#) 및 [원격 데스크톱 배포를 위한 Linux 시스템 준비](#)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](#)의 내용을 참조하십시오.

절차

- 1 RHEL 6.x 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 디바이스**를 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.

사전 요구 사항

- PCI 디바이스가 RHEL 6.x 가상 시스템에 추가되었는지 확인하십시오. [RHEL 6.x 가상 시스템에 vDGA 패스스루 디바이스 추가](#)의 내용을 참조하십시오.

절차

1 기본 NVIDIA Nouveau 드라이버를 비활성화하고 블랙리스트에 추가합니다.

a grub.conf 파일을 편집합니다.

RHEL 6.x의 경우 이 파일은 `/boot/grub/grub.conf`입니다.

RHEL 버전	명령
6.x	<code>sudo vi /boot/grub/grub.conf</code>

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: GRID K2)을 선택합니다.
운영 체제	Linux 64비트 또는 Linux 32비트

5 가상 시스템에 연결하려면 원격 터미널을 열거나 **Ctrl+Alt+F2**를 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 실행합니다.

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

- 9 NVIDIA 소프트웨어 라이선스 계약에 동의하고 예를 선택하여 X 구성 설정을 업데이트합니다.

다음에 수행할 작업

Linux 가상 시스템에 Horizon Agent를 설치합니다. [Linux 가상 시스템에 Horizon Agent 설치](#)의 내용을 참조하십시오.

구성된 Linux 가상 시스템을 포함하는 데스크톱 풀을 생성합니다. [Linux용 수동 데스크톱 풀 생성](#)의 내용을 참조하십시오.

NVIDIA 디스플레이 드라이버의 설치 여부 확인

Horizon 데스크톱 세션에 NVIDIA 드라이버 출력을 표시하여 Linux 가상 시스템에 NVIDIA 디스플레이 드라이버가 설치되어 있는지 확인할 수 있습니다.

사전 요구 사항

- NVIDIA 디스플레이 드라이버를 설치했는지 확인합니다.
- Linux 가상 시스템에 Horizon Agent가 설치되어 있는지 확인합니다. [Linux 가상 시스템에 Horizon Agent 설치](#)의 내용을 참조하십시오.
- 데스크톱 풀에 Linux 가상 시스템이 배포되어 있는지 확인합니다. [Linux용 수동 데스크톱 풀 생성](#)의 내용을 참조하십시오.

절차

- 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
```

결과

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 Horizon 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.

Horizon Agent 설치

5

Horizon Connection Server가 데스크톱과 통신하고 데스크톱을 관리할 수 있도록 Linux 데스크톱에서 Horizon Agent를 설치해야 합니다.

본 장은 다음 항목을 포함합니다.

- Linux 가상 시스템에 Horizon Agent 설치
- Linux Agent용 인증서 구성
- Linux 가상 시스템에서 Horizon Agent 업그레이드
- Horizon 7 for Linux 시스템 제거

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 Linux 데스크톱 배포를 수행하는 경우 기본 가상 시스템에서 Horizon Agent를 설치할 수 있습니다.

경고 NVIDIA GRID vGPU 또는 vDGA를 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 Linux 배포 구성 또는 vDGA용 RHEL 6.x 구성 항목을 참조하십시오. 3D 그래픽 구성이 완료된 후에 Horizon Agent를 설치합니다.

2D 그래픽 구성의 경우, 원격 데스크톱 배포를 위한 Linux 시스템 준비에서 단계를 완료한 후에 Horizon Agent를 설치할 수 있습니다.

사전 요구 사항

- Linux 게스트 운영 체제를 데스크톱에서 사용할 준비가 되었는지 확인하십시오. [원격 데스크톱 배포를 위한 Linux 시스템 준비](#)의 내용을 참조하십시오.
- Linux용 Horizon Agent 설치 관리자 스크립트를 숙지합니다. [install_viewagent.sh 명령줄 옵션](#)의 내용을 참조하십시오.

절차

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

[데스크톱 및 최종 사용자 컴퓨팅] 섹션에서 VMware Horizon에 대해 [다운로드 구성 요소 보기]를 선택합니다. [Horizon 7 for Linux]에서 64비트 Linux 시스템용 VMware Horizon 7에 대한 다운로드 페이지를 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 "VMware-horizonagent-linux-x86_64-y.y.y-xxxxxxx.tar.gz"
```

- 3 tar ball 폴더로 이동합니다.
- 4 고급 사용자로 install_viewagent.sh 스크립트를 실행합니다.

명령줄 옵션의 목록은 [install_viewagent.sh 명령줄 옵션](#)을 참조하십시오.

예:

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

- 5 -A 옵션을 지정하지 않고 install_viewagent.sh를 실행한 경우는 Yes를 입력하여 EULA를 수락합니다.

설치 관리자는 사용자가 EULA에 동의해야만 계속 실행됩니다.

- 6 변경 내용을 적용하려면 Linux를 재부팅합니다.

결과

설치 후에 **viewagent** 서비스가 시작됩니다. `sudo service viewagent status`를 사용하여 서비스가 시작되었는지 확인합니다.

다음에 수행할 작업

데스크톱 풀에서 가상 시스템을 배포합니다. [Linux용 수동 데스크톱 풀 생성](#)의 내용을 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 no	EULA(최종 사용자 라이선스 계약) 및 FIPS(Federal Information Processing Standards) 선언을 수락하거나 거부합니다. 설치를 계속하려면 yes 를 지정해야 합니다.

표 5-2. install_viewagent.sh 옵션 매개 변수

옵션 매개 변수	설명
-a yes no	오디오 입력 리디렉션 지원을 설치하거나 우회합니다. 기본값은 yes 입니다.
-f yes no	FIPS(Federal Information Processing Standards) 140-2용으로 고안된 암호화 모듈의 지원을 설치하거나 우회합니다. 기본값은 no 입니다. 자세한 내용은 Horizon Linux 데스크톱의 기능 의 FIPS 140-2 모드 설명을 참조하십시오.
-j	JMS SSL 키 저장소 암호입니다. 기본적으로 설치 관리자는 임의 문자열을 생성합니다.
-m yes no	스마트 카드 리디렉션 지원을 설치하거나 우회합니다. 기본값은 no 입니다.
-r yes no	설치 후에 시스템을 자동으로 다시 시작합니다. 기본값은 no 입니다.
-s	자체 서명된 인증서 주체 DN입니다. 기본적으로 설치 관리자는 Blast 를 사용합니다.
-C yes no	클립보드 리디렉션 지원을 설치하거나 우회합니다. 기본값은 yes 입니다.
-F yes no	CDR 지원을 설치하거나 우회합니다. 기본값은 yes 입니다.
-M yes no	Linux Agent를 관리되는 에이전트나 관리되지 않는 에이전트로 업그레이드합니다. 기본값은 yes 입니다.
-S yes no	SSO(Single Sign-On) 지원을 설치하거나 우회합니다. 기본값은 yes 입니다.
-T yes no	True SSO(Single Sign-On) 지원을 설치하거나 우회합니다. 기본값은 no 입니다.
-U yes no	USB 지원을 설치하거나 우회합니다. 기본값은 no 입니다.

표 5-3. install_viewagent.sh 매개 변수의 예

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

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.

절차

- 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/ssl`을 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 Horizon Connection Server에서 구성을 수정할 필요가 없습니다.
- 관리되지 않는 가상 시스템 배포에서 관리되는 가상 시스템 배포의 임의 버전으로 업그레이드합니다. 이러한 유형의 업그레이드에서는 Horizon Connection Server에 데스크톱 풀을 새로 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 yes 를 지정해야 합니다. 이 매개 변수를 지정하지 않으면 설치 중에 값을 묻는 메시지가 표시됩니다.
-a yes no	오디오 입력 리더렉션 지원을 설치하거나 우회합니다.
-f yes no	FIPS(Federal Information Processing Standards) 140-2용으로 고안된 암호화 모듈의 지원을 설치하거나 우회합니다. 기본값은 no 입니다. 자세한 내용은 Horizon Linux 데스크톱의 기능의 FIPS 140-2 모드 설명 을 참조하십시오.
-m yes no	스마트 카드 리더렉션 지원을 설치하거나 우회합니다. 기본값은 no 입니다.

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

매개 변수	설명
-r yes no	설치 후에 운영 체제를 재부팅합니다. 기본값은 no 입니다.
-C yes no	클립보드 리디렉션 지원을 설치하거나 우회합니다. 기본값은 yes 입니다.
-F yes no	CDR 지원을 설치하거나 우회합니다. 기본값은 yes 입니다.
-M yes no	Linux Agent를 관리되는 관리되지 않는 에이전트로 업그레이드합니다. 기본값은 yes 입니다.
-S yes no	SSO(Single Sign-On) 지원을 설치하거나 우회합니다. 기본값은 yes 입니다.
-U yes no	USB 지원을 설치하거나 우회합니다. 기본값은 no 입니다.

Linux 가상 시스템에서 Horizon Agent 업그레이드

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

사전 요구 사항

- VMwareBlastServer 프로세스를 실행하고 있지 않은지 확인합니다.

이 프로세스를 중지하려면 사용자가 시스템에서 로그오프하고 활성 상태인 데스크톱 세션이 없도록 하거나 시스템을 다시 부팅합니다.

절차

- 1 VMware 다운로드 사이트 <https://my.vmware.com/web/vmware/downloads>에서 Horizon Agent for Linux에 대한 최신 설치 관리자 파일을 다운로드합니다.

[데스크톱 및 최종 사용자 컴퓨팅]에서 Horizon Agent for Linux에 대한 설치 관리자를 포함하는 VMware Horizon 7을 선택하여 다운로드합니다.

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.

- 2 게스트 운영 체제에서 Linux 배포용 tarball의 압축을 풉니다.

예:

```
tar -xzf < "Horizon Agent tar ball" >
```

- 3 tar ball 폴더로 이동합니다.

- 4 관리되지 않는 가상 시스템을 업그레이드하려면 다음 배포 시나리오 중 하나를 사용하여 `install_viewagent.sh` 스크립트를 실행합니다.

옵션	설명
관리되지 않는 가상 시스템 배포 업그레이드 및 관리되지 않는 가상 시스템 배포 유지	<pre>sudo ./install_viewagent.sh -A yes -M no</pre> <p>참고 가능한 최고의 성능을 유지하려면 관리되지 않는 가상 시스템을 사용하지 마십시오.</p>
관리되지 않는 가상 시스템 배포 업그레이드 및 이를 관리되는 가상 시스템 배포로 변경	<pre>sudo ./install_viewagent.sh -A yes -M yes</pre> <p>참고 Horizon Console에서 관리되지 않는 가상 시스템 배포에 대한 기존 데스크톱 풀을 삭제하고 관리되는 가상 시스템 배포에 대한 데스크톱 풀을 생성합니다. 자세한 내용은 Linux용 수동 데스크톱 풀 생성을 참조하십시오.</p>
관리되는 가상 시스템 배포 업그레이드	<pre>sudo ./install_viewagent.sh -A yes -M yes</pre> <p>참고 업그레이드 후에 기존 데스크톱 풀을 재사용할 수 있습니다.</p>

Horizon 7 for Linux 시스템 제거

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

사전 요구 사항

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

절차

- 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.

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

Linux 데스크톱용 구성 옵션

6

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

본 장은 다음 항목을 포함합니다.

- Linux 데스크톱의 구성 파일에서 옵션 설정
- 스마트 정책 사용
- Linux 데스크톱용 Blast 설정 예
- Linux 데스크톱에 대한 클라이언트 드라이브 리더렉션 옵션의 예

Linux 데스크톱의 구성 파일에서 옵션 설정

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

Horizon Agent를 설치하는 동안 설치 관리자가 두 개의 구성 템플릿 파일 `config.template` 및 `viewagent-custom.conf.template`을 /etc/vmware에 복사합니다. 또한, /etc/vmware/config 및 /etc/vmware/viewagent-custom.conf가 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/config에서 다음 줄은 무손실 PNG 모드로 빌드되도록 설정합니다.

```
RemoteDisplay.buildToPNG=TRUE
```

구성을 변경하고 나면 변경이 적용되도록 Linux를 재부팅합니다.

/etc/vmware/config의 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 해석 및 강제 적용을 위해 설정을 Horizon Client에 전달합니다. 이러한 강제 적용은 **merge (m)** 수정자를 지정하여 Horizon Client 필터 정책 설정 외에 Horizon Agent 필터 정책 설정을 적용할지 또는 **(o)** 수정자를 사용하여 Horizon Client 필터 정책 설정 대신 Horizon Agent 필터 정책 설정을 사용할지 여부를 기준으로 합니다.

표 6-1. /etc/vmware/config의 구성 옵션

옵션	값/형식	기본값	설명
Clipboard.Direction	0, 1, 2, 또는 3	2	클립보드 리디렉션 정책을 지정하려면 이 옵션을 사용합니다. 유효한 값은 다음과 같습니다. <ul style="list-style-type: none"> ■ 0 - 클립보드 리디렉션을 사용하지 않도록 설정합니다. ■ 1 - 클립보드 리디렉션을 양방향 모두에서 사용하도록 설정합니다. ■ 2 - 클립보드 리디렉션을 클라이언트에서 원격 데스크톱 방향으로만 사용하도록 설정합니다. ■ 3 - 클립보드 리디렉션을 원격 데스크톱에서 클라이언트 방향으로만 사용하도록 설정합니다.
RemoteDisplay.allowAudio	true 또는 false	true	오디오 출력을 사용/사용하지 않으려면 이 옵션을 설정합니다.
RemoteDisplay.allowH264	true 또는 false	true	이 옵션을 설정하여 H.264 인코딩을 사용하거나 사용하지 않도록 설정합니다.
RemoteDisplay.buildToPNG	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 모드 빌드를 구성할 수 있습니다. 이 기능에서는 클라이언트와 ESXi 호스트 사이에 추가 대역폭을 사용합니다. 이 옵션을 사용하도록 설정하면 H.264 인코딩이 사용되지 않도록 설정됩니다.
RemoteDisplay.enableNetworkContinuity	true 또는 false	true	Horizon Agent for Linux에서 네트워크 연속성 기능을 사용하거나 사용하지 않도록 설정하려면 이 옵션을 설정합니다.
RemoteDisplay.enableNetworkIntelligence	true 또는 false	true	Horizon Agent for Linux에서 네트워크 인텔리전스 기능을 사용하거나 사용하지 않도록 설정하려면 이 옵션을 설정합니다.
RemoteDisplay.enableStats	true 또는 false	false	mks 로그에서 대역폭, FPS, RTT 등과 같은 VMware Blast 디스플레이 프로토콜 통계를 사용하거나 사용하지 않도록 설정합니다.

표 6-1. /etc/vmware/config의 구성 옵션 (계속)

옵션	값/형식	기본값	설명
RemoteDisplay.enableUDP	true 또는 false	true	Horizon Agent for Linux에서 UDP 프로토콜 지원을 사용하거나 사용하지 않도록 설정하려면 이 옵션을 설정합니다.
RemoteDisplay.maxBandwidthKbps	정수	1000000	VMware Blast 세션에 대해 초당 킬로비트(kbps)로 최대 대역폭을 지정합니다. 대역폭은 모든 이미징, 오디오, 가상 채널 및 VMware Blast 제어 트래픽을 포함합니다. 유효한 값은 4Gbps(4096000) 미만이어야 합니다.
RemoteDisplay.minBandwidthKbps	정수	256	VMware Blast 세션에 대해 초당 킬로비트(kbps)로 최소 대역폭을 지정합니다. 대역폭은 모든 이미징, 오디오, 가상 채널 및 VMware Blast 제어 트래픽을 포함합니다.
RemoteDisplay.maxFPS	정수	30	화면 업데이트의 최대 속도를 지정합니다. 사용자가 소비하는 평균 대역폭을 관리하려면 이 설정을 사용합니다. 값은 3에서 60 사이여야 합니다. 기본값은 초당 30 회 업데이트입니다.
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.
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.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.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.
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.
VMWPKcs11Plugin.log.enable	true 또는 false	false	True SSO 기능에 대해 로깅 모드를 사용하거나 사용하지 않도록 설정하려면 이 옵션을 설정합니다.

표 6-1. /etc/vmware/config의 구성 옵션 (계속)

옵션	값/형식	기본값	설명
VMWPKcs11Plugin.log.logLevel	error, warn, info, debug, trace 또는 verbose	info	True SSO 기능의 로그 수준을 설정하려면 이 옵션을 사용합니다.
VVC.RTAV.Enable	true 또는 false	true	오디오 입력을 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 프록시 노드의 로그 수준을 설정하려면 이 옵션을 사용합니다.
cdrserver.cacheEnable	true 또는 false	true	에이전트에서 클라이언트 측으로의 쓰기 캐싱 기능을 사용하거나 사용하지 않도록 설정하려면 이 옵션을 설정합니다.
cdrserver.customizedSharedFolderPath	folder_path	/home/	<p>이 옵션을 사용하여 CDR(클라이언트 드라이브 리디렉션) 공유 폴더 위치를 기본 /home/user/tsclient 디렉토리에서 사용자 지정 디렉토리로 변경합니다.</p> <p>예를 들어, 사용자 test가 CDR 공유 폴더를 /home/test/tsclient 대신 /mnt/test/tsclient에 배치하려는 경우</p> <p>cdrserver.customizedSharedFolderPath=/mnt/를 지정할 수 있습니다.</p> <p>참고 이 옵션을 적용하려면 지정된 폴더가 존재하고 올바른 사용자 사용 권한으로 구성되어야 합니다.</p>
cdrserver.forcedByAdmin	true 또는 false	false	클라이언트가 cdrserver.shareFolders 옵션에 지정되지 않은 추가 폴더를 공유할 수 있는지 여부를 제어하려면 이 옵션을 설정합니다.
cdrserver.logLevel	error, warn, info, debug, trace 또는 verbose	info	vmware-CDRserver.log 파일의 로그 수준을 설정하려면 이 옵션을 사용합니다.

표 6-1. /etc/vmware/config의 구성 옵션 (계속)

옵션	값/형식	기본값	설명
cdserver.permissions	R	RW	<p>Horizon Client가 공유하는 폴더에 대해 Horizon Agent가 갖는 추가 읽기/쓰기 사용 권한을 적용하려면 이 옵션을 사용합니다. 예:</p> <ul style="list-style-type: none"> ■ Horizon Client가 공유하는 폴더에 read 및 write 사용 권한이 있으며 cdserver.permissions=R를 설정했으면 Horizon Agent는 read 액세스 사용 권한만 갖습니다. ■ Horizon Client가 공유하는 폴더에 read 사용 권한만 있으며 cdserver.permissions=RW를 설정했으면 Horizon Agent는 read 액세스 권한을 계속 갖습니다. Horizon Agent에서는 Horizon Client에서 설정한 read 전용 특성을 변경할 수 없습니다. Horizon Agent는 쓰기 액세스 권한만 제거할 수 있습니다. <p>일반적인 사용법은 다음과 같습니다.</p> <ul style="list-style-type: none"> ■ cdserver.permissions=R ■ #cdserver.permissions=R(항목의 주석 처리를 해제하거나 삭제)
cdserver.sharedFolders	<i>file_path1,R; file_path2,; file_path3,R; . .</i>	정의되지 않음	<p>클라이언트가 Linux 데스크톱과 공유할 수 있는 폴더에 대한 하나 이상의 파일 경로를 지정합니다. 예:</p> <ul style="list-style-type: none"> ■ Windows 클라이언트: C:\spreadsheets,;D:\ebooks,R ■ 비 Windows 클라이언트: /tmp/spreadsheets;/tmp/ebooks,;/home/finance,R
collaboration.logLevel	error, info 또는 debug	info	이 옵션을 사용하여 공동 작업 세션에 사용되는 로그 수준을 설정합니다. 로그 수준이 debug이면 collabui 함수에 대한 모든 호출과 collabor 목록 콘텐츠가 기록됩니다.
collaboration.maxCollabors	10보다 작은 정수	5	세션에 가입하도록 초대할 수 있는 공동 작업자의 최대 수를 지정합니다.
collaboration.enableEmail	true 또는 false	true	설치된 e-메일 애플리케이션을 사용하여 공동 작업 초대 보내기를 사용하거나 사용하지 않도록 설정하려면 이 옵션을 설정합니다. 이 옵션을 사용하지 않도록 설정된 경우, e-메일 애플리케이션이 설치되어 있더라도 e-메일을 사용하여 공동 작업자를 초대할 수 없습니다.
collaboration.serverUrl	[URL]	정의되지 않음	공동 작업 초대에 포함할 서버 URL을 지정합니다.
collaboration.enableControlPassing	true 또는 false	true	공동 작업자의 Linux 데스크톱 제어를 허용하거나 제한하려면 이 옵션을 설정합니다. 읽기 전용 공동 작업 세션을 지정하려면 이 옵션을 false 로 설정합니다.

표 6-1. /etc/vmware/config의 구성 옵션 (계속)

옵션	값/형식	기본값	설명
mksVNCServer.useUIInputButton Mapping	true 또는 false	false	Ubuntu 또는 RHEL 7.x에서 왼쪽 마우스를 지원하도록 설정하려면 이 옵션을 설정합니다. CentOS 및 RHEL 6.x는 왼쪽 마우스를 지원하므로 이 옵션을 설정할 필요가 없습니다.
mksvhan.clipboardSize	정수	1024	이 옵션을 사용하여 복사하여 붙여넣을 클립보드 최대 크기를 지정합니다.
vdpservice.log.logLevel	fatal error, warn, info, debug 또는 trace	info	vdpservice의 로그 수준을 설정하려면 이 옵션을 사용합니다.
viewusb.AllowAudioIn	{m o}: {true false}	정의되지 않음, true와 동일	오디오 입력 디바이스가 리디렉션되도록 허용하거나 허용하지 않으려면 이 옵션을 사용합니다. 예: o:false
viewusb.AllowAudioOut	{m o}: {true false}	정의되지 않음, false와 동일	오디오 출력 디바이스의 리디렉션을 허용하거나 허용하지 않으려면 이 옵션을 설정합니다.
viewusb.AllowAutoDeviceSplitting	{m o}: {true false}	정의되지 않음, false와 동일	복합 USB 디바이스의 자동 분할을 허용하거나 허용하지 않으려면 이 옵션을 설정합니다. 예: m:true
viewusb.AllowDevDescFailsafe	{m o}: {true false}	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 IncludeVidPid 또는 IncludePath 와 같은 Include 필터에 포함하십시오.
viewusb.AllowHIDBootable	{m o}: {true false}	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.AllowKeyboardMouse	{m o}: {true false}	정의되지 않음, false와 동일	통합형 포인팅 디바이스(예: 마우스, 트랙볼 또는 터치패드)를 사용하여 키보드의 리디렉션을 허용하거나 허용하지 않으려면 이 옵션을 설정합니다.
viewusb.AllowSmartcard	{m o}: {true false}	정의되지 않음, false와 동일	스마트 카드 디바이스가 리디렉션되도록 허용하거나 허용하지 않으려면 이 옵션을 설정합니다.
viewusb.AllowVideo	{m o}: {true false}	정의되지 않음, true와 동일	비디오 디바이스가 리디렉션되도록 허용하거나 허용하지 않으려면 이 옵션을 사용합니다.
viewusb.DisableRemoteConfig	{m o}: {true false}	정의되지 않음, false와 동일	USB 디바이스 필터링을 수행할 때 Horizon Agent 설정을 사용하거나 사용하지 않도록 설정하려면 이 옵션을 설정합니다.

표 6-1. /etc/vmware/config의 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. true 로 설정되면 기타 정책 설정을 사용하여 특정 디바이스 또는 디바이스 제품군을 리디렉션할 수 있습니다. false 로 설정되면 기타 정책 설정을 사용하여 특정 디바이스 또는 디바이스 제품군이 리디렉션되지 않도록 방지할 수 있습니다. Horizon Agent에서 ExcludeAllDevices 값을 true 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: m:bluetooth;smart-card 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. 참고 기본적으로 마우스 및 키보드는 리디렉션에서 제외되므로 이 설정을 사용하여 제외할 필요가 없습니다.
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: m:bus-1/2/3_port-02;bus-1/1/1/4_port-ff
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 숫자 대신 와일드카드 문자(*)를 사용할 수 있습니다. 예: o:vid-0781_pid-****;vid-0561_pid-554c
viewusb.IncludeFamily	{m o}:family_name_1[;family_name_2]...	정의되지 않음	리디렉션될 수 있는 디바이스 제품군을 포함하려면 이 옵션을 설정합니다. 예: o:storage; smart-card
viewusb.IncludePath	{m o}:bus-x1[/y1].../port-z1[;bus-x2[/y2].../portz2;...]	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: m:bus-1/2_port-02;bus-1/7/1/4_port-0f
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: o:vid-***_pid-0001;vid-0561_pid-554c

표 6-1. /etc/vmware/config의 구성 옵션 (계속)

옵션	값/형식	기본값	설명
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 vid-xxx1_pid-yyy1[;vid-xxx2_pid-yyy2;...] 입니다. ID 번호는 16진수로 지정해야 합니다. ID에서 개별 자릿수 대신 와일드카드 문자(*)를 사용할 수 있습니다. 예: m:vid-0f0f_pid-55**
viewusb.SplitVidPid	{m o}: vid-xxxx_pid-yyyy([exintf:zz[;exintf:ww]])[;...]	정의되지 않음	벤더 및 제품 ID별로 지정된 복합 USB 디바이스의 구성 요소를 개별 디바이스로 처리하려면 이 옵션을 설정합니다. 이 설정의 형식은 vid-xxxx_pid-yyyy(exintf:zz[;exintf:ww]) 입니다. exintf 키워드를 사용하면 인터페이스 번호를 지정하여 구성 요소를 리더렉션에서 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: o:vid-0f0f_pid-***([exintf-01];vid-0781_pid-554c(exintf:01;exintf:02)) 참고 Horizon은 명시적으로 제외하지 않은 구성 요소를 자동으로 포함시키지 않습니다. VidPid 디바이스 포함 과 같은 필터 정책을 지정하여 해당 구성 요소를 포함시켜야 합니다.

/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.conf의 구성 옵션

옵션	값	기본값	설명
CDREnable	true 또는 false	true	CDR(클라이언트 드라이브 리더렉션) 기능을 사용하거나 사용하지 않도록 설정하려면 이 옵션을 사용합니다.
CollaborationEnable	true 또는 false	true	Linux 데스크톱에서 세션 공동 작업 기능을 사용하거나 사용하지 않도록 설정하려면 이 옵션을 사용합니다.
EndpointVPNEnable	true 또는 false	false	Dynamic Environment Manager 콘솔에서 사용되는 끝점 IP 주소 범위에 대해 끝점 IP 주소를 평가할 때 클라이언트의 물리적 네트워크 카드 IP 주소를 사용할지 또는 VPN IP 주소를 사용할지를 지정하려면 이 옵션을 설정합니다. 이 옵션을 false로 설정하는 경우 클라이언트의 물리적 네트워크 카드 IP 주소가 사용됩니다. 그렇지 않은 경우 VPN IP 주소가 사용됩니다.

표 6-2. /etc/vmware/viewagent-custom.conf의 구성 옵션 (계속)

옵션	값	기본값	설명
HelpDeskEnable	true 또는 false	true	헬프 데스크 도구 기능을 사용하거나 사용하지 않도록 설정하려면 이 옵션을 설정합니다.
KeyboardLayoutSync	true 또는 false	true	클라이언트의 시스템 로캘 목록 및 현재 키보드 레이아웃을 Horizon Agent for Linux 데스크톱과 동기화할지 여부를 지정하려면 이 옵션을 사용합니다. 이 설정이 사용되도록 설정되거나 구성되지 않은 경우, 동기화가 허용됩니다. 이 설정이 사용되지 않도록 설정되면 동기화가 허용되지 않습니다. 이 기능은 Windows용 Horizon Client 에서만 지원되고 영어, 프랑스어, 독일어, 일본어, 한국어, 스페인어, 중국어 간체 및 중국어 번체 로캘에서만 지원됩니다.
LogCnt	정수	-1	이 옵션을 사용하여 /tmp/vmware-root에서 예약된 로그 파일 수를 설정합니다. ■ -1 - 모두 유지 ■ 0 - 모두 삭제 ■ > 0 - 예약된 로그 수
NetbiosDomain	모두 대문자로 된 텍스트 스트링		True SSO를 구성할 때 이 옵션을 사용하여 조직 도메인의 NetBIOS 이름을 설정합니다.
OfflineJoinDomain	pbis 또는 samba	pbis	이 옵션을 사용하여 인스턴트 클론 오프라인 도메인 가입을 설정합니다. 오프라인 도메인 가입을 수행하는 데 사용할 수 있는 방법은 PBISO(PowerBroker Identity Services Open) 인증 및 Samba 오프라인 도메인 가입입니다. 이 속성값이 pbis 또는 samba가 아닌 경우 오프라인 도메인 가입은 무시됩니다.
RunOnceScript			복제된 가상 시스템을 Active Directory에 다시 가입하려면 이 옵션을 사용합니다. 호스트 이름이 변경된 후에 RunOnceScript 옵션을 설정합니다. 지정된 스크립트는 첫 번째 호스트 이름이 변경된 후에만 실행됩니다. 스크립트는 에이전트 설치 후에 에이전트 서비스가 시작되고 호스트 이름이 변경되면 루트 사용 권한으로 실행됩니다. 예를 들어 Winbind 솔루션의 경우 winbind를 사용하여 기본 가상 시스템을 Active Directory에 가입시키고 이 옵션을 스크립트 경로로 설정해야 합니다. 스크립트에는 도메인 다시 가입 명령(/usr/bin/net ads join -U <ADUserName> %<ADUserPassword>)이 포함되어야 합니다. VM 복제 후에 운영 체제 사용자 지정에 따라 호스트 이름이 변경됩니다. 에이전트 서비스가 시작되면 복제된 가상 시스템을 Active Directory에 가입시키기 위해 이 스크립트가 실행됩니다.
RunOnceScriptTimeout		120	RunOnceScript 옵션의 시간 초과 값을 초 단위로 설정하려면 이 옵션을 사용합니다. 예를 들어 RunOnceScriptTimeout=120을 설정합니다.

표 6-2. /etc/vmware/viewagent-custom.conf의 구성 옵션 (계속)

옵션	값	기본값	설명
SSLCiphers	텍스트 문 자열	!aNULL:kECDH +AESGCM:ECDH +AESGCM:RSA +AESGCM:kECDH +AES:ECDH+AES:RSA +AES	암호 목록을 지정하려면 이 옵션을 사용합니다. https://www.openssl.org/docs/manmaster/man1/ciphers.html 에 정의된 형식을 사용해야 합니다.
SSLProtocols	텍스트 문 자열	TLSv1_1:TLSv1_2	보안 프로토콜을 지정하려면 이 옵션을 사용합니다. 지원되는 프로토콜은 TLSv1.0, TLSv1.1 및 TLSv1.2입니다.
SSODesktopType	UseGnomeClassic , UseGnomeFlashback , UseGnomeUbuntu , UseMATE 또는 UseKdePlasma	N/A	SSO가 사용되도록 설정되면 이 옵션은 기본 데스크톱 환경 대신 사용할 데스크톱 환경을 지정합니다. 데스크톱 환경을 사용하도록 지정하기 전에 선택한 데스크톱 환경이 데스크톱에 설치되어 있는지 확인해야 합니다. Ubuntu 16.04/18.04 데스크톱에서 이 옵션을 설정하면 SSO 기능이 사용되도록 설정되었는지에 관계없이 이 옵션이 적용됩니다. RHEL/CentOS 7.x 데스크톱에서 이 옵션을 지정한 경우 SSO가 사용되도록 설정된 경우에만 선택한 데스크톱 환경이 사용됩니다. 참고 이 옵션은 RHEL/CentOS 8.x 및 RHEL/CentOS 6.x 데스크톱에서 지원되지 않습니다. Horizon 7은 RHEL/CentOS 8.x 데스크톱에서 Gnome 데스크톱 환경만 지원합니다. RHEL/CentOS 6.x 데스크톱에서 SSO가 사용되도록 설정된 경우 KDE를 기본 데스크톱 환경으로 설정하는 방법에 대한 자세한 내용은 데스크톱 환경 의 내용을 참조하십시오.
SSOEnable	true 또는 false	true	SSO(Single Sign On)를 사용하거나 사용하지 않도록 설정하려면 이 옵션을 설정합니다.
SSOUserFormat	텍스트 문 자열	[username]	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 예는 다음과 같습니다. ■ SSOUserFormat=[domain]\\[username] ■ SSOUserFormat=[domain]+[username] ■ SSOUserFormat=[username]@[domain]
서브넷	CIDR IP 주소 형식 값	[subnet]	다른 시스템이 Horizon Agent for Linux에 연결하는 데 사용할 수 있는 서브넷에 이 옵션을 설정합니다. 서브넷이 서로 다른 둘 이상의 로컬 IP 주소가 있는 경우 구성된 서브넷의 로컬 IP 주소가 Horizon Agent for Linux에 연결하는 데 사용됩니다. CIDR IP 주소 형식으로 값을 지정해야 합니다. 예: Subnet=123.456.7.8/24.

표 6-2. /etc/vmware/viewagent-custom.conf의 구성 옵션 (계속)

옵션	값	기본값	설명
UEMEnable	true 또는 false	false	Dynamic Environment Manager 스마트 정책을 사용하거나 사용하지 않도록 설정하려면 이 옵션을 설정합니다. 이 옵션을 사용하도록 설정하고 Dynamic Environment Manager 스마트 정책에 대한 조건이 충족되면 정책이 적용됩니다.
UEMNetworkPath	텍스트 문자열		이 옵션을 Dynamic Environment Manager 콘솔에서 설정된 동일한 네트워크 경로로 설정해야 합니다. 경로는 //10.111.22.333/view/LinuxAgent/UEMConfig와 유사한 형식이어야 합니다.

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.

스마트 정책 사용

스마트 정책을 사용하여 특정 원격 Linux 데스크톱에서 USB 리디렉션, 클립보드 리디렉션 및 클라이언트 드라이브 리디렉션 기능의 동작을 제어하는 정책을 생성할 수 있습니다.

동작 범위를 제어하는 사용자 환경 설정에 대한 정책을 생성할 수 있습니다. 사용자 환경 설정에 대한 Horizon 스마트 정책은 로그인 중에 적용되며 세션을 다시 연결하는 동안 새로 고쳐질 수 있습니다. 사용자가 세션에 다시 연결할 때 Horizon 스마트 정책을 다시 적용하려면 트리거된 작업을 구성할 수 있습니다.

최종 사용자의 컴퓨터가 부팅되는 동안 Dynamic Environment Manager가 적용하는 컴퓨터 환경 설정에 대한 정책을 생성할 수 있습니다. 컴퓨터 환경 설정에 대한 Horizon 스마트 정책은 컴퓨터 부팅 중에 적용되며 세션을 다시 연결하는 동안 새로 고쳐질 수 있습니다.

스마트 정책을 사용하면 특정 조건이 충족된 경우에만 적용되는 정책을 만들 수 있습니다. 예를 들어, 사용자가 회사 네트워크 외부에서 원격 데스크톱에 연결한 경우 클라이언트 드라이브 리디렉션 기능을 사용하지 않도록 설정하는 정책을 구성할 수 있습니다.

스마트 정책 요구 사항

스마트 정책을 사용하려면 Horizon 7 환경이 특정한 요구 사항을 충족해야 합니다.

- 스마트 정책으로 관리할 원격 데스크톱에 Horizon Agent 7.5 이상 및 VMware Dynamic Environment Manager 9.4 이상을 설치해야 합니다.
- 사용자는 Horizon Client 4.8 이상을 사용하여 스마트 정책으로 관리하는 원격 Linux 데스크톱에 연결해야 합니다.

- DEMEnable 옵션을 사용하도록 설정하고 `/etc/vmware/viewagent-custom.conf` 파일에서 DEMNetworkPath 옵션을 설정해야 합니다. [Linux 데스크톱의 구성 파일에서 옵션 설정](#)의 내용을 참조하십시오.
- 네트워크 공유 스토리지에 액세스하려면 클라이언트 패키지를 설치해야 합니다. 예를 들어, Ubuntu 18.04 시스템에서 NFS 지원 공유 스토리지에 대해서는 `nfs-common` 패키지를 설치하고, Samba 지원 스토리지에 대해서는 `cifs-utils` 패키지를 설치합니다.

Dynamic Environment Manager 설치

Horizon스마트 정책을 사용하여 원격 Linux 데스크톱에서 원격 데스크톱 기능의 동작을 제어하려면 원격 Windows 데스크톱에 Dynamic Environment Manager 9.4 이상을 설치해야 합니다.

VMware 다운로드 페이지에서 Dynamic Environment Manager 설치 관리자를 다운로드할 수 있습니다. Dynamic Environment Manager 환경을 관리할 Windows 데스크톱에 임의의 Dynamic Environment Manager 관리 콘솔 구성 요소를 설치할 수 있습니다. Windows 데스크톱의 Dynamic Environment Manager 관리 콘솔에서 원격 Linux 데스크톱의 원격 데스크톱 기능 동작을 제어할 수 있습니다.

RDS 데스크톱 풀의 경우 게시된 데스크톱 세션을 제공하는 RDS 호스트에 Dynamic Environment Manager를 설치합니다.

Dynamic Environment Manager 시스템 요구 사항과 전체 설치 지침은 "VMware Dynamic Environment Manager 설치 및 구성" 문서를 참조하십시오.

Dynamic Environment Manager 구성

원격 데스크톱 기능에 대해 스마트 정책을 만들려면 먼저 Dynamic Environment Manager를 구성해야 합니다.

Dynamic Environment Manager를 구성하려면 "VMware Dynamic Environment Manager 관리 가이드"에 있는 구성 지침을 따릅니다.

Horizon 스마트 정책 설정

Horizon 스마트 정책을 생성하면 Dynamic Environment Manager에서 원격 기능의 동작을 제어할 수 있습니다.

동작 범위를 제어하는 사용자 환경 설정에 대한 정책을 생성할 수 있습니다. 사용자 환경 설정에 대한 Horizon 스마트 정책은 로그인 중에 적용되며 세션을 다시 연결하는 동안 새로 고쳐질 수 있습니다. 사용자가 세션에 다시 연결할 때 Horizon 스마트 정책을 다시 적용하려면 트리거된 작업을 구성할 수 있습니다.

"VMware Dynamic Environment Manager 관리 가이드"의 "사용자 환경 설정에 대한 Horizon 스마트 정책 구성" 항목에서 전체 정책 목록을 참조하십시오.

최종 사용자의 컴퓨터가 부팅되는 동안 Dynamic Environment Manager가 적용하는 컴퓨터 환경 설정에 대한 정책을 생성할 수 있습니다. 컴퓨터 환경 설정에 대한 Horizon 스마트 정책은 컴퓨터 부팅 중에 적용되며 세션을 다시 연결하는 동안 새로 고쳐질 수 있습니다. "VMware Dynamic Environment Manager 관리 가이드"의 "컴퓨터 환경 설정에 대한 Horizon 스마트 정책 구성" 항목에서 전체 정책 목록을 참조하십시오.

일반적으로 **Dynamic Environment Manager**에서 원격 기능에 대해 구성된 **Horizon** 스마트 정책 설정이 이와 동등한 레지스트리 키 및 그룹 정책 설정을 재정의합니다.

Horizon 스마트 정책 정의에 조건 추가

Dynamic Environment Manager에서 **Horizon** 스마트 정책을 정의할 때 정책을 적용하기 위해 충족해야 할 조건을 추가할 수 있습니다. 예를 들어, 사용자가 회사 네트워크 외부에서 원격 데스크톱에 연결한 경우에만 클라이언트 드라이브 리디렉션 기능을 사용하지 않도록 설정하는 조건을 추가할 수 있습니다.

중요 지원되는 정책 설정이 원격 **Linux** 데스크톱에서 적용되려면 다음 조건을 **Horizon** 스마트 정책 정의에 추가해야 합니다. 현재 지원되는 조건만 해당됩니다. 다른 조건이 설정된 경우 조건 평가의 최종 결과는 **false**입니다.

표 6-3. 원격 **Linux** 데스크톱에 대한 필수 조건

조건	설명
Operating System Architecture	운영 체제 아키텍처를 확인합니다. 이 값은 Linux 로 설정되어야 합니다.
Endpoint IP address	끝점 IP 주소가 지정된 범위에 있는지 여부를 확인합니다. 범위 시작 부분의 빈 필드는 0 으로 해석되고 끝 부분의 빈 필드는 255 로 해석됩니다.

그러나 다음 예에 표시된 것처럼 여러 **Endpoint IP address** 조건을 설정할 수 있습니다.

```
Operating system is Linux
AND Endpoint IP address is in range 11.22.33.44 - 11.22.33.54
OR Endpoint IP address is in range 11.22.33.66 - 11.22.33.77
```

Dynamic Environment Manager 관리 콘솔에서의 조건 추가 및 편집에 대한 자세한 내용은 "**VMware Dynamic Environment Manager 관리 가이드**"를 참조하십시오.

Dynamic Environment Manager에 Horizon 스마트 정책 생성

Dynamic Environment Manager 관리 콘솔을 사용하여 **Dynamic Environment Manager**에 **Horizon** 스마트 정책을 생성합니다. **Horizon** 스마트 정책을 정의할 때, 스마트 정책을 적용하기 위해 충족해야 할 조건을 추가할 수 있습니다.

사전 요구 사항

- **Dynamic Environment Manager**를 설치 및 구성합니다. 자세한 내용은 [Dynamic Environment Manager 설치](#) 및 [Dynamic Environment Manager 구성](#)의 내용을 참조하십시오.
- **Horizon** 스마트 정책 정의에 추가할 수 있는 조건을 익힙니다. [Horizon 스마트 정책 정의에 조건 추가](#)의 내용을 참조하십시오.

- DEMEnable 옵션을 사용하도록 설정하고 `/etc/vmware/viewagent-custom.conf` 파일에서 DEMNetworkPath 옵션을 구성합니다. [Linux 데스크톱의 구성 파일에서 옵션 설정](#)의 내용을 참조하십시오.

참고 지연 시간이 긴 네트워크에서는 업데이트되거나 새로운 스마트 정책을 저장한 후, 최종 사용자에게 영향을 받는 데스크톱에 연결하도록 알리기 전에 **Dynamic Environment Manager**에서 변경 사항 처리를 완료하도록 1분 이상 허용합니다.

동작 범위를 제어하는 사용자 환경 설정에 대한 정책을 생성할 수 있습니다. 사용자 환경 설정에 대한 **Horizon** 스마트 정책은 로그인 중에 적용되며 세션을 다시 연결하는 동안 새로 고쳐질 수 있습니다. 사용자가 세션에 다시 연결할 때 **Horizon** 스마트 정책을 다시 적용하려면 트리거된 작업을 구성할 수 있습니다.

최종 사용자의 컴퓨터가 부팅되는 동안 **Dynamic Environment Manager**가 적용하는 컴퓨터 환경 설정에 대한 정책을 생성할 수 있습니다. 컴퓨터 환경 설정에 대한 **Horizon** 스마트 정책은 컴퓨터 부팅 중에 적용되며 세션을 다시 연결하는 동안 새로 고쳐질 수 있습니다.

Dynamic Environment Manager 관리 콘솔 사용에 대한 완전한 정보를 보려면 "VMware Dynamic Environment Manager 관리 가이드" 문서를 참조하십시오.

절차

- 1 **Dynamic Environment Manager** 관리 콘솔에서 **사용자 환경**을 선택하여 사용자 환경 설정에 대한 정책을 생성하거나 **컴퓨터 환경** 탭을 사용하여 컴퓨터 환경 설정에 대한 정책을 생성합니다.
기존 **Horizon** 스마트 정책 정의는 **Horizon** 스마트 정책 창에 나타납니다(있는 경우).
- 2 **Horizon 스마트 정책**을 선택하고 **생성**을 클릭하여 새 스마트 정책을 생성합니다.
- 3 **설정** 탭을 선택하고 스마트 정책 설정을 정의합니다.
 - a [일반 설정] 섹션에서 **이름** 텍스트 상자에 스마트 정책의 이름을 입력합니다.
예를 들어, 스마트 정책이 클라이언트 드라이브 리디렉션 기능에 영향을 준다면 스마트 정책 이름을 **CDR**로 지정할 수 있습니다.
 - b **Horizon** 스마트 정책 설정 섹션에서 스마트 정책에 포함할 원격 데스크톱 기능 및 설정을 선택합니다.
여러 개의 원격 데스크톱 기능을 선택할 수 있습니다.
- 4 원격 **Linux** 데스크톱에 새 스마트 정책을 사용하는 데 필요한 조건을 추가합니다.
 - a **조건** 탭을 선택하고 **추가**를 클릭한 후 **운영 체제 아키텍처** 조건을 선택합니다.
 - b 값을 **Linux**로 설정합니다.

Operating System is Linux

- c **추가**를 클릭하고 **끝점 IP 주소** 조건을 선택합니다.

기본적으로 **AND** 연산자가 추가됩니다.

- d [끝점 IP 주소] 대화 상자에서 끝점 IP 주소 범위를 설정하고 **확인**을 클릭합니다.

다음은 조건문의 예시입니다.

```
Operating System is Linux
AND Endpoint IP address is in range 11.22.33.44 - 11.22.33.54
```

- 5 **저장**을 클릭하여 스마트 정책을 저장합니다.

결과

Dynamic Environment Manager에서는 사용자가 원격 데스크톱에 연결하거나 다시 연결할 때마다 Horizon 스마트 정책을 처리합니다.

Dynamic Environment Manager에서는 스마트 정책 이름에 따라 여러 스마트 정책을 알파벳 순서로 처리합니다. Horizon 스마트 정책은 Horizon 스마트 정책 창에 알파벳 순으로 표시됩니다. 스마트 정책이 충돌하는 경우에는 마지막으로 처리한 스마트 정책의 우선 순위가 높습니다. 예를 들어 Sue라는 사용자가 USB 리디렉션을 사용할 수 있도록 설정하는 Sue라는 스마트 정책이 있고 Ubuntu1604라는 이름의 데스크톱 풀에 대해 USB 리디렉션을 사용하지 않도록 설정하는 Pool이라는 스마트 정책이 있는 경우, Ubuntu1604 데스크톱 풀에 있는 원격 데스크톱에 Sue가 연결하면 USB 리디렉션 기능이 사용되도록 설정됩니다.

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/config를 사용합니다.

표 6-4. /etc/vmware/config의 Blast 구성 옵션 예

옵션 이름	매개 변수	고속 LAN	LAN	전용 WAN	광대역 WAN	저속 WAN	초저속
대역폭 설정	RemoteDisplay.maxBandwidthKbps	1000000(1 Gbps)	1000000(1 Gbps)	1000000(1 Gbps)	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

표 6-4. /etc/vmware/config의 Blast 구성 옵션 예 (계속)

옵션 이름	매개 변수	고속 LAN	LAN	전용 WAN	광대역 WAN	저속 WAN	초저속
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:\ebooks 및 C:\spreadsheets 폴더를 공유하고, 두 폴더 모두를 읽기 전용으로 만들고, 클라이언트가 추가 폴더를 공유하지 못하도록 합니다.

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

Linux 데스크톱 풀 생성 및 관리

7

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
- 인스턴트 클론 플로팅 데스크톱 풀

vCenter 가상 시스템으로 수동 데스크톱 풀을 생성하려면 모든 가상 시스템에 Horizon Agent를 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](#)를 참조하십시오.

자동화된 전체 클론 데스크톱 풀을 생성하려면 Linux 가상 시스템 템플릿에 Horizon 7 에이전트를 설치해야 합니다. 그런 다음 연결 서버 데스크톱 풀 생성 마법사를 사용하여 전체 가상 시스템을 복제합니다.

인스턴트 클론 플로팅 데스크톱 풀을 생성하려면 PBIS Open 환경이 설정된 Linux 가상 시스템에 Horizon 7 Agent를 설치하고 템플릿을 생성해야 합니다. 그런 다음 연결 서버 데스크톱 풀 생성 마법사를 사용하여 인스턴트 클론 플로팅 데스크톱 풀을 생성합니다.

본 장은 다음 항목을 포함합니다.

- [Linux용 수동 데스크톱 풀 생성](#)
- [Linux 데스크톱 풀 관리](#)
- [Linux용 자동화된 전체 클론 데스크톱 풀 생성](#)
- [Linux용 인스턴트 클론 플로팅 데스크톱 풀 생성](#)
- [브로커 PowerCLI 명령](#)

Linux용 수동 데스크톱 풀 생성

Linux 가상 시스템용으로 수동 데스크톱 풀을 생성할 수 있습니다.

다음 절차에서는 Linux 기반 수동 데스크톱 풀에 대한 필수 설정을 구성하기 위한 지침을 제공합니다. 수동 데스크톱 풀 생성에 대한 자세한 내용은 "Horizon Console에서 가상 데스크톱 설정" 항목을 참조하십시오.

사전 요구 사항

- Linux 게스트 운영 체제에 Horizon Agent가 설치되어 있는지 확인합니다. [Linux 가상 시스템에 Horizon Agent 설치](#)의 내용을 참조하십시오.
- VMware vCenter Server가 Horizon Connection Server에 추가되었는지 확인합니다.

절차

- 1 Horizon Console에서 수동 데스크톱 풀을 추가합니다.

인벤토리 > 데스크톱 > 추가를 선택합니다.

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

- 2 수동 데스크톱 풀을 선택합니다.
- 3 vCenter Server에서 관리 또는 관리되지 않는 가상 시스템을 선택하고 **다음**을 클릭합니다.
- 4 데스크톱 풀의 시스템에 대해 전용 또는 부동 사용자 할당을 선택하고 **다음**을 클릭합니다.
- 5 풀을 생성하려면 마법사의 메시지를 따르십시오.

[데스크톱 풀 설정] 페이지에서 다음 옵션을 설정합니다.

옵션	설명
기본 디스플레이 프로토콜	VMware Blast
사용자가 프로토콜을 선택할 수 있도록 허용함	아니요
3D 렌더러	2D 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.

- 6 데스크톱 풀을 생성한 후에 사용자에게 데스크톱 풀의 시스템에 대한 사용 권한을 부여합니다.
Horizon Console에서 데스크톱 풀을 선택하고 **사용 권한 > 권한 추가**를 선택하고 사용자나 그룹을 추가합니다.

결과

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

Linux 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

Horizon 7에서 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"> ■ 전원 작업 수행 안 함 ■ 시스템 전원이 항상 켜져 있는지 확인 ■ 일시 중단 ■ 전원 끄기
연결 해제 후 자동 로그오프	<ul style="list-style-type: none"> ■ 즉시 ■ 안 함 ■ n분 후
사용자가 시스템을 재설정/다시 시작할 수 있도록 허용	<ul style="list-style-type: none"> ■ 예 ■ 아니요
사용자가 여러 클라이언트 디바이스에서 별도의 세션을 초기화할 수 있도록 허용	<ul style="list-style-type: none"> ■ 예 ■ 아니요
전체 클론 및 부동을 사용하는 자동화된 데스크톱 풀에서 [로그오프 후 시스템 삭제]	<ul style="list-style-type: none"> ■ 예 ■ 아니요

Horizon Console 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 Horizon PowerCLI Cmdlet을 사용해야 합니다.

- Update-UserOwnership

■ Remove-UserOwnership

참고 원격 디스플레이 프로토콜 설정은 변경하지 마십시오. 이러한 설정은 데스크톱 풀 생성 중에 지정된 것과 동일하게 유지되어야 합니다.

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

자세한 내용은 "VMware Horizon Console 관리" 설명서를 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.

다음 절차에서는 Linux 기반 자동 전체 클론 데스크톱 풀에 대한 필수 설정을 구성하기 위한 지침을 제공합니다. 자동 전체 클론 데스크톱 풀 생성에 대한 자세한 내용은 "Horizon Console에서 가상 데스크톱 설정" 항목을 참조하십시오.

사전 요구 사항

- Linux 게스트 운영 체제에 Horizon Agent가 설치되어 있는지 확인합니다. [Linux 가상 시스템에 Horizon Agent 설치](#)의 내용을 참조하십시오.
- 가상 시스템 복제를 수행하기 전에 복제의 기반이 되는 가상 시스템 템플릿을 생성합니다. [Linux 데스크톱 시스템의 복제를 위한 가상 시스템 템플릿 만들기](#)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](#)의 내용을 참조하십시오.
- vCenter Server가 Horizon 연결 서버에 추가되었는지 확인합니다.

절차

- 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	Horizon 7 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 Console에서 자동화된 데스크톱 풀을 추가합니다.
인벤토리 > 데스크톱 > 추가를 선택합니다.
- 3 자동화된 데스크톱 풀을 선택하고 다음을 클릭합니다.
- 4 전체 가상 시스템을 선택하고 vCenter Server 인스턴스를 선택한 후 다음을 클릭합니다.

5 풀을 생성하려면 마법사의 메시지를 따르십시오.

- a [데스크톱 풀 설정] 페이지에서 다음 옵션을 설정합니다.

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

- b 메시지가 표시되면 **가상 시스템 이름 지정** 옵션을 설정합니다.

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

- c 메시지가 표시되면 **vCenter Server** 설정을 순서대로 선택합니다.

vCenter Server 설정은 건너뛸 수 없습니다.

- 1 템플릿
 - 2 VM 폴더 위치
 - 3 호스트 또는 클러스터
 - 4 리소스 풀
 - 5 데이터스토어
- 6 데스크톱 풀을 생성한 후에 사용자에게 데스크톱 풀의 시스템에 대한 사용 권한을 부여합니다.
Horizon Console에서 데스크톱 풀을 선택하고 **사용 권한 > 권한 추가**를 선택하고 사용자나 그룹을 추가합니다.
- 7 데스크톱 풀에 있는 모든 Linux 가상 시스템을 사용할 수 있게 될 때까지 기다립니다.

Linux용 인스턴트 클론 플로팅 데스크톱 풀 생성

데스크톱 풀 추가 마법사를 사용하여 Linux 가상 시스템용 인스턴트 클론 플로팅 데스크톱 풀을 생성할 수 있습니다. 인스턴트 클론 플로팅 데스크톱 풀을 생성한 후 Horizon 7 배포에서 Linux 가상 시스템을 원격 데스크톱으로 사용할 수 있습니다.

Linux용 Horizon 7 Agent는 Ubuntu 18.04/16.04, RHEL 7.1 이상, RHEL 8.x 또는 SLED/SLES 12.x가 있는 시스템에서만 인스턴트 클론 데스크톱 풀을 지원합니다.

참고 vGPU 그래픽 기능은 Linux 데스크톱에서 생성된 인스턴트 클론 데스크톱 풀에서 지원되지 않습니다.

다음 절차에서는 Linux 기반 인스턴트 클론 데스크톱 풀에 대한 필수 설정을 구성하기 위한 지침을 제공합니다. 인스턴트 클론 데스크톱 풀 생성에 대한 자세한 내용은 "Horizon Console에서 가상 데스크톱 설정" 항목을 참조하십시오.

사전 요구 사항

- vCenter Server에서 가상 시스템을 생성하고 Linux 운영 체제를 설치하는 단계를 숙지하십시오. 자세한 내용은 [가상 시스템 생성 및 Linux 설치](#)의 내용을 참조하십시오.
- PBISO 인증 솔루션 또는 Samba Winbind 오프라인 가입을 사용하여 AD 통합 단계를 이해하십시오. 자세한 내용은 [PBISO\(PowerBroker Identity Services Open\) 인증 구성](#) 또는 [Samba 오프라인 도메인 가입 구성](#) 항목을 참조하십시오.

참고 RHEL 8.x를 실행하는 Linux 가상 시스템에서 인스턴트 클론 데스크톱 풀을 생성하려면 Samba Winbind 오프라인 가입을 사용하여 AD 통합을 수행하십시오. PBISO 인증을 사용하는 RHEL 8.x 가상 시스템에서는 인스턴트 클론 데스크톱 풀이 지원되지 않습니다.

- Linux용 Horizon 7 Agent에 대한 설치 단계를 숙지하십시오. 자세한 내용은 [Linux 가상 시스템에 Horizon Agent 설치](#)의 내용을 참조하십시오.
- VMware vSphere Web Client를 사용하여 전원이 꺼진 Linux VM의 스냅샷을 생성하는 단계를 이해하십시오. "vSphere 단일 호스트 관리 - VMware Host Client"에서 "VMware Host Client에서 스냅샷 생성"을 참조하십시오.
- vCenter Server가 Horizon 연결 서버에 추가되었는지 확인합니다.

절차

- 1 Ubuntu 18.04/16.04, RHEL 7.1 이상, RHEL 8.x 또는 SLED/SLES 12.x가 설치된 Linux VM(가상 시스템)을 생성합니다.

자세한 내용은 [가상 시스템 생성 및 Linux 설치](#)의 내용을 참조하십시오.

- 2 다음 명령을 사용하여 Ubuntu 18.04/16.04 시스템에 OVT(Open VMware Tools)를 수동으로 설치하십시오.

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

자세한 내용은 [원격 데스크톱 배포를 위한 Linux 시스템 준비](#)를 참조하십시오.

- 3 Linux 배포에 필요한 모든 종속성 패키지를 설치합니다.

자세한 내용은 [Horizon Agent에 대한 종속성 패키지 설치](#)의 내용을 참조하십시오.

4 Linux VM에서 Linux용 Horizon Agent를 설치합니다.

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

자세한 내용은 [Linux 가상 시스템에 Horizon Agent 설치](#)의 내용을 참조하십시오.

5 Linux VM을 Active Directory에 통합합니다.

- PBISO 인증 솔루션을 사용하려면 다음 단계를 수행합니다.

- a <https://www.beyondtrust.com/products/powerbroker-identity-services-open/>에서 PBIS Open 8.5.6 이상을 다운로드한 후 Linux VM에 설치합니다.

```
# sudo ./pbis-open-8.5.6.2029.linux.x86_64.deb.sh
```

- b [Linux와 Active Directory 통합](#)의 PBISO(PowerBroker Identity Services Open) 인증 섹션의 정보를 사용하여 Linux VM을 Active Directory에 통합합니다.

- Samba Winbind 오프라인 가입을 사용하려면 OfflineJoinDomain을 `/etc/vmware/viewagent-custom.conf` 파일의 **samba**로 설정하십시오.

참고 RHEL 8.x VM을 Active Directory와 통합하려면 Samba Winbind를 사용해야 합니다. 그렇지 않으면 인스턴트 클론 부동 데스크톱 풀의 생성이 실패합니다.

- 오프라인 도메인 가입을 사용하지 않도록 설정하려면 `/etc/vmware/viewagent-custom.conf` 파일에서 OfflineJoinDomain 옵션을 **없음**으로 설정해야 합니다. 그렇지 않으면 인스턴트 클론 부동 데스크톱 풀의 생성이 실패합니다.

6 DHCP 서버가 DNS 서버에 브로드캐스트하지 않는 경우 Linux 시스템에 대한 DNS 서버를 지정합니다.

새 인스턴트 클론 VM이 생성될 때 새 가상 네트워크 어댑터가 추가됩니다. 인스턴트 클론 VM에 새 네트워크 어댑터를 추가되면 VM 템플릿에서 DNS 서버와 같은 네트워크 어댑터의 모든 설정이 손실됩니다. PBIS에는 올바른 DNS 서버가 필요하고 `/etc/hosts`의 FQDN 매핑은 허용되지 않습니다. 복제된 VM에 새 네트워크 어댑터를 추가할 때 DNS 서버 설정이 손실되지 않도록 하려면 Linux 시스템에서 DNS 서버를 지정해야 합니다. 예를 들어 Ubuntu 16.04 시스템에서 `/etc/resolvconf/resolv.conf.d/head` 파일에 다음 줄을 추가하여 DNS 서버를 지정합니다.

```
nameserver 10.10.10.10
search mydomain.org
```

7 (선택 사항) 마스터 Linux VDI 인스턴트 클론 에이전트의 `/etc/fstab` 파일에 NFS 마운트를 추가하려는 경우 다음 방법 중 하나를 사용합니다.

- 다음과 같이 `/etc/fstab`에 'soft' 플래그를 추가합니다.

```
10.111.222.333:/share /home/nfsmount nfs
size=8192,ws=8192,timeo=14,soft,intr,tcp
```

- `/etc/fstab`에 'soft' 플래그를 사용하지 않으면 마스터 Linux VM 이미지에 `/etc/fstab`를 구성할 수 없습니다. 전원 끄기 스크립트를 작성하여 `/etc/fstab` 파일을 구성한 다음, ClonePrep 도구에 대해 이 전원 끄기 스크립트를 지정할 수 있습니다. 자세한 내용은 "VMware Horizon Console 관리" 문서를 참조하십시오.

- 8 Linux VM을 종료하고, VMware vSphere® Web Client를 사용하여 전원이 꺼진 Linux VM의 스냅샷을 생성하는 방식으로 마스터 이미지를 생성하십시오.

자세한 내용은 "vSphere 단일 호스트 관리 - VMware Host Client" 에서 "VMware Host Client에서 스냅샷 생성"을 참조하십시오.

- 9 Horizon Console에서 자동화된 데스크톱 풀을 추가합니다.

인벤토리 > 데스크톱 > 추가를 선택합니다.

- 10 자동화된 데스크톱 풀을 선택하고 다음을 클릭합니다.

- 11 인스턴트 클론을 선택하고 vCenter Server 인스턴스를 선택한 후 다음을 클릭합니다.

- 12 풀을 생성하려면 마법사의 메시지를 따르십시오.

- a 메시지가 표시되면 **가상 시스템 이름 지정** 옵션을 설정합니다.

옵션	설명
프로비저닝 사용	이 옵션을 선택합니다.
오류 시 프로비저닝 중지	이 옵션을 선택합니다.
이름 지정 패턴	Horizon 7에서 모든 데스크톱 VM 이름에서 접두사로 사용하는 패턴과 고유한 번호를 지정합니다. 예를 들어 LinuxVM-{n} 을 지정합니다.
최대 시스템 수	풀에 있는 전체 시스템 수를 지정합니다.
예비(전원 켜짐) 시스템 수	사용자가 사용할 수 있는 데스크톱 VM의 수를 지정합니다.
모든 시스템을 미리 프로비저닝	Horizon 7에서 최대 시스템 수 에 지정된 VM 수를 프로비저닝하려면 이 옵션을 선택합니다.

- b 메시지가 표시되면 스토리지 관리 정책에 대해 **VMware Virtual SAN 사용**을 선택합니다.
- c 메시지가 표시되면 도메인 설정 및 AD 컨테이너를 지정하고, VM을 복제한 후 실행해야 하는 모든 추가 사용자 지정 스크립트를 지정합니다.

중요 ClonePrep 전원 끄기 또는 사후 동기화 스크립트를 사용하는 경우 스크립트가 루트 사용자가 소유하는 `/var/userScript` 폴더에 있는지 확인하고 파일 사용 권한을 700으로 설정합니다.

결과

Horizon Console에서 **인벤토리 > 데스크톱**을 선택하면 데스크톱 VM이 풀에 추가되는 것을 확인할 수 있습니다.

풀을 생성한 후 풀이 존재하는 경우 마스터 이미지를 삭제하거나 vCenter Server 인벤토리에서 제거하지 마십시오. vCenter Server의 인벤토리에서 마스터 이미지 VM을 실수로 제거한 경우에는 이를 다시 추가한 다음 현재 이미지를 사용하여 푸시 이미지를 수행해야 합니다.

다음에 수행할 작업

풀에 액세스하려면 사용자에게 권한을 부여하십시오. "Horizon Console에서 가상 데스크톱 설정"에서 "데스크톱 풀에 사용 권한 추가"를 참조하십시오.

브로커 PowerCLI 명령

연결 서버 및 Windows 데스크톱에서 다양한 관리 작업을 수행하기 위한 Horizon PowerCLI cmdlet은 Linux 데스크톱에서도 사용됩니다.

수동 데스크톱 풀 생성

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

다음 옵션 및 값은 Linux 데스크톱에 대해 필수입니다.

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

예제

- VM(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 -threedRender 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]
```

다음 옵션 및 값은 Linux 데스크톱에 대해 필수입니다.

- DefaultProtocol Blast
- AllowProtocolOverride \$false
- threedRender usevc|vgpu vGPU 데스크톱의 경우 -threedRender vgpu를 사용하고 2D 데스크톱의 경우 -threedRender usevc를 사용합니다.

예]

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

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 VM에서 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

- **myuser**의 데스크톱 세션에서 로그아웃합니다.

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

브로커 PowerCLI cmdlet에 대한 자세한 내용은 "Horizon 7 통합" 에서 "Horizon PowerCLI 모듈 사용"을 참조하십시오.

수동 데스크톱 풀의 Horizon 7 대량 배포

8

Horizon Console를 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.

본 장은 다음 항목을 포함합니다.

- Linux 데스크톱의 대량 배포 개요
- Linux 데스크톱의 대량 업그레이드 개요
- Linux 데스크톱 시스템의 복제를 위한 가상 시스템 템플릿 만들기
- Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일
- Linux 가상 시스템을 복제하기 위한 샘플 스크립트
- 복제된 가상 시스템을 AD 도메인에 연결하기 위한 샘플 스크립트
- SSH를 사용하여 복제된 가상 시스템을 AD 도메인에 연결하기 위한 샘플 스크립트
- 구성 파일을 Linux 가상 시스템에 업로드하기 위한 샘플 스크립트
- SSH를 사용하여 구성 파일을 Linux 가상 시스템에 업로드하기 위한 샘플 스크립트
- Linux 데스크톱 시스템에서 Horizon Agent를 업그레이드하기 위한 샘플 PowerCLI 스크립트
- SSH를 사용하여 Linux 가상 시스템에서 Horizon Agent를 업그레이드하기 위한 샘플 스크립트
- Linux 가상 시스템에서 작업을 수행하기 위한 샘플 스크립트

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"> ■ 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. ■ SSH 클라이언트 애플리케이션 <code>pscp.exe</code> 및 <code>plink.exe</code>를 다운로드하고 PowerCLI 스크립트와 같은 폴더에 넣어야 합니다.
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](#)의 내용을 참조하십시오.

- 2 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	Horizon 7 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](#)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 또는 SSH를 사용하여 복제된 가상 시스템을 AD 도메인에 연결하기 위한 샘플 스크립트에 나와 있습니다.

- 5 가상 시스템에서 구성 옵션을 업데이트합니다.

자세한 내용은 구성 파일을 Linux 가상 시스템에 업로드하기 위한 샘플 스크립트 또는 SSH를 사용하여 구성 파일을 Linux 가상 시스템에 업로드하기 위한 샘플 스크립트에 나와 있습니다.

- 6 데스크톱 풀을 만듭니다.

Linux용 수동 데스크톱 풀 생성의 내용을 참조하십시오.

전원 켜기, 종료, 다시 시작 또는 가상 시스템 삭제와 같은 작업을 수행하는 샘플 스크립트는 Linux 가상 시스템에서 작업을 수행하기 위한 샘플 스크립트를 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 업그레이드의 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"> ■ 사용자 데이터 및 프로파일이 NFS 서버와 같은 공유 서버에 있지 않는 한, 기존 가상 시스템이 삭제되었으므로 사용자 데이터 및 프로파일은 손실됩니다. ■ 가상 시스템 교체 후에 View Administrator의 가상 시스템 상태가 누락될 수 있습니다. 이 문제를 해결하려면 브로커 서비스를 다시 시작해야 합니다.
업그레이드 샘플 스크립트를 사용하여 기존 가상 시스템에 새 Horizon Agent를 업로드하고 업그레이드 명령을 실행합니다.	사용자 데이터 및 프로파일은 보존됩니다.

Linux 데스크톱 시스템의 복제를 위한 가상 시스템 템플릿 만들기

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

사전 요구 사항

- 배포가 Linux 데스크톱을 지원하기 위한 요구 사항을 충족하는지 확인하십시오. [Horizon 7 for Linux에 대한 시스템 요구 사항](#)의 내용을 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 필요한 비디오 메모리(vRAM) 값을 숙지하십시오. [2D 그래픽을 위한 가상 시스템 설정](#)의 내용을 참조하십시오.
- AD 통합 단계를 숙지합니다. [장 3 Linux 데스크톱에 대한 Active Directory 통합 및 사용자 인증 기능 설정](#)의 내용을 참조하십시오.
- Linux에 Horizon Agent를 설치하는 단계를 숙지하십시오. [장 5 Horizon Agent 설치](#)의 내용을 참조하십시오.
- 필요한 경우 Horizon 7 구성 파일을 사용하여 옵션을 구성하는 단계를 숙지합니다. [장 6 Linux 데스크톱용 구성 옵션](#)의 내용을 참조하십시오.
- 그래픽을 설정하려는 경우에는 관련 단계를 숙지합니다. [장 4 Linux 데스크톱의 그래픽 설정](#)의 내용을 참조하십시오.

절차

- 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.

Linux 배포에 대한 설치 가이드에서 vCPU 및 vMemory 크기 지침을 따르십시오.

예를 들어 Ubuntu 18.04에서는 vMemory 2048MB 및 vCPU 2개를 구성하도록 지정합니다.

- c 비디오 카드를 선택하고 디스플레이 수 및 총 비디오 메모리(vRAM)를 지정하십시오.

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

2D 그래픽을 위한 가상 시스템 설정의 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/sudoers`를 편집하고 ViewUser ALL=(ALL) NOPASSWD:ALL 줄을 추가합니다.
`/etc/sudoers`에 이 줄이 포함되어 있으면 `sudo`를 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/sudoers`를 편집하고 다음 줄을 주석 처리합니다.

```
Defaults requiretty
Defaults !visiblepw
```

- 7 Linux 배포가 RHEL/CentOS 8.x, RHEL/CentOS 7.x 또는 SLED/SLES 12.x가 아닌 경우 VMware Tools를 설치합니다.

RHEL/CentOS 8.x, RHEL/CentOS 7.x 및 SLED/SLES 12.x에는 기본적으로 Open VM Tools가 설치되어 있습니다.

8 종속성 패키지를 설치 및 구성합니다.

- a Linux 배포에서 9.10 이전 버전의 Open VM Tools 버전을 실행하는 경우 deployPkg 플러그인을 설치합니다.

지침은 <http://kb.vmware.com/kb/2075048>에서 확인할 수 있습니다.

- b Linux 배포가 Ubuntu인 경우 다음 KB 문서를 참조하여 VM에서 설치 및 구성할 종속성 패키지를 확인하십시오.

- Ubuntu 18.04 및 16.04인 경우 KB 문서 <https://kb.vmware.com/s/article/2051469> 및 <https://kb.vmware.com/s/article/59687>을 참조하십시오.
- Ubuntu 18.04인 경우 KB 문서 <https://kb.vmware.com/s/article/56409>를 참조하십시오.

9 RHEL 및 CentOS의 경우 네트워크 연결 설정 자동으로 연결을 사용하도록 설정합니다.

10 AD 통합 작업을 수행합니다.

11 그래픽 설정 단계를 수행합니다.

12 Horizon Agent를 설치합니다.

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

장 5 Horizon Agent 설치의 내용을 참조하십시오.

13 Horizon 7 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,Ubuntu1804x64,linuxagent,datastore1,10.117.44.172,snapshot1,TRUE
linux-002,Ubuntu1804x64,linuxagent,datastore1,10.117.44.172,snapshot1,TRUE
linux-003,Ubuntu1804x64,linuxagent,datastore1,10.117.44.172,snapshot1,TRUE
linux-004,Ubuntu1804x64,linuxagent,datastore1,10.117.44.172,snapshot1,TRUE
linux-005,Ubuntu1804x64,linuxagent,datastore1,10.117.44.172,snapshot1,TRUE
```

샘플 스크립트는 이 입력 파일의 이름이 **CloneVMs.csv**이고 이 파일이 스크립트와 동일한 폴더에 있다고 간주합니다.

Linux 가상 시스템을 복제하기 위한 샘플 스크립트

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

페이지 구분 없이 스크립트 내용을 복사한 후 붙여넣으려면 Horizon 7 설명서 페이지(<https://docs.vmware.com/kr/VMware-Horizon-7/index.html>)에서 이 항목의 HTML 버전을 사용하십시오.

스크립트 입력

이 스크립트는 [Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트](#)의 입력 파일에 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(전체만 가능)
- vSphere VM 콘솔을 비활성화할지 여부

스크립트 내용

```
<#
Create Clones from a Master VM

The Tool supports creation of Full clone from Master VM.
#>
#----- Functions -----
function GetInput
{
    Param($prompt, $IsPassword = $false)
    $prompt = $prompt + ": "
    Write-Host $prompt -NoNewLine
    [Console]::ForegroundColor = "Blue"
    if ($IsPassword)
    {
        $input = Read-Host -AsSecureString
```



```

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

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

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

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

function Disable_VM_Console()
{
    Param($VMToDisableConsole)
    $vmConfigSpec = New-Object VMware.Vim.VirtualMachineConfigSpec
    $extra = New-Object VMware.Vim.optionvalue
    $extra.Key="RemoteDisplay.maxConnections"
    $extra.Value="0"
    $vmConfigSpec.extraconfig += $extra
    $vm = Get-VM $VMToDisableConsole | Get-View
    $vm.ReconfigVM($vmConfigSpec)
}

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

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

$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 ("full")' -IsPassword $false
$disableVMConsole = GetInput -prompt 'Disable vSphere VM Console ("yes" or "no", recommend "yes")' -
IsPassword $false
"-----"
$csvFile = '.\CloneVMs.csv'

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

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

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

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

    $destVMName=$line.VMName
    $srcVM = $line.Parentvm
    $cSpec = $line.CustomSpec
    $targetDSName = $line.Datastore
    $destHost = $line.Host

```

```

$srcSnapshot = $line.FromSnapshot
$deleteExisting = $line.DeleteIfPresent
if (IsVMExists ($destVMName))
{
    Write-Host "VM $destVMName Already Exists in VC $vcAddress"
    if($deleteExisting -eq "TRUE")
    {
        Delete_VM ($destVMName)
    }
    else
    {
        Write-Host "Skip clone for $destVMName"
        continue
    }
}
$vm = get-vm $srcvm -ErrorAction Stop | get-view -ErrorAction Stop
$cloneSpec = new-object VMware.VIM.VirtualMachineCloneSpec
$cloneSpec.Location = new-object VMware.VIM.VirtualMachineRelocateSpec
Write-Host "Using Datastore $targetDSName"
$newDS = Get-Datastore $targetDSName | Get-View
$cloneSpec.Location.Datastore = $newDS.summary.Datastore
Set-VM -vm $srcVM -snapshot (Get-Snapshot -vm $srcVM -Name $srcSnapshot) -confirm:$false
$cloneSpec.Snapshot = $vm.Snapshot.CurrentSnapshot
$cloneSpec.Location.Host = (get-vmhost -Name $destHost).Extensiondata.MoRef
$cloneSpec.Location.Pool = (Get-ResourcePool -Name Resources -Location (Get-VMHost -Name
$destHost)).Extensiondata.MoRef
# Start the Clone task using the above parameters
$task = $vm.CloneVM_Task($vm.parent, $destVMName, $cloneSpec)
# Get the task object
$task = Get-Task | where { $_.id -eq $task }
#Wait for the taks to Complete
Wait-Task -Task $task

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

```

스크립트 실행

스크립트를 실행하면 다음 메시지가 표시됩니다.

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

클로닝 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.html의 Horizon 7 설명서 페이지에서 이 항목의 HTML 버전을 사용하십시오.

스크립트 입력

이 스크립트는 Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일에 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"

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

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

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

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

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

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

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

$destFolder = "/home/$guestUser/"

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

    $cmd = "sudo /usr/bin/net ads join -U $adUser%$adUserPassword"
    Write-Host "Run cmd 'sudo /usr/bin/net ads join' in VM '$VMName' with user '$guestUser'"
    Invoke-VMScript -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -GuestUser
$guestUser -GuestPassword $guestPassword -Confirm:$false -ScriptType Bash -ScriptText $cmd
}

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

스크립트 실행

스크립트를 실행하면 다음 메시지가 표시됩니다.

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

-----
Your vCenter address: 10.117.44.17
Your vCenter admin user name: administrator
Your vCenter admin user password: *****
-----
Your ESXi host admin user name, such as root: root
Your ESXi host admin user password: *****
-----
Your VM guest OS user name: ViewUser
Your VM guest OS user password: *****
-----
```

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

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.html의 Horizon 7 설명서 페이지에서 이 항목의 HTML 버전을 사용하십시오.

스크립트 입력

이 스크립트는 Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일에 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

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

.NOTES
#>
#----- Functions -----
function GetInput
{
```

```

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

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

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

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

```



```

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

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

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

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

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

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

#Connect to vCenter
$VC_Conn_State = Connect-VIServer $vcAddress -user $vcAdmin -password $vcPassword
if([string]::IsNullOrEmpty($VC_Conn_State))

```

```

{
    Write-Host 'Exit since failed to login vCenter'
    exit
}
else
{
    Write-Host 'vCenter is connected'
}

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

$destFolder = "/home/$guestUser/"

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

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

Disconnect-VIServer $vcAddress -Confirm:$false
exit

```

스크립트 실행

스크립트를 실행하면 다음 메시지가 표시됩니다.

```

PowerCLI C:\scripts> .\ClonedVMs_JoinDomain_SSH.ps1

-----
Your vCenter address: 10.117.44.17
Your vCenter admin user name: administrator
Your vCenter admin user password: *****
-----
Your VM guest OS user name: ViewUser
Your VM guest OS user password: *****
-----
Type the AD user name to join the AD: viewadmin
Please type the AD user password.
Please note that special character should be escaped. For example, $ should be \$
Your AD user password: *****

```

구성 파일을 Linux 가상 시스템에 업로드하기 위한 샘플 스크립트

다음 샘플 스크립트를 사용자 지정 및 사용하여 구성 파일 `config` 및 `viewagent-custom.conf`를 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.html의 Horizon 7 설명서 페이지에서 이 항목의 HTML 버전을 사용하십시오.

스크립트 입력

이 스크립트는 [Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일](#)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 = '.\CloneVMs.csv'
$setConfig = $false
$setCustomConf = $false
$config_File = "config"
$customConf_File = "viewagent-custom.conf"

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

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

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

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

```

```

}

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

$destFolder = "/home/$guestUser/"

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

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

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

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

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

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

```

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 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.conf`를 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.html의 Horizon 7 설명서 페이지에서 이 항목의 HTML 버전을 사용하십시오.

스크립트 입력

이 스크립트는 [Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일](#)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
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBS
TR($input))
    }
    else
    {
        $input = Read-Host
    }

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

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

    $VM= Get-VM $VM_Name
    $IP = $VM.guest.IPAddress[0]

```

```

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

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

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

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

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

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

```



```

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

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

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

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

$destFolder = "/home/$guestUser/"

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

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

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

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

```

```

}

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

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

```

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 PowerCLI 스크립트

다음 샘플 스크립트를 사용자 지정 한 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://docs.vmware.com/kr/VMware-Horizon-7/index.html>의 Horizon 7 설명서 페이지에서 이 항목의 HTML 버전을 사용하십시오.

스크립트 입력

이 스크립트는 [Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일](#)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
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBS
TR($input))
    }
    else
    {
        $input = Read-Host
    }

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

```

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

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

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

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

```

```

Functions-----
function GetSourceInstallerMD5()
{
    $agentInstallerPath = Convert-Path $agentInstaller;
    $md5 = New-Object -TypeName System.Security.Cryptography.MD5CryptoServiceProvider;
    $md5HashWithFormat =
[System.BitConverter]::ToString($md5.ComputeHash([System.IO.File]::ReadAllBytes($agentInstallerPath)))
;
    $md5Hash = ($md5HashWithFormat.replace("-", "")).ToLower();
    return $md5Hash;
}

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

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

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

$destFolder = "/home/$guestUser/"

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

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

    #Upload installer tar ball to Linux VM
    Write-Host "Upload File '$agentInstaller' to '$destFolder' of VM '$VMName' with user '$guestUser'"
    Copy-VMGuestFile -HostUser $hostAdmin -HostPassword $hostPassword -VM $VMName -GuestUser
$guestUser -GuestPassword $guestPassword -Confirm:$false -LocalToGuest -Destination $destFolder -
Source $agentInstaller

    #Check the uploaded installer md5sum
    $cmd = "md5sum VMware-*linux-*"
    Write-Host "Run cmd '$cmd' in VM '$VMName' with user '$guestUser'"

```

```

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

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

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

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

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

    $cmd = "sudo shutdown -r +1&"
    Write-Host "Reboot to apply the Horizon 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:\scripts> .\InstallAgent.ps1
-----
Accept Linux Horizon Agent EULA in tar bundle ("yes" or "no"): yes
Your vCenter address: 10.117.44.17
Your vCenter admin user name: administrator

```

```

Your vCenter admin user password: *****
-----
Your ESXi host admin user name, such as root: root
Your ESXi host admin user password: *****
-----
Your VM guest OS user name: HorizonUser
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 지정한 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.html의 Horizon 7 설명서 페이지에서 이 항목의 HTML 버전을 사용하십시오.

스크립트 입력

이 스크립트는 [Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일](#)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 리더렉션 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
        $input =
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBS
TR($input))
    }
    else
    {
        $input = Read-Host
    }

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



```

official web site'
    exit
}
}

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

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

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

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

#-----Handle
input-----
"-----"
Check_SSH_Client -IsPlink $true -IsPSCP $true
"-----"
$acceptEULA = GetInput -prompt 'Accept Linux View Agent EULA in tar bundle ("yes" or "no")' -
IsPassword $false
if ($acceptEULA -ne "yes")
{
    write-host -ForegroundColor Red "You need accept the EULA with 'yes'(case sensitive)"
    exit
}
$svcAddress = GetInput -prompt "Your vCenter address" -IsPassword $false
$svcAdmin = GetInput -prompt "Your vCenter admin user name" -IsPassword $false
$svcPassword = GetInput -prompt "Your vCenter admin user password" -IsPassword $true
"-----"
$guestUser = GetInput -prompt 'Your VM guest OS user name' -IsPassword $false
$guestPassword = GetInput -prompt 'Your VM guest OS user password' -IsPassword $true

```

```

"-----"
$agentInstaller = GetInput -prompt 'Type the View Agent tar ball path' -IsPassword $false
"-----"
$UpgradeToManagedVM = GetInput -prompt 'Upgrade to managed VM ("yes" or "no")' -IsPassword $false
if (($UpgradeToManagedVM -ne "yes") -AND $UpgradeToManagedVM -ne "no")
{
    write-host -ForegroundColor Red "You need select 'yes' or 'no'(case sensitive)"
    exit
}
$installSmartcard = GetInput -prompt 'Install the Smartcard redirection feature ("yes" or "no")' -
IsPassword $false
if (($installSmartcard -ne "yes") -AND $installSmartcard -ne "no")
{
    write-host -ForegroundColor Red "You need select 'yes' or 'no'(case sensitive)"
    exit
}
"-----"

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

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

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

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

#Connect to vCenter
$VC_Conn_State = Connect-VIServer $vcAddress -user $vcAdmin -password $vcPassword
if([string]::IsNullOrEmpty($VC_Conn_State))

```

```

{
    Write-Host 'Exit since failed to login vCenter'
    exit
}
else
{
    Write-Host 'vCenter is connected'
}

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

$destFolder = "/home/$guestUser/"

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

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

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

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

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

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

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

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

        #Run the upgrade command.
    }
}

```

```

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

```

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 지정한 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>의 Horizon 7 설명서 페이지에서 이 항목의 HTML 버전을 사용하십시오.

스크립트 입력

이 스크립트는 [Linux 데스크톱 배포를 위한 샘플 PowerCLI 스크립트의 입력 파일](#)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 = '.\CloneVMs.csv'

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

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

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

```

```

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

Disconnect-VIServer $vcAddress -Confirm:$false
exit

```

스크립트 실행

스크립트를 실행하면 다음 메시지가 표시됩니다.

```

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

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

Your selection is 6). Delete VM

```

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

9

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

본 장은 다음 항목을 포함합니다.

- [Horizon Console](#)에서 [Horizon Help Desk Tool](#) 사용
- [Horizon 7 for Linux](#) 시스템에 대한 진단 정보 수집
- [Horizon Agent](#)에서 [iPad Pro Horizon Client](#)에 연결 해제하지 못함
- [SLES 12 SP1](#) 데스크톱이 자동으로 새로 고쳐지지 않음
- [SSO](#)에서 [PowerOff](#) 에이전트에 연결하지 못함
- Linux용 수동 데스크톱 풀을 생성한 후에 VM에 연결할 수 없음

Horizon Console에서 Horizon Help Desk Tool 사용

Horizon Help Desk Tool은 Horizon 7 사용자 세션 상태를 가져오고 문제 해결 및 유지 보수 작업을 수행하는 데 사용할 수 있는 웹 애플리케이션입니다.

Horizon Help Desk Tool에서 사용자 세션을 조회하여 문제를 해결하고 데스크톱 다시 시작 또는 재설정과 같은 데스크톱 유지 보수 작업을 수행할 수 있습니다.

Horizon Help Desk Tool를 구성하려면 다음 요구 사항을 충족해야 합니다.

- Horizon 7에 대한 [Horizon Enterprise Edition](#) 라이선스 또는 [Horizon Apps Advanced Edition](#) 라이선스. 올바른 라이선스가 있는지 확인하려면 "[Horizon 7 관리](#)" 문서를 참조하십시오.
- Horizon 7 구성 요소에 대한 정보를 저장하는 이벤트 데이터베이스. 이벤트 데이터베이스 구성에 대한 자세한 내용은 "[Horizon 7 관리](#)" 문서를 참조하십시오.
- Horizon Help Desk Tool에 로그인하기 위한 헬프 데스크 관리자 역할 또는 헬프 데스크 관리자(읽기 전용) 역할. 이러한 역할에 대한 자세한 내용은 "[Horizon 7 관리](#)" 문서를 참조하십시오.
- 로그인 세그먼트를 보려면 각 연결 서버 인스턴스에서 타이밍 프로파일러를 사용하도록 설정합니다.

각 연결 서버 인스턴스에서 타이밍 프로파일러를 사용하도록 설정하려면 다음 **vdmadmin** 명령을 사용합니다.

```
vdmadmin -I -timingProfiler -enable
```

관리 포트를 사용하는 연결 서버 인스턴스에서 타이밍 프로파일러를 사용하도록 설정하려면 다음 **vdmadmin** 명령을 사용합니다.

```
vdmadmin -I -timingProfiler -enable -server {ip/server}
```

- **/etc/vmware/viewagent-custom.conf** 구성 파일에서 **HelpDeskEnable** 옵션을 사용하도록 설정합니다.

Horizon Console에서 Horizon Help Desk Tool 시작

Horizon Help Desk Tool이(가) Horizon Console에 통합되었습니다. Horizon Help Desk Tool에서 문제를 해결하려는 사용자를 검색할 수 있습니다.

절차

- 1 [사용자 검색] 텍스트 상자에서 사용자 이름을 검색하거나 Horizon Help Desk Tool 도구로 바로 이동할 수 있습니다.

- Horizon Console에서 [사용자 검색] 텍스트 상자에 사용자 이름을 입력합니다.
- **모니터 > 헬프 데스크**를 선택하고 [사용자 검색] 텍스트 상자에 사용자 이름을 입력합니다.

Horizon Console의 검색 결과에 사용자 목록이 표시됩니다. 검색 시 최대 100개의 일치하는 결과가 반환될 수 있습니다.

- 2 사용자 이름을 선택합니다.

사용자 정보가 사용자 카드에 표시됩니다.

다음에 수행할 작업

문제를 해결하려면 사용자 카드에서 관련 탭을 클릭합니다.

Horizon Help Desk Tool에서 사용자 문제 해결

Horizon Help Desk Tool에서는 사용자 카드의 기본 사용자 정보를 볼 수 있습니다. 사용자 카드에서 탭을 클릭하여 특정 구성 요소에 대한 자세한 정보를 얻을 수 있습니다.

경우에 따라 사용자 세부 정보가 표에 제공될 수도 있습니다. 표 열을 기준으로 이러한 사용자 세부 정보를 정렬할 수 있습니다.

- 열을 오름차순으로 정렬하려면 열을 한 번 클릭합니다.
- 열을 내림차순으로 정렬하려면 열을 두 번 클릭합니다.
- 열을 정렬하지 않으려면 열을 세 번 클릭합니다.

기본 사용자 정보

사용자의 사용자 이름, 전화 번호 및 이메일 주소와 같은 기본 사용자 정보와 사용자의 연결 또는 연결 해제 상태를 표시합니다. 사용자에게 데스크톱 세션이 있는 경우 사용자는 연결된 상태입니다. 사용자에게 데스크톱 세션이 없는 경우 사용자는 연결 해제된 상태입니다.

이메일 주소를 클릭하여 사용자에게 메시지를 전송할 수 있습니다.

세션

세션 탭에는 사용자가 연결되는 데스크톱 세션에 대한 정보가 표시됩니다.

필터 텍스트 상자를 사용하여 데스크톱 세션을 필터링할 수 있습니다.

참고 세션 탭에는 vSphere Client 또는 ESXi에서 VM에 액세스하는 세션에 대한 세션 정보는 표시되지 않습니다.

세션 탭에는 다음 정보가 포함됩니다.

표 9-1. 세션 탭

옵션	설명
상태	데스크톱 세션의 상태에 대한 정보를 표시합니다. <ul style="list-style-type: none"> ■ 세션이 연결된 경우 녹색으로 나타납니다. ■ L: 세션이 로컬 세션이거나 로컬 포트에서 실행되는 세션인 경우.
컴퓨터 이름	데스크톱 세션의 이름입니다. 이름을 클릭하여 카드의 세션 정보를 엽니다. <p>세션 카드에 있는 탭을 클릭하여 추가 정보를 볼 수 있습니다.</p> <ul style="list-style-type: none"> ■ 세부 정보 탭에는 VM 정보, CPU 또는 메모리 사용량 같은 사용자 정보가 표시됩니다. ■ 프로세스 탭에는 CPU 및 메모리 관련 프로세스에 대한 정보가 표시됩니다.
프로토콜	데스크톱 세션에 대한 디스플레이 프로토콜입니다.
유형	데스크톱이 게시된 데스크톱인지 또는 가상 시스템 데스크톱인지를 표시합니다.
연결 시간	연결 서버에 세션이 연결된 시간입니다.
세션 기간	세션이 연결 서버와 연결된 상태를 유지하는 기간입니다.

데스크톱

데스크톱 탭에는 사용자에게 사용 권한이 부여된, 게시된 데스크톱 또는 가상 데스크톱에 대한 정보가 표시됩니다.

표 9-2. 데스크톱

옵션	설명
상태	데스크톱 세션의 상태에 대한 정보를 표시합니다. ■ 세션이 연결된 경우 녹색으로 나타납니다.
데스크톱 풀 이름	세션의 데스크톱 풀 이름입니다.
데스크톱 유형	데스크톱이 게시된 데스크톱인지 또는 가상 시스템 데스크톱인지를 표시합니다. 참고 세션이 포드 페더레이션의 다른 포드에서 실행되는 경우 어떤 정보도 표시되지 않습니다.
유형	데스크톱 권한 유형에 대한 정보를 표시합니다. ■ 로컬: 로컬 권한의 경우
vCenter	vCenter Server에 있는 가상 시스템 이름을 표시합니다. 참고 세션이 포드 페더레이션의 다른 포드에서 실행되는 경우 어떤 정보도 표시되지 않습니다.
기본 프로토콜	데스크톱 세션에 대한 기본 디스플레이 프로토콜입니다.

활동

활동 탭에는 사용자 활동에 대한 이벤트 로그 정보가 표시됩니다. 최근 12시간, 최근 30일 같은 시간 범위 또는 관리자 이름을 기준으로 활동을 필터링할 수 있습니다. Horizon Help Desk Tool 활동만을 기준으로 필터링하려면 **기술 지원 이벤트만 해당**을 클릭합니다. 새로 고침 아이콘을 클릭하여 이벤트 로그를 새로 고칩니다. 이벤트 로그를 파일로 내보내려면 내보내기 아이콘을 클릭합니다.

참고 Cloud Pod 아키텍처 환경의 사용자에게 대해서는 이벤트 로그 정보가 표시되지 않습니다.

표 9-3. 활동

옵션	설명
시간	시간 범위를 선택합니다. 기본값은 최근 12시간입니다. ■ 최근 12시간 ■ 최근 24시간 ■ 최근 7일 ■ 최근 30일 ■ 모두
관리자	관리자 사용자의 이름입니다.
메시지	사용자 또는 관리자가 수행한 활동에 국한되는 사용자 또는 관리자에 대한 메시지를 표시합니다.
리소스 이름	활동이 수행된 데스크톱 풀 또는 가상 시스템 이름에 대한 정보를 표시합니다.

Horizon Help Desk Tool에 대한 세션 세부 정보

세션 세부 정보는 **세션** 탭에서 **컴퓨터 이름** 옵션에 있는 사용자 이름을 클릭하면 **세부 정보** 탭에 나타납니다. Horizon Client, 가상 또는 게시된 데스크톱에 대한 세부 정보와 CPU 및 메모리 세부 정보를 볼 수 있습니다.

클라이언트

Horizon Client 유형에 따라 정보를 표시하고 사용자 이름, Horizon Client 버전, 클라이언트 시스템의 IP 주소 및 클라이언트 시스템의 운영 체제와 같은 세부 정보를 포함합니다.

참고 Horizon Agent를 업그레이드한 경우 Horizon Client도 최신 버전으로 업그레이드해야 합니다. 그렇지 않으면 Horizon Client에 대한 버전이 표시되지 않습니다. Horizon Client 업그레이드에 대한 자세한 내용은 "Horizon 7 업그레이드" 문서를 참조하십시오.

VM

가상 데스크톱 또는 게시된 데스크톱에 대한 정보를 표시합니다.

표 9-4. VM 세부 정보

옵션	설명
컴퓨터 이름	데스크톱 세션의 이름입니다.
에이전트 버전	Horizon Agent 버전입니다.
OS 버전	운영 체제 버전입니다.
연결 서버	세션이 연결된 연결 서버입니다.
풀	데스크톱 풀 이름
vCenter	vCenter Server의 IP 주소입니다.
세션 상태	데스크톱 세션의 상태입니다. 세션 상태는 연결됨 또는 연결 해제됨일 수 있습니다.
세션 기간	세션이 연결 서버에 연결된 상태를 유지하는 시간입니다.
상태 기간	세션이 동일한 상태를 유지하는 시간입니다.
로그온 시간	세션에 로그인한 사용자의 로그인 시간입니다.
로그온 기간	사용자가 Linux 데스크톱에 로그인되어 있는 시간입니다.

사용자 환경 메트릭

VMware Blast 디스플레이 프로토콜을 사용하는 가상 또는 게시된 데스크톱 세션에 대한 성능 세부 정보를 표시합니다. 이러한 성능 세부 정보를 보려면 **자세히**를 클릭합니다. 이러한 세부 정보를 새로 고치려면 새로 고침 아이콘을 클릭합니다.

표 9-5. Blast 디스플레이 프로토콜 세부 정보

옵션	설명
프레임 속도	Blast 세션의 프레임 속도(초당 프레임 수)입니다.
Skype 상태	Linux 데스크톱 세션의 경우 이 옵션이 해당 없음으로 표시됩니다.
Blast 세션 카운터	<ul style="list-style-type: none"> ■ 예상 대역폭(업링크). 업링크 신호의 예상 대역폭입니다. ■ 패킷 손실(업링크). 업링크 신호의 패킷 손실 백분율입니다.
Blast 이미징 카운터	<ul style="list-style-type: none"> ■ 전송된 바이트. Blast 세션에 대해 전송된 이미징 데이터의 총 바이트 수입니다. ■ 수신된 바이트. Blast 세션에 대해 수신된 이미징 데이터의 총 바이트 수입니다.
Blast 오디오 카운터	<ul style="list-style-type: none"> ■ 전송된 바이트. Blast 세션에 대해 전송된 오디오 데이터의 총 바이트 수입니다. ■ 수신된 바이트. Blast 세션에 대해 수신된 오디오 데이터의 총 바이트 수입니다.
Blast CDR 카운터	<ul style="list-style-type: none"> ■ 전송된 바이트. Blast 세션에 대해 전송된 클라이언트 드라이브 리더렉션 데이터의 총 바이트 수입니다. ■ 수신된 바이트. Blast 세션에 대해 수신된 클라이언트 드라이브 리더렉션 데이터의 총 바이트 수입니다.

CPU 및 메모리 사용량과 네트워크 및 디스크 성능

가상 또는 게시된 데스크톱의 CPU 및 메모리 사용량과 Blast 디스플레이 프로토콜에 대한 네트워크 또는 디스크 성능을 차트로 표시합니다.

참고 데스크톱에서 Horizon Agent를 시작하거나 다시 시작한 후 성능 차트에 타임라인이 즉시 표시되지 않을 수 있습니다. 몇 분 후에 타임라인이 나타납니다.

표 9-6. CPU 사용량

옵션	설명
세션 CPU	현재 세션의 CPU 사용량입니다.
호스트 CPU	세션이 할당된 가상 시스템의 CPU 사용량입니다.

표 9-7. 메모리 사용량

옵션	설명
세션 메모리	현재 세션의 메모리 사용량입니다.
호스트 메모리	세션이 할당된 가상 시스템의 메모리 사용량입니다.

표 9-8. 네트워크 성능

옵션	설명
지연 시간	PCoIP 또는 Blast 세션의 지연 시간 차트를 표시합니다. 지연 시간은 왕복 시간(밀리초)입니다. 이 지연 시간을 추적하는 성능 카운터는 VMware Blast 세션 카운터 > RTT 입니다.

표 9-9. 디스크 성능

옵션	설명
읽기	초당 읽기 I/O(입출력) 작업 수입니다.
쓰기	초당 쓰기 I/O 작업 수입니다.
디스크 지연 시간	디스크 지연 시간에 대한 차트를 표시합니다. 디스크 지연 시간은 Windows 성능 카운터에서 검색된 IOPS(초당 입출력 작업) 데이터의 시간(밀리초)입니다.
평균 읽기	초당 임의 읽기 I/O 작업의 평균 수입니다.
평균 쓰기	초당 임의 쓰기 I/O 작업의 평균 수입니다.
평균 지연 시간	Windows 성능 카운터에서 검색된 IOPS 데이터의 평균 지연 시간(밀리초)입니다.

세션 로그온 세그먼트

로그온 동안 생성된 로그온 기간 및 사용량 세그먼트를 표시합니다.

표 9-10. 세션 로그인 세그먼트

옵션	설명
로그온 기간	사용자가 데스크톱 풀을 클릭한 시간부터 사용자가 Linux 데스크톱에 로그인한 시간까지 계산된 기간입니다.
세션 로그인 시간	사용자가 세션에 로그인된 기간입니다.
로그온 세그먼트	<p>로그온 동안 생성된 세그먼트를 표시합니다.</p> <ul style="list-style-type: none"> ■ 브로커링. 연결 서버에서 세션 연결 또는 다시 연결을 처리하는 총 시간. 사용자가 데스크톱 풀을 클릭하는 시간부터 터널 연결이 설정되는 시간까지 계산됩니다. 사용자 인증, 시스템 선택 및 터널 연결 설정을 위한 시스템 준비 등의 연결 서버 작업 시간이 포함됩니다. ■ 대화형. Horizon Agent에서 세션 연결 또는 다시 연결을 처리하는 총 시간. Blast Extreme이 터널 연결을 사용하는 시간부터 사용자가 Linux 데스크톱에 로그인한 시간까지 계산됩니다. ■ 프로토콜 연결. 로그인 프로세스 동안 PCoIP 또는 Blast 프로토콜 연결을 완료하는 데 소요된 총 시간입니다. ■ 로그온 스크립트. 시작에서 완료까지 로그인 스크립트를 실행하는 데 소요된 총 시간입니다. ■ 인증. 연결 서버에서 세션을 인증하는 데 소요되는 총 시간입니다. ■ VM 시작. VM을 시작하는 데 소요된 총 시간입니다. 이 시간에는 운영 체제 부팅, 일시 중단된 시스템 재개에 소요되는 시간과 Horizon Agent가 연결 준비가 완료되었음을 신호로 알리는 데 걸리는 시간이 포함됩니다.

Horizon Help Desk Tool에 대한 세션 프로세스

세션 프로세스는 **세션** 탭에서 **컴퓨터 이름** 옵션에 있는 사용자 이름을 클릭하면 **프로세스** 탭에 나타납니다.

프로세스

세션 프로세스 목록을 따라 스크롤하지 않으려면 [검색 필터] 텍스트 상자에 프로세스 이름을 입력하여 세션 프로세스를 검색할 수 있습니다.

각 세션에 대해 CPU 및 메모리 관련 프로세스에 대한 추가 세부 정보를 볼 수 있습니다. 예를 들어 세션에 대한 CPU 및 메모리 사용량이 비정상적으로 높다는 것을 알게 된 경우 **프로세스** 탭에서 프로세스에 대한 세부 정보를 볼 수 있습니다.

RDS 호스트 세션의 경우 **프로세스** 탭에 현재 사용자 또는 현재 시스템 프로세스에 의해 시작된 현재 RDS 호스트 세션 프로세스가 표시됩니다.

표 9-11. 세션 프로세스 세부 정보

옵션	설명
프로세스 이름	세션 프로세스의 이름입니다. 예: chrome.exe.
CPU	프로세스의 CPU 사용량(백분율)입니다.

표 9-11. 세션 프로세스 세부 정보 (계속)

옵션	설명
메모리	프로세스의 메모리 사용량(KB)입니다.
디스크	메모리 디스크 IOPS입니다. 다음 수식을 사용하여 계산됩니다. (현재 시간의 총 I/O 바이트) - (현재 시간 1초 전의 총 I/O 바이트) 작업 관리자가 양수 값을 표시하는 경우 이 계산은 초당 0KB의 값을 표시할 수 있습니다.
사용자 이름	프로세스를 소유하는 사용자의 사용자 이름입니다.
호스트 CPU	세션이 할당된 가상 시스템의 CPU 사용량입니다.
호스트 메모리	세션이 할당된 가상 시스템의 메모리 사용량입니다.
프로세스	가상 시스템의 프로세스의 수입니다.
새로 고침	새로 고침 아이콘은 프로세스의 목록을 새로 고칩니다.
프로세스 종료	실행 중인 프로세스를 종료합니다. 참고 프로세스를 종료하려면 헬프 데스크 관리자 역할이 있어야 합니다. 프로세스를 종료하려면 프로세스를 선택하고 프로세스 종료 버튼을 클릭합니다. 프로세스 탭에 나열될 수 있는 Windows 코어 프로세스와 같은 중요한 프로세스는 종료할 수 없습니다. 중요한 프로세스를 종료하는 경우 Horizon Help Desk Tool에 시스템 프로세스를 종료할 수 없다는 메시지가 표시됩니다.

Horizon Help Desk Tool에서 Linux 데스크톱 세션 문제 해결

Horizon Help Desk Tool에서 사용자의 연결 상태를 기준으로 Linux 데스크톱 세션 문제를 해결할 수 있습니다.

사전 요구 사항

- Horizon Help Desk Tool를 시작합니다.

절차

- 1 사용자 카드에서 **세션** 탭을 클릭합니다.

CPU 및 메모리 사용량을 표시하고 Horizon Client, 가상 또는 게시된 데스크톱에 대한 정보를 포함하는 성능 카드가 표시됩니다.

2 문제 해결 옵션을 선택합니다.

옵션	조치
메시지 보내기	<p>게시된 데스크톱 또는 가상 데스크톱의 사용자에게 메시지를 전송합니다. [주의], [정보] 또는 [오류]를 포함하도록 메시지 심각도를 선택할 수 있습니다.</p> <p>메시지 보내기를 클릭하고 심각도 유형 및 메시지 세부 정보를 입력한 후 제출을 클릭합니다.</p>
다시 시작	<p>가상 데스크톱에서 다시 시작 프로세스를 시작합니다. 이 기능은 게시된 데스크톱 세션에는 사용할 수 없습니다.</p> <p>VDI 다시 시작을 클릭합니다.</p>
연결 끊기	<p>데스크톱 또는 애플리케이션 세션을 연결 해제합니다.</p> <p>자세히 > 연결 끊기를 클릭합니다.</p>
로그오프	<p>게시된 데스크톱 또는 가상 데스크톱에 대해 로그오프 프로세스를 시작합니다.</p> <p>자세히 > 로그오프를 클릭합니다.</p>
재설정	<p>가상 시스템의 재설정을 시작합니다. 이 기능은 게시된 데스크톱에는 사용할 수 없습니다.</p> <p>자세히 > VM 재설정을 클릭합니다.</p>
참고 저장되지 않은 작업은 손실될 수 있습니다.	

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.

절차

- 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
```

결과

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.

Horizon Agent에서 iPad Pro Horizon Client에 연결 해제하지 못함

iPad Pro Horizon Client에서 다시 시작 또는 종료를 수행한 후에 SUSE Horizon Agent 연결이 연결을 해제하지 못합니다.

문제

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

원인

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

해결책

- ◆ iPad Pro Horizon Client에서 데스크톱 연결을 수동으로 해제합니다.

SLES 12 SP1 데스크톱이 자동으로 새로 고쳐지지 않음

GNOME 터미널을 끌어들 때 다중 모니터 모드에서 SLES 12 SP1이 자동으로 새로 고쳐지지 않습니다.

문제

다중 모니터 모드에서 SLES 12 SP1을 시작한 후 창 모드로 돌아갈 경우 GNOME 터미널을 끌어들 때 데스크톱은 자동으로 새로 고쳐지지 않습니다.

원인

GNOME 터미널은 끌어들기 작업에 응답하지 않습니다.

해결책

- 1 GNOME 셸 세션을 종료합니다.

```
kill -9 <process id of gnome-shell>
```

- 2 GNOME 셸 세션을 다시 시작합니다.

SSO에서 PowerOff 에이전트에 연결하지 못함

SSO(Single Sign-On)에서 PowerOff 에이전트에 연결하지 못합니다.

문제

브로커로 로그인하고 에이전트에 연결할 경우 SSO는 PowerOff 에이전트에 연결하지 못합니다.

해결책

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

Linux용 수동 데스크톱 풀을 생성한 후에 VM에 연결할 수 없음

가상 시스템 상태가 응답하지 않습니다.

문제

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.