



# Red Hat Virtualization 4.3

## 관리 가이드

Red Hat Virtualization의 관리 작업



## Red Hat Virtualization 4.3 관리 가이드

---

Red Hat Virtualization의 관리 작업

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

## 법적 공지

Copyright © 2022 | You need to change the HOLDER entity in the en-US/Administration\_Guide.ent file |.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## 초록

이 문서에서는 Red Hat Virtualization 관리자와 관련된 정보 및 절차를 설명합니다.

## 차례

I 부. RED HAT VIRTUALIZATION 환경 유지 및 유지 관리 .....	14
1장. 글로벌 구성 .....	15
1.1. 역할 .....	15
1.1.1. 새 역할 생성 .....	15
1.1.2. 역할 편집 또는 복사 .....	15
1.1.3. 사용자 역할 및 권한 부여 예 .....	16
1.2. 시스템 권한 .....	17
1.2.1. 사용자 속성 .....	18
1.2.2. 사용자 및 관리자 역할 .....	18
1.2.3. 설명되는 사용자 역할 .....	19
1.2.4. 설명된 관리자 역할 .....	20
1.2.5. 리소스에 관리자 또는 사용자 역할 할당 .....	22
1.2.6. 리소스에서 관리자 또는 사용자 역할 제거 .....	22
1.2.7. 데이터 센터의 시스템 권한 관리 .....	22
1.2.8. 데이터 센터 관리자 역할 설명 .....	23
1.2.9. 클러스터의 시스템 권한 관리 .....	23
1.2.10. 설명된 클러스터 관리자 역할 .....	24
1.2.11. 네트워크에 대한 시스템 권한 관리 .....	24
1.2.12. 네트워크 관리자 및 사용자 역할 설명 .....	25
1.2.13. 호스트의 시스템 권한 관리 .....	25
1.2.14. 설명된 호스트 관리자 역할 .....	26
1.2.15. 스토리지 도메인의 시스템 권한 관리 .....	26
1.2.16. 설명된 스토리지 관리자 역할 .....	27
1.2.17. 가상 머신 풀의 시스템 권한 관리 .....	27
1.2.18. 가상 머신 풀 관리자 역할 설명 .....	27
1.2.19. 가상 디스크에 대한 시스템 권한 관리 .....	28
1.2.20. 설명된 가상 디스크 사용자 역할 .....	28
1.2.21. 레거시 SPICE Cipher 설정 .....	29
1.3. 정책 예약 .....	29
1.3.1. 스케줄링 정책 생성 .....	31
1.3.2. 새 스케줄링 정책 및 스케줄링 정책 편집 창의 설정에 대한 설명 .....	31
1.4. 인스턴스 유형 .....	35
1.4.1. 인스턴스 유형 생성 .....	35
1.4.2. 인스턴스 유형 편집 .....	36
1.4.3. 인스턴스 유형 제거 .....	36
1.5. MAC 주소 풀 .....	37
1.5.1. MAC 주소 풀 생성 .....	37
1.5.2. MAC 주소 풀 편집 .....	38
1.5.3. MAC 주소 풀 권한 편집 .....	38
1.5.4. MAC 주소 풀 제거 .....	39
2장. 대시보드 .....	40
2.1. 사전 요구 사항 .....	40
2.2. 글로벌 인벤토리 .....	40
2.3. 글로벌 사용률 .....	42
2.3.1. 가장 많이 사용되는 리소스 .....	42
2.4. 클러스터 사용률 .....	43
2.4.1. CPU .....	43
2.4.2. 메모리 .....	43
2.5. 스토리지 사용률 .....	44

<b>3장. SEARCH</b> .....	<b>45</b>
3.1. RED HAT VIRTUALIZATION에서 검색 수행	45
3.2. 검색 구문 및 예	45
3.3. 자동 완료 검색	45
3.4. 결과 유형 옵션	46
3.5. 검색 RESULTS	46
3.6. 검색: 여러KNATIVE 및 와일드 카드	48
3.7. SEARCH ORDER(검색 순서 결정)	48
3.8. 데이터 센터 검색	49
3.9. 클러스터 검색	49
3.10. 호스트 검색	50
3.11. 네트워크 검색	52
3.12. 스토리지 검색	53
3.13. 디스크 검색	54
3.14. 볼륨 검색	56
3.15. 가상 머신 검색	57
3.16. 풀 검색	60
3.17. 템플릿 검색	60
3.18. 사용자 검색	62
3.19. 이벤트 검색	63
<b>4장. 북마크</b> .....	<b>65</b>
4.1. 쿼리 문자열을 BOOKMARK로 저장	65
4.2. 북마크 편집	65
4.3. 북마크 삭제	66
<b>5장. 태그</b> .....	<b>67</b>
5.1. 태그를 사용하여 RED HAT VIRTUALIZATION으로 상호 작용 사용자 지정	67
5.2. 태그 생성	67
5.3. 태그 수정	67
5.4. 태그 삭제	68
5.5. 오브젝트에서 태그 추가 및 제거	69
5.6. 태그를 사용하여 오브젝트 검색	69
5.7. 태그를 사용하여 호스트 사용자 정의	69
<b>II 부. 리소스 확보</b> .....	<b>71</b>
<b>6장. 서비스 품질</b> .....	<b>72</b>
6.1. 스토리지 서비스 품질	72
6.1.1. 서비스 입력의 스토리지 품질 생성	72
6.1.2. 서비스 입력의 스토리지 품질 제거	73
6.2. 가상 머신 네트워크 서비스 품질	74
6.2.1. 가상 머신 네트워크 서비스 품질 생성	74
6.2.2. 새로운 가상 머신 네트워크 QoS의 설정 및 가상 머신 네트워크 QoS(Virtual Machine Network QoS 설명) 편집	75
6.2.3. 가상 머신 네트워크 서비스 품질 제거	76
6.3. 호스트 네트워크 서비스 품질	76
6.3.1. 호스트 네트워크 서비스 품질 생성	76
6.3.2. 새로운 호스트 네트워크 서비스 품질의 설정 및 Windows의 호스트 네트워크 품질 편집	77
6.3.3. 호스트 네트워크 서비스 품질 제거	78
6.4. CPU 서비스 품질	79
6.4.1. CPU Quality of Service Entry 생성	79
6.4.2. CPU Quality of Service Entry 제거	80

<b>7장. 데이터 센터</b> .....	<b>81</b>
7.1. 데이터 센터 소개	81
7.2. 스토리지 풀 관리자	81
7.3. ECDHE 우선 순위	82
7.4. 데이터 센터 작업	82
7.4.1. 새 데이터 센터 생성	82
7.4.2. 새 데이터 센터 및 편집 데이터 센터 Windows의 설정에 대한 설명	83
7.4.3. 데이터 센터 재 시작: recovery process	84
7.4.4. 데이터 센터 제거	85
7.4.5. 데이터 센터 제거	86
7.4.6. 데이터 센터 스토리지 유형 변경	86
7.4.7. 데이터 센터 호환성 버전 변경	87
7.5. 데이터 센터 및 스토리지 도메인	88
7.5.1. 데이터 센터에 기존 데이터 도메인 연결	88
7.5.2. 데이터 센터에 기존 ISO 도메인 연결	89
7.5.3. 데이터 센터에 기존 내보내기 도메인 연결	89
7.5.4. 데이터 센터에서 스토리지 도메인 분리	90
<b>8장. 클러스터</b> .....	<b>92</b>
8.1. 클러스터 소개	92
8.2. 클러스터 작업	93
8.2.1. 새 클러스터 생성	93
8.2.2. 일반 클러스터 설정 설명	96
8.2.3. 최적화 설정 설명	98
8.2.4. 설명된 마이그레이션 정책 설정	100
8.2.5. 설명된 스케줄링 정책 설정	103
8.2.6. 클러스터 콘솔 설정 설명	107
8.2.7. 펜싱 정책 설정 설명	107
8.2.8. 클러스터에서 호스트에 대한 로드 및 전원 관리 정책 설정	108
8.2.9. 클러스터의 호스트에서 MoM 정책 업데이트	111
8.2.10. CPU 프로파일 생성	112
8.2.11. CPU 프로파일 제거	113
8.2.12. 기존 Red Hat Gluster Storage 클러스터 가져오기	113
8.2.13. 호스트 추가 창에서 설정에 대한 설명	115
8.2.14. 클러스터 제거	116
8.2.15. 메모리 최적화	116
8.2.15.1. 메모리 최적화 및 메모리 과다 할당	117
8.2.15.2. 스왑 공간 및 메모리 과다 할당	118
8.2.15.3. Memory Overcommit Manager (MoM)	118
8.2.15.4. 메모리(Memory)	119
8.2.15.5. 커널 동일 페이지 병합(KSM)	120
8.2.16. 클러스터 호환성 버전 변경	121
<b>9장. 논리 네트워크</b> .....	<b>123</b>
9.1. 논리적 네트워크 작업	123
9.1.1. 네트워킹 작업 수행	123
9.1.2. 데이터 센터 또는 클러스터에서 새 논리 네트워크 생성	124
9.1.3. 논리 네트워크 편집	126
9.1.4. 논리 네트워크 제거	127
9.1.5. 기본 경로로 비관리 논리 네트워크 구성	128
9.1.6. 논리 네트워크의 게이트웨이 보기 또는 편집	129
9.1.7. 설명되는 논리적 네트워크 일반 설정	130
9.1.8. 설명되는 논리적 네트워크 클러스터 설정	131

9.1.9. 설명된 논리 네트워크 vNIC 프로필 설정	132
9.1.10. Manage Networks(네트워크 관리 창)를 사용하여 논리 네트워크의 특정 트래픽 유형 지정	132
9.1.11. 네트워크 관리 창에서 설정에 대한 설명	133
9.1.12. NIC에서 가상 기능 구성 편집	134
9.2. 가상 네트워크 인터페이스 카드	135
9.2.1. vNIC 프로필 개요	135
9.2.2. vNIC 프로파일 생성 또는 편집	136
9.2.3. VM 인터페이스 프로필 창에서 설정 설명	137
9.2.4. vNIC 프로파일에서 패스스루 활성화	139
9.2.5. vNIC 프로필 제거	140
9.2.6. vNIC 프로필에 보안 그룹 할당	141
9.2.7. vNIC 프로필에 대한 사용자 권한	142
9.2.8. UCS Integration을 위한 vNIC 프로필 구성	143
9.3. 외부 공급자 네트워크	144
9.3.1. 외부 공급자에서 네트워크 가져오기	144
9.3.2. 외부 공급자 네트워크 사용 제한 사항	145
9.3.3. 외부 공급자 논리 네트워크에서 서브넷 구성	146
9.3.4. 외부 공급자 논리 네트워크에 서브넷 추가	146
9.3.5. 외부 공급자 논리 네트워크에서 서브넷 제거	147
9.3.6. 논리 네트워크 및 포트에 보안 그룹 할당	148
9.4. 호스트 및 네트워킹	149
9.4.1. 호스트 기능 새로 고침	149
9.4.2. 호스트 네트워크 인터페이스 편집 및 호스트에 논리 네트워크 할당	149
9.4.3. 호스트 네트워크 동기화	155
9.4.4. 호스트의 VLAN 설정 편집	157
9.4.5. 논리 네트워크를 사용하여 단일 네트워크 인터페이스에 여러 VLAN 추가	158
9.4.6. 호스트 네트워크에 추가 IPv4 주소 할당	160
9.4.7. 호스트 네트워크 인터페이스에 네트워크 라벨 추가	161
9.4.8. 호스트의 FQDN 변경	164
9.4.9. IPv6 네트워킹 지원	164
9.4.10. SR-IOV 설정 및 구성	165
9.4.10.1. 사전 요구 사항	165
9.4.10.2. SR-IOV 설정 및 구성	165
9.4.10.3. 추가 리소스	167
9.5. 네트워크 본딩	167
9.5.1. 관리 포털에서 본딩 장치 생성	168
9.5.2. LLDP 라벨러 서비스를 사용하여 본딩 장치 생성	170
9.5.3. 본딩 모드	172
9.6. 네트워크 연결 분석 및 모니터링	173
9.6.1. Skydive 소개	173
9.6.2. Skydive 설치	174
9.6.3. Skydive를 사용하여 네트워크 연결 테스트	176
<b>10장. 호스트</b> .....	<b>178</b>
10.1. 호스트 소개	178
10.2. RED HAT VIRTUALIZATION HOST	179
10.3. RED HAT ENTERPRISE LINUX 호스트	180
10.4. SATELLITE 호스트 공급자 호스트	180
10.5. 호스트 작업	181
10.5.1. Red Hat Virtualization Manager에 표준 호스트 추가	181
10.5.2. Satellite 호스트 공급자 호스트 추가	182
10.5.3. 호스트에 대한 Satellite 에라타 관리 구성	184
10.5.4. 새 호스트 및 Edit Host Windows의 설정 및 제어 설명	185



10.5.5. 호스트 일반 설정 설명	185
10.5.6. 호스트 전원 관리 설정 설명	187
10.5.7. Restic 우선 순위 설정 설명	190
10.5.8. 호스트 콘솔 설정 설명	190
10.5.9. 설명된 네트워크 공급자 설정	191
10.5.10. 커널 설정 설명	191
10.5.11. 호스트 엔진 설정 설명	192
10.5.12. 호스트 전원 관리 설정 구성	193
10.5.13. 호스트 스토리지 풀 관리자 설정 구성	195
10.5.14. PCI 패스스루용 호스트 구성	196
10.5.15. 호스트를 유지 관리 모드로 전환	198
10.5.16. 유지 관리 모드에서 호스트 활성화	200
10.5.17. 호스트 방화벽 규칙 구성	201
10.5.18. 호스트 제거	201
10.5.19. 마이너 릴리스 간 호스트 업데이트	202
10.5.19.1. 클러스터의 모든 호스트 업데이트	202
10.5.19.2. 개별 호스트 업데이트	204
10.5.19.3. 호스트 수동 업데이트	206
10.5.20. 호스트 재설치	208
10.5.21. 호스트 에라타 보기	209
10.5.22. 호스트의 상태 보기	209
10.5.23. 호스트 장치 보기	210
10.5.24. 관리 포털에서 Cockpit에 액세스	211
10.5.25. 레거시 SPICE Cipher 설정	212
10.6. 호스트 복원	213
10.6.1. 호스트 고가용성	213
10.6.2. Red Hat Virtualization의 Proxy에 의한 전원 관리	214
10.6.3. 호스트에서 펜싱 매개 변수 설정	214
10.6.4. fence_kdump 고급 구성	217
10.6.4.1. fence_kdump 리스너 구성	218
10.6.4.2. 관리자에서 fence_kdump 구성	219
10.6.5. 소프트 펜싱 호스트	221
10.6.6. 호스트 전원 관리 기능 사용	222
10.6.7. 비 응답 호스트 수동 펜싱 또는 분리	223
<b>11장. 스토리지</b> .....	<b>225</b>
11.1. 스토리지 도메인 이해	226
11.2. NFS 스토리지 준비 및 추가	227
11.2.1. NFS 스토리지 준비	227
11.2.2. NFS 스토리지 추가	228
11.2.3. NFS 스토리지 증가	229
11.3. 로컬 스토리지 준비 및 추가	230
11.3.1. 로컬 스토리지 준비	230
11.3.2. 로컬 스토리지 추가	231
11.4. POSIX 호환 파일 시스템 스토리지 준비 및 추가	232
11.4.1. POSIX 호환 파일 시스템 스토리지 준비	232
11.4.2. POSIX 호환 파일 시스템 스토리지 추가	232
11.5. 블록 스토리지 준비 및 추가	234
11.5.1. iSCSI 스토리지 준비	234
11.5.2. iSCSI 스토리지 추가	235
11.5.3. iSCSI 다중 경로 구성	238
11.5.4. 논리 네트워크를 iSCSI 본딩으로 마이그레이션	239
11.5.5. FCP 스토리지 준비	241

11.5.6. FCP 스토리지 추가	242
11.5.7. iSCSI 또는 FCP 스토리지 증가	244
11.5.8. LUN 재사용	246
11.6. RED HAT GLUSTER STORAGE 준비 및 추가	246
11.6.1. Red Hat Gluster Storage 준비	247
11.6.2. Red Hat Gluster Storage 추가	247
11.7. 기존 스토리지 도메인 가져오기	247
11.7.1. 기존 스토리지 도메인 가져오기 개요	247
11.7.2. 스토리지 도메인 가져오기	248
11.7.3. 동일한 환경에서 데이터 센터 간 스토리지 도메인 마이그레이션	250
11.7.4. 다른 환경에서 데이터 센터 간 스토리지 도메인 마이그레이션	251
11.7.5. 가져온 데이터 스토리지 도메인에서 가상 머신 가져오기	254
11.7.6. 가져온 데이터 스토리지 도메인에서 템플릿 가져오기	255
11.8. 스토리지 작업	256
11.8.1. 데이터 스토리지 도메인에 이미지 업로드	256
11.8.2. 스토리지 도메인을 유지 관리 모드로 이동	258
11.8.3. 스토리지 도메인 편집	259
11.8.4. OVF 업데이트	261
11.8.5. 유지 관리 모드에서 스토리지 도메인 활성화	262
11.8.6. 데이터 센터에서 스토리지 도메인 분리	262
11.8.7. 데이터 센터에 스토리지 도메인 연결	263
11.8.8. 스토리지 도메인 제거	264
11.8.9. 스토리지 도메인 삭제	265
11.8.10. 디스크 프로필 생성	265
11.8.11. 디스크 프로필 제거	266
11.8.12. 스토리지 도메인의 상태 보기	267
11.8.13. 스토리지 도메인에 대한 삭제 후 카드 설정	267
11.8.14. 250개 이상의 호스트가 있는 환경에서 4K 지원 활성화	268
11.8.15. 4K 지원 비활성화	269
<b>12장. 풀</b>	<b>271</b>
12.1. 가상 머신 풀 소개	271
12.2. 가상 머신 풀 생성	271
12.3. 새 풀 및 편집 풀 WINDOWS의 설정 및 제어 설명	277
12.3.1. 새 풀 및 편집 풀 일반 설정 설명	278
12.3.2. 새로운 풀 및 편집 풀 유형 설정 설명	278
12.3.3. 새 풀 및 Edit Pool Console 설정 설명	279
12.3.4. 가상 머신 풀 호스트 설정 설명	279
12.3.5. 새로운 풀 및 편집 풀 리소스 설정 설명	284
12.4. 가상 머신 풀 편집	285
12.5. 풀에서 가상 머신 사전 시작	285
12.6. 가상 머신 풀에 가상 머신 추가	286
12.7. 가상 머신 풀에서 가상 머신 분리	286
12.8. 가상 머신 풀 제거	287
12.9. 신뢰할 수 있는 컴퓨팅 풀	288
12.9.1. OpenAttestation Server를 Manager에 연결	288
12.9.2. 신뢰할 수 있는 클러스터 생성	289
12.9.3. 신뢰할 수 있는 호스트 추가	290
<b>13장. 가상 디스크</b>	<b>292</b>
13.1. 가상 머신 스토리지 이해	292
13.2. 가상 디스크 이해	292
13.3. 삭제 후 가상 디스크 설정	294

13.4. RED HAT VIRTUALIZATION의 공유 가능 디스크	296
13.5. READ ONLY DISKS IN RED HAT VIRTUALIZATION	296
13.6. 가상 디스크 작업	297
13.6.1. 가상 디스크 생성	297
13.6.2. 새 가상 디스크 창에서 설정에 대한 설명	298
13.6.3. 실시간 스토리지 마이그레이션 개요	305
13.6.4. 가상 디스크 이동	305
13.6.5. 디스크 인터페이스 유형 변경	306
13.6.6. 가상 디스크 복사	308
13.6.7. 데이터 스토리지 도메인에 이미지 업로드	308
13.6.8. 가져오기 스토리지 도메인에서 디스크 이미지 가져오기	309
13.6.9. 가져온 스토리지 도메인에서 등록되지 않은 디스크 이미지 가져오기	309
13.6.10. OpenStack 이미지 서비스에서 가상 디스크 가져오기	310
13.6.11. OpenStack 이미지 서비스로 가상 디스크 내보내기	311
13.6.12. 가상 디스크 공간 회수	312
<b>14장. 외부 공급자</b> .....	<b>314</b>
14.1. RED HAT VIRTUALIZATION의 외부 공급자 소개	314
14.2. 외부 공급자 추가	315
14.2.1. 호스트 프로비저닝을 위한 Red Hat Satellite 인스턴스 추가	315
14.2.2. 이미지 관리를 위해 OpenStack Image(Glance) 인스턴스 추가	316
14.2.3. 네트워크 프로비저닝을 위해 OpenStack Networking(Neutron) 인스턴스 추가	318
14.2.4. 스토리지 관리를 위해 OpenStack Block Storage(Cinder) 인스턴스 추가	322
14.2.5. VMware 인스턴스를 가상 머신 공급자로 추가	324
14.2.6. RHEL 5ECDHE Host를 가상 머신 공급자로 추가	326
14.2.7. KVM 호스트를 가상 머신 공급자로 추가	327
14.2.8. OVN(Open Virtual Network)을 외부 네트워크 공급자로 추가	329
14.2.8.1. 새 OVN 네트워크 공급자 설치	330
14.2.8.2. 기존 OVN 네트워크 공급자 추가	332
14.2.8.3. Ansible 플레이북을 사용하여 OVN 터널 네트워크 수정	336
14.2.8.4. OVNECDHE 네트워크의 호스트 구성	337
14.2.8.5. OVN 네트워크를 물리적 네트워크에 연결	339
14.2.9. 외부 네트워크 공급자 추가	341
14.2.10. 설명되는 공급자 일반 설정 추가	343
14.2.11. 설명된 공급자 에이전트 구성 설정 추가	348
14.3. 외부 공급자 편집	349
14.4. 외부 공급자 제거	349
<b>III 부. 환경</b> .....	<b>351</b>
<b>15장. 자체 호스트 엔진</b> .....	<b>352</b>
15.1. 셀프 호스트 엔진 유지 관리	352
셀프 호스트 엔진 유지 관리 모드	352
로컬 유지 관리 설정	352
글로벌 유지 관리 설정	353
15.2. MANAGER 가상 머신	354
15.2.1. 셀프 호스트 엔진 구성 업데이트	354
15.2.2. 이메일 알림 구성	355
15.3. 추가 호스트의 셀프 호스트 엔진에 예약된 메모리 슬롯 구성	356
15.4. RED HAT VIRTUALIZATION MANAGER에 자체 호스팅 엔진 노드 추가	357
15.5. 기존 호스트를 셀프 호스트 엔진 노드로 다시 설치	358
15.6. 복구 모드에서 관리자 가상 머신 부팅	359
15.7. 자체 호스트 엔진 환경에서 호스트 제거	361
15.8. 자체 호스트 엔진 업데이트	361

글로벌 유지 관리 모드 활성화	362
Red Hat Virtualization Manager 업데이트	362
글로벌 유지 관리 모드 비활성화	364
15.9. 셸프 호스트 엔진에서 관리자의 FQDN 변경	364
<b>16장. 백업 및 마이그레이션</b>	<b>366</b>
16.1. RED HAT VIRTUALIZATION MANAGER 백업 및 복원	366
16.1.1. Red Hat Virtualization Manager 백업 - 개요	366
16.1.2. engine-backup 명령의 구문	366
16.1.3. engine-backup 명령을 사용하여 백업 생성	367
16.1.4. engine-backup 명령을 사용하여 백업 복원	369
16.1.5. 새로 고침으로 백업 복원	369
16.1.6. 기존 설치를 덮어쓰도록 백업 복원	370
16.1.7. 다른 자격 증명을 사용하여 백업 복원	372
16.1.8. 셸프 호스트 엔진 백업 및 복원	374
16.1.8.1. 원래 관리자 백업	375
16.1.8.2. 새 자체 호스트 엔진에서 백업 복원	377
16.1.8.3. Red Hat Virtualization Manager 리포지토리 활성화	381
16.1.8.4. 호스트 재설치	383
16.1.8.5. 스토리지 도메인 제거	384
16.1.9. 기존 백업에서 자체 호스팅 엔진 복구	385
16.1.9.1. 새 자체 호스트 엔진에서 백업 복원	386
16.1.9.2. Red Hat Virtualization Manager 리포지토리 활성화	390
16.1.9.3. 호스트 재설치	392
16.1.9.4. 스토리지 도메인 제거	393
16.1.10. 기존 백업에서 자체 호스트 엔진 덮어쓰기	394
16.1.10.1. 글로벌 유지 관리 모드 활성화	394
16.1.10.2. 기존 설치를 덮어쓰도록 백업 복원	395
16.1.10.3. 글로벌 유지 관리 모드 비활성화	396
16.2. RED HAT VIRTUALIZATION 데이터베이스를 원격 서버로 마이그레이션	397
16.2.1. Manager 데이터베이스를 원격 서버로 마이그레이션	397
16.2.1.1. Red Hat Virtualization Manager 리포지토리 활성화	397
Manager 데이터베이스를 원격 서버로 마이그레이션	399
16.2.2. 셸프 호스트 엔진 데이터베이스를 원격 서버로 마이그레이션	399
Red Hat Virtualization Manager 리포지토리 활성화	400
셸프 호스트 엔진 데이터베이스를 원격 서버로 마이그레이션	401
16.2.3. 분리 머신으로 데이터 마이그레이션	402
16.2.3.1. 분리 머신으로 데이터 데이터베이스 마이그레이션	402
Red Hat Virtualization Manager 리포지토리 활성화	403
분리 머신으로 데이터 데이터베이스 마이그레이션	404
16.2.3.2. DataECDHE 서비스를 분리 머신으로 마이그레이션	405
16.2.3.2.1. 새 데이터 제공 시스템 설정	406
16.2.3.2.2. 관리자 머신에서 데이터 sharing 서비스 중지	407
16.2.3.2.3. 새 데이터 제공 시스템 구성	408
16.2.3.2.4. 관리자 머신에서 데이터 hieradata 서비스 비활성화	409
16.3. 백업 스토리지 도메인을 사용하여 가상 머신 백업 및 복원	410
16.3.1. 백업 스토리지 도메인 설명	410
16.3.2. 데이터 스토리지 도메인을 백업 도메인으로 설정	412
16.3.3. 백업 도메인을 사용하여 가상 머신 또는 스냅샷 백업 또는 복원	413
16.4. 백업 및 복원 API를 사용하여 가상 머신 백업 및 복원	414
16.4.1. 백업 및 복원 API	414
16.4.2. 가상 머신 백업	414
16.4.3. 가상 머신 복원	417

<b>17장. RED HAT SATELLITE를 통한 에라타 관리</b> .....	<b>420</b>
<b>18장. ANSIBLE을 사용하여 구성 작업 자동화</b> .....	<b>422</b>
18.1. ANSIBLE 역할	422
18.1.1. Ansible 역할 설치	423
18.1.2. Ansible 역할을 사용하여 Red Hat Virtualization 구성	423
<b>19장. 사용자 및 역할</b> .....	<b>426</b>
19.1. 사용자 소개	426
19.2. 디렉터리 서버 소개	426
19.3. 외부 LDAP 공급자 구성	428
19.3.1. 외부 LDAP 공급자 구성(Interactive Setup)	428
19.3.2. Active Directory 연결	434
19.3.3. 외부 LDAP 공급자 구성(Manual Method)	438
19.3.4. 외부 LDAP 공급자 제거	441
19.4. SSO(SINGLE SIGN-ON)의 LDAP 및 KERBEROS 구성	442
19.5. RED HAT SINGLE SIGN-ON 설치 및 구성	448
19.5.1. Red Hat Single Sign-On 설치	449
19.5.2. LDAP 그룹 매핑 구성	449
19.5.3. 관리자에서 Apache 구성	451
19.5.4. OVN 구성	453
19.6. 사용자 인증	454
19.6.1. 사용자 권한 부여 모델	454
19.6.2. 사용자 작업	454
19.7. 관리 포털에서 사용자 작업	455
19.7.1. 사용자 추가 및 VM 포털 권한 할당	455
19.7.2. 사용자 정보 보기	455
19.7.3. 리소스에서 사용자 권한 보기	456
19.7.4. 사용자 제거	456
19.7.5. 로그인한 사용자 보기	457
19.7.6. 사용자 세션 종료	457
19.8. 명령줄에서 사용자 작업	457
19.8.1. 새 사용자 생성	458
19.8.2. 사용자 암호 설정	458
19.8.3. 사용자 시간 제한 설정	458
19.8.4. 사용자 암호 사전 암호화	459
19.8.5. 사용자 정보 보기	460
19.8.6. 사용자 정보 편집	460
19.8.7. 사용자 제거	460
19.8.8. 내부 사용자 비활성화	460
19.8.9. 그룹 관리	461
19.8.10. 사용자 및 그룹 쿼리	463
19.8.11. 계정 설정 관리	464
19.9. 추가 로컬 도메인 구성	465
<b>20장. 할당량 및 서비스 수준 계약 정책</b> .....	<b>466</b>
20.1. 할당량 소개	466
20.2. 공유 할당량 및 개별적으로 정의 할당량	468
20.3. 할당량 계정	468
20.4. 데이터 센터에서 할당량 모드 활성화 및 변경	469
20.5. 새 할당량 정책 생성	470
20.6. 할당량 임계값 설정에 대한 설명	471
20.7. 오브젝트에 할당량 할당	472
20.8. 할당량을 사용하여 사용자별 리소스 제한 사용	473

20.9. 할당량 편집	474
20.10. 할당량 제거	474
20.11. 서비스 수준 계약 정책 시행	474
<b>21장. 이벤트 알림</b>	<b>476</b>
21.1. 관리 포털에서 이벤트 알림 구성	476
21.2. 관리 포털에서 이벤트 알림 취소	478
21.3. OVIRT-ENGINE-NOTIFIER.CONF의 이벤트 알림 매개변수	479
21.4. SNMP TRAPS를 전송하도록 RED HAT VIRTUALIZATION MANAGER 구성	483
<b>22장. 유틸리티</b>	<b>486</b>
22.1. OVIRT ENGINE RENAME TOOL	486
22.1.1. oVirt Engine Rename Tool	486
22.1.2. oVirt Engine Rename 명령의 구문	487
22.1.3. oVirt Engine Rename Tool으로 관리자 이름 변경	488
22.2. 엔진 구성 도구	490
22.2.1. 엔진 구성 도구	490
22.2.2. engine-config 명령의 구문	490
22.3. USB 필터 편집기	491
22.3.1. USB 필터 편집기 설치	492
22.3.2. USB 필터 편집기 인터페이스	492
22.3.3. USB 정책 추가	493
22.3.4. USB 정책 제거	494
22.3.5. USB 장치 정책 검색	495
22.3.6. USB 정책 내보내기	495
22.3.7. USB 정책 가져오기	496
22.4. 로그 수집기 툴	497
22.4.1. 로그 수집기	497
22.4.2. ovirt-log-collector 명령의 구문	497
22.4.3. 기본 로그 수집기 사용량	501
22.5. ISO 업로더 툴	501
22.5.1. ISO 업로더 툴	501
22.5.2. engine-iso-uploader 명령의 구문	502
22.5.3. NFS 서버 지정	505
22.5.4. 기본 ISO 업로더 사용	505
22.5.5. ISO 스토리지 도메인에 VirtIO 및 게스트 툴 이미지 파일 업로드	505
22.6. 엔진 VACUUM 툴	507
22.6.1. 엔진 Vacuum 툴	507
22.6.2. 엔진 Vacuum 모드	507
22.6.3. engine-vacuum 명령의 구문	508
22.7. VDSM에서 네트워크 이름 매핑 툴	509
22.7.1. VDSM 이름을 논리 네트워크 이름에 매핑	509
<b>IV 부. 환경에 대한 정보 수집</b>	<b>511</b>
<b>23장. RED HAT VIRTUALIZATION MANAGER에 INSIGHTS 배포</b>	<b>512</b>
<b>24장. 로그 파일</b>	<b>514</b>
24.1. 관리자 설치 로그 파일	514
24.2. RED HAT VIRTUALIZATION MANAGER 로그 파일	514
24.3. SPICE 로그 파일	515
24.3.1. Hypervisor SPICE Server를 위한 SPICE 로그	515
24.3.2. 게스트 머신용 SPICE 로그	515
24.3.3. console.vv 파일을 사용하여 시작된 SPICE 클라이언트용 SPICE 로그	516

24.4. 호스트 로그 파일	517
24.5. 호스트 로깅 서버 설정	517
24.6. OVIRT ENGINE EXTENSION LOGGER LOG4J 활성화	519
<b>부록 A. VDSM 및 후크</b>	<b>522</b>
A.1. VDSM	522
A.2. VDSM 후크	522
A.3. HOOKS를 사용하여 VDSM 확장	522
A.4. 지원되는 VDSM 이벤트	522
A.5. VDSM 후크 환경	525
A.6. VDSM 후크 도메인 XML 오브젝트	525
A.7. 사용자 정의 속성 정의	526
A.8. 가상 머신 사용자 정의 속성 설정	528
A.9. VDSM 후크에서 가상 머신 사용자 정의 속성 평가	528
A.10. VDSM 후크 모듈 사용	529
A.11. VDSM 후크 실행	529
A.12. VDSM 후크 반환 코드	530
A.13. VDSM 후크 예	531
<b>부록 B. 사용자 정의 네트워크 속성</b>	<b>534</b>
B.1. BRIDGE_OPTS 매개변수 설명	534
B.2. RED HAT VIRTUALIZATION MANAGER를 ETHTOOL을 사용하도록 설정하는 방법	536
B.3. FCOE를 사용하도록 RED HAT VIRTUALIZATION MANAGER 설정 방법	537
<b>부록 C. RED HAT VIRTUALIZATION 사용자 인터페이스 플러그인</b>	<b>539</b>
C.1. RED HAT VIRTUALIZATION 사용자 인터페이스 플러그인	539
C.2. RED HAT VIRTUALIZATION USER INTERFACE PLUGIN LIFECYCLE	539
C.2.1. Red Hat Virtualization 사용자 인터페이스 플러그인 라이프 사이클	539
C.2.2. Red Hat Virtualization 사용자 인터페이스 플러그인 검색	539
C.2.3. Red Hat Virtualization 사용자 인터페이스 플러그인 로드	540
C.2.4. Red Hat Virtualization 사용자 인터페이스 플러그인 부트 스트랩	540
C.3. 사용자 인터페이스 플러그인 관련 파일 및 해당 위치	541
C.4. 사용자 인터페이스 플러그인 배포 예	542
<b>부록 D. RED HAT VIRTUALIZATION 및 암호화된 통신</b>	<b>544</b>
D.1. RED HAT VIRTUALIZATION MANAGER CA 인증서 교체	544
D.2. 관리자와 LDAP 서버 간 암호화 통신 설정	548
D.3. VDSM을 위해 수동으로 암호화된 통신 설정 절차	549 550
<b>부록 E. 프록시</b>	<b>551</b>
E.1. SPICE 프록시	551
E.1.1. SPICE 프록시 개요	551
E.1.2. SPICE 프록시 머신 설정	551
E.1.3. SPICE 프록시 켜기	552
E.1.4. SPICE 프록시 끄기	553
E.2. SQUID PROXY	554
E.2.1. Squid 프록시 설치 및 구성	554
E.3. WEBSOCKET 프록시	557
E.3.1. WebSocket 프록시 개요	557
E.3.2. Websocket 프록시를 분리 머신으로 마이그레이션 Manager 시스템에서 Websocket 프록시 제거 별도의 컴퓨터에 Websocket 프록시 설치	558 558 559
<b>부록 F. 브랜딩</b>	<b>562</b>

F.1. 브랜딩	562
F.1.1. 관리자의 다시 시작	562
F.1.2. 로그인 화면	562
F.1.3. 관리 포털 화면	562
F.1.4. VM 포털 화면	563
F.1.5. 팝업 Windows	564
F.1.6. 탭	564
F.1.7. The welcome page	564
F.1.8. 페이지를 찾을 수 없음 페이지	565
<b>부록 G. 시스템 계정</b> .....	<b>567</b>
G.1. 시스템 계정	567
G.1.1. Red Hat Virtualization Manager 사용자 계정	567
G.1.2. Red Hat Virtualization Manager 그룹	567
G.1.3. 가상화 호스트 사용자 계정	568
G.1.4. 가상화 호스트 그룹	568





## I 부. RED HAT VIRTUALIZATION 환경 유지 및 유지 관리

Red Hat Virtualization 환경을 사용하려면 관리자가 이를 계속 실행해야 합니다. 관리자는 다음이 포함됩니다.

- 호스트 및 가상 시스템과 같은 물리적 리소스 및 가상 리소스 관리. 여기에는 호스트 업그레이드 및 추가, 도메인 가져오기, 외부 하이퍼바이저에서 생성된 가상 머신 변환, 가상 머신 풀 관리가 포함됩니다.
- 호스트 중 하나에서 과도한 부하, 메모리 또는 디스크 공간이 부족하고 필요한 작업을 수행하는 등 전체 시스템 리소스를 모니터링하여 다른 호스트로 가상 머신을 마이그레이션하여 시스템을 종료하여 리소스를 줄일 수 있습니다.
- 가상 머신의 새로운 요구 사항에 대응(예: 운영 체제를 업그레이드하거나 더 많은 메모리를 할당).
- 태그를 사용하여 사용자 지정 오브젝트 속성 관리.
- [공용 북마크](#) 로 저장된 검색 관리
- 사용자 설정 관리 및 권한 수준 설정.
- 전체 시스템 기능을 위해 특정 사용자 또는 가상 머신 문제 해결
- 일반 및 특정 보고서를 생성합니다.

## 1장. 글로벌 구성

**Administration → Configure** ) 창을 클릭하면 사용자, 역할, 시스템 권한, 스케줄링 정책, 인스턴스 유형, MAC 주소 풀과 같은 Red Hat Virtualization 환경에 대한 여러 글로벌 리소스를 구성할 수 있습니다. 이 창을 사용하면 사용자가 환경의 리소스와 상호 작용하는 방법을 사용자 정의하고 여러 클러스터에 적용할 수 있는 옵션을 구성하기 위한 중앙 위치를 제공할 수 있습니다.

### 1.1. 역할

역할은 **Red Hat Virtualization Manager**에서 구성할 수 있는 사전 정의된 권한 집합입니다. 역할은 데이터 센터의 다양한 리소스 수준과 특정 물리적 및 가상 리소스에 액세스 및 관리 권한을 제공합니다.

다중 수준 관리를 사용하면 컨테이너 오브젝트에 적용되는 모든 권한이 해당 컨테이너 내의 모든 개별 오브젝트에도 적용됩니다. 예를 들어, 호스트 관리자 역할이 특정 호스트의 사용자에게 할당되면 사용자는 할당된 호스트에서만 사용 가능한 호스트 작업을 수행할 수 있는 권한을 얻습니다. 그러나 호스트 관리자 역할이 데이터 센터의 사용자에게 할당되면 사용자는 데이터 센터 클러스터 내의 모든 호스트에서 호스트 작업을 수행할 수 있는 권한을 얻습니다.

#### 1.1.1. 새 역할 생성

필요한 역할이 **Red Hat Virtualization**의 기본 역할 목록에 없는 경우 새 역할을 생성하고 목적에 맞게 사용자 지정할 수 있습니다.

새 역할 생성

1. **Administration → Configure** (관리 구성)를 클릭하여 **Configure** (구성) 창을 엽니다. 역할 탭은 기본적으로 선택되어 기본 사용자 및 관리자 역할 목록과 사용자 지정 역할 목록을 표시합니다.
2. 새로 생성을 클릭합니다.
3. 새 역할의 이름 및 설명을 입력합니다.
4. **Admin** 또는 **User** 를 계정 유형으로 선택합니다.
5. 모두 확장 또는 모두 확장버튼을 사용하여 **Check Boxes to Allow Action** 목록에서 나열된 개체에 대한 권한을 더 많거나 적게 볼 수 있습니다. 또한 각 개체의 옵션을 확장하거나 축소할 수 있습니다.
6. 각 오브젝트에 대해 설정 중인 역할에 대해 허용 또는 거부하려는 작업을 선택하거나 지웁니다.
7. 확인을 클릭하여 변경 사항을 적용합니다. 새 역할이 역할 목록에 표시됩니다.

#### 1.1.2. 역할 편집 또는 복사

생성한 역할의 설정은 변경할 수 있지만 기본 역할은 변경할 수 없습니다. 기본 역할을 변경하려면 요구 사항에 맞게 복제 및 수정합니다.

역할 편집 또는 복사

1. **Administration → Configure** (관리 구성)를 클릭하여 **Configure** (구성) 창을 엽니다. 창에 기본 사용자 및 관리자 역할 목록과 사용자 지정 역할 목록이 표시됩니다.
2. 변경할 역할을 선택합니다. **Edit** (편집)를 클릭하여 역할 편집 창을 열거나 복사를 클릭하여 역할 복사 창을 엽니다.
3. 필요한 경우 역할의 이름 및 설명을 편집합니다.

4. 모두 확장 또는 모두 확장버튼을 사용하여 나열된 오브젝트에 대한 사용 권한을 더 많거나 적게 볼 수 있습니다. 또한 각 개체의 옵션을 확장하거나 축소할 수 있습니다.
5. 각 오브젝트에 대해 편집 중인 역할에 대해 허용 또는 거부하려는 작업을 선택하거나 지웁니다.
6. 확인 을 클릭하여 변경한 내용을 적용합니다.**Click OK to apply the changes you have made.**

### 1.1.3. 사용자 역할 및 권한 부여 예

다음 예제에서는 이 장에서 설명하는 권한 부여 시스템의 다양한 기능을 사용하여 다양한 시나리오에 권한 부여 제어를 적용하는 방법을 보여줍니다.

#### 예 1.1. 클러스터 권한

**Sarah**는 회사의 계정 부서의 시스템 관리자입니다. 부서의 모든 가상 리소스는 **Accounts** 라는 **Red Hat Virtualization** 클러스터에서 구성됩니다. **he**에는 **accounts** 클러스터에 대한 **ClusterAdmin** 역할이 할당됩니다. 이를 통해 가상 머신이 클러스터의 하위 오브젝트이므로 클러스터의 모든 가상 머신을 관리할 수 있습니다. 가상 머신 관리에는 디스크와 같은 가상 리소스 편집, 추가, 제거, 스냅샷이 포함됩니다. 따라서 클러스터 외부의 모든 리소스를 관리할 수 없습니다. **ClusterAdmin** 은 관리자 역할이므로 관리 포털 또는 **VM** 포털을 사용하여 이러한 리소스를 관리할 수 있습니다.

#### 예 1.2. VM PowerUser 권한

**BOOM**은 계정 부서의 소프트웨어 개발자입니다. 가상 머신을 사용하여 소프트웨어를 빌드하고 테스트합니다. **Sarah**는 **johnECDHE**라는 가상 데스크탑 을 생성했습니다. **BOOM**에는 **john ECDHE** 가상 시스템에 대한 **UserVmManager** 역할이 할당됩니다. 이렇게 하면 **VM** 포털을 사용하여 이 단일 가상 머신에 액세스할 수 있습니다. **UserVmManager** 권한이 있으므로 가상 시스템을 수정할 수 있습니다. **UserVmManager** 는 사용자 역할이므로 관리 포털을 사용하도록 허용하지 않습니다.

#### 예 1.3. 데이터 센터 전원 사용자 역할 권한

**Penelope**는 사무실 관리자입니다. 자신의 책임이 있을 뿐만 아니라, 잠시 후사나이는 때론 조사 일정을 예약하고 참고 검사를 수행하는 것과 같은 **task**를 가진 **BOOM** 관리자에게 도움을 줄 수 있습니다. 기업 정책에 따라 **Penelope**는 작업에 특정 애플리케이션을 사용해야 합니다.

**Penelope**에는 사무실 관리 작업을 위한 자체 시스템이 있지만, 이 애플리케이션은 실행할 별도의 가상 머신을 만들고자 합니다. **he**는 새 가상 머신이 상주하는 데이터 센터에 대한 **ECDHE** 권한이 할당됩니다. 이는 새 가상 머신을 생성하기 때문에 스토리지 도메인에서 가상 디스크 생성을 포함하여 데이터 센터 내의 여러 구성 요소를 변경해야 하기 때문입니다.

이는 **Penelope**에 **DataCenterAdmin** 권한을 할당하는 것과 동일하지 않습니다. 데이터 센터의 **PowerUser**로 **Penelope**는 **VM** 포털에 로그인하여 데이터 센터 내의 가상 머신에 대한 가상 머신 관련 작업을 수행할 수 있습니다. 데이터 센터에 호스트 또는 스토리지 연결과 같은 데이터 센터 수준 작업을 수행할 수 없습니다.

#### 예 1.4. 네트워크 관리자 권한

**Chris**는 IT 부서의 네트워크 관리자로 작업하고 있습니다. 부서의 **Red Hat Virtualization** 환경에서 네트워크를 생성, 조작 및 제거하는 작업을 담당합니다. 역할에 대해 리소스 및 각 리소스의 네트워크에 대한 관리 권한이 필요합니다. 예를 들어 **Chris**가 IT 부서의 데이터 센터에 **NetworkAdmin** 권한이 있는 경우 데이터 센터에서 네트워크를 추가 및 제거하고 데이터 센터에 속한 모든 가상 머신의 네트워크를 연결 및 분리할 수 있습니다.

### 예 1.5. 사용자 정의 역할 권한

**Rachel**은 IT 부서에서 일하며 **Red Hat Virtualization**에서 사용자 계정을 관리합니다. 사용자 계정을 추가하고 적절한 역할 및 권한을 할당할 수 있는 권한이 필요합니다. **he**는 가상 머신을 사용하지 않으며 호스트, 가상 머신, 클러스터 또는 데이터 센터 관리에 액세스할 수 없어야 합니다. 이 특정 권한 세트를 제공하는 기본 제공 역할은 없습니다. **Rachel**의 위치에 적합한 권한 집합을 정의하려면 사용자 지정 역할을 생성해야 합니다.

그림 1.1. UserManager 사용자 정의 역할

The screenshot shows a 'New Role' dialog box with the following details:

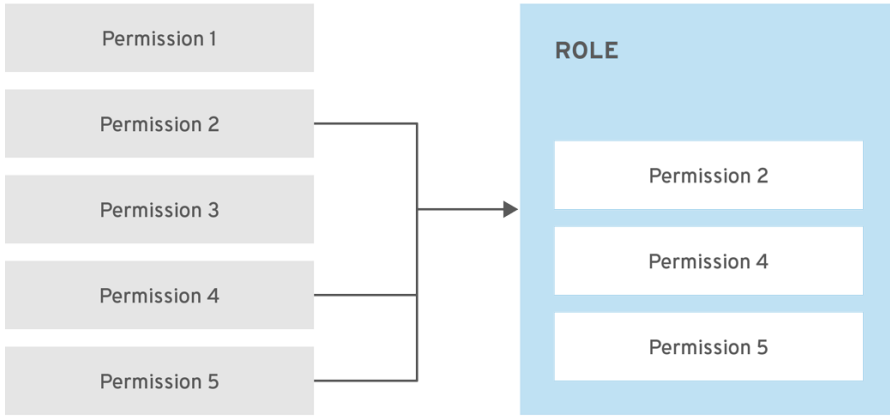
- Name:** UserManager
- Description:** (Empty field)
- Account Type:** Admin (Selected)
- Check Boxes to Allow Action:** Expand All, Collapse All
- System:**
  - Configure System
    - Manipulate Users (Checked)
    - Manipulate Permissions (Checked)
    - Add users and groups from directory while adding permissions (Unchecked)
    - Manipulate Roles (Checked)
    - Login Permissions (Unchecked)
    - Tag management Permissions (Unchecked)
    - Bookmark management Permissions (Checked)

위에 표시된 **UserManager** 사용자 지정 역할은 사용자, 권한 및 역할을 조작할 수 있습니다. 이러한 작업은 [그림 1.3. "Red Hat Virtualization 오브젝트 계층 구조"](#)에 표시된 계층 구조의 최상위 오브젝트인 **System** 하에서 구성됩니다. 즉, 시스템의 다른 모든 오브젝트에 적용됩니다. 역할은 계정 유형 **admin**으로 설정되어 있습니다. 즉, 서비스에 이 역할이 할당되면 **Rachel**에서 관리 포털과 **VM** 포털을 모두 사용할 수 있습니다.

## 1.2. 시스템 권한

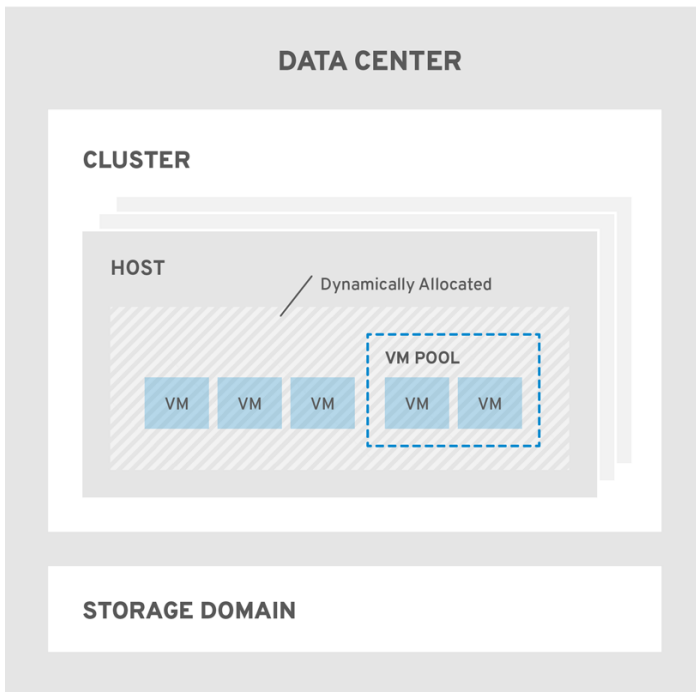
권한을 통해 사용자는 오브젝트가 개별 오브젝트 또는 컨테이너 오브젝트인 오브젝트에 대한 작업을 수행할 수 있습니다. 컨테이너 오브젝트에 적용되는 모든 권한은 해당 컨테이너의 모든 멤버에도 적용됩니다.

그림 1.2. 권한 및 역할



RHV\_453537\_0219

그림 1.3. Red Hat Virtualization 오브젝트 계층 구조



RHV\_453537\_0219

### 1.2.1. 사용자 속성

역할 및 권한은 사용자의 속성입니다. 역할은 다양한 수준의 물리적 및 가상 리소스에 대한 액세스를 허용하는 사전 정의된 권한 집합입니다. 다중 수준 관리에서는 권한의 세분화된 계층을 제공합니다. 예를 들어 데이터 센터 관리자에게는 데이터 센터의 모든 오브젝트를 관리할 수 있는 권한이 있으며 호스트 관리자는 단일 물리적 호스트에 대한 시스템 관리자 권한이 있습니다. 사용자는 단일 가상 머신을 사용할 수 있는 권한이 있지만 가상 머신 구성을 변경하지 않는 반면 다른 사용자는 가상 머신에 시스템 권한을 할당할 수 있습니다.

### 1.2.2. 사용자 및 관리자 역할

Red Hat Virtualization은 시스템 전체 권한이 있는 관리자부터 단일 가상 머신에 액세스할 수 있는 최종 사용자에게 다양한 사전 구성된 역할을 제공합니다. 기본 역할을 변경하거나 제거할 수는 없지만 요구 사항에 따라 복제 및 사용자 지정하거나 새 역할을 생성할 수 있습니다. 역할에는 다음 두 가지 유형이 있습니다.

- 관리자 역할: 관리 포털에 액세스하여 물리적 및 가상 리소스를 관리할 수 있습니다. 관리자 역할은 VM 포털에서 수행할 작업을 제한하지만, 사용자가 VM 포털에서 확인할 수 있는 권한은 없습니다.

- 사용자 역할: 가상 머신 및 템플릿을 관리하고 액세스하기 위해 VM 포털에 액세스할 수 있습니다. 사용자 역할에 따라 VM 포털에서 사용자가 볼 수 있는 내용이 결정됩니다. 관리자 역할을 가진 사용자에게 부여된 권한은 VM 포털에서 해당 사용자에게 사용 가능한 작업에 반영됩니다.

### 1.2.3. 설명되는 사용자 역할

아래 표에서는 VM 포털에서 가상 머신에 액세스하고 구성할 수 있는 권한을 부여하는 기본 사용자 역할을 설명합니다.

표 1.1. Red Hat Virtualization 사용자 역할 - 기본

역할	권한	참고
UserRole	가상 머신 및 풀에 액세스하고 사용할 수 있습니다.	VM 포털에 로그인하고 할당된 가상 머신 및 풀을 사용하고 가상 시스템 상태 및 세부 정보를 볼 수 있습니다.
PowerUserRole	가상 머신 및 템플릿을 생성하고 관리할 수 있습니다.	<b>Configure</b> 창이 있거나 특정 데이터 센터 또는 클러스터에 대해 전체 환경의 사용자에게 이 역할을 적용합니다. 예를 들어 데이터 센터 수준에 ECDHE가 적용되는 경우 PowerUser는 데이터 센터에서 가상 머신 및 템플릿을 생성할 수 있습니다.
UserVmManager	가상 머신의 시스템 관리자.	가상 머신을 관리하고 스냅샷을 생성하고 사용할 수 있습니다. VM 포털에서 가상 시스템을 생성하는 사용자에게는 시스템에 대한 UserVmManager 역할이 자동으로 할당됩니다.

아래 표에서는 VM 포털에서 리소스에 대한 권한을 더 세밀하게 조정할 수 있는 고급 사용자 역할을 설명합니다.

표 1.2. Red Hat Virtualization 사용자 역할 - 고급

역할	권한	참고
UserTemplateBasedVm	템플릿만 사용하도록 제한된 권한입니다.	템플릿을 사용하여 가상 머신을 생성할 수 있습니다.
DiskOperator	가상 디스크 사용자.	가상 디스크를 사용, 보기 및 편집할 수 있습니다. 가상 디스크가 연결된 가상 머신을 사용하는 권한을 상속합니다.

역할	권한	참고
VmCreator	VM 포털에서 가상 머신을 생성할 수 있습니다.	이 역할은 특정 가상 시스템에 적용되지 않습니다. <b>Configure</b> 창이 있는 전체 환경에 대해 이 역할을 사용자에게 적용합니다. 또는 특정 데이터 센터 또는 클러스터에 이 역할을 적용합니다. 이 역할을 클러스터에 적용할 때 전체 데이터 센터 또는 특정 스토리지 도메인에 DiskCreator 역할도 적용해야 합니다.
TemplateCreator	할당된 리소스 내에서 가상 머신 템플릿을 생성, 편집, 관리 및 제거할 수 있습니다.	이 역할은 특정 템플릿에 적용되지 않습니다. <b>Configure</b> 창을 사용하여 전체 환경에 대해 이 역할을 사용자에게 적용합니다. 또는 특정 데이터 센터, 클러스터 또는 스토리지 도메인에 이 역할을 적용합니다.
DiskCreator	할당된 클러스터 또는 데이터 센터 내에서 가상 디스크를 생성, 편집, 관리 및 제거할 수 있습니다.	이 역할은 특정 가상 디스크에 적용되지 않습니다. <b>Configure</b> 창이 있는 전체 환경에 대해 이 역할을 사용자에게 적용합니다. 또는 특정 데이터 센터 또는 스토리지 도메인에 이 역할을 적용합니다.
TemplateOwner	템플릿을 편집하고 삭제하고 템플릿에 대한 사용자 권한을 할당하고 관리할 수 있습니다.	이 역할은 템플릿을 생성하는 사용자에게 자동으로 할당됩니다. 템플릿에 대한 <b>TemplateOwner</b> 권한이 없는 다른 사용자는 템플릿을 보거나 사용할 수 없습니다.
VnicProfileUser	가상 시스템 및 템플릿용 논리적 네트워크 및 네트워크 인터페이스 사용자입니다.	특정 논리 네트워크에서 네트워크 인터페이스를 연결하거나 분리할 수 있습니다.

### 1.2.4. 설명된 관리자 역할

아래 표에서는 관리 포털에서 리소스에 액세스하고 구성할 수 있는 권한을 부여하는 기본 관리자 역할을 설명합니다.

표 1.3. Red Hat Virtualization System Administrator Roles - Basic

역할	권한	참고
슈퍼 유저	Red Hat Virtualization 환경의 시스템 관리자.	모든 오브젝트와 수준에 대한 전체 권한이 있어야 모든 데이터 센터의 모든 오브젝트를 관리할 수 있습니다.



역할	권한	참고
ClusterAdmin	클러스터 관리자.	특정 클러스터 아래의 모든 오브젝트에 대한 관리 권한이 있어야 합니다.
DataCenterAdmin	데이터 센터 관리자.	스토리지를 제외한 특정 데이터 센터 아래에 있는 모든 오브젝트에 대한 관리 권한이 있어야 합니다.



### 중요

디렉터리 서버에 대한 관리 사용자를 **Red Hat Virtualization** 관리 사용자로 사용하지 마십시오. 특히 **Red Hat Virtualization** 관리 사용자로 사용하기 위해 디렉터리 서버에 사용자를 생성합니다.

아래 표에서는 관리 포털에서 리소스에 대한 권한을 더 세밀하게 조정할 수 있는 고급 관리자 역할을 설명합니다.

표 1.4. Red Hat Virtualization System Administrator Roles - Advanced

역할	권한	참고
TemplateAdmin	가상 머신 템플릿의 관리자입니다.	템플릿의 스토리지 도메인 및 네트워크 세부 정보를 생성, 삭제 및 구성하고 도메인 간에 템플릿을 이동할 수 있습니다.
StorageAdmin	스토리지 관리자.	할당된 스토리지 도메인을 생성, 삭제, 구성 및 관리할 수 있습니다.
HostAdmin	호스트 관리자.	특정 호스트를 연결, 제거, 구성 및 관리할 수 있습니다.
NetworkAdmin	네트워크 관리자.	특정 데이터 센터 또는 클러스터의 네트워크를 구성하고 관리할 수 있습니다. 데이터 센터 또는 클러스터의 네트워크 관리자는 클러스터 내의 가상 풀에 대한 네트워크 권한을 상속합니다.
VmPoolAdmin	가상 풀의 시스템 관리자.	가상 풀을 생성, 삭제 및 구성하고, 가상 풀 사용자를 할당 및 제거하고, 풀의 가상 머신에서 기본 작업을 수행할 수 있습니다.
GlusterAdmin	Gluster Storage 관리자.	Gluster 스토리지 볼륨을 생성, 삭제, 구성 및 관리할 수 있습니다.

역할	권한	참고
VmlImporterExporter	가상 머신의 가져오기 및 내보내기 관리자입니다.	가상 머신을 가져오고 내보낼 수 있습니다. 다른 사용자가 내보낸 모든 가상 머신 및 템플릿을 볼 수 있습니다.

### 1.2.5. 리소스에 관리자 또는 사용자 역할 할당

리소스에 관리자 또는 사용자 역할을 할당하여 사용자가 해당 리소스에 액세스하거나 관리할 수 있습니다.

리소스에 역할 할당

1. 리소스 이름을 찾아서 클릭하여 세부 정보 보기를 엽니다.
2. 권한 탭을 클릭하여 할당된 사용자, 사용자의 역할 및 선택한 리소스에 대한 상속된 권한을 나열합니다.
3. 추가를 클릭합니다.
4. 기존 사용자의 이름 또는 사용자 이름을 검색 텍스트 상자에 입력하고 **Go** 를 클릭합니다. 결과 일치 목록에서 사용자를 선택합니다.
5. **Role to Assign** 드롭다운 목록에서 역할을 선택합니다.
6. **OK**를 클릭합니다.

이제 사용자에게 해당 리소스에 대해 활성화된 해당 역할의 상속된 권한이 있습니다.

### 1.2.6. 리소스에서 관리자 또는 사용자 역할 제거

리소스에서 관리자 또는 사용자 역할을 제거합니다. 사용자는 해당 리소스에 대한 역할과 연관된 상속된 권한이 손실됩니다.

리소스에서 역할 제거

1. 리소스 이름을 찾아서 클릭하여 세부 정보 보기를 엽니다.
2. 권한 탭을 클릭하여 할당된 사용자, 사용자의 역할 및 선택한 리소스에 대한 상속된 권한을 나열합니다.
3. 리소스에서 제거할 사용자를 선택합니다.
4. 제거를 클릭합니다.
5. **OK**를 클릭합니다.

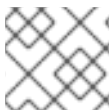
### 1.2.7. 데이터 센터의 시스템 권한 관리

슈퍼 유저로서 시스템 관리자는 관리 포털의 모든 측면을 관리합니다. 보다 구체적인 관리 역할을 다른 사용자에게 할당할 수 있습니다. 이러한 제한된 관리자 역할은 특정 리소스로 제한하는 사용자 관리 권한을 부여하는 데 유용합니다. 예를 들어 **DataCenterAdmin** 역할에는 해당 데이터 센터의 스토리지를 제외하고 할당된 데이터 센터에 대해서만 관리자 권한이 있으며 **ClusterAdmin**에는 할당된 클러스터에 대해서만 관리자 권한이 있습니다.

데이터 센터 관리자는 특정 데이터 센터에 대해서만 시스템 관리 역할입니다. 이는 각 데이터 센터에 관리자가 필요한 여러 데이터 센터가 있는 가상화 환경에서 유용합니다. **DataCenterAdmin** 역할은 계층적 모델입니다. 데이터 센터에 대해 할당된 데이터 센터 관리자 역할이 할당된 사용자는 해당 데이터 센터의 스토리지를 제외하고 데이터 센터의 모든 오브젝트를 관리할 수 있습니다. 헤더 표시줄의 **Configure** 버튼을 사용하여 환경의 모든 데이터 센터에 데이터 센터 관리자를 할당합니다.

데이터 센터 관리자 역할은 다음 작업을 허용합니다.

- 데이터 센터와 연결된 클러스터를 생성하고 제거합니다.
- 데이터 센터와 연결된 호스트, 가상 시스템 및 풀을 추가하고 제거합니다.
- 데이터 센터와 연결된 가상 머신에 대한 사용자 권한을 편집합니다.



## 참고

기본 사용자에게 역할 및 권한만 할당할 수 있습니다.

기본 시스템 관리자를 제거하고 새 시스템 관리자를 추가하여 데이터 센터의 시스템 관리자를 변경할 수 있습니다.

### 1.2.8. 데이터 센터 관리자 역할 설명

데이터 센터 권한 역할

아래 표에서는 데이터 센터 관리에 적용되는 관리자 역할 및 권한을 설명합니다.

표 1.5. Red Hat Virtualization System Administrator Roles

역할	권한	참고
DataCenterAdmin	데이터 센터 관리자	클러스터, 호스트, 템플릿 및 가상 머신 등 스토리지를 제외하고 특정 데이터 센터 내의 모든 물리적 및 가상 리소스를 사용, 생성, 삭제, 관리할 수 있습니다.
NetworkAdmin	네트워크 관리자	특정 데이터 센터의 네트워크를 구성하고 관리할 수 있습니다. 데이터 센터의 네트워크 관리자는 데이터 센터 내의 가상 머신에 대한 네트워크 권한도 상속합니다.

### 1.2.9. 클러스터의 시스템 권한 관리

슈퍼 유저로서 시스템 관리자는 관리 포털의 모든 측면을 관리합니다. 보다 구체적인 관리 역할을 다른 사용자에게 할당할 수 있습니다. 이러한 제한된 관리자 역할은 특정 리소스로 제한하는 사용자 관리 권한을 부여하는 데 유용합니다. 예를 들어 **DataCenterAdmin** 역할에는 해당 데이터 센터의 스토리지를 제외하고 할당된 데이터 센터에 대해서만 관리자 권한이 있으며 **ClusterAdmin**에는 할당된 클러스터에 대해서만 관리자 권한이 있습니다.

클러스터 관리자는 특정 클러스터에 대해서만 시스템 관리 역할입니다. 이는 각 클러스터에 시스템 관리자가 필요한 여러 클러스터가 있는 데이터 센터에서 유용합니다. **ClusterAdmin** 역할은 계층적 모델입니다. 클

러스터에 대해 클러스터 관리자 역할이 할당된 사용자는 클러스터의 모든 오브젝트를 관리할 수 있습니다. 헤더 표시줄의 **Configure** 버튼을 사용하여 환경의 모든 클러스터에 클러스터 관리자를 할당합니다.

클러스터 관리자 역할은 다음 작업을 허용합니다.

- 관련 클러스터를 생성하고 제거합니다.
- 클러스터와 연결된 호스트, 가상 시스템 및 풀을 추가하고 제거합니다.
- 클러스터와 연결된 가상 머신에 대한 사용자 권한을 편집합니다.



**참고**

기존 사용자에게 역할 및 권한만 할당할 수 있습니다.

기존 시스템 관리자를 제거하고 새 시스템 관리자를 추가하여 클러스터의 시스템 관리자를 변경할 수도 있습니다.

**1.2.10. 설명된 클러스터 관리자 역할**

클러스터 권한 역할

아래 표에는 클러스터 관리에 적용되는 관리자 역할과 권한이 설명되어 있습니다.

**표 1.6. Red Hat Virtualization System Administrator Roles**

역할	권한	참고
ClusterAdmin	클러스터 관리자	호스트, 템플릿 및 가상 머신을 포함하여 특정 클러스터의 모든 물리적 및 가상 리소스를 사용, 생성, 삭제, 관리할 수 있습니다. 디스플레이 네트워크 지정 또는 필수로 표시하는 등 클러스터 내에서 네트워크 속성을 구성할 수 있습니다.  그러나 <b>ClusterAdmin</b> 에는 클러스터에서 네트워크를 연결하거나 분리할 수 있는 권한이 없으므로 <b>NetworkAdmin</b> 권한이 필요합니다.
NetworkAdmin	네트워크 관리자	특정 클러스터의 네트워크를 구성하고 관리할 수 있습니다. 클러스터의 네트워크 관리자는 클러스터 내의 가상 머신에 대한 네트워크 권한도 상속합니다.

**1.2.11. 네트워크에 대한 시스템 권한 관리**

슈퍼 유저로서 시스템 관리자는 관리 포털의 모든 측면을 관리합니다. 보다 구체적인 관리 역할을 다른 사용자에게 할당할 수 있습니다. 이러한 제한된 관리자 역할은 특정 리소스로 제한하는 사용자 관리 권한을 부여하는 데 유용합니다. 예를 들어 **DataCenterAdmin** 역할에는 해당 데이터 센터의 스토리지를 제외하고 할당

된 데이터 센터에 대해서만 관리자 권한이 있으며 **ClusterAdmin**에는 할당된 클러스터에 대해서만 관리자 권한이 있습니다.

네트워크 관리자는 특정 네트워크 또는 데이터 센터, 클러스터, 호스트, 가상 시스템 또는 템플릿의 모든 네트워크에 적용할 수 있는 시스템 관리 역할입니다. 네트워크 사용자는 특정 가상 머신 또는 템플릿에서 네트워크 보기 및 연결과 같은 제한된 관리 역할을 수행할 수 있습니다. 헤더 표시줄의 **Configure** 버튼을 사용하여 환경의 모든 네트워크에 대해 네트워크 관리자를 할당할 수 있습니다.

네트워크 관리자 역할은 다음 작업을 허용합니다.

- 네트워크를 생성, 편집 및 제거합니다.
- 포트 미러링 구성을 포함하여 네트워크의 구성을 편집합니다.
- 클러스터 및 가상 머신을 포함한 리소스에서 네트워크 연결 및 분리.

네트워크를 생성하는 사용자에게는 생성된 네트워크에 대한 **NetworkAdmin** 권한이 자동으로 할당됩니다. 기존 관리자를 제거하고 새 관리자를 추가하여 네트워크 관리자를 변경할 수도 있습니다.

## 1.2.12. 네트워크 관리자 및 사용자 역할 설명

네트워크 권한 역할

아래 표에서는 네트워크 관리에 적용되는 관리자 및 사용자 역할 및 권한을 설명합니다.

표 1.7. Red Hat Virtualization Network 관리자 및 사용자 역할

역할	권한	참고
NetworkAdmin	데이터 센터, 클러스터, 호스트, 가상 시스템 또는 템플릿용 네트워크 관리자. 네트워크를 생성하는 사용자에게는 생성된 네트워크에 대한 <b>NetworkAdmin</b> 권한이 자동으로 할당됩니다.	특정 데이터 센터, 클러스터, 호스트, 가상 시스템 또는 템플릿의 네트워크를 구성하고 관리할 수 있습니다. 데이터 센터 또는 클러스터의 네트워크 관리자는 클러스터 내의 가상 풀에 대한 네트워크 권한을 상속합니다. 가상 머신 네트워크에서 포트 미러링을 구성하려면 네트워크에 <b>NetworkAdmin</b> 역할을 적용하고 가상 머신에 <b>UserVmManager</b> 역할을 적용합니다.
VnicProfileUser	가상 시스템 및 템플릿용 논리적 네트워크 및 네트워크 인터페이스 사용자입니다.	특정 논리 네트워크에서 네트워크 인터페이스를 연결하거나 분리할 수 있습니다.

## 1.2.13. 호스트의 시스템 권한 관리

슈퍼 유저로서 시스템 관리자는 관리 포털의 모든 측면을 관리합니다. 보다 구체적인 관리 역할을 다른 사용자에게 할당할 수 있습니다. 이러한 제한된 관리자 역할은 특정 리소스로 제한하는 사용자 관리 권한을 부여하는 데 유용합니다. 예를 들어 **DataCenterAdmin** 역할에는 해당 데이터 센터의 스토리지를 제외하고 할당된 데이터 센터에 대해서만 관리자 권한이 있으며 **ClusterAdmin**에는 할당된 클러스터에 대해서만 관리자 권한이 있습니다.

호스트 관리자는 특정 호스트에 대한 시스템 관리 역할입니다. 이는 각 호스트에 시스템 관리자가 필요한 여러 호스트가 있는 클러스터에서 유용합니다. 헤더 표시줄의 **Configure** 버튼을 사용하여 환경의 모든 호스트에 호스트 관리자를 할당할 수 있습니다.

호스트 관리자 역할은 다음 작업을 허용합니다.

- 호스트의 구성을 편집합니다.
- 논리 네트워크를 설정합니다.
- 호스트를 제거합니다.

기존 시스템 관리자를 제거하고 새 시스템 관리자를 추가하여 호스트의 시스템 관리자를 변경할 수도 있습니다.

### 1.2.14. 설명된 호스트 관리자 역할

호스트 권한 역할

아래 표에서는 호스트 관리에 적용되는 관리자 역할 및 권한을 설명합니다.

표 1.8. Red Hat Virtualization System Administrator Roles

역할	권한	참고
HostAdmin	호스트 관리자	특정 호스트를 구성, 관리 및 제거할 수 있습니다. 특정 호스트에서 네트워크 관련 작업을 수행할 수도 있습니다.

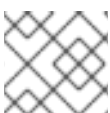
### 1.2.15. 스토리지 도메인의 시스템 권한 관리

슈퍼 유저로서 시스템 관리자는 관리 포털의 모든 측면을 관리합니다. 보다 구체적인 관리 역할을 다른 사용자에게 할당할 수 있습니다. 이러한 제한된 관리자 역할은 특정 리소스로 제한하는 사용자 관리 권한을 부여하는 데 유용합니다. 예를 들어 **DataCenterAdmin** 역할에는 해당 데이터 센터의 스토리지를 제외하고 할당된 데이터 센터에 대해서만 관리자 권한이 있으며 **ClusterAdmin**에는 할당된 클러스터에 대해서만 관리자 권한이 있습니다.

스토리지 관리자는 특정 스토리지 도메인에 대해서만 시스템 관리 역할입니다. 이는 각 스토리지 도메인에 시스템 관리자가 필요한 여러 스토리지 도메인이 있는 데이터 센터에서 유용합니다. 헤더 표시줄의 **Configure** 버튼을 사용하여 환경의 모든 스토리지 도메인에 스토리지 관리자를 할당합니다.

스토리지 도메인 관리자 역할은 다음 작업을 허용합니다.

- 스토리지 도메인의 구성을 편집합니다.
- 스토리지 도메인을 유지 관리 모드로 이동합니다.
- 스토리지 도메인을 제거합니다.



**참고**

기존 사용자에게 역할 및 권한만 할당할 수 있습니다.

기존 시스템 관리자를 제거하고 새 시스템 관리자를 추가하여 스토리지 도메인의 시스템 관리자를 변경할 수도 있습니다.

## 1.2.16. 설명된 스토리지 관리자 역할

스토리지 도메인 권한 역할

아래 표에서는 스토리지 도메인 관리에 적용되는 관리자 역할 및 권한을 설명합니다.

표 1.9. Red Hat Virtualization System Administrator Roles

역할	권한	참고
StorageAdmin	스토리지 관리자	특정 스토리지 도메인을 생성, 삭제, 구성 및 관리할 수 있습니다.
GlusterAdmin	Gluster Storage 관리자	Gluster 스토리지 볼륨을 생성, 삭제, 구성 및 관리할 수 있습니다.

## 1.2.17. 가상 머신 풀의 시스템 권한 관리

슈퍼 유저로서 시스템 관리자는 관리 포털의 모든 측면을 관리합니다. 보다 구체적인 관리 역할을 다른 사용자에게 할당할 수 있습니다. 이러한 제한된 관리자 역할은 특정 리소스로 제한하는 사용자 관리 권한을 부여하는 데 유용합니다. 예를 들어 **DataCenterAdmin** 역할에는 해당 데이터 센터의 스토리지를 제외하고 할당된 데이터 센터에 대해서만 관리자 권한이 있으며 **ClusterAdmin**에는 할당된 클러스터에 대해서만 관리자 권한이 있습니다.

가상 시스템 풀 관리자는 데이터 센터의 가상 머신 풀의 시스템 관리 역할입니다. 이 역할은 특정 가상 머신 풀, 데이터 센터 또는 전체 가상화 환경에 적용할 수 있습니다. 이는 다른 사용자가 특정 가상 머신 풀 리소스를 관리할 수 있도록 하는 데 유용합니다.

가상 머신 풀 관리자 역할은 다음 작업을 허용합니다.

- 풀을 생성, 편집 및 제거합니다.
- 풀에서 가상 머신을 추가 및 분리합니다.



### 참고

기존 사용자에게 역할 및 권한만 할당할 수 있습니다.

## 1.2.18. 가상 머신 풀 관리자 역할 설명

풀 권한 역할

아래 표에서는 풀 관리에 적용되는 관리자 역할 및 권한을 설명합니다.

표 1.10. Red Hat Virtualization System Administrator Roles

역할	권한	참고
----	----	----

역할	권한	참고
VmPoolAdmin	가상 풀의 시스템 관리자 역할입니다.	가상 풀을 생성, 삭제 및 구성하고, 가상 풀 사용자를 할당 및 제거하고, 가상 머신에서 기본 작업을 수행할 수 있습니다.
ClusterAdmin	클러스터 관리자	특정 클러스터의 모든 가상 머신 풀을 사용, 생성, 삭제, 관리할 수 있습니다.

### 1.2.19. 가상 디스크에 대한 시스템 권한 관리

슈퍼 유저로서 시스템 관리자는 관리 포털의 모든 측면을 관리합니다. 보다 구체적인 관리 역할을 다른 사용자에게 할당할 수 있습니다. 이러한 제한된 관리자 역할은 특정 리소스로 제한하는 사용자 관리 권한을 부여하는 데 유용합니다. 예를 들어 **DataCenterAdmin** 역할에는 해당 데이터 센터의 스토리지를 제외하고 할당된 데이터 센터에 대해서만 관리자 권한이 있으며 **ClusterAdmin**에는 할당된 클러스터에 대해서만 관리자 권한이 있습니다.

**Red Hat Virtualization Manager**는 두 개의 기본 가상 디스크 사용자 역할을 제공하지만 기본 가상 디스크 관리자 역할은 제공하지 않습니다. **DiskCreator** 역할인 **DiskCreator** 역할 중 하나를 사용하면 **VM** 포털에서 가상 디스크를 관리할 수 있습니다. 이 역할은 특정 가상 시스템, 데이터 센터, 특정 스토리지 도메인에 또는 전체 가상화 환경에 적용할 수 있습니다. 이는 다른 사용자가 다른 가상 리소스를 관리할 수 있도록 하는 데 유용합니다.

가상 디스크 작성자 역할은 다음 작업을 허용합니다.

- 가상 머신 또는 기타 리소스와 관련된 가상 디스크를 생성, 편집 및 제거합니다.
- 가상 디스크에 대한 사용자 권한을 편집합니다.



#### 참고

기존 사용자에게 역할 및 권한만 할당할 수 있습니다.

### 1.2.20. 설명된 가상 디스크 사용자 역할

가상 디스크 사용자 권한 역할

아래 표는 **VM** 포털에서 가상 디스크를 사용하고 관리하는 데 적용되는 사용자 역할 및 권한을 설명합니다.

표 1.11. Red Hat Virtualization System Administrator Roles

역할	권한	참고
DiskOperator	가상 디스크 사용자.	가상 디스크를 사용, 보기 및 편집할 수 있습니다. 가상 디스크가 연결된 가상 머신을 사용하는 권한을 상속합니다.



역할	권한	참고
DiskCreator	할당된 클러스터 또는 데이터 센터 내에서 가상 디스크를 생성, 편집, 관리 및 제거할 수 있습니다.	이 역할은 특정 가상 디스크에 적용되지 않습니다. <b>Configure</b> 장이 있는 전체 환경에 대해 이 역할을 사용자에게 적용합니다. 또는 특정 데이터 센터, 클러스터 또는 스토리지 도메인에 이 역할을 적용합니다.

### 1.2.21. 레거시 SPICE Cipher 설정

SPICE 콘솔은 기본적으로 암호화 문자열과 함께 **FIPS** 호환 암호화를 사용합니다. 기본 **SPICE** 암호화 문자열: **kECDHE+FIPS:kDHE+FIPS:kRSA+FIPS:!eNULL:!aNULL**

이 문자열은 일반적으로 충분합니다. 그러나 이전 운영 체제 또는 **SPICE** 클라이언트가 있는 가상 머신이 있는 경우 하나 또는 다른 클라이언트가 **FIPS** 호환 암호화를 지원하지 않는 경우 약한 암호화 문자열을 사용해야 합니다. 그렇지 않으면 기존 클러스터에 새 클러스터 또는 새 호스트를 설치하고 해당 가상 머신에 연결을 시도하는 경우 연결 보안 오류가 발생할 수 있습니다.

**Ansible** 플레이북을 사용하여 암호화 문자열을 변경할 수 있습니다.

암호화 문자열 변경

1. **Manager** 시스템에서 **/usr/share/ovirt-engine/playbooks** 디렉토리에 파일을 만듭니다. 예를 들면 다음과 같습니다.

```
# vim /usr/share/ovirt-engine/playbooks/change-spice-cipher.yml
```

2. 파일에 다음을 입력하고 저장합니다.

```
name: oVirt - setup weaker SPICE encryption for old clients
hosts: hostname
vars:
  host_deploy_spice_cipher_string: 'DEFAULT:-RC4:-3DES:-DES'
roles:
  - ovirt-host-deploy-spice-encryption
```

3. 방금 만든 파일을 실행합니다.

```
# ansible-playbook -i hostname /usr/share/ovirt-engine/playbooks/change-spice-cipher.yml
```

또는 다음과 같이 **host\_deploy\_spice\_cipher\_string** 과 함께 **--extra-vars** 옵션을 사용하여 **Ansible** 플레이북 **ovirt-host-deploy** 를 사용하여 호스트를 재구성할 수 있습니다.

```
# ansible-playbook -i hostname \
  --extra-vars host_deploy_spice_cipher_string="DEFAULT:-RC4:-3DES:-DES" \
  /usr/share/ovirt-engine/playbooks/ovirt-host-deploy.yml
```

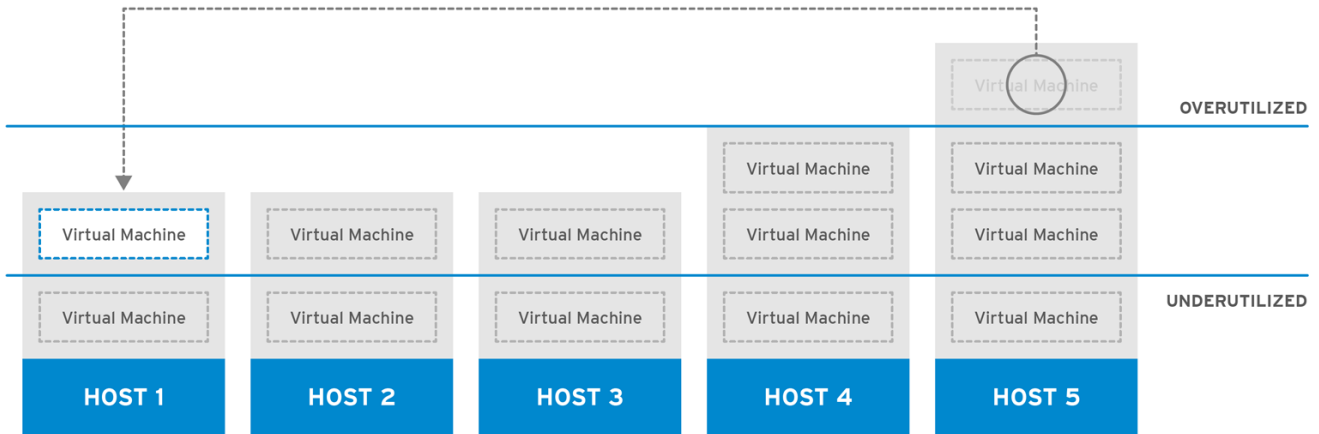
## 1.3. 정책 예약

스케줄링 정책은 스케줄링 정책이 적용되는 클러스터의 호스트에 가상 시스템이 배포되는 논리를 정의하는

규칙 집합입니다. 스케줄링 정책은 필터, 가중치 및 로드 밸런싱 정책의 조합을 통해 이 논리를 결정합니다. **filter** 모듈은 하드 시행을 적용하고 해당 필터에서 지정한 조건을 충족하지 않는 호스트를 필터링합니다. **weights** 모듈은 소프트 적용을 적용하며 가상 시스템을 실행할 수 있는 클러스터의 호스트를 결정할 때 고려되는 요인의 상대적 우선 순위를 제어하는 데 사용됩니다.

Red Hat Virtualization Manager는 **Evenly\_Distributed, Cluster\_Maintenance, None, Power\_Saving, VM\_Evenly\_Distributed**의 5가지 기본 스케줄링 정책을 제공합니다. 가상 시스템 배포에 대한 세분화된 제어 제공하는 새로운 스케줄링 정책을 정의할 수도 있습니다. 스케줄링 정책에 관계없이 과부하된 **CPU**가 있는 호스트에서 가상 머신이 시작되지 않습니다. 기본적으로 호스트의 **CPU**는 5분 동안 **80%**를 초과하는 부하가 있는 경우 과부하로 간주되지만 스케줄링 정책을 사용하여 이러한 값을 변경할 수 있습니다. 각 스케줄링 정책의 속성에 대한 자세한 내용은 **8.2.5절. "설명된 스케줄링 정책 설정"**를 참조하십시오.

그림 1.4. 균등하게 분산 스케줄링 정책

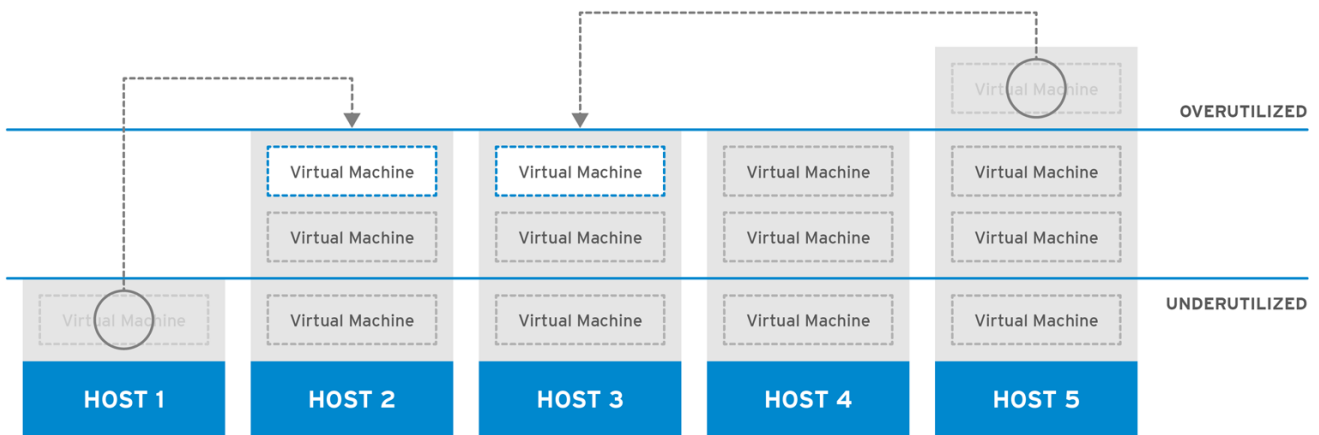


RHV\_ 444396\_0417

**Evenly\_Distributed** 스케줄링 정책은 클러스터의 모든 호스트에 메모리 및 **CPU** 처리 부하를 균등하게 분배합니다. 호스트가 정의된 **CpuOverCommitDurationMinutes, HighUtilization, MaxFreeMemoryForOverUtilized**에 도달하면 호스트에 연결된 추가 가상 머신이 시작되지 않습니다.

**VM\_Evenly\_Distributed** 스케줄링 정책 가상 머신은 가상 머신 수에 따라 호스트 간에 균등하게 사용됩니다. 호스트가 **HighVmCount**보다 더 많은 가상 머신을 실행 중이고 **MigrationThreshold**를 벗어나는 가상 머신 수가 있는 호스트가 한 개 이상 있는 경우 클러스터의 균형을 맞추지 않은 것으로 간주됩니다.

그림 1.5. Power Saving Scheduling Policy



RHV\_ 444396\_0417

**Power\_Saving** 스케줄링 정책은 사용 가능한 호스트의 하위 집합에 메모리 및 **CPU** 처리 부하를 분배하여 활용도가 낮은 호스트의 전원 소비를 줄입니다. **CPU** 부하가 정의된 시간 간격보다 긴 경우의 사용률이 낮은 호스트는 전원이 꺼질 수 있도록 모든 가상 시스템을 다른 호스트로 마이그레이션합니다. 호스트가 정의된 높은 사용률 값에 도달한 경우 호스트에 연결된 추가 가상 시스템이 시작되지 않습니다.

가상 시스템을 실행하기 위해 호스트 간에 로드 또는 전원 공유가 없도록 **None** 정책을 설정합니다. 이는 기본값 모드입니다. 가상 머신이 시작되면 클러스터의 모든 호스트에 메모리 및 **CPU** 처리 로드가 균등하게 분배됩니다. 호스트가 정의된 **CpuOverCommitDurationMinutes, HighUtilization, MaxFreeMemoryForOverUtilized** 에 도달하면 호스트에 연결된 추가 가상 머신이 시작되지 않습니다.

**Cluster\_Maintenance** 스케줄링 정책은 유지 관리 작업 중에 클러스터의 활동을 제한합니다.

**Cluster\_Maintenance** 정책이 설정되면 고가용성 가상 머신을 제외하고 새로운 가상 머신을 시작할 수 없습니다. 호스트 장애가 발생하면 고가용성 가상 머신이 제대로 재시작되고 모든 가상 머신을 마이그레이션할 수 있습니다.

### 1.3.1. 스케줄링 정책 생성

새로운 스케줄링 정책을 생성하여 **Red Hat Virtualization** 환경에서 지정된 클러스터에 가상 머신이 배포되는 논리를 제어할 수 있습니다.

스케줄링 정책 생성

1. **Administration** → **Configure** 클릭합니다.
2. 스케줄링 정책 탭을 클릭합니다.
3. 새로 생성을 클릭합니다.
4. 스케줄링 정책에 대한 이름 및 설명을 입력합니다.
5. 필터 모듈을 구성합니다.
  - a. 모듈 필터링 섹션의 **Disabled Filters** 섹션의 스케줄링 정책에 적용할 기본 필터 모듈을 드래그 앤 드롭합니다.
  - b. 특정 필터 모듈을 첫 번째로 설정하여 기본 최적화를 위해 우선 순위가 가장 높은 우선 순위 또는 마지막으로 설정할 수 있습니다. 우선 순위를 설정하려면 **filter** 모듈을 마우스 오른쪽 버튼으로 클릭하고 커서를 **0:0** 또는 마지막 을 선택합니다.
6. **weight** 모듈을 구성합니다.
  - a. **Weights Modules** 섹션에서 **Disabled Weights** 섹션의 스케줄링 정책에 적용할 기본 **weights** 모듈을 드래그 앤 드롭 합니다.
  - b. 활성화된 **weight** 모듈 왼쪽에 있는 + 및 - 버튼을 사용하여 해당 모듈의 가중치를 늘리거나 줄입니다.
7. 로드 밸런싱 정책을 지정합니다.
  - a. **Load Balancer** 섹션의 드롭다운 메뉴에서 스케줄링 정책에 적용할 로드 밸런싱 정책을 선택합니다.
  - b. 속성 섹션의 드롭다운 메뉴에서 로드 밸런싱 속성을 선택하여 스케줄링 정책에 적용하고 해당 속성 오른쪽의 텍스트 필드를 사용하여 값을 지정합니다.
  - c. + 및 - 버튼을 사용하여 추가 속성을 추가하거나 제거합니다.
8. **OK**를 클릭합니다.

### 1.3.2. 새 스케줄링 정책 및 스케줄링 정책 편집 창의 설정에 대한 설명

다음 표에서는 새 스케줄링 정책 및 스케줄링 정책편집 창에서 사용할 수 있는 옵션에 대해 자세히 설명합니다.

표 1.12. 새로운 스케줄링 정책 및 스케줄링 정책 설정 편집

필드 이름	설명
이름	스케줄링 정책의 이름입니다. 이는 Red Hat Virtualization Manager의 스케줄링 정책을 참조하는데 사용되는 이름입니다.
설명	스케줄링 정책에 대한 설명입니다. 이 필드는 권장되지만 필수는 아닙니다.
필터 모듈	<p>클러스터의 가상 머신을 실행할 수 있는 호스트를 제어하기 위한 필터 세트입니다. 필터를 활성화하면 아래에 설명된 대로 해당 필터에서 지정한 조건을 충족하지 않는 호스트를 필터링합니다.</p> <ul style="list-style-type: none"> <li>● <b>CpuPinning:</b> CPU 고정 정의를 충족하지 않는 호스트</li> <li>● <b>Migration:</b> 동일한 호스트로의 마이그레이션을 방지합니다.</li> <li>● <b>PinToHost:</b> 가상 머신이 고정되는 호스트 이외의 호스트입니다.</li> <li>● <b>cpu-Level:</b> 가상 머신의 CPU 토폴로지를 충족하지 않는 호스트입니다.</li> <li>● <b>cpu:</b> 가상 머신에 할당된 수보다 CPU 수가 적은 호스트입니다.</li> <li>● <b>memory:</b> 가상 머신을 실행하기에 충분한 메모리가 없는 호스트입니다.</li> <li>● <b>VmAffinityGroups:</b> 선호도 그룹의 멤버인 가상 머신에 지정된 조건을 충족하지 않는 호스트입니다. 예를 들어 선호도 그룹의 가상 머신은 동일한 호스트 또는 별도의 호스트에서 실행해야 합니다.</li> <li>● <b>VmToHostsAffinityGroups:</b> 선호도 그룹의 멤버인 가상 머신에 지정된 조건을 충족하지 않는 호스트 그룹입니다. 예를 들어 선호도 그룹의 가상 머신은 그룹 또는 그룹에서 제외된 별도의 호스트에서 실행되어야 합니다.</li> <li>● <b>InClusterUpgrade:</b> 가상 머신이 현재 실행되는 호스트보다 이전 운영 체제를 실행하는 호스트입니다.</li> <li>● <b>HostDevice:</b> 가상 머신에 필요한 호스트 장치를 지원하지 않는 호스트입니다.</li> <li>● <b>HA:</b> 자체 호스팅 엔진 환경에서 Manager 가상 시스템을 강제 시행하여 긍정적인 고가용성 점수가 있는 호스트에서만 실행됩니다.</li> </ul>

필드 이름	설명
	<ul style="list-style-type: none"> <li>● <b>emulated-Machine</b>: 적절한 에뮬레이션 머신 지원이 없는 호스트</li> <li>● <b>Network</b>: 가상 머신의 네트워크 인터페이스 컨트롤러에 필요한 네트워크 또는 클러스터의 디스플레이 네트워크가 설치되지 않은 호스트입니다.</li> <li>● <b>HostedEnginesSpares</b>: 지정된 수의 자체 호스팅 엔진 노드에서 Manager 가상 머신의 공간을 예약합니다.</li> <li>● 레이블: 필수 선호도 라벨이 없는 호스트입니다.</li> <li>● <b>compatibility -Version</b> : 올바른 호환성 버전 지원이 있는 호스트에서만 가상 머신을 실행합니다.</li> <li>● <b>cpuOverloaded</b> : CPU 과부하가 있는 호스트입니다.</li> </ul>

필드 이름	설명
<p>가중치 모듈</p>	<p>가상 머신을 실행할 수 있는 클러스터에서 호스트를 결정할 때 고려되는 상대적 우선 순위의 요인을 제어하기 위한 가중치 세트입니다.</p> <ul style="list-style-type: none"> <li>● <b>InClusterUpgrade:</b> 운영 체제 버전에 따라 호스트를 모니터링합니다. 가중치는 가상 시스템이 현재 실행 중인 호스트와 동일한 운영 체제를 사용하는 호스트보다 이전 운영 체제로 호스트를 더 많이 내는 것을 의미합니다. 이렇게 하면 우선 순위가 최신 운영 체제가 있는 호스트에 항상 부여됩니다.</li> <li>● <b>OptimalForHaReservation:</b> 고가용성 점수에 따라 호스트를 검색합니다.</li> <li>● <b>none:</b> even even 배포 모듈에 따라 호스트를 모니터링합니다.</li> <li>● <b>OptimalForEvenGuestDistribution:</b> 해당 호스트에서 실행되는 가상 머신의 수에 따라 호스트를 검색합니다.</li> <li>● <b>VmAffinityGroups:</b> 가상 머신에 대해 정의된 선호도 그룹에 따라 호스트를 모니터링합니다. 이 weight 모듈은 선호도 그룹의 가상 머신을 해당 선호도 그룹의 매개변수에 따라 동일한 호스트 또는 별도의 호스트에서 실행할 가능성을 결정합니다.</li> <li>● <b>VmToHostsAffinityGroups:</b> 가상 머신에 대해 정의된 선호도 그룹에 따라 호스트를 모니터링합니다. 이 weight 모듈은 선호도 그룹의 가상 머신을 그룹 또는 그룹에서 제외된 별도의 호스트의 호스트 중 하나에서 실행할 가능성을 결정합니다.</li> <li>● <b>OptimalForCPUPowerSaving:</b> CPU 사용량에 따라 호스트를 확인하여 CPU 사용량이 높은 호스트에 우선 순위를 부여합니다.</li> <li>● <b>OptimalForEvenCpuDistribution:</b> CPU 사용량에 따라 호스트를 확인하여 CPU 사용량이 낮은 호스트에 우선 순위를 부여합니다.</li> <li>● <b>HA:</b> 고가용성 점수에 따라 호스트를 모니터링합니다.</li> <li>● <b>PreferredHosts:</b> 가상 머신 설정 중에 Preferred Host가 우선합니다.</li> <li>● <b>OptimalForMemoryPowerSaving:</b> 메모리 사용량에 따라 호스트를 확인하여 사용 가능한 메모리가 낮은 호스트에 우선 순위를 부여합니다.</li> <li>● <b>OptimalForMemoryEvenDistribution:</b> 메모리 사용량에 따라 호스트를 확인하여 사용 가능한 메모리가 높은 호스트에 우선 순위를 부여합니다.</li> </ul>

필드 이름	설명
로드 밸런서	이 드롭다운 메뉴를 사용하면 적용할 로드 밸런싱 모듈을 선택할 수 있습니다. 로드 밸런싱 모듈에서는 사용량이 낮은 호스트로 가상 머신을 마이그레이션하는 데 사용되는 논리를 결정합니다.
속성	이 드롭다운 메뉴를 사용하면 로드 밸런싱 모듈에 대한 속성을 추가하거나 제거할 수 있으며 스케줄링 정책에 대한 로드 밸런싱 모듈을 선택한 경우에만 사용할 수 있습니다. 속성은 기본적으로 정의되어 있지 않으며 사용 가능한 속성은 선택한 로드 밸런싱 모듈에 따라 다릅니다. + 및 - 버튼을 사용하여 로드 밸런싱 모듈에 속성을 추가하거나 제거합니다.

## 1.4. 인스턴스 유형



인스턴스 유형을 사용하여 가상 시스템의 하드웨어 구성을 정의할 수 있습니다. 가상 머신을 생성하거나 편집할 때 인스턴스 유형을 선택하면 하드웨어 구성 필드가 자동으로 채워집니다. 이를 통해 사용자는 모든 필드를 수동으로 채우지 않고도 동일한 하드웨어 구성으로 여러 가상 머신을 생성할 수 있습니다.

다음 표에 설명된 대로 사전 정의된 인스턴스 유형 세트는 기본적으로 사용할 수 있습니다.

표 1.13. 사전 정의된 인스턴스 유형

이름	메모리	vCPU
작음	512MB	1
작음	2GB	1
중간	4GB	2
대규모	8GB	2
XLarge	16GB	4

관리자는 **Configure** 창의 **Instance Types** 탭에서 인스턴스 유형을 생성, 편집 및 제거할 수도 있습니다.

새 가상 머신의 필드와 인스턴스 유형에 바인딩된 가상 머신 편집 창의 필드에는 체인 링크 이미지(  )이 있습니다. 이러한 필드 중 하나의 값이 변경되면 인스턴스 유형에서 가상 머신이 분리되고 사용자 지정으로 변경되고 체인이 손상될 수 있습니다(  ). 그러나 값이 다시 변경되면 체인이 다시 링크되고 인스턴스 유형이 선택한 항목으로 다시 이동합니다.

### 1.4.1. 인스턴스 유형 생성

관리자는 가상 머신을 생성하거나 편집할 때 사용자가 선택할 수 있는 새 인스턴스 유형을 생성할 수 있습니다.

인스턴스 유형 생성

1. **Administration → Configure** 클릭합니다.
2. 인스턴스 유형 탭을 클릭합니다.
3. 새로 생성을 클릭합니다.
4. 인스턴스 유형에 대한 이름 및 설명을 입력합니다.
5. 고급 옵션 표시를 클릭하고 필요에 따라 인스턴스 유형의 설정을 구성합니다. **New Instance Type** (새 인스턴스 유형) 창에 표시되는 설정은 **New Virtual Machine** (새 가상 머신) 창의 설정과 동일하지만 관련 필드만 있습니다. 가상 머신 [관리 가이드](#)에서 [새 가상 머신의 설정 설명 및 가상 머신 Windows 편집](#) 을 참조하십시오.
6. **OK**를 클릭합니다.

새 인스턴스 유형은 **Configure** (구성) 창의 **Instance Types** 탭에 표시되며, 가상 머신을 생성하거나 편집할 때 **Instance Type** 드롭다운 목록에서 선택할 수 있습니다.

### 1.4.2. 인스턴스 유형 편집

관리자는 **Configure** 창에서 기존 인스턴스 유형을 편집할 수 있습니다.

인스턴스 유형 속성 편집

1. **Administration → Configure** 클릭합니다.
2. 인스턴스 유형 탭을 클릭합니다.
3. 편집할 인스턴스 유형을 선택합니다.
4. **편집** 을 클릭합니다.
5. 필요에 따라 설정을 변경합니다.
6. **OK**를 클릭합니다.

인스턴스 유형의 구성이 업데이트되었습니다. 이 인스턴스 유형을 기반으로 새 가상 머신이 작성되거나 이 인스턴스 유형을 기반으로 하는 기존 가상 머신이 생성되면 새 구성이 적용됩니다.

이 인스턴스 유형을 기반으로 하는 기존 가상 머신은 체인 아이콘으로 표시된 필드가 표시됩니다. 이 필드는 업데이트됩니다. 인스턴스 유형이 변경될 때 기존 가상 머신이 실행 중인 경우 보류 중인 변경 사항 아이콘이 그 옆에 표시되고 체인 아이콘이 있는 필드가 다시 시작할 때 업데이트됩니다.

### 1.4.3. 인스턴스 유형 제거

인스턴스 유형 제거

1. **Administration → Configure** 클릭합니다.
2. 인스턴스 유형 탭을 클릭합니다.
3. 제거할 인스턴스 유형을 선택합니다.
4. **제거**를 클릭합니다.



5. 가상 시스템이 제거할 인스턴스 유형을 기반으로 하는 경우 연결된 가상 시스템이 나열된 경고 창이 표시됩니다. 인스턴스 유형을 계속 제거하려면 **Approve Operation** 확인란을 선택합니다. 그렇지 않으면 취소 를 클릭합니다.
6. **OK**를 클릭합니다.

인스턴스 유형은 **Instance Types** 목록에서 제거되며 새 가상 머신을 생성할 때 더 이상 사용할 수 없습니다. 제거된 인스턴스 유형에 연결된 가상 머신은 이제 **Custom** (인스턴스 유형 없음)에 연결됩니다.

## 1.5. MAC 주소 풀

**MAC** 주소 풀은 각 클러스터에 할당된 **MAC** 주소의 범위를 정의합니다. 각 클러스터에 대해 **MAC** 주소 풀이 지정됩니다. **Red Hat Virtualization**은 **MAC** 주소 풀을 사용하여 **MAC** 주소를 자동으로 생성하고 새 가상 네트워크 장치에 할당할 수 있으므로 **MAC** 주소 중복을 방지할 수 있습니다. 클러스터와 관련된 모든 **MAC** 주소가 할당된 **MAC** 주소 풀의 범위 내에 있으면 **MAC** 주소 풀의 메모리 효율성이 향상됩니다.

동일한 **MAC** 주소 풀을 여러 클러스터에서 공유할 수 있지만 각 클러스터에는 단일 **MAC** 주소 풀이 할당됩니다. 기본 **MAC** 주소 풀은 **Red Hat Virtualization**에서 생성되며 다른 **MAC** 주소 풀이 할당되지 않은 경우 사용됩니다. **MAC** 주소 풀을 클러스터에 할당하는 방법에 대한 자세한 내용은 [8.2.1절. "새 클러스터 생성"](#)을 참조하십시오.



### 참고

**Red Hat Virtualization** 클러스터가 네트워크를 두 개 이상 공유하는 경우 각 클러스터의 가상 머신에서 동일한 범위의 **MAC** 주소를 사용하려고 하므로 기본 **MAC** 주소 풀에만 의존하지 않고 충돌이 발생합니다. **MAC** 주소 충돌을 방지하려면 **MAC** 주소 풀 범위를 확인하여 각 클러스터에 고유한 **MAC** 주소 범위가 할당되었는지 확인합니다.

**MAC** 주소 풀은 풀에 반환된 마지막 주소 뒤에 사용 가능한 다음 **MAC** 주소를 할당합니다. 범위에 더 이상 남은 주소가 없으면 범위의 처음부터 검색이 다시 시작됩니다. 단일 **MAC** 주소 풀에 사용 가능한 **MAC** 주소가 있는 여러 **MAC** 주소 범위가 있는 경우, 범위는 사용 가능한 **MAC** 주소가 선택된 것과 동일한 방식으로 들어오는 요청을 처리합니다.

### 1.5.1. MAC 주소 풀 생성

새 **MAC** 주소 풀을 생성할 수 있습니다.

#### MAC 주소 풀 생성

1. **Administration** → **Configure** 클릭합니다.
2. **MAC** 주소 풀 탭을 클릭합니다.
3. 추가를 클릭합니다.
4. 새 **MAC** 주소 풀의 **Name** 및 **Description** 을 입력합니다.
5. **MAC** 주소를 풀에서 여러 번 사용할 수 있도록 하려면 **Allow Duplicates** 확인란을 선택합니다. **MAC** 주소 풀은 중복된 **MAC** 주소를 자동으로 사용하지 않지만 **duplicates** 옵션을 활성화하면 사용자가 중복된 **MAC** 주소를 수동으로 사용할 수 있습니다.



### 참고

하나의 **MAC** 주소 풀에 중복이 비활성화되고 다른 하나는 중복이 활성화된 경우 풀에서 각 **MAC** 주소를 한 번만 사용할 수 있으며 중복이 활성화된 풀에서 여러 번 사용할 수 있습니다.

6. 필요한 **MAC** 주소 범위를 입력합니다. 여러 범위를 입력하려면 **From** 및 **To** 필드 옆에 있는 더하기 버튼을 클릭합니다.
7. **OK**를 클릭합니다.

## 1.5.2. MAC 주소 풀 편집

**MAC** 주소 풀을 편집하여 풀에서 사용 가능한 **MAC** 주소 범위 및 중복 허용 여부를 포함하여 세부 정보를 변경할 수 있습니다.

### MAC 주소 풀 속성 편집

1. **Administration** → **Configure** 클릭합니다.
2. **MAC** 주소 풀 탭을 클릭합니다.
3. 편집할 **MAC** 주소 풀을 선택합니다.
4. 편집 을 클릭합니다.
5. 필요에 따라 이름, 설명, 중복 허용 및 **MAC** 주소 범위 필드를 변경합니다.



### 참고

**MAC** 주소 범위가 업데이트되면 기존 **NIC**의 **MAC** 주소가 다시 할당되지 않습니다. 이미 할당되었지만 새 **MAC** 주소 범위를 벗어나는 **MAC** 주소는 사용자 지정 **MAC** 주소로 추가되며 해당 **MAC** 주소 풀에서 계속 추적됩니다.

6. **OK**를 클릭합니다.

## 1.5.3. MAC 주소 풀 권한 편집

**MAC** 주소 풀이 생성되면 사용자 권한을 편집할 수 있습니다. 사용자 권한은 **MAC** 주소 풀을 사용할 수 있는 데이터 센터를 제어합니다. 새 사용자 권한 추가에 대한 자세한 내용은 [1.1절. "역할"](#) 을 참조하십시오.

### MAC 주소 풀 권한 편집

1. **Administration** → **Configure** 클릭합니다.
2. **MAC** 주소 풀 탭을 클릭합니다.
3. 필요한 **MAC** 주소 풀을 선택합니다.
4. **MAC** 주소 풀에 대한 사용자 권한을 편집합니다.
  - **MAC** 주소 풀에 사용자 권한을 추가하려면 다음을 수행합니다.
    - a. **Configure** (구성) 창 하단에 있는 사용자 권한 창에서 추가를 클릭합니다.
    - b. 필요한 사용자를 검색하고 선택합니다.

- c. **Role to Assign** 드롭다운 목록에서 필요한 역할을 선택합니다.
- d. 확인을 클릭하여 사용자 권한을 추가합니다.
- **MAC 주소 풀**에서 사용자 권한을 제거하려면 다음을 수행합니다.
  - a. **Configure (구성)** 창 하단에 있는 사용자 권한 창에서 제거할 사용자 권한을 선택합니다.
  - b. 제거를 클릭하여 사용자 권한을 제거합니다.

#### 1.5.4. MAC 주소 풀 제거

풀이 클러스터와 연결되지 않은 경우 생성된 **MAC** 주소 풀을 제거할 수 있지만 기본 **MAC** 주소 풀은 제거할 수 없습니다.

##### MAC 주소 풀 제거

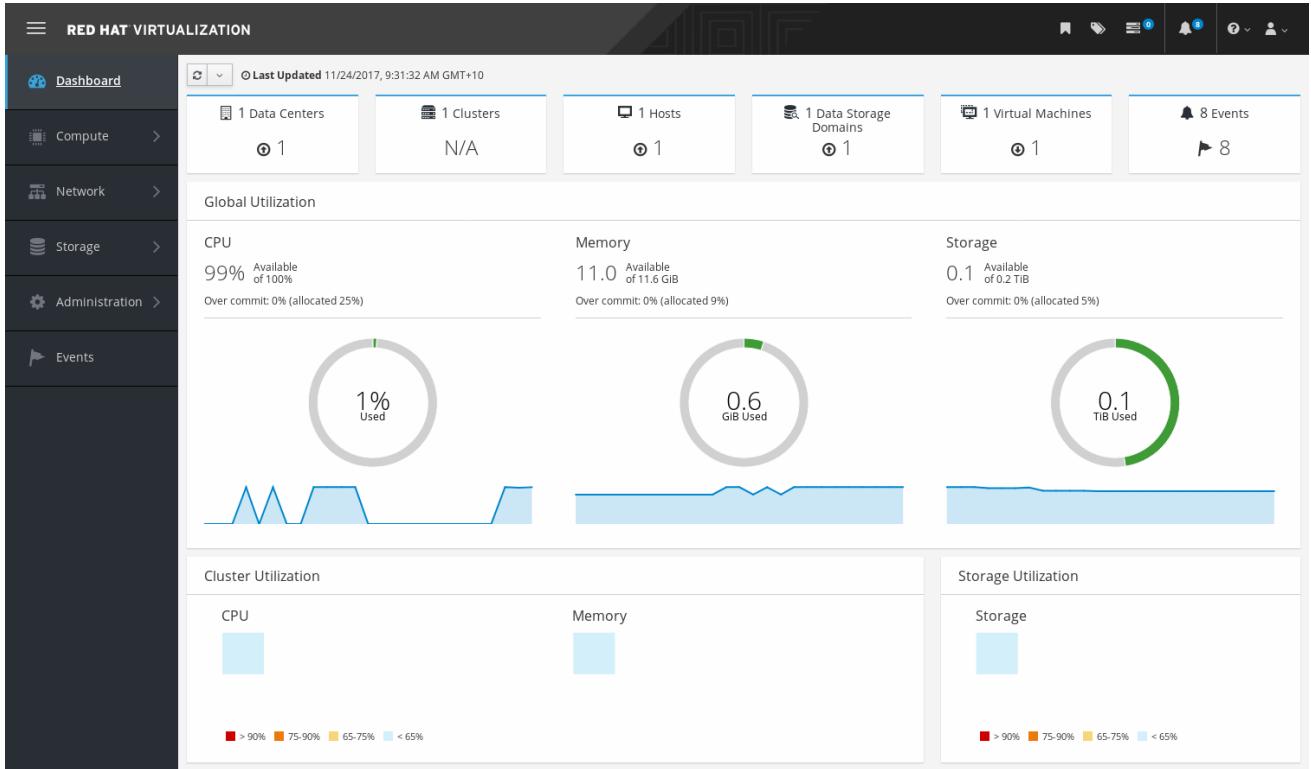
1. **Administration → Configure** 클릭합니다.
2. **MAC** 주소 풀 탭을 클릭합니다.
3. 제거할 **MAC** 주소 풀을 선택합니다.
4. 제거를 클릭합니다.
5. **OK**를 클릭합니다.

## 2장. 대시보드

대시보드는 **Red Hat Virtualization**의 리소스 및 사용률 요약을 표시하여 **Red Hat Virtualization** 시스템 상태에 대한 개요를 제공합니다. 이 요약에서는 문제를 경고하고 문제 영역을 분석할 수 있습니다.

대시보드의 정보는 기본적으로 데이터에서 **15분**마다 업데이트되며, **Manager API**를 통해 기본적으로 **15초**마다 또는 대시보드가 새로 고쳐집니다. 사용자가 다른 페이지에서 다시 변경되거나 수동으로 새로 고칠 때 대시보드가 새로 고쳐집니다. 대시보드가 자동으로 새로 고침되지 않습니다. 인벤토리 카드 정보는 **Manager API**에서 제공하며 사용률 정보는 **DataECDHE**에서 제공합니다. 대시보드는 UI 플러그인 구성 요소로 구현되며 **Manager**와 함께 자동으로 설치 및 업그레이드됩니다.

그림 2.1. 대시보드



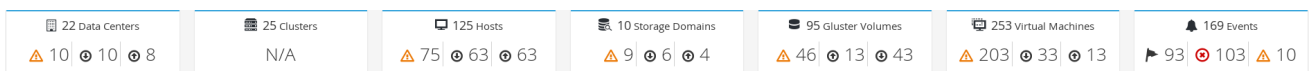
### 2.1. 사전 요구 사항

대시보드를 사용하려면 데이터가 설치 및 구성되어 있어야 합니다. 데이터란드바이저 [가이드에서 데이터 설치 및 구성](#)을 참조하십시오.

### 2.2. 글로벌 인벤토리





대시보드의 상단 섹션에서는 **Red Hat Virtualization** 리소스의 글로벌 인벤토리를 제공하며 데이터 센터, 클러스터, 호스트, 스토리지 도메인, 가상 시스템 및 이벤트에 대한 항목을 포함합니다. 아이콘은 각 리소스의 상태를 표시하고 숫자가 해당 상태로 각 리소스의 수량을 표시합니다.

그림 2.2. 글로벌 인벤토리



제목은 리소스 유형의 수를 표시하고 제목 아래에 상태가 표시됩니다. 리소스 제목을 클릭하면 **Red Hat Virtualization Manager**의 관련 페이지로 이동합니다. 클러스터의 상태는 항상 **N/A**로 표시됩니다.

표 2.1. 리소스 상태

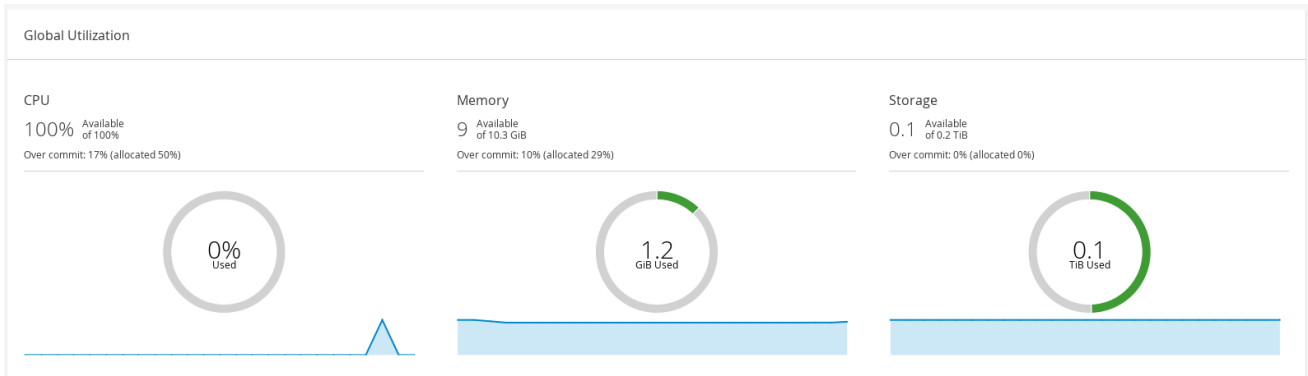
icon	상태
	<p>Red Hat Virtualization에 추가 리소스가 없습니다.</p>
	<p>경고 상태가 있는 리소스 수를 표시합니다. 아이콘을 클릭하면 검색이 경고 상태가 있는 해당 리소스로 제한되는 적절한 페이지로 이동합니다. 각 리소스에 대해 다른 방식으로 검색이 제한됩니다.</p> <ul style="list-style-type: none"> <li>● <b>데이터 센터:</b> 검색은 작동하지 않거나 응답하지 않는 데이터 센터로 제한됩니다.</li> <li>● <b>Gluster 볼륨:</b> 검색은 전원을 켜거나 일시 중지, 마이그레이션, 대기, 일시 중지 또는 전원을 끄는 Gluster 볼륨으로 제한됩니다.</li> <li>● <b>hosts:</b> 검색은 할당되지 않은 호스트, 유지 관리 모드, 설치, 재부팅, 유지 관리 준비, 승인 보류 중으로 제한됩니다.</li> <li>● <b>스토리지 도메인:</b> 검색은 초기화되지 않은, 연결 해제, 비활성 스토리지 도메인, 유지 관리 모드, 유지보수 준비, 분리 또는 활성화로 제한됩니다.</li> <li>● <b>가상 머신:</b> 검색은 전원을 켜거나 일시 중지, 마이그레이션, 대기, 일시 중지 또는 전원을 끄는 가상 머신으로 제한됩니다.</li> <li>● <b>events:</b> 검색은 경고 심각도가 있는 이벤트로 제한됩니다.</li> </ul>
	<p>up 상태의 리소스 수를 표시합니다. 아이콘을 클릭하면 검색이 up된 리소스로 제한되는 적절한 페이지로 이동합니다.</p>
	<p>down 상태의 리소스 수를 표시합니다. 아이콘을 클릭하면 검색 상태가 down인 리소스로 제한되는 적절한 페이지로 이동합니다. 각 리소스에 대해 다른 방식으로 검색이 제한됩니다.</p> <ul style="list-style-type: none"> <li>● <b>데이터 센터:</b> 검색은 초기화되지 않았거나 유지 관리 모드에서 또는 down 상태인 데이터 센터로 제한됩니다.</li> <li>● <b>Gluster 볼륨:</b> 검색은 분리 또는 비활성 볼륨으로 제한됩니다.</li> <li>● <b>hosts:</b> 검색은 무응답 호스트, 오류, 설치 오류, 비작동, 초기화 또는 다운된 호스트로 제한됩니다.</li> <li>● <b>스토리지 도메인:</b> 검색은 분리되거나 비활성 스토리지 도메인으로 제한됩니다.</li> <li>● <b>가상 머신:</b> 검색은 응답하지 않거나 재부팅하지 않는 가상 머신으로 제한됩니다.</li> </ul>

icon	상태
	경고 상태의 이벤트 수를 표시합니다. 아이콘을 클릭하면 검색이 경고 심각도가 있는 이벤트로 제한되는 <b>Events</b> 로 이동합니다.
	오류 상태의 이벤트 수를 표시합니다. 아이콘을 클릭하면 검색이 오류 심각도가 있는 <b>이벤트</b> 로 제한된 이벤트로 이동합니다.

## 2.3. 글로벌 사용률

글로벌 사용률 섹션에는 **CPU**, 메모리 및 스토리지의 시스템 사용률이 표시됩니다.

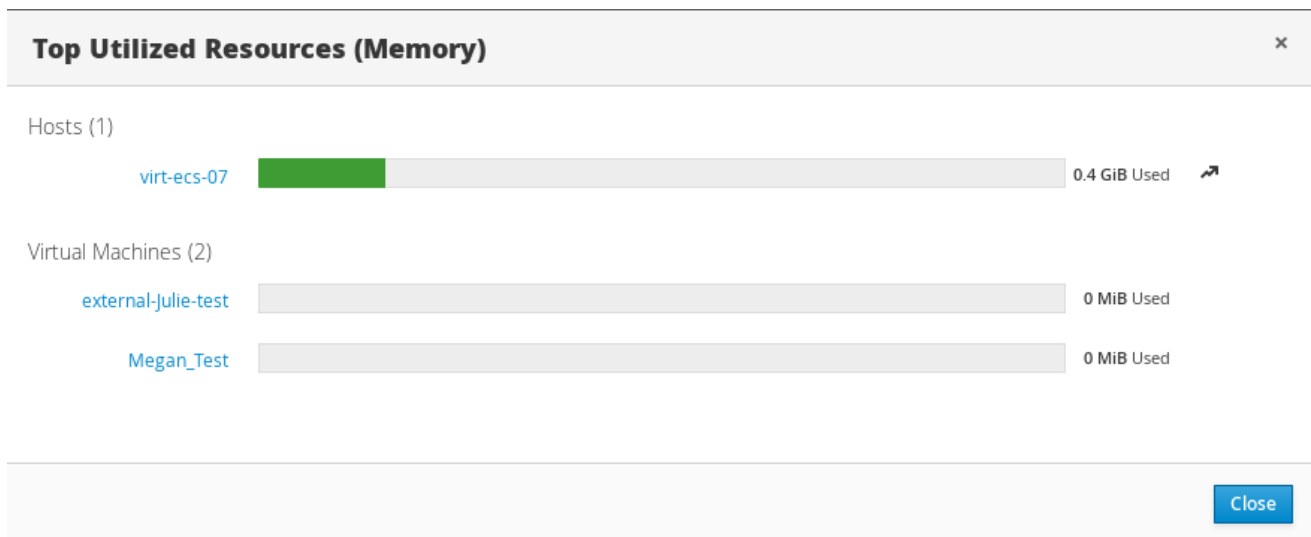
그림 2.3. 글로벌 사용률



- **top** 섹션은 사용 가능한 **CPU**, 메모리 또는 스토리지의 백분율과 오버 커밋 비율을 보여줍니다. 예를 들어 **CPU**의 최신 데이터를 기반으로 실행 중인 가상 머신에 사용할 수 있는 물리적 코어 수로 가상 코어 수를 나타냄으로써 **CPU**의 오버 커밋 비율이 계산됩니다.
- 도넛은 **CPU**, 메모리 또는 스토리지의 사용량을 표시하고 지난 **5분** 동안의 평균 사용량에 따라 모든 호스트의 평균 사용량을 표시합니다. 도넛 섹션을 마우스로 가리키면 선택한 섹션의 값이 표시됩니다.
- 하단에 있는 라인 그래프는 지난 **24시간** 동안의 추세를 보여줍니다. 각 데이터 포인트는 특정 시간 동안의 평균 사용량을 보여줍니다. 그래프 위로 마우스를 올리면 시간과 **CPU** 그래프에 사용된 백분율과 메모리 및 스토리지 그래프의 사용량이 표시됩니다.

### 2.3.1. 가장 많이 사용되는 리소스

그림 2.4. top Utilized Resources (Memory)

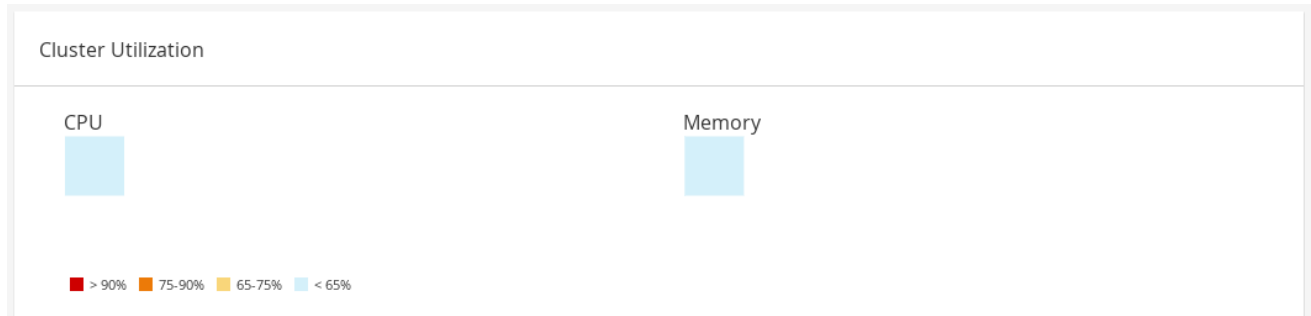


대시보드의 글로벌 사용률 섹션에서 도넛을 클릭하면 **CPU**, 메모리 또는 스토리지에 가장 많이 사용되는 리소스 목록이 표시됩니다. **CPU** 및 메모리의 경우 팝업에 사용량이 가장 많은 호스트 및 가상 머신 목록이 표시됩니다. 스토리지의 경우 팝업에는 사용된 상위 10개의 스토리지 도메인 및 가상 머신 목록이 표시됩니다. 사용 표시줄 오른쪽에 있는 화살표는 마지막 분 동안 해당 리소스에 대한 사용 추세를 보여줍니다.

## 2.4. 클러스터 사용률

**Cluster Utilization** 섹션에는 heatmap의 **CPU** 및 메모리에 대한 클러스터 사용률이 표시되어 있습니다.

그림 2.5. 클러스터 사용률



### 2.4.1. CPU

지난 24시간 동안 **CPU**의 평균 사용률을 보여주는 특정 클러스터의 **CPU** 사용률의 heatmap입니다. heatmap을 마우스로 가리키면 클러스터 이름이 표시됩니다. heatmap을 클릭하면 **Compute → Hosts** (컴퓨팅 호스트)로 이동하여 **CPU** 사용률에 따라 정렬된 특정 클러스터에서 검색 결과가 표시됩니다. 클러스터에서 **CPU** 사용량을 계산하는 데 사용되는 공식은 클러스터의 평균 호스트 **CPU** 사용률입니다. 이는 지난 24시간 동안 각 호스트의 평균 호스트 **CPU** 사용률을 사용하여 클러스터에서 **CPU**의 총 사용량을 찾는 방식으로 계산됩니다.

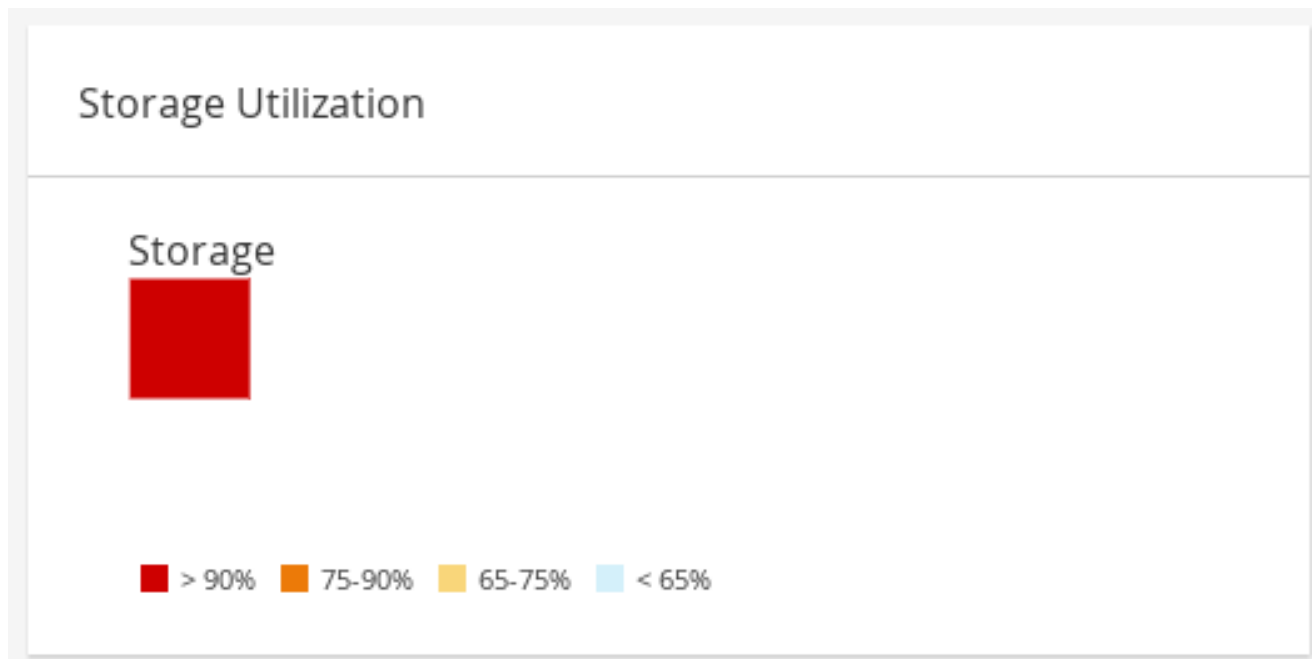
### 2.4.2. 메모리

지난 24시간 동안의 메모리의 평균 사용률을 보여주는 특정 클러스터의 메모리 사용률의 heatmap입니다. heatmap을 마우스로 가리키면 클러스터 이름이 표시됩니다. heatmap을 클릭하면 **Compute → Hosts** (컴퓨팅 호스트)로 이동하여 메모리 사용량에 따라 정렬된 특정 클러스터에서 검색 결과가 표시됩니다. 클러스터에서 메모리 사용량을 계산하는 데 사용되는 공식은 클러스터의 총 메모리 사용률(**GB**)입니다. 이는 지난 24시간 동안 각 호스트의 평균 호스트 메모리 사용률을 사용하여 클러스터의 총 메모리 사용량을 찾는 방식으로 계산됩니다.

## 2.5. 스토리지 사용률

스토리지 사용률 섹션에는 **heatmap**의 스토리지 사용률이 표시되어 있습니다.

그림 2.6. 스토리지 사용률



**heatmap**은 지난 **24**시간 동안의 스토리지의 평균 사용률을 보여줍니다. 클러스터에서 스토리지 사용량을 계산하는 데 사용되는 공식은 클러스터에서 스토리지의 총 사용률입니다. 이는 지난 **24**시간 동안 각 호스트의 평균 스토리지 사용률을 사용하여 클러스터의 스토리지 총 사용량을 찾는 방식으로 계산됩니다.

**heatmap**을 마우스로 가리키면 스토리지 도메인 이름이 표시됩니다. **heatmap**을 클릭하면 사용률에 따라 정렬된 스토리지 도메인이 있는 **Storage → Domains** 로 이동합니다.



## 3장. SEARCH

### 3.1. RED HAT VIRTUALIZATION에서 검색 수행

관리 포털을 사용하면 가상 머신, 호스트, 사용자 등과 같은 수천 개의 리소스를 관리할 수 있습니다. 검색을 수행하려면 각 리소스의 기본 페이지에서 사용할 수 있는 검색 쿼리(무료 텍스트 또는 구문 기반)를 검색 창에 입력합니다. 검색 쿼리는 향후 재사용을 위해 북마크로 저장할 수 있으므로 특정 검색 결과가 필요할 때마다 검색 쿼리를 다시 입력할 필요가 없습니다. 검색은 대소문자를 구분하지 않습니다.

### 3.2. 검색 구문 및 예

Red Hat Virtualization 리소스에 대한 검색 쿼리 구문은 다음과 같습니다.

결과 유형: `{crite:00:00} [sortby sort_spec]`

구문 예

다음 예제에서는 검색 쿼리 사용 방법을 설명하고 Red Hat Virtualization이 검색 쿼리 빌드에 도움이 되는 방법을 이해하는 데 도움이 됩니다.

표 3.1. 쿼리 검색 예

예제	결과
호스트: Vms.status = up page 2	가동 중인 가상 머신을 실행하는 모든 호스트 목록의 2 페이지를 표시합니다.
VM: domain =ECDHE.com	지정된 도메인에서 실행 중인 모든 가상 머신 목록을 표시합니다.
VMs: users.name =ECDHE	사용자 이름이 있는 사용자에 속하는 모든 가상 머신 목록을 표시합니다.
events: severity > Normal sortby time	심각도가 Normal보다 높은 모든 이벤트 목록을 시간순으로 정렬합니다.

### 3.3. 자동 완료 검색

관리 포털은 유효하고 강력한 검색 쿼리를 생성하는 데 도움이 되는 자동 완성 기능을 제공합니다. 검색 쿼리의 각 부분을 입력하면 검색바 아래에 검색 열기의 다음 부분에 대한 선택 드롭다운 목록이 표시됩니다. 목록에서 선택한 다음 검색의 다음 부분을 입력/선택하거나 옵션을 무시하고 계속 쿼리를 수동으로 입력할 수 있습니다.

다음 표는 관리 포털 자동 완성이 쿼리를 구성하는 데 도움이 되는 방법을 보여줍니다.

호스트: **Vms.status = down**

표 3.2. 자동 완료를 사용하여 쿼리 검색 예

INPUT	표시된 항목 나열	동작
h	호스트 (1 옵션만 해당)	호스트를 선택하거나 호스트 유형
호스트:	모든 호스트 속성	유형 v
호스트: v	v로 시작하는 호스트 속성	Vms 를 선택하거나 Vms를 입력합니다.
호스트: Vms	모든 가상 머신 속성	유형 s
호스트: Vms.s	s로 시작하는 모든 가상 머신 속성	상태 또는 유형 상태 선택
호스트: Vms.status	= !=	선택 또는 유형 =
Hosts: Vms.status =	모든 상태 값	선택 또는 유형

### 3.4. 결과 유형 옵션

결과 유형을 사용하면 다음 유형의 리소스를 검색할 수 있습니다.

- 가상 머신 목록용 VM
- 호스트 목록용 호스트
- 풀 목록용 풀
- 템플릿 목록에 대한 템플릿
- 이벤트 목록에 대한 이벤트
- 사용자 목록에 대한 사용자
- 클러스터 목록용 클러스터
- 데이터 센터 목록용 데이터 센터
- 스토리지 도메인 목록용 스토리지

각 리소스 유형에 고유한 속성 세트와 연결된 다른 리소스 유형 집합이 있으므로 각 검색 유형에는 유효한 구문 조합 세트가 있습니다. 자동 완성 기능을 사용하여 유효한 쿼리를 쉽게 생성할 수도 있습니다.

### 3.5. 검색 RESULTS

쿼리의 콜론 뒤에 검색 기준을 지정할 수 있습니다. {crite:00:00}의 구문은 다음과 같습니다.

<prop><operator><value>

또는

**<obj-type><prop><operator><value>**

예제

다음 표에서는 구문의 일부를 설명합니다.

표 3.3. 검색의 예

part	설명	값	예제	참고
Prop	검색된 리소스의 속성입니다. 리소스 유형의 속성( other other type 참조) 또는 <b>tag</b> (custom tag)일 수도 있습니다.	특정 속성이 있는 오브젝트로 검색을 제한합니다. 예를 들어 <b>상태</b> 속성이 있는 오브젝트를 검색합니다.	상태	해당 없음
obj-type	searched-for 리소스와 연결할 수 있는 리소스 유형입니다.	이는 데이터 센터 및 가상 시스템과 같은 시스템 오브젝트입니다.	사용자	해당 없음
operator	비교 연산자입니다.	=  != (동등하지 않음)  >  <  >=  <=	해당 없음	값 옵션은 속성에 따라 다릅니다.

part	설명	값	예제	참고
값	표현식이 비교되는 대상입니다.	문자열 정수 순위 날짜(지역 설정에 따라 포맷됨)	dotnet 256 Normal	<ul style="list-style-type: none"> <li>와일드카드는 문자열 내에서 사용할 수 있습니다.</li> <li>""(하나의 공백이 없는 따옴표 두 세트)를 사용하여 초기화되지 않은(비어 있음) 문자열을 나타낼 수 있습니다.</li> <li>따옴표를 문자열이나 공백을 포함하는 날짜를 사용해야 합니다.</li> </ul>

### 3.6. 검색: 여러 KNATIVE 및 와일드 카드

와일드카드는 문자열 구문의 `&lt;value&gt;` 부분에 사용할 수 있습니다. 예를 들어 `m` 으로 시작하는 모든 사용자를 찾으려면 `m*` 를 입력합니다.

부울 연산자 **AND** 및 **OR** 를 사용하여 두 개의 기준을 검색할 수 있습니다. 예를 들면 다음과 같습니다.

**VMs: users.name = m\* AND status = Up**

이 쿼리는 이름이 "m"로 시작하는 사용자의 실행 중인 모든 가상 머신을 반환합니다.

**VMs: users.name = m\* AND tag = "paris-loc"**

이 쿼리는 이름이 "m"로 시작하는 사용자에 대해 "paris-loc" 태그가 지정된 모든 가상 머신을 반환합니다.

**AND** 또는 **OR** 없이 두 기준이 지정되면 **AND**가 암시적으로 지정됩니다. **AND** 는 **OR** 앞에 있으며, **OR** 앞에 있는 **AND** .

### 3.7. SEARCH ORDER(검색 순서 결정)

`sortby` 를 사용하여 반환된 정보의 정렬 순서를 결정할 수 있습니다. 정렬 방향(예: `asc forECDHE` )을 포함할 수 있습니다.

예를 들면 다음과 같습니다.

**events: severity > 일반적인 분류 시간 desc**

이 쿼리는 심각도가 **Normal**보다 높은 모든 이벤트를 시간별 정렬(필요 순서)을 반환합니다.

### 3.8. 데이터 센터 검색

다음 표에서는 데이터 센터에 대한 모든 검색 옵션에 대해 설명합니다.

표 3.4. 데이터 센터 검색

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>cluster.cluster-prop</b>	속성 유형에 따라 다릅니다.	데이터 센터와 연결된 클러스터의 속성입니다.
<b>name</b>	문자열	데이터 센터의 이름입니다.
<b>description</b>	문자열	데이터 센터에 대한 설명입니다.
<b>type</b>	문자열	데이터 센터 유형입니다.
<b>status</b>	list	데이터 센터의 가용성.
<b>sortby</b>	list	반환된 결과를 리소스 속성 중 하나로 정렬합니다.
페이지	정수	표시할 결과 수입니다.

예제

**datacenter: type = nfs 및 status != up**

이 예에서는 스토리지 유형 **NFS** 및 **up** 이외의 상태의 데이터 센터 목록을 반환합니다.

### 3.9. 클러스터 검색

다음 표에서는 클러스터의 모든 검색 옵션을 설명합니다.

표 3.5. 클러스터 검색

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>datacenter.prop</b>	속성 유형에 따라 다릅니다.	클러스터와 연결된 데이터 센터의 속성입니다.
데이터 센터	문자열	클러스터가 속한 데이터 센터입니다.
<b>name</b>	문자열	네트워크에서 클러스터를 식별하는 고유 이름입니다.
<b>description</b>	문자열	클러스터에 대한 설명입니다.
초기화됨	문자열	클러스터 상태를 나타내는 true 또는 False입니다.
<b>sortby</b>	list	반환된 결과를 리소스 속성 중 하나로 정렬합니다.
페이지	정수	표시할 결과 수입니다.

예제

**clusters: initialized = true 또는 name = default**

이 예에서는 초기화되거나 **Default**라는 클러스터 목록을 반환합니다.

### 3.10. 호스트 검색

다음 표에서는 호스트의 모든 검색 옵션을 설명합니다.

표 3.6. 호스트 검색

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>VMs.Vms-prop</b>	속성 유형에 따라 다릅니다.	호스트와 연결된 가상 시스템의 속성입니다.

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>templates.templates-prop</b>	속성 유형에 따라 다릅니다.	호스트와 연결된 템플릿의 속성입니다.
<b>events.events-prop</b>	속성 유형에 따라 다릅니다.	호스트와 관련된 이벤트의 속성입니다.
<b>users.user-prop</b>	속성 유형에 따라 다릅니다.	호스트와 연결된 사용자의 속성입니다.
<b>name</b>	문자열	호스트 이름입니다.
<b>status</b>	list	호스트의 가용성입니다.
<b>external_status</b>	문자열	외부 시스템 및 플러그인에서 보고한 호스트의 상태.
<b>cluster</b>	문자열	호스트가 속한 클러스터입니다.
주소	문자열	네트워크에서 호스트를 식별하는 고유 이름입니다.
<b>cpu_usage</b>	정수	사용된 처리 성능의 백분율입니다.
<b>mem_usage</b>	정수	사용된 메모리 백분율입니다.
<b>network_usage</b>	정수	네트워크 사용 백분율입니다.
<b>load</b>	정수	지정된 시간 슬라이스의 프로세서당 실행 대기열에서 실행 대기 중인 작업입니다.
<b>version</b>	정수	운영 체제의 버전 번호입니다.
<b>cpus</b>	정수	호스트의 CPU 수입니다.
<b>memory</b>	정수	사용 가능한 메모리 양입니다.
<b>cpu_speed</b>	정수	CPU의 처리 속도
<b>cpu_model</b>	문자열	CPU 유형입니다.
<b>active_vms</b>	정수	현재 실행 중인 가상 머신 수입니다.
<b>migrating_vms</b>	정수	현재 마이그레이션 중인 가상 머신 수입니다.

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>committed_mem</b>	정수	커밋된 메모리의 백분율입니다.
<b>tag</b>	문자열	호스트에 할당된 태그입니다.
<b>type</b>	문자열	호스트 유형입니다.
데이터 센터	문자열	호스트가 속한 데이터 센터입니다.
<b>sortby</b>	list	반환된 결과를 리소스 속성 중 하나로 정렬합니다.
페이지	정수	표시할 결과 수입니다.

예제

호스트: **cluster = Default** 및 **Vms.os = rhel6**

이 예에서는 **Red Hat Enterprise Linux 6** 운영 체제를 실행하는 **Default** 클러스터 및 호스트 가상 머신의 일부인 호스트 목록을 반환합니다.

### 3.11. 네트워크 검색

다음 표에서는 네트워크의 모든 검색 옵션에 대해 설명합니다.

표 3.7. 네트워크 검색

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>cluster_network.clusternetw ork-prop</b>	속성 유형에 따라 다릅니다.	네트워크와 연결된 클러스터의 속성입니다.
<b>Host_Network.hostnetwork- prop</b>	속성 유형에 따라 다릅니다.	네트워크와 연결된 호스트의 속성입니다.
<b>name</b>	문자열	사람이 읽을 수 있는 이름을 네트워크를 식별합니다.
<b>description</b>	문자열	네트워크를 생성할 때 선택적으로 사용되는 네트워크를 설명하는 키워드 또는 텍스트입니다.



속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>vlanid</b>	정수	네트워크의 VLAN ID입니다.
<b>stp</b>	문자열	네트워크에서 STP(Spanning Tree Protocol)를 활성화 또는 비활성화 할지 여부입니다.
<b>mtu</b>	정수	논리 네트워크의 최대 전송 단위입니다.
<b>vmnetwork</b>	문자열	네트워크가 가상 머신 트래픽에만 사용되는지 여부입니다.
데이터 센터	문자열	네트워크가 연결된 데이터 센터입니다.
<b>sortby</b>	list	반환된 결과를 리소스 속성 중 하나로 정렬합니다.
페이지	정수	표시할 결과 수입니다.

예제

네트워크: **mtu > 1500** 및 **vmnetwork = true**

이 예에서는 최대 전송 단위가 **1500**바이트를 초과하는 네트워크 목록을 반환하며 가상 머신에서만 사용하도록 설정됩니다.

### 3.12. 스토리지 검색

다음 표에서는 스토리지에 대한 모든 검색 옵션에 대해 설명합니다.

표 3.8. 스토리지 검색

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>hosts-prop</b>	속성 유형에 따라 다릅니다.	스토리지와 연결된 호스트의 속성입니다.
<b>cluster.cluster-prop</b>	속성 유형에 따라 다릅니다.	스토리지와 연결된 클러스터의 속성입니다.

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>name</b>	문자열	네트워크에서 스토리지를 식별하는 고유 이름입니다.
<b>status</b>	문자열	스토리지 도메인의 상태입니다.
<b>external_status</b>	문자열	외부 시스템 및 플러그인에서 보고하는 스토리지 도메인의 상태.
데이터 센터	문자열	스토리지가 속한 데이터 센터입니다.
<b>type</b>	문자열	스토리지의 유형입니다.
크기	정수	스토리지의 크기입니다.
사용됨	정수	사용되는 스토리지의 양입니다.
<b>committed</b>	정수	커밋된 스토리지의 양입니다.
<b>sortby</b>	list	반환된 결과를 리소스 속성 중 하나로 정렬합니다.
페이지	정수	표시할 결과 수입니다.

예제

스토리지: 크기 > 200 또는 사용됨 < 50

이 예에서는 총 스토리지 공간이 **200GB**보다 크거나 사용된 스토리지 공간이 **50GB** 미만인 스토리지 목록을 반환합니다.

### 3.13. 디스크 검색

다음 표에서는 디스크에 대한 모든 검색 옵션에 대해 설명합니다.



## 참고

디스크 유형 및 콘텐츠 유형 필터링 옵션을 사용하여 표시된 가상 디스크의 수를 줄일 수 있습니다.

표 3.9. 디스크 검색

속성(리소스 또는 리소스 유형)	유형	설명(참조)
데이터 센터 ( <b>datacenter-prop</b> )	속성 유형에 따라 다릅니다.	디스크와 연결된 데이터 센터의 속성입니다.
스토리지. <b>스토리지-prop</b>	속성 유형에 따라 다릅니다.	디스크와 연결된 스토리지의 속성입니다.
별칭	문자열	네트워크에서 스토리지를 식별하는 사람이 읽을 수 있는 이름입니다.
<b>description</b>	문자열	디스크를 생성하는 경우 선택적으로 사용되는 키워드 또는 텍스트입니다.
<b>provisioned_size</b>	정수	디스크의 가상 크기입니다.
크기	정수	디스크의 크기입니다.
<b>actual_size</b>	정수	디스크에 할당된 실제 크기입니다.
<b>creation_date</b>	정수	디스크가 생성된 날짜입니다.
부팅 가능	문자열	디스크를 부팅할 수 있는지 여부입니다. 유효한 값은 <b>0,1,yes</b> 또는 <b>no</b> 중 하나입니다.
공유 가능	문자열	디스크를 한 번에 두 개 이상의 가상 머신에 연결할 수 있는지 여부입니다. 유효한 값은 <b>0,1,yes</b> 또는 <b>no</b> 중 하나입니다.
형식	문자열	디스크 형식입니다. 사용되지 않는, 할당되지 않은, 소가 또는 원시 중 하나일 수 있습니다.
<b>status</b>	문자열	디스크 상태입니다. 할당되지 않은, <b>ok,locked,invalid</b> 또는 <b>illegal</b> 중 하나일 수 있습니다.

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>disk_type</b>	문자열	디스크의 유형입니다. 이미지 또는 <b>lun</b> 중 하나일 수 있습니다.
<b>number_of_vms</b>	정수	디스크를 연결할 가상 머신 수입니다.
<b>vm_names</b>	문자열	디스크가 연결된 가상 시스템의 이름입니다.
할당량	문자열	가상 디스크에 적용되는 할당량의 이름입니다.
<b>sortby</b>	list	반환된 결과를 리소스 속성 중 하나로 정렬합니다.
페이지	정수	표시할 결과 수입니다.

예제

**disks: format = fraction 및 provisioned\_size > 8**

이 예에서는 **QCOW** 형식이 있는 가상 디스크 목록과 할당된 디스크 크기가 **8GB**보다 큼니다.

### 3.14. 볼륨 검색

다음 표에서는 볼륨에 대한 모든 검색 옵션에 대해 설명합니다.

표 3.10. 볼륨 검색

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>Cluster</b>	문자열	볼륨과 연결된 클러스터의 이름입니다.
<b>cluster-prop</b>	속성 유형(예: name, description, comment, architecture)	볼륨과 연결된 클러스터의 속성입니다.
<b>name</b>	문자열	볼륨을 식별하는 사람이 읽을 수 있는 이름입니다.

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>type</b>	문자열	distribute, replicate, distributed_replicate, strip 또는 distributed_stripe 중 하나일 수 있습니다.
<b>transport_type</b>	정수	TCP 또는 RDMA 중 하나일 수 있습니다.
<b>replica_count</b>	정수	복제본 수입니다.
<b>stripe_count</b>	정수	스트라이프 수.
<b>status</b>	문자열	블롭 상태. Up 또는 Down 중 하나일 수 있습니다.
<b>sortby</b>	list	반환된 결과를 리소스 속성 중 하나로 정렬합니다.
페이지	정수	표시할 결과 수입니다.

예제

블롭: **transport\_type = rdma** 및 **strip\_count >= 2**

이 예에서는 전송 유형이 **RDMA**로 설정된 블롭 목록과 **2**개 이상의 스트라이프를 반환합니다.

### 3.15. 가상 머신 검색

다음 표에서는 가상 머신에 대한 모든 검색 옵션을 설명합니다.



참고

현재 **Network Label, Custom Emulated Machine, Custom CPU Type** 및 **Custom CPU Type** 속성은 지원되지 않습니다.

표 3.11. 가상 머신 검색

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>hosts-prop</b>	속성 유형에 따라 다릅니다.	가상 시스템과 연결된 호스트의 속성입니다.
<b>templates.templates-prop</b>	속성 유형에 따라 다릅니다.	가상 머신과 연결된 템플릿의 속성입니다.
<b>events.events-prop</b>	속성 유형에 따라 다릅니다.	가상 머신과 관련된 이벤트의 속성입니다.
<b>users.user-prop</b>	속성 유형에 따라 다릅니다.	가상 시스템과 연결된 사용자의 속성입니다.
스토리지. <b>storage-prop</b>	속성 유형에 따라 다릅니다.	가상 시스템과 연결된 스토리지 장치의 속성입니다.
<b>vNIC.vnic-prop</b>	속성 유형에 따라 다릅니다.	가상 시스템과 연결된 vNIC의 속성입니다.
<b>name</b>	문자열	가상 머신의 이름입니다.
<b>status</b>	list	가상 머신의 가용성입니다.
<b>ip</b>	정수	가상 머신의 IP 주소입니다.
가동 시간	정수	가상 시스템이 실행 중인 시간(분)입니다.
<b>domain</b>	문자열	이러한 시스템을 그룹화하는 도메인(일반적으로 Active Directory 도메인)입니다.
<b>os</b>	문자열	가상 머신이 생성될 때 선택한 운영 체제입니다.
<b>creationDate</b>	날짜	가상 머신이 생성된 날짜입니다.
주소	문자열	네트워크에서 가상 머신을 식별하는 고유 이름입니다.
<b>cpu_usage</b>	정수	사용된 처리 성능의 백분율입니다.
<b>mem_usage</b>	정수	사용된 메모리 백분율입니다.
<b>network_usage</b>	정수	사용된 네트워크의 백분율입니다.
<b>memory</b>	정수	정의된 최대 메모리입니다.

속성(리소스 또는 리소스 유형)	유형	설명(참조)
앱	문자열	현재 가상 머신에 설치된 애플리케이션입니다.
<b>cluster</b>	list	가상 머신이 속한 클러스터입니다.
<b>pool</b>	list	가상 머신이 속한 가상 머신 풀입니다.
<b>loggedinuser</b>	문자열	가상 머신에 현재 로그인한 사용자의 이름입니다.
<b>tag</b>	list	가상 머신이 속한 태그입니다.
데이터 센터	문자열	가상 머신이 속한 데이터 센터입니다.
<b>type</b>	list	가상 머신 유형(서버 또는 데스크탑)입니다.
할당량	문자열	가상 머신과 연결된 할당량의 이름입니다.
<b>description</b>	문자열	가상 머신을 생성할 때 필요한 경우 가상 머신을 설명하는 키워드 또는 텍스트입니다.
<b>sortby</b>	list	반환된 결과를 리소스 속성 중 하나로 정렬합니다.
페이지	정수	표시할 결과 수입니다.
<b>next_run_configuration_exists</b>	부울	가상 머신에 보류 중인 구성 변경 사항이 있습니다.

예제

**VMs: template.name = Win\* 및 user.name = ""**

이 예에서는 기본 템플릿 이름이 **Win** 로 시작하고 모든 사용자에게 할당된 가상 머신 목록을 반환합니다.

예제

**VM: cluster = default 및 os = windows7**

이 예에서는 **Default** 클러스터에 속하고 **Windows 7**을 실행하는 가상 머신 목록을 반환합니다.

### 3.16. 풀 검색

다음 표에서는 풀에 대한 모든 검색 옵션에 대해 설명합니다.

표 3.12. 풀 검색

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>name</b>	문자열	풀의 이름입니다.
<b>description</b>	문자열	풀에 대한 설명입니다.
<b>type</b>	list	풀 유형입니다.
<b>sortby</b>	list	반환된 결과를 리소스 속성 중 하나로 정렬합니다.
페이지	정수	표시할 결과 수입니다.

예제

**pool: type = automatic**

이 예에서는 유형이 자동인 풀 목록을 반환합니다.

### 3.17. 템플릿 검색

다음 표에서는 템플릿에 대한 모든 검색 옵션에 대해 설명합니다.

표 3.13. 템플릿 검색



속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>VMs. Vms-prop</b>	문자열	템플릿과 관련된 가상 시스템의 속성입니다.
<b>hosts-prop</b>	문자열	템플릿과 관련된 호스트의 속성입니다.
<b>events.events-prop</b>	문자열	템플릿과 관련된 이벤트의 속성입니다.
<b>users. user-prop</b>	문자열	템플릿과 연결된 사용자의 속성입니다.
<b>name</b>	문자열	템플릿의 이름입니다.
<b>domain</b>	문자열	템플릿의 도메인입니다.
<b>os</b>	문자열	운영 체제 유형입니다.
<b>creationDate</b>	정수	템플릿이 생성된 날짜입니다. 날짜 형식은 <b>mm/dd/yy</b> 입니다.
<b>childcount</b>	정수	템플릿에서 생성된 가상 머신 수입니다.
<b>mem</b>	정수	정의된 메모리입니다.
<b>description</b>	문자열	템플릿에 대한 설명입니다.
<b>status</b>	문자열	템플릿의 상태입니다.
<b>cluster</b>	문자열	템플릿과 연결된 클러스터입니다.
데이터 센터	문자열	템플릿과 연결된 데이터 센터입니다.
할당량	문자열	템플릿과 연결된 할당량입니다.
<b>sortby</b>	list	반환된 결과를 리소스 속성 중 하나로 정렬합니다.
페이지	정수	표시할 결과 수입니다.

예제

템플릿: **Events.severity >=** 정상 및 **Vms.uptime > 0**

이 예에서는 템플릿에서 파생된 가상 머신에서 일반 또는 그 이상의 심각도의 이벤트가 발생하고 가상 머신이 여전히 실행 중인 템플릿 목록을 반환합니다.

### 3.18. 사용자 검색

다음 표에서는 사용자를 위한 모든 검색 옵션에 대해 설명합니다.

표 3.14. 사용자 검색

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>VMs.Vms-prop</b>	속성 유형에 따라 다릅니다.	사용자와 연결된 가상 시스템의 속성입니다.
<b>hosts-prop</b>	속성 유형에 따라 다릅니다.	사용자와 연결된 호스트의 속성입니다.
<b>templates.templates-prop</b>	속성 유형에 따라 다릅니다.	사용자와 연결된 템플릿의 속성입니다.
<b>events.events-prop</b>	속성 유형에 따라 다릅니다.	사용자와 관련된 이벤트의 속성입니다.
<b>name</b>	문자열	사용자 이름입니다.
<b>LastName</b>	문자열	사용자의 마지막 이름입니다.
<b>username</b>	문자열	사용자의 고유 이름입니다.
<b>department</b>	문자열	사용자가 속한 부서입니다.
<b>group</b>	문자열	사용자가 속한 그룹입니다.
<b>title</b>	문자열	사용자의 제목입니다.
<b>status</b>	문자열	사용자의 상태.
<b>role</b>	문자열	사용자의 역할입니다.
<b>tag</b>	문자열	사용자가 속한 태그입니다.
<b>pool</b>	문자열	사용자가 속한 풀입니다.

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>sortby</b>	list	반환된 결과를 리소스 속성 중 하나로 정렬합니다.
페이지	정수	표시할 결과 수입니다.

예제

사용자: **Events.severity > 정상** 및 **Vms.status = up** 또는 **Vms.status = pause**

이 예에서는 가상 머신에서 일반 심각도보다 큰 이벤트가 발생한 사용자 목록을 반환하고 가상 머신이 계속 실행 중이거나 사용자의 가상 머신이 일시 중지됩니다.

### 3.19. 이벤트 검색

다음 표에서는 이벤트를 검색하는 데 사용할 수 있는 모든 검색 옵션에 대해 설명합니다. 자동 완성은 많은 옵션에 적절하게 제공됩니다.

표 3.15. 이벤트 검색

속성(리소스 또는 리소스 유형)	유형	설명(참조)
<b>VMs.Vms-prop</b>	속성 유형에 따라 다릅니다.	이벤트와 관련된 가상 머신의 속성입니다.
<b>hosts-prop</b>	속성 유형에 따라 다릅니다.	이벤트와 관련된 호스트의 속성입니다.
<b>templates.templates-prop</b>	속성 유형에 따라 다릅니다.	이벤트와 관련된 템플릿의 속성입니다.
<b>users.user-prop</b>	속성 유형에 따라 다릅니다.	이벤트와 관련된 사용자의 속성입니다.
<b>cluster.cluster-prop</b>	속성 유형에 따라 다릅니다.	이벤트와 관련된 클러스터의 속성입니다.
<b>volumes.Volumes-prop</b>	속성 유형에 따라 다릅니다.	이벤트와 관련된 볼륨의 속성입니다.
<b>type</b>	list	이벤트 유형입니다.

속성(리소스 또는 리소스 유형)	유형	설명(참조)
심각도	list	이벤트의 심각도: Warning/Error/Normal.
message	문자열	이벤트 유형에 대한 설명입니다.
time	list	이벤트가 발생한 날입니다.
username	문자열	이벤트와 관련된 사용자 이름입니다.
event_host	문자열	이벤트와 관련된 호스트입니다.
event_vm	문자열	이벤트와 관련된 가상 머신입니다.
event_template	문자열	이벤트와 관련된 템플릿입니다.
event_storage	문자열	이벤트와 관련된 스토리지입니다.
event_datacenter	문자열	이벤트와 관련된 데이터 센터입니다.
event_volume	문자열	이벤트와 연결된 볼륨입니다.
correlation_id	정수	이벤트의 식별 번호입니다.
sortby	list	반환된 결과를 리소스 속성 중 하나로 정렬합니다.
페이지	정수	표시할 결과 수입니다.

예제

**events: Vms.name = testECDHE 및 Hosts.name = gonECDHE.example.com**

이 예에서는 호스트 **gonECDHE.example.com** 에서 실행되는 동안 **test 4.6.1**이라는 가상 머신에서 이벤트가 발생한 이벤트 목록을 반환합니다.

## 4장. 북마크

### 4.1. 쿼리 문자열을 **BOOKMARK**로 저장

북마크를 사용하여 검색 쿼리를 생각하고 다른 사용자와 공유할 수 있습니다.

쿼리 문자열을 **Bookmark**로 저장

1. 검색 창에 원하는 검색 쿼리를 입력하고 검색을 수행합니다.
2. 검색바 오른쪽에 있는 **star-shaped Bookmark** 버튼을 클릭하여 새 북마크 창을 엽니다.
3. 북마크의 이름을 입력합니다.
4. 필요한 경우 검색 문자열 필드를 편집합니다.
5. **OK**를 클릭합니다.


헤더 표시줄에서 **Bookmarks** 아이콘()

)을 클릭하여 북마크를 찾아서 선택합니다.

### 4.2. 북마크 편집

북마크의 이름 및 검색 문자열을 수정할 수 있습니다.

북마크 편집


1. 제목 표시줄에서 **Bookmarks** 아이콘()을 클릭합니다.

2. 북마크를 선택하고 편집 을 클릭합니다.
3. 필요에 따라 **Name** 및 **Search** 문자열 필드를 변경합니다.
4. **OK**를 클릭합니다.

### 4.3. 북마크 삭제

북마크가 더 이상 필요하지 않으면 제거합니다.

북마크 삭제

1. 제목 표시줄에서 **Bookmarks** 아이콘()을 클릭합니다.
2. 북마크를 선택하고 제거를 클릭합니다.
3. **OK**를 클릭합니다.

## 5장. 태그

### 5.1. 태그를 사용하여 RED HAT VIRTUALIZATION으로 상호 작용 사용자 지정

**Red Hat Virtualization** 플랫폼을 요구 사항에 맞게 설정하고 구성된 후에는 태그를 사용하여 작업 방식을 사용자 지정할 수 있습니다. 태그를 사용하면 시스템 리소스를 그룹 또는 카테고리로 구성할 수 있습니다. 이는 가상화 환경에 많은 개체가 있고 관리자가 특정 세트에 집중하려는 경우 유용합니다.

이 섹션에서는 태그를 생성 및 편집하고 호스트 또는 가상 시스템에 할당한 다음 태그를 기준으로 사용하여 검색하는 방법에 대해 설명합니다. 태그는 엔터프라이즈의 요구 사항에 맞게 구조와 일치하는 계층으로 정렬될 수 있습니다.

관리 포털 태그를 생성, 수정, 제거하려면 헤더 표시줄에서 태그 아이콘(




)을 클릭합니다.

### 5.2. 태그 생성

태그를 사용하여 검색 결과를 필터링할 수 있도록 태그를 생성합니다.


태그 생성

1.
  - 제목 표시줄에서 태그 아이콘(
 
)을 클릭합니다.
2.
  - 추가를 클릭하여 새 태그를 만들거나 태그를 선택하고 새로 생성을 클릭하여 하위 태그를 만듭니다.
3.
  - 새 태그의 이름 및 설명을 입력합니다.
4.
  - OK**를 클릭합니다.

### 5.3. 태그 수정

태그의 이름 및 설명을 편집할 수 있습니다.


#### 태그 수정

1. 제목 표시줄에서 태그 아이콘()을 클릭합니다.
2. 수정할 태그를 선택하고 편집 을 클릭합니다.
3. 필요에 따라 **Name** 및 **Description** 필드를 변경합니다.
4. **OK**를 클릭합니다.

#### 5.4. 태그 삭제

태그가 더 이상 필요하지 않으면 제거합니다.

#### 태그 삭제

1. 제목 표시줄에서 태그 아이콘()을 클릭합니다.
2. 삭제할 태그를 선택하고 제거를 클릭합니다. 이 태그를 제거하면 태그의 하위 항목도 모두 제거된다는 메시지가 경고됩니다.
3. **OK**를 클릭합니다.


태그와 모든 하위 항목이 제거되었습니다. 이 태그는 연결된 모든 오브젝트에서도 제거됩니다.



## 5.5. 오브젝트에서 태그 추가 및 제거

호스트, 가상 시스템 및 사용자로부터 태그를 할당하고 제거할 수 있습니다.

오브젝트에서 태그 추가 및 제거

1. 태그하거나 태그 해제할 오브젝트를 선택합니다.
2. 더 많은 작업 (  )을 클릭한 다음 태그 할당 을 클릭합니다.
3. 확인란을 선택하여 개체에 태그를 할당하거나 확인란을 선택하여 개체에서 태그를 분리합니다.
4. **OK**를 클릭합니다.

지정된 태그가 선택한 오브젝트의 사용자 지정 속성으로 추가 또는 제거되었습니다.

## 5.6. 태그를 사용하여 오브젝트 검색

**tag** 를 속성으로 사용하고 원하는 값 또는 검색 기준으로 값을 사용하여 검색 쿼리를 입력합니다.

지정된 기준에 태그된 오브젝트가 결과 목록에 나열됩니다.



참고



**tag** 를 속성으로 사용하고, **!=!** 태그를 사용하여 오브젝트를 검색하는 경우(예: **Host: Vms.tag!=server1**) 결과 목록에 태그되지 않은 오브젝트가 포함되지 않습니다.

## 5.7. 태그를 사용하여 호스트 사용자 정의

태그를 사용하여 호스트에 대한 정보를 저장할 수 있습니다. 그런 다음 태그를 기반으로 호스트를 검색

할 수 있습니다. 검색에 대한 자세한 내용은 [3장. Search](#) 에서 참조하십시오.

태그로 호스트 사용자 정의

1. **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.
2.  더 많은 작업 (  )을 클릭한 다음 태그 할당 을 클릭합니다.
3. 적용 가능한 태그의 확인란을 선택합니다.
4. **OK**를 클릭합니다.

호스트에 대한 검색 가능한 추가 정보를 태그로 추가했습니다.

II 부. 리소스 확보

## 6장. 서비스 품질

**Red Hat Virtualization**을 사용하면 환경에 액세스할 수 있는 입력 및 출력, 처리 및 네트워킹 기능 수준을 세밀하게 제어할 수 있는 서비스 품질의 기능을 정의할 수 있습니다. 서비스 품질 항목은 데이터 센터 수준에서 정의되며 클러스터 및 스토리지 도메인에서 생성된 프로필에 할당됩니다. 그러면 프로필이 생성된 클러스터 및 스토리지 도메인의 개별 리소스에 이러한 프로필이 할당됩니다.

### 6.1. 스토리지 서비스 품질

스토리지 품질은 최대 처리량 수준과 스토리지 도메인의 가상 디스크에 대한 최대 입력 및 출력 작업을 정의합니다. 스토리지 품질의 서비스를 가상 디스크에 할당하면 스토리지 도메인의 성능을 미세 조정하고 한 가상 디스크와 관련된 스토리지 작업이 동일한 스토리지 도메인에 호스팅되는 다른 가상 디스크에 사용 가능한 스토리지 기능에 영향을 미치지 않습니다.

#### 6.1.1. 서비스 입력의 스토리지 품질 생성

서비스 입력의 스토리지 품질 생성

1. **Compute** → **Data Centers** (데이터 센터)를 클릭합니다.
2. 데이터 센터의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **QoS** 탭을 클릭합니다.
4. 스토리지 에서 새로 생성을 클릭합니다.
5. **QoS** 이름 및 서비스 항목에 대한 설명을 입력합니다.
6. 라디오 버튼 중 하나를 클릭하여 처리량 품질을 지정합니다.
  - 없음
  - **total - MB/s** 필드에서 허용되는 최대 처리량을 입력합니다.

- **read /write** - 왼쪽 **MB/s** 필드에서 읽기 작업에 허용되는 최대 처리량과 오른쪽 **MB/s** 필드에서 쓰기 작업에 허용되는 최대 처리량을 입력합니다.
7. 라디오 버튼 중 하나를 클릭하여 입력 및 출력 (**I/Ops**) 서비스 품질을 지정합니다.
- 없음
  - **total - I/Ops** 필드에서 허용되는 최대 입력 및 출력 작업 수를 입력합니다.
  - 읽기/쓰기 - 왼쪽 **I/Ops** 필드에서 초당 허용되는 최대 입력 작업 수와 오른쪽 **I/Ops** 필드에서 초당 허용되는 최대 출력 작업 수를 입력합니다.
8. **OK**를 클릭합니다.

스토리지 품질의 서비스 항목을 생성하고 데이터 센터에 속한 데이터 스토리지 도메인의 해당 항목을 기반으로 디스크 프로필을 생성할 수 있습니다.

### 6.1.2. 서비스 입력의 스토리지 품질 제거

기존 스토리지 품질의 서비스 항목을 제거합니다.

서비스 입력의 스토리지 품질 제거

1. **Compute** → **Data Centers** (데이터 센터)를 클릭합니다.
2. 데이터 센터의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **QoS** 탭을 클릭합니다.
4. 스토리지 에서 서비스 항목의 스토리지 품질을 선택하고 제거를 클릭합니다.

5.

**OK**를 클릭합니다.

디스크 프로필이 해당 항목을 기반으로 한 경우 해당 프로필에 대한 서비스 품질 항목이 자동으로 **[unlimited]**로 설정됩니다.

## 6.2. 가상 머신 네트워크 서비스 품질

가상 머신 네트워크 품질 서비스는 개별 가상 네트워크 인터페이스 컨트롤러의 인바운드 및 아웃바운드 트래픽을 모두 제한하기 위한 프로필을 생성할 수 있는 기능입니다. 이 기능을 사용하면 네트워크 리소스 사용을 제어하는 여러 계층에서 대역폭을 제한할 수 있습니다.

### 6.2.1. 가상 머신 네트워크 서비스 품질 생성

가상 머신 네트워크 인터페이스 프로필이라고도 하는 **vNIC**(가상 네트워크 인터페이스 컨트롤러) 프로필에 적용할 때 네트워크 트래픽을 규제하기 위해 가상 머신 네트워크 품질 항목을 생성합니다.

가상 머신 네트워크 서비스 품질 생성

1.

**Compute** → **Data Centers** (데이터 센터)를 클릭합니다.

2.

데이터 센터의 이름을 클릭하여 세부 정보 보기를 엽니다.

3.

**QoS** 탭을 클릭합니다.

4.

**VM Network** 에서 새로 생성을 클릭합니다.

5.

가상 머신 네트워크 품질의 서비스 항목의 이름을 입력합니다.

6.

인바운드 및 아웃 바운드 네트워크 트래픽에 대한 제한을 입력합니다.

7.

**OK**를 클릭합니다.

가상 네트워크 인터페이스 컨트롤러에서 사용할 수 있는 가상 머신 네트워크 서비스 품질 항목을 생성했습니다.

## 6.2.2. 새로운 가상 머신 네트워크 QoS의 설정 및 가상 머신 네트워크 QoS(Virtual Machine Network QoS 설명) 편집

가상 머신 네트워크 품질 설정을 사용하면 세 가지 별도의 수준에서 인바운드 및 아웃바운드 트래픽에 대한 대역폭 제한을 구성할 수 있습니다.

표 6.1. 가상 머신 네트워크 QoS 설정

필드 이름	설명
데이터 센터	가상 머신 네트워크 QoS 정책을 추가할 데이터 센터입니다. 이 필드는 선택한 데이터 센터에 따라 자동으로 구성됩니다.
이름	Manager 내에서 가상 머신 네트워크 QoS 정책을 나타내는 이름입니다.
인바운드	인바운드 트래픽에 적용할 설정입니다. <b>인바운드</b> 확인란을 선택하거나 지우면 이러한 설정을 활성화하거나 비활성화합니다. <ul style="list-style-type: none"> <li>● <b>평균</b>: 인바운드 트래픽의 평균 속도입니다.</li> <li>● <b>peak</b>: 최대 시간 동안 인바운드 트래픽의 속도입니다.</li> <li>● <b>burst</b>: 버스트 중 인바운드 트래픽의 속도</li> </ul>
아웃바운드	아웃바운드 트래픽에 적용할 설정입니다. 이러한 설정을 활성화하거나 비활성화하려면 <b>아웃 바운드</b> 확인란을 선택하거나 지웁니다. <ul style="list-style-type: none"> <li>● <b>평균</b>: 아웃 바운드 트래픽의 평균 속도입니다.</li> <li>● <b>peak</b>: 최대 시간 동안 아웃 바운드 트래픽의 속도입니다.</li> <li>● <b>burst</b>: 버스트 중 아웃 바운드 트래픽의 속도.</li> </ul>

**Average, Peak** 또는 **Burst** 필드에서 허용되는 최대 값을 변경하려면 **engine-config** 명령을 사용하여 **MaxAverageNetworkQoSValue, MaxPeak NetworkQoSValue** 구성 키의 값을 변경합니다. 변경 사항을 적용하려면 **ovirt-engine** 서비스를 다시 시작해야 합니다. 예를 들면 다음과 같습니다.

```
# engine-config -s MaxAverageNetworkQoSValue=2048
# systemctl restart ovirt-engine
```

### 6.2.3. 가상 머신 네트워크 서비스 품질 제거

기존 가상 머신 네트워크 서비스 품질 항목을 제거합니다.

가상 머신 네트워크 서비스 품질 제거

1. **Compute** → **Data Centers** (데이터 센터)를 클릭합니다.
2. 데이터 센터의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **QoS** 탭을 클릭합니다.
4. **VM Network** 에서 가상 머신 네트워크 서비스 품질 항목을 선택하고 제거를 클릭합니다.
5. **OK**를 클릭합니다.

### 6.3. 호스트 네트워크 서비스 품질

호스트 네트워크 품질 서비스는 물리적 인터페이스를 통해 네트워크 트래픽을 제어할 수 있도록 호스트에서 네트워크를 구성합니다. 호스트 네트워크 품질 서비스를 사용하면 동일한 물리적 네트워크 인터페이스 컨트롤러에서 네트워크 리소스의 사용을 제어하여 네트워크 성능을 미세 조정할 수 있습니다. 이를 통해 한 네트워크에서 동일한 물리적 네트워크 인터페이스 컨트롤러에 연결된 다른 네트워크가 더 이상 트래픽이 많이 작동하지 않는 상황을 방지할 수 있습니다. 이러한 네트워크는 호스트 네트워크 서비스 품질을 구성함으로써 이제 정체 문제 없이 동일한 물리적 네트워크 인터페이스 컨트롤러에서 작동할 수 있습니다.

#### 6.3.1. 호스트 네트워크 서비스 품질 생성

호스트 네트워크 서비스 품질 항목을 생성합니다.

호스트 네트워크 서비스 품질 생성



1. **Compute** → **Data Centers** (데이터 센터)를 클릭합니다.
2. 데이터 센터의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **QoS** 탭을 클릭합니다.
4. **Host Network** 에서 새로 생성을 클릭합니다.
5. **Qos Name** 을 입력하고 서비스 항목에 대한 설명을 입력합니다.
6. **Weighted -4.6 ,Rate Limit [Mbps]** 및 **Rate Limit [Mbps]** 에 대해 원하는 값을 입력합니다.
7. **OK**를 클릭합니다.

### 6.3.2. 새로운 호스트 네트워크 서비스 품질의 설정 및 Windows의 호스트 네트워크 품질 편집

호스트 네트워크 품질의 서비스 설정을 사용하면 아웃바운드 트래픽에 대한 대역폭 제한을 구성할 수 있습니다.

표 6.2. 호스트 네트워크 QoS 설정

필드 이름	설명
데이터 센터	호스트 네트워크 QoS 정책을 추가할 데이터 센터입니다. 이 필드는 선택한 데이터 센터에 따라 자동으로 구성됩니다.
QoS 이름	Manager 내에서 호스트 네트워크 QoS 정책을 나타내는 이름입니다.
설명	호스트 네트워크 QoS 정책에 대한 설명입니다.

필드 이름	설명
아웃바운드	<p>아웃바운드 트래픽에 적용할 설정입니다.</p> <ul style="list-style-type: none"> <li>● <b>weighted DestinationRule:</b> 동일한 논리 링크에 연결된 다른 네트워크를 기준으로 특정 네트워크를 할당해야 하는 논리 링크 용량의 양을 나타냅니다. 정확한 공유는 해당 링크의 모든 네트워크 공유 합계에 따라 다릅니다. 기본적으로 이 값은 1~100 범위의 숫자입니다.</li> <li>● <b>속도 제한 [Mbps]:</b> 네트워크에서 사용할 최대 대역폭입니다.</li> <li>● <b>committed Rate [Mbps]:</b> 네트워크에 필요한 최소 대역폭입니다. 요청된 Rate는 보장되지 않으며 동일한 논리 링크의 다른 네트워크에서 요청하는 네트워크 인프라 및 요청된 비율에 따라 다릅니다.</li> </ul>

**Rate Limit [Mbps]** 또는 **ECDHEted Rate [Mbps]** 필드에서 허용되는 최대 값을 변경하려면 **engine-config** 명령을 사용하여 **MaxAverageNetworkQoSValue** 구성 키의 값을 변경합니다. 변경 사항을 적용하려면 **ovirt-engine** 서비스를 다시 시작해야 합니다. 예를 들면 다음과 같습니다.

```
# engine-config -s MaxAverageNetworkQoSValue=2048
# systemctl restart ovirt-engine
```

### 6.3.3. 호스트 네트워크 서비스 품질 제거

기존 네트워크 서비스 품질 항목을 제거합니다.

호스트 네트워크 서비스 품질 제거

1. **Compute** → **Data Centers** (데이터 센터)를 클릭합니다.
2. 데이터 센터의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **QoS** 탭을 클릭합니다.
4. **Host Network** 에서 호스트 네트워크 서비스 항목을 선택하고 제거를 클릭합니다.

5. 메시지가 표시되면 **OK** 를 클릭합니다.

## 6.4. CPU 서비스 품질

**CPU** 품질은 가상 머신이 실행되는 호스트에서 액세스할 수 있는 최대 처리 기능을 정의합니다. 이는 해당 호스트에서 사용 가능한 총 처리 기능의 백분율로 표시됩니다. 가상 머신에 **CPU** 품질을 할당하면 클러스터의 한 가상 머신의 워크로드가 해당 클러스터의 다른 가상 머신에 사용 가능한 처리 리소스에 영향을 미치지 않도록 할 수 있습니다.

### 6.4.1. CPU Quality of Service Entry 생성

**CPU** 품질의 서비스 항목을 생성합니다.

#### CPU Quality of Service Entry 생성

1. **Compute** → **Data Centers** (데이터 센터)를 클릭합니다.
2. 데이터 센터의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **QoS** 탭을 클릭합니다.
4. **CPU** 에서 새로 생성을 클릭합니다.
5. **QoS** 이름 및 서비스 항목에 대한 설명을 입력합니다.
6. **Limit(제한)** 필드에 서비스 항목이 허용하는 최대 처리 기능을 입력합니다. % 기호는 포함하지 마십시오.
7. **OK**를 클릭합니다.

**CPU** 품질의 서비스 항목을 생성했으며 데이터 센터에 속한 클러스터의 해당 항목을 기반으로 **CPU** 프로필을 생성할 수 있습니다.

## 6.4.2. CPU Quality of Service Entry 제거

기존 **CPU** 품질의 서비스 항목을 제거합니다.

### CPU Quality of Service Entry 제거

1. **Compute** → **Data Centers** (데이터 센터)를 클릭합니다.
2. 데이터 센터의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **QoS** 탭을 클릭합니다.
4. **CPU** 에서 **CPU** 품질의 서비스 항목을 선택하고 제거를 클릭합니다.
5. **OK**를 클릭합니다.

모든 **CPU** 프로필이 해당 항목을 기반으로 하는 경우 해당 프로필에 대한 서비스 항목의 **CPU** 품질이 자동으로 **[unlimited]** 로 설정됩니다.

## 7장. 데이터 센터

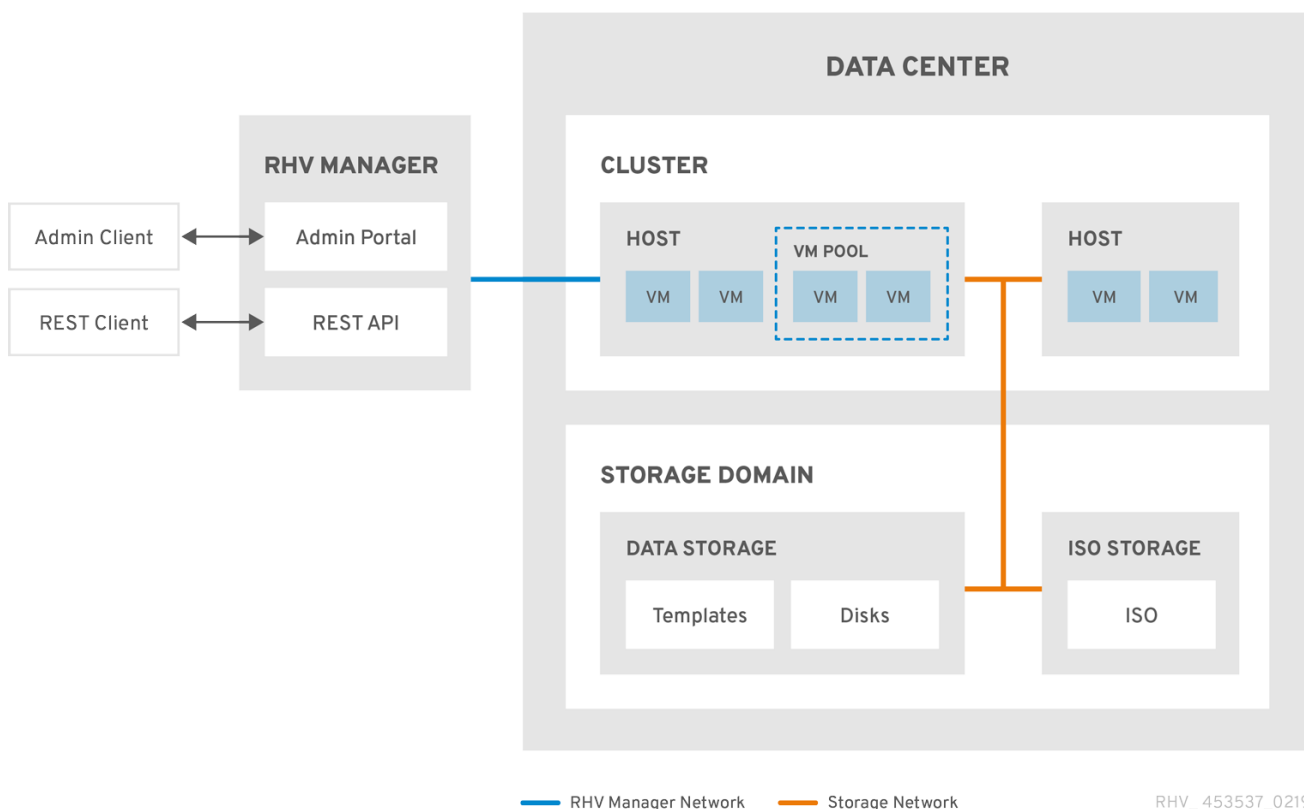
### 7.1. 데이터 센터 소개

데이터센터는 특정 환경에서 사용되는 리소스 집합을 정의하는 논리 엔티티입니다. 데이터 센터에는 클러스터 및 호스트의 형태로 논리 리소스, 즉 논리적 네트워크 및 물리적 **NIC** 형식의 네트워크 리소스, 스토리지 도메인 형태로 스토리지 리소스로 구성됩니다.

데이터 센터에는 여러 호스트를 포함할 수 있는 여러 클러스터가 포함될 수 있습니다. 이 클러스터에는 여러 스토리지 도메인이 연결될 수 있으며, 각 호스트에서 여러 가상 시스템을 지원할 수 있습니다. **Red Hat Virtualization** 환경은 여러 데이터 센터를 포함할 수 있습니다. 데이터 센터 인프라를 사용하면 이러한 센터를 별도로 유지할 수 있습니다.

모든 데이터센터는 단일 관리 포털에서 관리됩니다.

그림 7.1. 데이터 센터



**Red Hat Virtualization**은 설치 중에 기본 데이터 센터를 생성합니다. 기본 데이터 센터를 구성하거나 적절하게 이름이 지정된 데이터 센터를 새로 설정할 수 있습니다.

### 7.2. 스토리지 풀 관리자

스토리지 풀 관리자(**SPM**)는 데이터 센터의 스토리지 도메인을 관리할 수 있도록 데이터 센터의 호스트 중 하나에 지정된 역할입니다. **Red Hat Virtualization Manager**는 데이터 센터의 모든 호스트에서 실행할 수 있습니다. **Red Hat Virtualization Manager**는 호스트 중 하나에 역할을 부여합니다. **ECDHE**는 호스트가 표준 작업에서 제외되지 않습니다. **gradle**으로 실행되는 호스트는 가상 리소스를 계속 호스팅할 수 있습니다.

**FlexVolume** 엔티티는 스토리지 도메인의 메타데이터를 조정하여 스토리지에 대한 액세스를 제어합니다. 여기에는 가상 디스크(이미지), 스냅샷 및 템플릿 생성, 삭제, 조작, 스파스 블록 장치(**SAN**에서)에 대한 스토리지 할당이 포함됩니다. 이는 한 번에 하나의 호스트만 데이터 센터에서 메타데이터 무결성을 보장하기 위해 한 번에 하나의 호스트일 수 있습니다.

**Red Hat Virtualization Manager**를 통해 항상 사용할 수 있는지 확인합니다. **Manager**가 **storage**에 액세스하는 데 문제가 발생하면 **Manager** 역할을 다른 호스트로 이동합니다. **BOOM**이 시작되면 해당 역할이 유일한 호스트임을 보장하므로 스토리지 중심 리스를 취득하게 됩니다. 이 프로세스에는 시간이 다소 걸릴 수 있습니다.

### 7.3. ECDHE 우선 순위

**DestinationRule** 역할은 호스트의 사용 가능한 일부 리소스를 사용합니다. 호스트의 **priority** 설정으로 인해 **host**가 할당될 가능성을 변경합니다. 우선 순위가 높은 호스트에는 우선 순위가 낮은 호스트보다 우선 순위가 높은 호스트에 **seccomp** 역할이 할당됩니다. 우선 순위가 낮은 호스트의 중요한 가상 머신은 호스트 리소스에 대해 **ECDHE** 작업을 수행할 필요가 없습니다.

**Edit Host** (호스트 편집) 창에서 호스트 탭에서 호스트의 **priority**를 변경할 수 있습니다.

## 7.4. 데이터 센터 작업

### 7.4.1. 새 데이터 센터 생성

이 절차에서는 가상화 환경에 데이터 센터를 생성합니다. 데이터 센터에는 작동할 클러스터, 호스트 및 스토리지 도메인이 필요합니다.




## 참고

호환성 버전이 설정되면 나중에 이 버전을 줄일 수 없으며 버전 회귀는 허용되지 않습니다.

데이터 센터의 **MAC** 풀 범위를 지정하는 옵션이 비활성화되어 클러스터 수준에서 수행됩니다.

## 새 데이터 센터 생성

1. **Compute** → **Data Centers** (데이터 센터)를 클릭합니다.
2. 새로 생성을 클릭합니다.
3. 데이터 센터의 이름 및 설명을 입력합니다.
4. 드롭다운 메뉴에서 데이터 센터의 스토리지 유형, 호환성 버전, 할당량 모드를 선택합니다.
5. **OK** 를 클릭하여 데이터 센터를 생성하고 **Data Center - Guide Me** 창을 엽니다.
6. **Guide Me** 창에는 데이터 센터에 대해 구성해야 하는 엔티티가 나열됩니다. **Configure Later** (나중에 구성) 버튼을 클릭하여 이러한 엔티티 또는 **postpone** 구성을 구성합니다. 데이터 센터를 선택하고 추가 작업 (  )을 클릭한 다음 **Guide Me** 를 클릭하여 설정을 다시 시작할 수 있습니다.

새 데이터 센터는 클러스터, 호스트 및 스토리지 도메인이 구성될 때까지 초기화되지 않은 상태로 유지됩니다. **Guide Me** 를 사용하여 이러한 엔티티를 구성합니다.

7.4.2. 새 데이터 센터 및 편집 데이터 센터 **Windows**의 설정에 대한 설명

아래 표에서는 **New Data Center** (새 데이터 센터) 및 **Edit Data Center** (데이터 센터 편집) 창에 표시되는 데이터 센터 설정을 설명합니다. 유효하지 않은 항목은 **OK** 를 클릭하면 변경 사항이 허용되지 않습니다. 또한 필드 프롬프트는 예상되는 값 또는 값 범위를 나타냅니다.

표 7.1. 데이터 센터 속성

필드	설명/Action
이름	데이터 센터의 이름입니다. 이 텍스트 필드에는 40자 제한이 있으며 대문자 및 소문자, 숫자, 하이픈 및 밑줄이 조합된 고유한 이름이어야 합니다.
설명	데이터 센터에 대한 설명입니다. 이 필드는 권장되지만 필수는 아닙니다.
스토리지 유형	공유 또는 로컬 스토리지 유형을 선택합니다.  다양한 유형의 스토리지 도메인(iSCSI, NFS, FC, POSIX, Gluster)을 동일한 데이터 센터에 추가할 수 있습니다. 그러나 로컬 및 공유 도메인은 혼합할 수 없습니다.  데이터 센터를 초기화한 후 스토리지 유형을 변경할 수 있습니다. 7.4.6절. "데이터 센터 스토리지 유형 변경"을 참조하십시오.
호환성 버전	Red Hat Virtualization의 버전입니다.  Red Hat Virtualization Manager를 업그레이드한 후에도 호스트, 클러스터 및 데이터 센터가 이전 버전에 있을 수 있습니다. 데이터 센터의 호환성 수준을 업그레이드하기 전에 모든 호스트, 클러스터를 업그레이드했는지 확인하십시오.
할당량 모드	할당량은 Red Hat Virtualization에서 제공하는 리소스 제한 틀입니다. 다음 중 하나를 선택합니다. <ul style="list-style-type: none"> <li>● <b>Disabled:</b> 할당량을 구현하지 않으려면 선택</li> <li>● <b>audit:</b> 할당량 설정을 편집하려는 경우 선택합니다.</li> <li>● <b>Enforced:</b> 할당량 구현을 선택합니다.</li> </ul>
주석	선택적으로 데이터 센터에 대한 일반 텍스트 주석을 추가합니다.


### 7.4.3. 데이터 센터 재 시작: recovery process

이 복구 절차에서는 데이터 센터의 마스터 데이터 도메인을 새 마스터 데이터 도메인으로 교체합니다. 데이터가 손상된 경우 마스터 데이터 도메인을 다시 초기화해야 합니다. 데이터 센터를 다시 초기화하면 클러스터, 호스트 및 비**Problematic** 스토리지 도메인을 포함하여 데이터 센터와 관련된 기타 모든 리소스를 복원할 수 있습니다.



백업 또는 내보낸 가상 머신 또는 템플릿을 새 마스터 데이터 도메인으로 가져올 수 있습니다.

데이터 센터 다시 시작

1. **Compute** → **Data Centers (데이터 센터)**를 클릭하고 데이터 센터를 선택합니다.
2. 데이터 센터에 연결된 모든 스토리지 도메인이 유지 관리 모드에 있는지 확인합니다.
3. **More Actions (  )**를 클릭한 다음 **Re-ECDHE Data Center** 를 클릭합니다.
4. **Data Center Re-ECDHE** 창에 는 사용 가능한 모든(유지 관리 모드에서) 스토리지 도메인이 나열됩니다. 데이터 센터에 추가할 스토리지 도메인의 라디오 버튼을 클릭합니다.
5. 승인 작업 확인란을 선택합니다.
6. **OK**를 클릭합니다.

스토리지 도메인은 마스터 데이터 도메인으로 데이터 센터에 연결되고 활성화됩니다. 이제 백업 또는 내보낸 가상 머신 또는 템플릿을 새 마스터 데이터 도메인으로 가져올 수 있습니다.

#### 7.4.4. 데이터 센터 제거

데이터 센터를 제거하려면 활성 호스트가 필요합니다. 데이터 센터를 제거해도 관련 리소스는 제거되지 않습니다.

데이터 센터 제거

1. 데이터 센터에 연결된 스토리지 도메인이 유지 관리 모드에 있는지 확인합니다.

2. **Compute** → **Data Centers** (데이터 센터)를 클릭하고 제거할 데이터 센터를 선택합니다.
3. 제거를 클릭합니다.
4. **OK**를 클릭합니다.


#### 7.4.5. 데이터 센터 제거

연결된 스토리지 도메인이 손상되거나 호스트가 무응답 이 되는 경우 데이터 센터가 무응답이 됩니다. 데이터 센터를 제거할 수 없습니다.

강제 제거 에는 활성 호스트가 필요하지 않습니다. 연결된 스토리지 도메인도 영구적으로 제거합니다.

데이터 센터를 강제로 제거하기 전에 손상된 스토리지 도메인을 제거해야 할 수 있습니다.

데이터 센터 제거

1. **Compute** → **Data Centers** (데이터 센터)를 클릭하고 제거할 데이터 센터를 선택합니다.
2. 더 많은 작업 (  )을 클릭한 다음 강제 제거를 클릭합니다.
3. 승인 작업 확인란을 선택합니다.
4. **OK**를 클릭합니다.

데이터 센터 및 연결된 스토리지 도메인은 **Red Hat Virtualization** 환경에서 영구적으로 제거됩니다.

#### 7.4.6. 데이터 센터 스토리지 유형 변경

데이터 센터의 스토리지 유형을 초기화한 후 변경할 수 있습니다. 이는 가상 머신 또는 템플릿을 이동하는 데 사용되는 데이터 도메인에 유용합니다.

#### 제한 사항

- **Local**과 공유 - 로컬 데이터 센터에서 지원하지 않기 때문에 둘 이상의 호스트와 두 개 이상의 클러스터가 없는 데이터 센터의 경우입니다.
- 로컬에서 공유 - 로컬 스토리지 도메인이 없는 데이터 센터의 경우입니다.

#### 데이터 센터 스토리지 유형 변경

1. **Compute** → **Data Centers** (데이터 센터)를 클릭하고 변경할 데이터 센터를 선택합니다.
2. 편집 을 클릭합니다.
3. 스토리지 유형을 원하는 값으로 변경합니다.
4. **OK**를 클릭합니다.

#### 7.4.7. 데이터 센터 호환성 버전 변경

**Red Hat Virtualization** 데이터 센터에는 호환성 버전이 있습니다. 호환성 버전은 데이터 센터를 호환하려는 **Red Hat Virtualization** 버전을 나타냅니다. 데이터 센터의 모든 클러스터는 원하는 호환성 수준을 지원해야 합니다.



#### 중요

데이터 센터 호환성 버전을 변경하려면 먼저 데이터 센터의 모든 클러스터 및 가상 머신의 호환성 버전을 업데이트해야 합니다.

#### 절차

1. 관리 포털에서 **Compute** → **Data Centers** 를 클릭합니다.
2. 변경할 데이터 센터를 선택하고 편집 을 클릭합니다.
3. 호환성 버전을 원하는 값으로 변경합니다.
4. **OK**를 클릭합니다. 데이터 센터 호환성 버전 확인 대화 상자가 열립니다.
5. 확인을 클릭하여 확인합니다.

## 7.5. 데이터 센터 및 스토리지 도메인

### 7.5.1. 데이터 센터에 기존 데이터 도메인 연결

연결 해제된 데이터 도메인을 데이터 센터에 연결할 수 있습니다. 여러 유형의 공유 스토리지 도메인 (**iSCSI, NFS, FC, POSIX, Gluster**)을 동일한 데이터 센터에 추가할 수 있습니다.

데이터 센터에 기존 데이터 도메인 연결

1. **Compute** → **Data Centers** (데이터 센터)를 클릭합니다.
2. 데이터 센터의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 스토리지 탭을 클릭하여 데이터 센터에 이미 연결된 스토리지 도메인을 나열합니다.
4. 데이터 연결을 클릭합니다.
5. 데이터 센터에 연결할 데이터 도메인의 확인란을 선택합니다. 여러 데이터 도메인을 연결하려면 여러 확인란을 선택할 수 있습니다.

6. **OK**를 클릭합니다.

데이터 도메인이 데이터 센터에 연결되고 자동으로 활성화됩니다.

### 7.5.2. 데이터 센터에 기존 ISO 도메인 연결

연결 해제인 **ISO** 도메인은 데이터 센터에 연결할 수 있습니다. **ISO** 도메인은 데이터 센터와 동일한 스토리지 유형 이어야 합니다.

하나의 **ISO** 도메인만 데이터 센터에 연결할 수 있습니다.

데이터 센터에 기존 **ISO** 도메인 연결

1. **Compute** → **Data Centers** (데이터 센터)를 클릭합니다.
2. 데이터 센터의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 스토리지 탭을 클릭하여 데이터 센터에 이미 연결된 스토리지 도메인을 나열합니다.
4. **ISO** 첨부를 클릭합니다.
5. 적절한 **ISO** 도메인의 라디오 버튼을 클릭합니다.
6. **OK**를 클릭합니다.

**ISO** 도메인은 데이터 센터에 연결되며 자동으로 활성화됩니다.

### 7.5.3. 데이터 센터에 기존 내보내기 도메인 연결



## 참고

내보내기 스토리지 도메인은 더 이상 사용되지 않습니다. 스토리지 데이터 도메인은 데이터 센터에서 연결 해제되어 동일한 환경 또는 다른 환경의 다른 데이터 센터로 가져올 수 있습니다. 그런 다음 가져온 스토리지 도메인에서 연결된 데이터 센터로 가상 머신, 유동 가상 디스크 및 템플릿을 업로드할 수 있습니다. 스토리지 도메인 가져오기에 대한 자세한 내용은 [11.7절. “기존 스토리지 도메인 가져오기”](#)를 참조하십시오.

**Unattached** (연결되지 않음)인 내보내기 도메인을 데이터 센터에 연결할 수 있습니다. 하나의 내보내기 도메인만 데이터 센터에 연결할 수 있습니다.

데이터 센터에 기존 내보내기 도메인 연결

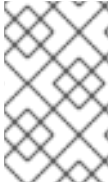
1. **Compute** → **Data Centers** (데이터 센터)를 클릭합니다.
2. 데이터 센터의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 스토리지 탭을 클릭하여 데이터 센터에 이미 연결된 스토리지 도메인을 나열합니다.
4. **Attach Export** 를 클릭합니다.
5. 적절한 내보내기 도메인에 대한 라디오 버튼을 클릭합니다.
6. **OK**를 클릭합니다.

내보내기 도메인은 데이터 센터에 연결되고 자동으로 활성화됩니다.

#### 7.5.4. 데이터 센터에서 스토리지 도메인 분리

데이터 센터에서 스토리지 도메인을 분리하면 데이터 센터가 해당 스토리지 도메인과 연결할 수 없습니다. 스토리지 도메인은 **Red Hat Virtualization** 환경에서 제거되지 않으며 다른 데이터 센터에 연결할 수 있습니다.

가상 머신 및 템플릿과 같은 데이터는 스토리지 도메인에 연결된 상태로 유지됩니다.



#### 참고

마스터 스토리지는 사용 가능한 마지막 스토리지 도메인인 경우 제거할 수 없습니다.

데이터 센터에서 스토리지 도메인 분리

1. **Compute** → **Data Centers** (데이터 센터)를 클릭합니다.
2. 데이터 센터의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 스토리지 탭을 클릭하여 데이터 센터에 연결된 스토리지 도메인을 나열합니다.
4. 분리할 스토리지 도메인을 선택합니다. 스토리지 도메인이 활성 상태이면 유지 관리를 클릭합니다.
5. 확인 을 클릭하여 유지보수 모드를 시작합니다.
6. 분리 를 클릭합니다.
7. **OK**를 클릭합니다.

세부 정보 보기에서 스토리지 도메인이 사라질 때까지 최대 몇 분이 걸릴 수 있습니다.

## 8장. 클러스터

### 8.1. 클러스터 소개

클러스터는 동일한 스토리지 도메인을 공유하고 동일한 유형의 **CPU(Intel 또는 AMD)**를 갖는 호스트의 논리적 그룹입니다. 호스트에 다른 세대의 **CPU** 모델이 있는 경우 모든 모델에 있는 기능만 사용합니다.

시스템의 각 클러스터는 데이터 센터에 속해야 하며 시스템의 각 호스트는 클러스터에 속해야 합니다. 가상 머신은 클러스터의 모든 호스트에 동적으로 할당되며, 가상 머신의 클러스터 및 설정에 따라 해당 호스트 간에 마이그레이션할 수 있습니다. 클러스터는 전원 및 로드 밸런싱 정책을 정의할 수 있는 최고 수준입니다.

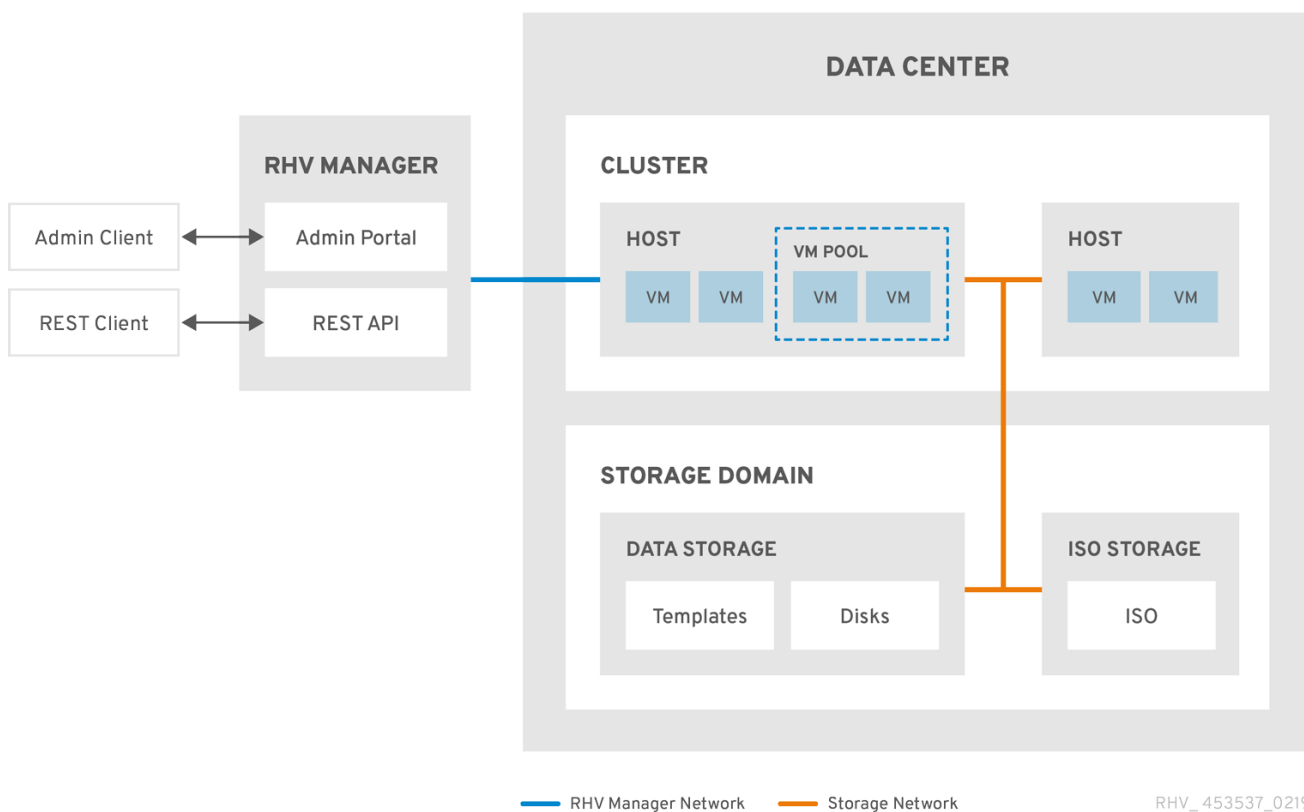
클러스터에 속한 가상 시스템의 호스트 수와 수는 각각 호스트 수 및 **VM** 수 아래의 결과 목록에 표시됩니다.

클러스터는 가상 머신 또는 **Red Hat Gluster Storage Server**를 실행합니다. 이러한 두 가지 용도는 상호 배타적입니다. 단일 클러스터는 가상화 및 스토리지 호스트를 함께 지원할 수 없습니다.

**Red Hat Virtualization**은 설치 중에 기본 데이터 센터에 기본 클러스터를 생성합니다.



그림 8.1. Cluster



## 8.2. 클러스터 작업



참고

일부 클러스터 옵션은 **Gluster** 클러스터에 적용되지 않습니다. **Red Hat Virtualization** 에서 **Red Hat Gluster Storage**를 사용하는 방법에 대한 자세한 내용은 **Red Hat *Gluster Storage*를 사용하여 Red Hat Virtualization 구성** 을 참조하십시오.

### 8.2.1. 새 클러스터 생성

데이터 센터에는 여러 클러스터를 포함할 수 있으며 클러스터는 여러 호스트를 포함할 수 있습니다. 클러스터의 모든 호스트는 동일한 **CPU** 유형(**Intel** 또는 **AMD**)이어야 합니다. **CPU** 유형 최적화를 위해 클러스터를 생성하기 전에 호스트를 생성하는 것이 좋습니다. 그러나 나중에 **Guide Me** 버튼을 사용하여 호스트를 구성할 수 있습니다.

새 클러스터 생성

1.

**Compute** → **Clusters** 를 클릭합니다.

2. 새로 생성을 클릭합니다.
3. 드롭다운 목록에서 클러스터가 속할 데이터 센터를 선택합니다.
4. 클러스터의 이름 및 설명을 입력합니다.
5. **Management Network** 드롭다운 목록에서 네트워크를 선택하여 관리 네트워크 역할을 할당합니다.
6. 드롭다운 목록에서 **CPU** 아키텍처 및 **CPU** 유형을 선택합니다. **CPU** 프로세서 제품군을 클러스터에 연결하려는 호스트의 최소 **CPU** 프로세서 유형과 일치해야 합니다. 그렇지 않으면 호스트가 작동하지 않습니다.



참고

**Intel** 및 **AMD CPU** 유형 모두에서 **CPU** 모델은 가장 오래된 것부터 최신 버전까지 논리 순서로 지정됩니다. 클러스터에 다른 **CPU** 모델이 있는 호스트가 포함된 경우 가장 오래된 **CPU** 모델을 선택합니다. 각 **CPU** 모델에 대한 자세한 내용은 <https://access.redhat.com/solutions/634853> 를 참조하십시오.

7. 드롭다운 목록에서 클러스터의 호환성 버전을 선택합니다.
8. 드롭다운 목록에서 스위치 유형을 선택합니다.
9. 클러스터에서 호스트의 **Firewall Type ( iptables 또는 firewalld )**을 선택합니다.



참고

**iptables** 는 더 이상 사용되지 않습니다.

10. **Enable Virt Service or Enable Gluster Service** 확인란을 선택하여 클러스터가 가상 시스템 호스트 또는 **Gluster** 지원 노드로 채워질지 여부를 정의합니다.

11. 필요한 경우 **Enable to set VM maintenance reason** 확인란을 선택하여 **Manager**에서 가상 머신을 종료할 때 선택적 이유 필드를 활성화하여 관리자가 유지 관리에 대한 설명을 제공할 수 있습니다.
12. 선택적으로 **Manager**에서 호스트가 유지보수 모드에 배치될 때 선택적 이유 필드를 활성화하도록 **Enable to set Host maintenance reason** 확인란을 선택하여 관리자가 유지 관리에 대한 설명을 제공할 수 있습니다.
13. 선택적으로 **/dev/hwrng** 소스 (외부 하드웨어 장치) 확인란을 선택하여 클러스터의 모든 호스트가 사용할 난수 생성기를 지정합니다. **/dev/urandom** 소스 (**Linux** 제공 장치)는 기본적으로 활성화되어 있습니다.
14. 최적화 탭을 클릭하여 클러스터의 임계값을 공유하는 메모리 페이지를 선택하고, 선택적으로 클러스터의 호스트에서 **CPU** 스레드 처리 및 메모리 불루닝을 활성화합니다.
15. **Migration Policy** 탭을 클릭하여 클러스터에 대한 가상 머신 마이그레이션 정책을 정의합니다.
16. 스케줄링 정책 탭을 클릭하여 선택 옵션으로 스케줄링 정책을 구성하고, 스케줄러 최적화 설정을 구성하고, 클러스터의 호스트에 대해 신뢰할 수 있는 서비스를 활성화하며, **HAECDHE**을 활성화한 다음 사용자 정의 일련 번호 정책을 추가합니다.
17. 콘솔 탭을 클릭하여 필요한 경우 글로벌 **SPICE** 프록시를 재정의하고 클러스터의 호스트에 대한 **SPICE** 프록시 주소를 지정합니다.
18. 클러스터에서 펜싱을 활성화하거나 비활성화하려면 **Fencing policy** 탭을 클릭하고 펜싱 옵션을 선택합니다.
19. **MAC** 주소 풀 탭을 클릭하여 클러스터의 기본 풀 이외의 **MAC** 주소 풀을 지정합니다. **MAC** 주소 풀 생성, 편집 또는 제거에 대한 자세한 내용은 **1.5절. "MAC 주소 풀"**의 내용을 참조하십시오.
20. 확인을 클릭하여 클러스터를 생성하고 **Cluster - Guide Me** 창을 엽니다.
21. **Guide Me** 창에는 클러스터에 대해 구성해야 하는 엔티티가 나열됩니다. **Configure Later** (나중에 구성) 버튼을 클릭하여 이러한 엔티티 또는 **postpone** 구성을 구성합니다. 클러스터를 선

택하고 추가 작업 (



)을 클릭한 다음 **Guide Me** 를 클릭하여 설정을 다시 시작할 수 있습니다.

### 8.2.2. 일반 클러스터 설정 설명

아래 표에서는 새 클러스터 및 클러스터 편집 창의 일반 탭에 대한 설정을 설명합니다. 유효하지 않은 항목은 **OK** 를 클릭하면 변경 사항이 허용되지 않습니다. 또한 필드 프롬프트는 예상되는 값 또는 값 범위를 나타냅니다.

표 8.1. 일반 클러스터 설정

필드	설명/Action
데이터 센터	클러스터를 포함할 데이터 센터입니다. 클러스터를 추가하기 전에 데이터 센터를 생성해야 합니다.
이름	클러스터의 이름입니다. 이 텍스트 필드에는 40자 제한이 있으며 대문자 및 소문자, 숫자, 하이픈 및 밑줄이 조합된 고유한 이름이어야 합니다.
설명 / 댓글	클러스터 또는 추가 노트에 대한 설명입니다. 이러한 필드는 권장되지만 필수는 아닙니다.
관리 네트워크	관리 네트워크 역할이 할당될 논리적 네트워크입니다. 기본값은 <b>ovirtmgmt</b> 입니다. 마이그레이션 네트워크가 소스 또는 대상 호스트에 제대로 연결되지 않은 경우 이 네트워크는 가상 머신을 마이그레이션하는 데도 사용됩니다.  기존 클러스터에서는 세부 정보 보기의 <b>Logical Networks</b> 탭에서 <b>Manage Networks</b> (네트워크 관리) 버튼을 통해서만 관리 네트워크를 변경할 수 있습니다.
CPU 아키텍처	클러스터의 CPU 아키텍처입니다. 선택한 CPU 아키텍처에 따라 다른 CPU 유형을 사용할 수 있습니다. <ul style="list-style-type: none"> <li>● <b>undefined</b>: 모든 CPU 유형을 사용할 수 있습니다.</li> <li>● <b>x86_64</b>: 모든 Intel 및 AMD CPU 유형을 사용할 수 있습니다.</li> <li>● <b>ppc64</b>: IBM POWER 8만 사용할 수 있습니다.</li> </ul>

필드	설명/Action
CPU 유형	클러스터의 CPU 유형입니다. 지원되는 CPU 유형 목록은 <a href="#">계획 및 사전 요구 사항 가이드</a> 의 CPU 요구 사항을 참조하십시오. 클러스터의 모든 호스트는 Intel, AMD 또는 IBM POWER 8 CPU 유형을 실행해야 합니다. 생성 후에는 심각한 중단없이 변경할 수 없습니다. CPU 유형을 클러스터에서 가장 오래된 CPU 모델로 설정해야 합니다. 모든 모델에 존재하는 기능만 사용할 수 있습니다. Intel 및 AMD CPU 유형 모두에서 CPU 모델은 가장 오래된 것부터 최신 버전까지 논리 순서로 지정됩니다.
호환성 버전	Red Hat Virtualization의 버전입니다. 데이터 센터에 지정된 버전보다 이전 버전을 선택할 수 없습니다.
스위치 유형	클러스터에서 사용하는 스위치 유형입니다. <b>Linux Bridge</b> 는 표준 Red Hat Virtualization 스위치입니다. <b>OVS</b> 는 Open vSwitch 네트워킹 기능을 지원합니다.
방화벽 유형	클러스터의 호스트의 방화벽 유형을 <b>iptables</b> 또는 <b>firewalld</b> 로 지정합니다.  <b>참고 iptables</b> 는 더 이상 사용되지 않습니다.  기존 클러스터의 방화벽 유형을 변경하는 경우 클러스터의 <b>모든 호스트를 다시 설치</b> 하여 변경 사항을 적용해야 합니다.
기본 네트워크 공급자	클러스터가 사용할 기본 외부 네트워크 공급자를 지정합니다. OVN(Open Virtual Network)을 선택하면 클러스터에 추가된 호스트가 OVN 공급자와 통신하도록 자동으로 구성됩니다.  기본 네트워크 공급자를 변경하는 경우 변경 사항을 적용하려면 클러스터의 <b>모든 호스트를 다시 설치</b> 해야 합니다.
최대 로그 메모리 임계값	최대 메모리 사용량에 대한 로깅 임계값을 백분율 또는 절대 값(MB)으로 지정합니다. 호스트의 메모리 사용량이 백분율 값을 초과하거나 호스트의 사용 가능한 메모리가 절대 값(MB) 미만인 경우 메시지가 기록됩니다. 기본값은 <b>95%</b> 입니다.
Enable Virt Service	이 라디오 버튼을 선택하면 이 클러스터의 호스트가 가상 머신을 실행하는 데 사용됩니다.
Gluster 서비스 활성화	이 라디오 버튼을 선택하면 이 클러스터의 호스트가 Red Hat Gluster Storage Server 노드로 사용되며 가상 머신 실행에는 사용할 수 없습니다.

필드	설명/Action
<p>기존 triggeruster 구성 가져오기</p>	<p>이 확인란은 <b>Gluster Service 활성화</b> 버튼을 선택한 경우에만 사용할 수 있습니다. 이 옵션을 사용하면 기존 Gluster 지원 클러스터와 모든 연결된 호스트를 Red Hat Virtualization Manager에 가져올 수 있습니다.</p> <p>클러스터의 각 호스트에 다음 옵션을 가져와야 합니다.</p> <ul style="list-style-type: none"> <li>● <b>address:</b> Gluster 호스트 서버의 IP 또는 정규화된 도메인 이름을 입력합니다.</li> <li>● <b>지문:</b> Red Hat Virtualization Manager는 호스트의 지문을 가져와서 올바른 호스트에 연결되어 있는지 확인합니다.</li> <li>● <b>Root password:</b> 호스트와 통신하는 데 필요한 root 암호를 입력합니다.</li> </ul>
<p>VM 유지 관리 이유 설정 가능</p>	<p>이 확인란을 선택하면 Manager에서 클러스터의 가상 머신이 종료될 때 선택적 이유 필드가 표시됩니다. 이를 통해 로그에 표시되고 가상 머신의 전원이 다시 켜지면 유지 관리에 대한 설명을 제공할 수 있습니다.</p>
<p>호스트 유지 관리 이유 설정 가능</p>	<p>이 확인란을 선택하면 클러스터의 호스트가 Manager에서 유지보수 모드로 전환될 때 선택적 이유 필드가 나타납니다. 이를 통해 로그에 표시되고 호스트가 다시 활성화될 때 유지 관리에 대한 설명을 제공할 수 있습니다.</p>
<p>추가 Random Number Generator 소스</p>	<p>확인란을 선택하면 클러스터의 모든 호스트에 사용할 수 있는 추가 난수 생성기 장치가 있습니다. 이를 통해 임의의 수 생성기 장치에서 가상 머신으로 엔트로피를 통과할 수 있습니다.</p>

### 8.2.3. 최적화 설정 설명

#### 메모리 고려 사항

메모리 페이지 공유를 사용하면 가상 머신에서 다른 가상 머신에서 사용되지 않는 메모리를 활용하여 할당된 메모리의 최대 **200%**를 사용할 수 있습니다. 이 프로세스는 **Red Hat Virtualization** 환경의 가상 머신이 동시에 모든 가상 머신을 동시에 실행하는 것은 아니므로 사용하지 않는 메모리를 특정 가상 머신에 일시적으로 할당할 수 있습니다.

#### CPU 고려 사항

- CPU 사용량이 많은 워크로드의 경우 호스트의 코어 수보다 총 프로세서 코어 수를 사용하여 가상 머신을 실행할 수 있습니다. 이렇게 하면 다음이 활성화됩니다.
  - 더 많은 수의 가상 머신을 실행하여 하드웨어 요구 사항을 줄일 수 있습니다.
  - 가상 코어 수가 호스트 코어 수와 호스트 스레드 수 사이에 있는 경우와 같이 CPU 토폴로지를 사용하여 가상 머신을 구성할 수 있습니다.
- 최상의 성능과 CPU 집약적 워크로드의 경우 특히 호스트에서와 동일한 토폴로지를 사용해야 하므로 호스트와 가상 머신의 캐시 사용량도 동일합니다. 호스트에 하이퍼 스레딩이 활성화된 경우 QEMU는 호스트의 하이퍼 스레딩을 코어로 처리하므로 가상 머신은 여러 스레드가 있는 단일 코어에서 실행되고 있음을 인식하지 못합니다. 이 동작은 호스트 코어의 하이퍼 스레딩에 실제로 해당하는 가상 코어가 동일한 호스트 코어의 다른 하이퍼스레드와 단일 캐시를 공유하는 반면 가상 머신은 가상 머신의 성능에 영향을 미칠 수 있습니다.

아래 표에서는 새 클러스터 및 클러스터 편집 창의 최적화 탭에 대한 설정을 설명합니다.

표 8.2. 최적화 설정

필드	설명/Action
메모리 최적화	<ul style="list-style-type: none"> <li>● none - 메모리 과다 할당 비활성화: 메모리 페이지 공유를 비활성화합니다.</li> <li>● 서버 로드 - 실제 메모리의 150%의 스케줄링 허용: 각 호스트에서 시스템 메모리의 150%로 메모리 페이지 공유 임계값을 설정합니다.</li> <li>● 데스크탑 로드 - 물리적 메모리의 200%의 스케줄링 허용: 각 호스트에서 시스템 메모리의 200%로 메모리 페이지 공유 임계값을 설정합니다.</li> </ul>
CPU 스레드	<p><b>Count Threads As Cores</b> 확인란을 선택하면 호스트에서 코어 수보다 총 프로세서 코어 수가 있는 가상 머신을 실행할 수 있습니다.</p> <p>이 확인란을 선택하면 노출된 호스트 스레드가 가상 시스템이 사용할 수 있는 코어로 취급됩니다. 예를 들어, 코어당 2개 스레드(48 스레드 합계)가 있는 24코어 시스템은 각각 최대 48개의 코어가 있는 가상 머신을 실행할 수 있으며, 호스트 CPU 로드를 계산하는 알고리즘은 잠재적인 사용 가능한 코어 수에 비해 부하 두 배를 비교합니다.</p>

필드	설명/Action
<p>메모리란?</p>	<p><b>Enable MemoryECDHEoon Optimization</b> 확인란을 선택하면 이 클러스터의 호스트에서 실행되는 가상 머신에서 메모리 과다 할당이 가능합니다. 이 확인란을 선택하면 MoM(Memory Overcommit Manager)이 모든 가상 머신의 보장된 메모리 크기 제한으로 가능한 경우 불링을 시작합니다.</p> <p>balloon을 실행하려면 가상 머신에 관련 드라이버가 있는 balloon 장치가 있어야 합니다. 각 가상 머신에는 구체적으로 제거되지 않는 한 balloon 장치가 포함되어 있습니다. 이 클러스터의 각 호스트는 상태가 <b>Up</b> 으로 변경될 때 balloon 정책 업데이트를 받습니다. 필요한 경우 상태를 변경하지 않고도 호스트에서 balloon 정책을 수동으로 업데이트할 수 있습니다. <a href="#">8.2.9절. "클러스터의 호스트에서 MoM 정책 업데이트"</a> 을 참조하십시오.</p> <p>일부 시나리오에서 불링이 KSM과 충돌할 수 있다는 것을 이해하는 것이 중요합니다. 이러한 경우 MoM은 충돌을 최소화하기 위해 balloon 크기를 조정하려고 합니다. 또한 일부 시나리오에서는 가상 머신에 대한 하위 최적화 성능이 발생할 수 있습니다. 관리자는 볼루닝 최적화를 신중하게 사용하는 것이 좋습니다.</p>
<p>KSM 컨트롤</p>	<p><b>KSM 사용</b> 확인란을 선택하면 MoM이 필요한 경우 KSM (Kernel Same-page Merging)을 실행할 수 있으며 CPU 비용보다 큰 메모리 절약 이점을 얻을 수 있습니다.</p>

### 8.2.4. 설명된 마이그레이션 정책 설정

마이그레이션 정책은 호스트 장애 발생 시 가상 머신을 실시간 마이그레이션하는 조건을 정의합니다. 이러한 조건에는 마이그레이션 중에 가상 시스템의 다운타임, 네트워크 대역폭 및 가상 머신의 우선 순위가 지정되는 방법이 포함됩니다.

표 8.3. 설명된 마이그레이션 정책

정책	설명
<p>legacy</p>	<p>3.6 버전의 기존 동작. <b>vdsm.conf</b> 의 덮어쓰기는 계속 적용됩니다. 게스트 에이전트 후크 메커니즘이 비활성화되어 있습니다.</p>



정책	설명
다운 타임 최소화	<p>일반적인 상황에서 가상 머신을 마이그레이션할 수 있는 정책입니다. 가상 머신에 심각한 다운타임이 발생하지 않아야 합니다. 가상 머신 마이그레이션이 긴 시간(QEMU 반복에 따라 최대 500밀리초) 후에 통합되지 않으면 마이그레이션이 중단됩니다. 게스트 에이전트 후크 메커니즘이 활성화되어 있습니다.</p>
복사 후 마이그레이션	<p>가상 머신은 최소 가동 중지 시간 정책과 유사하게 상당한 다운타임이 발생하지 않아야 합니다. Post-copy 정책은 먼저 통합이 발생할 수 있는지 확인하기 위해 사전 복사를 시도합니다. 가상 머신 마이그레이션이 오랜 시간 경과 후 통합되지 않는 경우 마이그레이션이 post-copy로 전환됩니다. 이 정책의 단점은 복사 후 단계에서 메모리의 누락된 부분이 호스트 간에 전송되므로 가상 머신의 속도가 크게 느려질 수 있다는 것입니다.</p> <p>호스트 간에 마이그레이션 네트워크의 네트워크 오류와 같이 복사 후 단계에서 문제가 발생하면 마이그레이션 프로세스가 일관되지 않고 일시 정지된 VM으로 이어지며 이로 인해 VM이 손실됩니다. 따라서 복사 후 단계에서 마이그레이션을 중단할 수 없습니다.</p> <div data-bbox="815 1025 1428 1496" style="background-color: #fff9c4; padding: 10px; border: 1px solid #ccc;"> <p style="text-align: center;"> <b>경고</b></p> <p>복사 후 프로세스가 완료되기 전에 네트워크 연결이 끊어지면 Manager가 일시 중지되고 실행 중인 VM을 종료합니다. VM 가용성이 중요하거나 마이그레이션 네트워크가 불안정한 경우 복사 후 마이그레이션을 사용하지 마십시오.</p> </div>
필요한 경우 워크로드 일시 중지	<p>많은 워크로드를 실행하는 가상 머신을 포함하여 대부분의 상황에서 가상 머신을 마이그레이션할 수 있는 정책입니다. 이로 인해 가상 머신의 다운타임이 다른 일부 설정보다 발생할 수 있습니다. 극단적인 워크로드에서 마이그레이션이 중단될 수 있습니다. 게스트 에이전트 후크 메커니즘이 활성화되어 있습니다.</p>

대역폭 설정은 호스트당 발신 및 수신 마이그레이션의 최대 대역폭을 정의합니다.

표 8.4. 설명된 대역폭

정책	설명
auto	대역폭은 데이터 센터 <b>Host Network QoS의 Rate Limit [Mbps]</b> 설정에서 복사됩니다. 속도 제한이 정의되지 않은 경우 네트워크 인터페이스를 전송하고 수신하는 최소 링크 속도로 계산됩니다. 속도 제한이 설정되지 않았으며 링크 속도를 사용할 수 없는 경우 호스트 전송 시 로컬 VDSM 설정에 따라 결정됩니다.
하이퍼바이저 기본값	대역폭은 호스트 전송 시 로컬 VDSM 설정에 의해 제어됩니다.
사용자 지정	<p>사용자(Mbps)에 의해 정의됩니다. 이 값은 동시 마이그레이션 수로 나뉩니다(기본값은 2로, ingoing 및 발신 마이그레이션을 고려합니다). 따라서 사용자 정의 대역폭은 모든 동시 마이그레이션을 수용할 수 있을 만큼 커야 합니다.</p> <p>예를 들어 <b>Custom</b> 대역폭이 600Mbps로 정의되면 가상 머신 마이그레이션의 최대 대역폭은 실제로 300Mbps입니다.</p>

복원력 정책은 마이그레이션에서 가상 머신의 우선 순위를 지정하는 방법을 정의합니다.

표 8.5. 복구 정책 설정

필드	설명/Action
가상 머신 마이그레이션	정의된 우선 순위의 모든 가상 머신을 마이그레이션합니다.
고가용성 가상 머신만 마이그레이션	다른 호스트 과부하를 방지하기 위해 고가용성 가상 머신만 마이그레이션합니다.
가상 머신 마이그레이션하지 않음	가상 머신이 마이그레이션되지 않도록 합니다.

추가 속성은 레거시 마이그레이션 정책에만 적용할 수 있습니다.

표 8.6. 설명되는 추가 속성

속성	설명
----	----

속성	설명
<p><b>자동 컨버지 마이그레이션</b></p>	<p>가상 머신의 실시간 마이그레이션 중에 자동 불일치 사용 여부를 설정할 수 있습니다. 워크로드가 많은 대규모 가상 머신은 실시간 마이그레이션 중에 달성된 전송 속도보다 더 빨리 메모리를 더 빠르게 수행할 수 있으며 마이그레이션이 중단되지 않을 수 있습니다. QEMU의 자동 일관성 기능을 사용하면 가상 머신 마이그레이션을 강제 통합할 수 있습니다. QEMU는 가상 머신의 vCPU가 제한되는 것과 트리거가 통합되지 않음을 자동으로 감지합니다. Auto-convergence는 기본적으로 전역에서 비활성화되어 있습니다.</p> <ul style="list-style-type: none"> <li>● <b>전역 설정에서 Inherit를 선택하여</b> 글로벌 수준에서 설정된 자동 일치 설정을 사용합니다. 이 옵션은 기본적으로 선택됩니다.</li> <li>● <b>Auto Converge</b> 를 선택하여 글로벌 설정을 재정의하고 가상 머신에 대해 자동 일관성을 허용합니다.</li> <li>● <b>don't Auto Converge</b> 를 선택하여 글로벌 설정을 재정의하고 가상 머신의 자동 일관성을 방지합니다.</li> </ul>
<p><b>마이그레이션 압축 사용</b></p>	<p>가상 머신의 실시간 마이그레이션 중에 마이그레이션 압축을 사용할지 여부를 설정할 수 있습니다. 이 기능은 Xor Binary Zero Run-Length-Encoding을 사용하여 메모리 쓰기 집약적인 워크로드 또는 스파스 메모리 업데이트 패턴을 실행하는 가상 머신의 다운타임과 총 실시간 마이그레이션 시간을 줄입니다. 마이그레이션 압축은 기본적으로 전역적으로 비활성화되어 있습니다.</p> <ul style="list-style-type: none"> <li>● <b>전역 수준에서 설정된 압축 설정을 사용하려면 글로벌 설정에서 상속을</b> 선택합니다. 이 옵션은 기본적으로 선택됩니다.</li> <li>● <b>압축</b> 을 선택하여 글로벌 설정을 재정의하고 가상 머신에 대한 압축을 허용합니다.</li> <li>● <b>don't compress</b> 를 선택하여 글로벌 설정을 재정의하고 가상 머신의 압축을 방지합니다.</li> </ul>

### 8.2.5. 설명된 스케줄링 정책 설정

스케줄링 정책을 사용하면 사용 가능한 호스트 간에 가상 머신의 사용량 및 배포를 지정할 수 있습니다. 클러스터의 호스트 간에 자동 로드 밸런싱을 사용하도록 스케줄링 정책을 정의합니다. 스케줄링 정책에 관계없이 과부하된 CPU가 있는 호스트에서 가상 머신이 시작되지 않습니다. 기본적으로 호스트의 CPU는 5분 동안 80%를 초과하는 부하가 있는 경우 과부하로 간주되지만 스케줄링 정책을 사용하여 이러한 값을 변경할 수 있습니다. 스케줄링 정책에 대한 자세한 내용은 [1.3절. “정책 예약”](#) 를 참조하십시오.

표 8.7. 스케줄링 정책 탭 속성

필드	설명/Action
<p>정책 선택</p>	<p>드롭다운 목록에서 정책을 선택합니다.</p> <ul style="list-style-type: none"> <li>● <b>none:</b> 이미 실행 중인 가상 머신에 대해 호스트 간에 로드 밸런싱 또는 전원 조정이 없는 정책 값을 <b>none</b> 으로 설정합니다. 기본 모드입니다. 가상 머신이 시작되면 클러스터의 모든 호스트에 메모리 및 CPU 처리 로드가 균등하게 분배됩니다. 호스트가 정의된 <b>CpuOverCommitDurationMinutes, HighUtilization, MaxFreeMemoryForOverUtilized</b> 에 도달하면 호스트에 연결된 추가 가상 머신이 시작되지 않습니다.</li> <li>● <b>evenly_Distributed :</b> 클러스터의 모든 호스트에서 메모리 및 CPU 처리 부하를 균등하게 분산 합니다. 호스트가 정의된 <b>CpuOverCommitDurationMinutes, HighUtilization, MaxFreeMemoryForOverUtilized</b> 에 도달하면 호스트에 연결된 추가 가상 머신이 시작되지 않습니다.</li> <li>● <b>Cluster_Maintenance :</b> 유지 관리 작업 중에 클러스터의 작업 제한 고가용성 가상 머신을 제외하고 새로운 가상 머신을 시작할 수 없습니다. 호스트 장애가 발생하면 고가용성 가상 머신이 제대로 재시작되고 모든 가상 머신을 마이그레이션할 수 있습니다.</li> <li>● <b>Power_Saving:</b> 활용도가 낮은 호스트의 전원 소비를 줄이기 위해 사용 가능한 호스트의 하위 집합에서 메모리 및 CPU 처리 부하를 분산합니다. CPU 부하가 정의된 시간 간격보다 긴 경우의 사용률이 낮은 호스트는 전원이 꺼질 수 있도록 모든 가상 시스템을 다른 호스트로 마이그레이션합니다. 호스트가 정의된 높은 사용률 값에 도달한 경우 호스트에 연결된 추가 가상 시스템이 시작되지 않습니다.</li> <li>● <b>vm_evenly_distributed:</b> 가상 머신 수에 따라 호스트 간에 가상 머신을 균등하게 분산합니다. 호스트가 <b>HighVmCount</b> 보다 더 많은 가상 머신을 실행 중이고 <b>MigrationThreshold</b> 를 벗어나는 가상 머신 수가 있는 호스트가 한 개 이상 있는 경우 클러스터의 균형을 맞추지 않은 것으로 간주됩니다.</li> </ul>
<p>속성</p>	<p>선택한 정책에 따라 다음 속성이 표시되고 필요한 경우 편집할 수 있습니다.</p> <ul style="list-style-type: none"> <li>● <b>HighVmCount:</b> 로드 밸런싱을 활성화하기 위해 호스트별로 실행해야 하는 최소 가상 머신 수를 설정합니다. 기본값은 하나의 호스트에서 실행 중인 가상 머신 10입니다. 로드 밸런싱은 클러스터에 하나 이상의 호스트가 가상 머신을 실행하는 HighVmCount가 있는 경우에 만 활성화됩니다.</li> <li>● <b>MigrationThreshold:</b> 호스트에서 가상 머신을 마이그레이션하기 전에 버퍼를 정의합니다. 가장 많이 사용되는 호스트와 최소 사용률</li> </ul>

필드	설명/Action
	<p>호스트 간의 가상 시스템 수의 최대 포함 차이입니다. 클러스터의 모든 호스트에 마이그레이션 임계값에 속하는 가상 머신 수가 있을 때 클러스터가 균형을 유지합니다. 기본값은 5입니다.</p> <ul style="list-style-type: none"> <li>● <b>SpmVmGrace:ECDHE</b>: 호스트에서 예약할 가상 머신의 슬롯 수를 정의합니다. ECDHE 호스트는 다른 호스트보다 부하가 낮기 때문에 이 변수는 다른 호스트와 비교하여 실행할 수 있는 가상 시스템 수를 정의합니다. 기본값은 5입니다.</li> <li>● <b>CpuOverCommitDurationMinutes</b>: 스케줄링 정책이 작업을 수행하기 전에 호스트에서 정의된 사용률 값 외부에서 CPU 로드를 실행할 수 있는 시간(분)을 설정합니다. 정의된 시간 간격은 CPU 로드 활성화 정책의 일시적인 급증으로부터 보호되고 불필요한 가상 머신 마이그레이션을 유도합니다. 최대 두 개의 문자입니다. 기본값은 2입니다.</li> <li>● <b>HighUtilization</b>: 백분율로 표시합니다. 호스트가 정의된 시간 간격에 대해 높은 사용률 값을 초과하여 CPU 사용량을 사용하여 실행하는 경우 Red Hat Virtualization Manager는 호스트의 CPU 로드가 최대 서비스 임계값보다 낮을 때까지 가상 머신을 클러스터의 다른 호스트로 마이그레이션합니다. 기본값은 80입니다.</li> <li>● <b>LowUtilization</b>: 백분율로 표시합니다. 호스트가 정의된 시간 간격의 사용률이 낮은 값보다 낮은 CPU 사용량으로 실행되는 경우 Red Hat Virtualization Manager는 가상 머신을 클러스터의 다른 호스트로 마이그레이션합니다. Manager는 원래 호스트 시스템의 전원을 끄고 로드 밸런싱이 필요한 경우 다시 시작하거나 클러스터에 사용 가능한 호스트가 충분하지 않은 경우 다시 시작합니다. 기본값은 20입니다.</li> <li>● <b>scaleDown</b>: 호스트의 점수를 지정된 양으로 나누어 HA weight 함수의 영향을 줄입니다. 이는 none 을 포함하여 모든 정책에 추가할 수 있는 선택적 속성입니다.</li> <li>● <b>HostsInReserve</b>: 실행 중인 가상 머신이 없는 경우에도 계속 실행하도록 여러 호스트를 지정합니다. power_ECDHE 정책에 추가할 수 있는 선택적 속성입니다.</li> <li>● <b>EnableAutomaticHostPowerManagement</b>: 클러스터의 모든 호스트에 대해 자동 전원 관리를 활성화합니다. power_ECDHE 정책에 추가할 수 있는 선택적 속성입니다. 기본값은 true 입니다.</li> <li>● <b>MaxFreeMemoryForOverUtilized</b>: 최소 서비스 수준에 필요한 최소 사용 가능한 메모리 (MB)를 설정합니다. 호스트의 사용 가능한 메모리가 이 값 이하에서 실행되는 경우 Red Hat Virtualization Manager는 가상 머신을 클러스터의 다른 호스트로 마이그레이션하지만 호스트의 사용 가능한 메모리는 최소 서비스 임계값 미만입니다.</li> </ul>

필드	설명/Action
	<p><b>MaxFreeMemoryForOverUtilized</b> 및 <b>MinFreeMemoryForOverUtilized</b> 를 OMB 로 설정하면 메모리 기반 밸런싱이 비활성화 됩니다. <b>MaxFreeMemoryForOverUtilized</b> 가 설정된 경우 예기치 않은 동작을 방지하려면 <b>MinFreeMemoryForUnderUtilized</b> 도 설정해야 합니다. 이는 <b>power_ECDHE</b> 및 <b>evenly_distributed</b> 정책에 추가할 수 있는 선택적 속성입니다.</p> <ul style="list-style-type: none"> <li>● <b>MinFreeMemoryForUnderUtilized</b>: 호스트가 활용도가 낮은 것으로 간주되기 전에 최소 사용 가능한 메모리(MB)를 설정합니다. 호스트의 사용 가능한 메모리가 이 값 이상으로 실행되는 경우 Red Hat Virtualization Manager 는 가상 머신을 클러스터의 다른 호스트로 마이그레이션하고 호스트 시스템의 전원을 자동으로 끄고 로드 밸런싱에 필요한 호스트가 필요하거나 클러스터에 사용 가능한 호스트가 충분하지 않을 때 다시 시작합니다.</li> </ul> <p><b>MaxFreeMemoryForOverUtilized</b> 및 <b>MinFreeMemoryForOverUtilized</b> 를 OMB 로 설정하면 메모리 기반 밸런싱이 비활성화 됩니다. <b>MinFreeMemoryForUnderUtilized</b> 가 설정된 경우 <b>MaxFreeMemoryForOverUtilized</b> 도 예기치 않은 동작을 방지하도록 설정해야 합니다. 이는 <b>power_ECDHE</b> 및 <b>evenly_distributed</b> 정책에 추가할 수 있는 선택적 속성입니다.</p> <ul style="list-style-type: none"> <li>● <b>HeSparesCount</b>: 마이그레이션 또는 종료 시 Manager 가상 머신을 시작하기 위해 사용 가능한 메모리를 충분히 예약해야 하는 추가 자체 호스팅 엔진 노드 수를 설정합니다. 이렇게 하면 Manager 가상 시스템에 사용 가능한 메모리를 충분히 남겨 두지 않는 경우 다른 가상 시스템이 자체 호스팅 엔진 노드에서 시작되지 않습니다. 이는 <b>power_ECDHE</b>, <b>vm_evenly_distributed</b>, <b>evenly_distributed</b> 정책에 추가할 수 있는 선택적 속성입니다. 기본값은 0 입니다.</li> </ul>
스케줄러 최적화	<p>호스트 조정/순서에 대한 스케줄링을 최적화합니다.</p> <ul style="list-style-type: none"> <li>● <b>Utilization에 최적화</b>: 최상의 선택을 할 수 있도록 예약에 가중치 모듈이 포함됩니다.</li> <li>● <b>속도 최적화</b>: Skips 호스트 가중치는 보류중인 10 개 이상의 요청이있는 경우입니다.</li> </ul>
신뢰할 수 있는 서비스 사용	<p>OpenAttestation 서버와의 통합을 활성화합니다. 이를 활성화하려면 <b>engine-config</b> 툴을 사용하여 OpenAttestation 서버의 세부 정보를 입력합니다. 자세한 내용은 <a href="#">12.9절</a>. "신뢰할 수 있는 컴퓨팅 풀"의 내용을 참조하십시오.</p>

필드	설명/Action
HA 사용	Manager를 활성화하여 고가용성 가상 머신의 클러스터 용량을 모니터링합니다. Manager를 사용하면 기존 호스트가 예기치 않게 실패하는 경우 마이그레이션하기 위해 고가용성으로 지정된 가상 머신의 클러스터 내에 적절한 용량이 있는지 확인합니다.
사용자 정의 일련 번호 정책 제공	이 확인란을 사용하면 클러스터의 가상 머신에 대한 일련 번호 정책을 지정할 수 있습니다. 다음 옵션 중 하나를 선택합니다. <ul style="list-style-type: none"> <li>● <b>host ID:</b> 호스트의 UUID를 가상 시스템의 일련 번호로 설정합니다.</li> <li>● <b>VM ID:</b> 가상 머신의 UUID를 일련 번호로 설정합니다.</li> <li>● <b>사용자 정의 일련 번호:</b> 사용자 정의 일련 번호를 지정할 수 있습니다.</li> </ul>

호스트의 사용 가능한 메모리가 **20%** 미만으로 떨어지면 **oboons .Controllers.Balloonning guest:half1**에서 **1096400**에서 **ECDHE580** 으로의 **ballooning** 명령은 **/var/log/vdsm/mom.log** 에 기록됩니다. **/var/log/vdsm/mom.log** 는 **Memory Overcommit Manager** 로그 파일입니다.

### 8.2.6. 클러스터 콘솔 설정 설명

아래 표에서는 **New Cluster** (새 클러스터) 및 **Edit Cluster** (클러스터 편집) 창의 **Console** 탭에 대한 설정을 설명합니다.

표 8.8. 콘솔 설정

필드	설명/Action
클러스터에 SPICE 프록시 정의	글로벌 구성에 정의된 SPICE 프록시를 재정의하려면 이 확인란을 선택합니다. 이 기능은 사용자(예: VM 포털을 통해 연결)가 하이퍼바이저가 상주하는 네트워크 외부에 있는 경우에 유용합니다.
재정의된 SPICE 프록시 주소	SPICE 클라이언트가 가상 머신에 연결하는 프록시입니다. 주소는 다음 형식이어야 합니다. <p>protocol://[host]:[port]</p>

### 8.2.7. 펜싱 정책 설정 설명

아래 표에서는 새 클러스터 및 클러스터 편집 창의 **Fencing Policy** 탭에 대한 설정을 설명합니다.

표 8.9. 펜싱 정책 설정

필드	설명/Action
펜싱 활성화	클러스터에서 펜싱을 활성화합니다. 펜싱은 기본적으로 활성화되어 있지만 필요한 경우 비활성화할 수 있습니다. 예를 들어 임시 네트워크 문제가 발생하거나 예상되는 경우 관리자는 진단 또는 유지 관리 작업이 완료될 때까지 펜싱을 비활성화할 수 있습니다. 펜싱이 비활성화되면 응답하지 않는 호스트에서 실행되는 고가용성 가상 머신은 다른 위치에서 다시 시작되지 않습니다.
호스트에서 스토리지에서 라이브 리스가 있는 경우 펜싱을 건너뛵니다.	이 확인란을 선택하면 Non Responsive이고 여전히 스토리지에 연결된 클러스터의 모든 호스트가 펜싱되지 않습니다.
클러스터 연결 문제에서 펜싱 건너뛰기	이 확인란을 선택하면 연결 문제가 발생한 호스트의 백분율이 정의된 <b>임계값</b> 보다 크거나 같은 경우 펜싱이 일시적으로 비활성화됩니다. <b>Threshold</b> 값은 드롭다운 목록에서 선택됩니다. 사용 가능한 값은 <b>25,50,75,100</b> 입니다.
veluster Clair가 가동된 경우 펜싱 건너 뛰기	이 옵션은 Red Hat Gluster Storage 기능이 활성화된 경우에만 사용할 수 있습니다. 이 확인란을 선택하면 반란이 실행 중이고 다른 피어에서 도달할 수 있는 경우 펜싱을 건너뛵니다. <a href="#">Chapter 2</a> 를 참조하십시오. 자세한 내용은 <a href="#">Fencing Policies</a> 및 <a href="#">부록 A. Red Hat Gluster Storage에 대한 Fencing Policies for Red Hat Hyperconverged Infrastructure</a> 를 사용하여 고가용성을 구성합니다.
ClusterLogginguster 쿼럼이 충족되지 않는 경우 펜싱 건너뛰기	이 옵션은 Red Hat Gluster Storage 기능이 활성화된 경우에만 사용할 수 있습니다. 이 확인란을 선택하면 유예가 실행 중이고 호스트를 종료하면 쿼럼이 손실되면 펜싱을 건너뛵니다. <a href="#">Chapter 2</a> 를 참조하십시오. 자세한 내용은 <a href="#">Fencing Policies</a> 및 <a href="#">부록 A. Red Hat Gluster Storage에 대한 Fencing Policies for Red Hat Hyperconverged Infrastructure</a> 를 사용하여 고가용성을 구성합니다.

**8.2.8. 클러스터에서 호스트에 대한 로드 및 전원 관리 정책 설정**

**evenly\_distributed** 및 **power\_ECDHE** 스케줄링 정책을 사용하면 허용 가능한 메모리 및 **CPU** 사용량을 지정하고 가상 머신을 호스트에서 또는 호스트로 마이그레이션해야 하는 지점을 지정할 수 있습니다. **vm\_evenly\_distributed** 예약 정책은 가상 머신 수에 따라 가상 머신을 균등하게 분배합니다. 클러스터의 호스트 간에 자동 로드 밸런싱을 사용하도록 스케줄링 정책을 정의합니다. 각 스케줄링 정책에 대한 자세한 설명은 [8.2.5절. “설명된 스케줄링 정책 설정”](#)을 참조하십시오.



## 호스트에 대한 로드 및 전원 관리 정책 설정

1. **Compute** → **Clusters** (클러스터)를 클릭하고 클러스터를 선택합니다.
2. 편집 을 클릭합니다.
3. 스케줄링 정책 탭을 클릭합니다.
4. 다음 정책 중 하나를 선택합니다.
  - **none**
  - **vm\_evenly\_distributed**
    - a. **HighVmCount** 필드에서 부하 분산을 활성화하려면 하나 이상의 호스트에서 실행해야 하는 가상 머신의 최소 수를 설정합니다.
    - b. 가장 많이 사용되는 호스트의 가상 머신 수와 **MigrationThreshold** 필드의 최소 호스트에서 가상 머신 수 간의 최대 허용 차이를 정의합니다.
    - c. **SpmVmGrace** 필드의 **ECDHE** 호스트에 예약할 가상 머신의 슬롯 수를 정의합니다.
    - d. 필요한 경우 **HeSparesCount** 필드에 마이그레이션 또는 종료 시 **Manager** 가상 시스템을 시작할 수 있는 충분한 사용 가능한 메모리를 예약할 수 있는 추가 셀프 호스트 엔진 노드 수를 입력합니다. 자세한 내용은 [15.3절. “추가 호스트의 셀프 호스트 엔진에 예약된 메모리 슬롯 구성”](#)를 참조하십시오.
- **evenly\_distributed**
  - a. 스케줄링 정책이 **CpuOverCommitDurationMinutes** 필드에서 작업을 수행하기 전에 호스트에서 정의된 사용률 값 외부에서 **CPU** 부하를 실행할 수 있는 시간(분)을 설정

합니다.

- b. **HighUtilization** 필드에 가상 머신이 다른 호스트로 마이그레이션을 시작하는 CPU 사용률 백분율을 입력합니다.
- c. 가상 머신이 **MinFreeMemoryForUnderUtilized** 에서 다른 호스트로 마이그레이션하기 시작하는 최소 필요한 메모리(MB)를 입력합니다.
- d. 가상 머신이 **MaxFreeMemoryForOverUtilized** 에서 다른 호스트로 마이그레이션을 시작하는 최대 필요한 메모리(MB)를 입력합니다.
- e. 필요한 경우 **HeSparesCount** 필드에 마이그레이션 또는 종료 시 **Manager** 가상 시스템을 시작할 수 있는 충분한 사용 가능한 메모리를 예약할 수 있는 추가 셀프 호스트 엔진 노드 수를 입력합니다. 자세한 내용은 [15.3절. “추가 호스트의 셀프 호스트 엔진에 예약된 메모리 슬롯 구성”](#)를 참조하십시오.

#### power\_saving

- a. 스케줄링 정책이 **CpuOverCommitDurationMinutes** 필드에서 작업을 수행하기 전에 호스트에서 정의된 사용률 값 외부에서 CPU 부하를 실행할 수 있는 시간(분)을 설정합니다.
- b. **LowUtilization** 필드에 호스트가 활용도가 낮은 것으로 간주될 CPU 사용률을 입력합니다.
- c. **HighUtilization** 필드에 가상 머신이 다른 호스트로 마이그레이션을 시작하는 CPU 사용률 백분율을 입력합니다.
- d. 가상 머신이 **MinFreeMemoryForUnderUtilized** 에서 다른 호스트로 마이그레이션하기 시작하는 최소 필요한 메모리(MB)를 입력합니다.
- e. 가상 머신이 **MaxFreeMemoryForOverUtilized** 에서 다른 호스트로 마이그레이션을 시작하는 최대 필요한 메모리(MB)를 입력합니다.
- f. 필요한 경우 **HeSparesCount** 필드에 마이그레이션 또는 종료 시 **Manager** 가상

시스템을 시작할 수 있는 충분한 사용 가능한 메모리를 예약할 수 있는 추가 셀프 호스트 엔진 노드 수를 입력합니다. 자세한 내용은 [15.3절. “추가 호스트의 셀프 호스트 엔진에 예약된 메모리 슬롯 구성”](#)를 참조하십시오.

5. 다음 중 하나를 클러스터에 대한 스케줄러 최적화 로 선택합니다.
  - 최상의 선택을 허용하도록 예약에 가중치 모듈을 포함하려면 **Utilization(사용용)**을 선택합니다.
  - 보류 중인 요청이 **10**개 이상인 경우 **host weighting**을 건너뛰려면 **Speed** 를 선택합니다.
6. **OpenAttestation** 서버를 사용하여 호스트를 확인하고 **engine-config** 도구를 사용하여 서버의 세부 정보를 설정한 경우 신뢰할 수 있는 서비스 사용 확인란을 선택합니다.
7. 필요에 따라 **HAECDE** 사용 확인란을 선택하여 **Manager**에서고가용성 가상 시스템의 클러스터 용량을 모니터링할 수 있습니다.
8. 선택적으로 사용자 정의 일련 번호 정책 확인란을 선택하여 클러스터의 가상 머신에 대한 일련 번호 정책을 지정한 다음 다음 옵션 중 하나를 선택합니다.
  - **Host ID** 를 선택하여 호스트의 **UUID**를 가상 시스템의 일련 번호로 설정합니다.
  - **Vm ID** 를 선택하여 가상 시스템의 **UUID**를 일련 번호로 설정합니다.
  - 사용자 정의 일련번호 를 선택한 다음 텍스트 필드에 사용자 정의 일련 번호를 지정합니다.
9. **OK**를 클릭합니다.

### 8.2.9. 클러스터의 호스트에서 MoM 정책 업데이트

**Memory Overcommit Manager**는 호스트에서 메모리 **balloon** 및 **KSM** 기능을 처리합니다. 다음 번에 호스트가 재부팅 또는 유지 관리 모드로 전환된 후 호스트가 **Up** 상태로 전환될 때만 클러스터 수준에서 이러한 기능에 대한 변경 사항이 호스트에 전달됩니다. 그러나 필요한 경우 호스트가 가동 되는 동안

**MoM** 정책을 동기화하여 호스트에 중요한 변경 사항을 즉시 적용할 수 있습니다. 각 호스트에서 다음 절차를 개별적으로 수행해야 합니다.

호스트에서 **MoM** 정책 동기화

1. **Compute** → **Clusters** 를 클릭합니다.
2. 클러스터 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **Hosts** 탭을 클릭하고 업데이트된 **MoM** 정책이 필요한 호스트를 선택합니다.
4. **Sync MoM Policy** 를 클릭합니다.

호스트의 **MoM** 정책은 호스트를 유지 관리 모드로 이동하고 백업할 필요 없이 업데이트됩니다.

### 8.2.10. CPU 프로파일 생성

**CPU** 프로파일은 클러스터의 가상 머신이 실행되는 호스트에서 액세스할 수 있는 최대 처리 기능을 정의합니다. 이 기능은 해당 호스트에 사용 가능한 총 처리 기능의 백분율로 표시됩니다. **CPU** 프로파일은 데이터 센터에 정의된 **CPU** 프로파일을 기반으로 생성되며 클러스터의 모든 가상 머신에 자동으로 적용되지 않습니다. 프로파일을 적용하려면 개별 가상 머신에 수동으로 할당해야 합니다.

이 절차에서는 클러스터가 속한 데이터 센터 아래에 하나 이상의 **CPU** 품질의 서비스 항목을 이미 정의했다고 가정합니다.

**CPU** 프로파일 생성

1. **Compute** → **Clusters** 를 클릭합니다.
2. 클러스터 이름을 클릭하여 세부 정보 보기를 엽니다.

3. **CPU** 프로필 탭을 클릭합니다.
4. 새로 생성을 클릭합니다.
5. **CPU** 프로필에 대한 이름 및 설명을 입력합니다.
6. **QoS** 목록에서 **CPU** 프로필에 적용할 서비스 품질을 선택합니다.
7. **OK**를 클릭합니다.

### 8.2.11. CPU 프로필 제거

**Red Hat Virtualization** 환경에서 기존 **CPU** 프로필을 제거합니다.

#### **CPU** 프로필 제거

1. **Compute** → **Clusters** 를 클릭합니다.
2. 클러스터 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **CPU** 프로필 탭을 클릭하고 제거할 **CPU** 프로필을 선택합니다.
4. 제거를 클릭합니다.
5. **OK**를 클릭합니다.

**CPU** 프로필이 가상 머신에 할당되면 해당 가상 머신에 자동으로 기본 **CPU** 프로필이 할당됩니다.

### 8.2.12. 기존 **Red Hat Gluster Storage** 클러스터 가져오기

**Red Hat Gluster Storage** 클러스터와 클러스터에 속한 모든 호스트를 **Red Hat Virtualization Manager**로 가져올 수 있습니다.

클러스터에 있는 모든 호스트의 **IP** 주소 또는 호스트 이름 및 암호와 같은 세부 정보를 제공하면 **SSH** 를 통해 해당 호스트에서 **ovsuster peer status** 명령이 실행된 다음 클러스터의 일부인 호스트 목록을 표시합니다. 각 호스트의 지문을 수동으로 확인하고 암호를 제공해야 합니다. 클러스터의 호스트 중 하나가 다운되거나 연결할 수 없는 경우 클러스터를 가져올 수 없습니다. 새로 가져온 호스트에 **VDSM**이 설치되지 않았으므로 부트스트랩 스크립트는 가져온 후 호스트에 필요한 모든 **VDSM** 패키지를 설치하고 재부팅합니다.

### Red Hat Virtualization Manager로 기존 Red Hat Gluster Storage 클러스터 가져오기

1. **Compute** → **Clusters** 를 클릭합니다.
2. 새로 생성을 클릭합니다.
3. 클러스터가 속할 데이터 센터를 선택합니다.
4. 클러스터의 이름 및 설명을 입력합니다.
5. **Enable Gluster Service** 확인란을 선택하고 기존 **Glusteruster** 구성 가져오기 확인란을 선택합니다.  
  
기존 **Gluster Service** 가져오기 구성 필드는 **Gluster Service** 활성화 를 선택한 경우에만 표시됩니다.
6. **Hostname** 필드에 클러스터에 있는 서버의 호스트 이름 또는 **IP** 주소를 입력합니다.  
  
올바른 호스트에 연결 중인지 확인하기 위해 호스트 **SSH Runningerprint**가 표시됩니다. 호스트에 연결할 수 없거나 네트워크 오류가 있는 경우 지문 가져오기에 오류가 표시됩니다.
7. 서버에 암호 를 입력하고 확인을 클릭합니다.

8.

**Add Hosts** (호스트 추가) 창이 열리며 클러스터의 일부인 호스트 목록이 표시됩니다.

9.

각 호스트에 대해 이름 및 루트 암호를 입력합니다.

10.

모든 호스트에 동일한 암호를 사용하려면 일반 암호 사용 확인란을 선택하여 제공된 텍스트 필드에 암호를 입력합니다.

적용을 클릭하여 입력한 암호 모든 호스트를 설정합니다.

지문이 유효한지 확인하고 확인을 클릭하여 변경 사항을 제출합니다.

부트스트랩 스크립트는 가져온 후 호스트에 필요한 모든 **VDSM** 패키지를 설치하고 재부팅합니다. 기존 **Red Hat Gluster Storage** 클러스터를 **Red Hat Virtualization Manager**로 성공적으로 가져왔습니다.

### 8.2.13. 호스트 추가 창에서 설정에 대한 설명

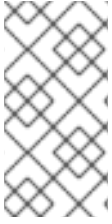
**Add Hosts** (호스트 추가) 창을 사용하면 **Gluster** 지원 클러스터의 일부로 가져온 호스트의 세부 정보를 지정할 수 있습니다. 이 창은 새 클러스터 창에서 **Gluster Service** 활성화 확인란을 선택하고 필요한 호스트 세부 정보를 제공한 후 표시됩니다.

표 8.10. **Gluster** 호스트 설정 추가

필드	설명
일반 암호 사용	클러스터에 속하는 모든 호스트에 대해 동일한 암호를 사용하려면 이 확인란을 선택합니다. <b>Password</b> 필드에 암호를 입력한 다음 <b>Apply</b> 버튼을 클릭하여 모든 호스트에서 암호를 설정합니다.
이름	호스트 이름을 입력합니다.
Hostname/IP	이 필드는 새 클러스터 창에서 제공한 호스트의 정규화된 도메인 이름 또는 IP로 자동으로 채워집니다.
루트 암호	이 필드에 암호를 입력하여 각 호스트에 다른 루트 암호를 사용합니다. 이 필드는 클러스터의 모든 호스트에 제공된 공통 암호를 재정의합니다.
지문	올바른 호스트와 연결 중인지 확인하기 위해 호스트 지문이 표시됩니다. 이 필드는 새 클러스터 창에서 제공한 호스트의 지문으로 자동으로 채워집니다.

### 8.2.14. 클러스터 제거

제거하기 전에 모든 호스트를 클러스터에서 이동합니다.



참고

**Blank** 템플릿이 있으므로 **Default** 클러스터를 제거할 수 없습니다. 그러나 **Default** 클러스터의 이름을 변경하고 새 데이터 센터에 추가할 수 있습니다.

클러스터 제거

1. **Compute** → **Clusters** (클러스터)를 클릭하고 클러스터를 선택합니다.
2. 클러스터에 호스트가 없는지 확인합니다.
3. 제거를 클릭합니다.
4. **OK**를 클릭합니다.

### 8.2.15. 메모리 최적화

호스트의 가상 머신 수를 늘리려면 *메모리 과다* 할당을 사용하여 가상 머신에 할당한 메모리가 **RAM**을 초과하고 스왑 공간을 사용할 수 있습니다.

그러나 메모리 과다 할당에 잠재적인 문제가 있습니다.

- 성능 스왑 - 메모리가 느리고 **RAM**보다 더 많은 **CPU** 리소스를 소비하여 가상 머신 성능에 영향을 미칩니다. 과도한 스왑으로 인해 **CPU** 충돌이 발생할 수 있습니다.
- **OOM**(메모리 부족) 종료 - 호스트가 스왑 공간이 부족하면 새 프로세스를 시작할 수 없으며 커널의 **OOM** 종료 데몬에서 가상 머신 게스트와 같은 활성 프로세스 종료를 시작합니다.



이러한 단점을 극복하기 위해 다음을 수행할 수 있습니다.

- **Memory Optimization** 설정 및 **Memory Overcommit Manager (MoM)** 를 사용하여 메모리 과다 할당을 제한합니다.
- 스왑 공간이 가상 메모리에 대한 최대 잠재적 수요를 수용할 수 있을 만큼 커지고 안전 여유가 남아 있게 합니다.
- **KSM(Memory ballooning and Kernel Same-page Merging)** 을 활성화하여 가상 메모리 크기를 줄입니다.

### 8.2.15.1. 메모리 최적화 및 메모리 과다 할당

메모리 최적화 설정 중 하나를 선택하여 메모리 과다 할당을 제한할 수 있습니다( 없음(없음 ), 150%, 200% ).

각 설정은 **RAM**의 백분율을 나타냅니다. 예를 들어 호스트에 **64GB RAM**이 있는 경우 **150%** 를 선택하면 가상 메모리에서 총 **96GB**의 메모리를 추가로 **32GB**로 과다 할당할 수 있습니다. 호스트가 총 **4GB**를 사용하는 경우 나머지 **92GB**를 사용할 수 있습니다. 대부분의 가상 머신( 시스템 탭의 메모리 크기 )을 할당할 수 있지만, 일부 가상 머신은 할당 해제된 안전 범위로 두는 것을 고려합니다.

가상 메모리에 대한 수요가 급증하는 경우 **MoM**, 메모리 **ballooning** 및 **KSM**이 가상 메모리를 다시 최적화하기 전에 성능에 영향을 미칠 수 있습니다. 이러한 영향을 줄이려면 실행 중인 애플리케이션 및 워크로드에 적합한 제한을 선택하십시오.

- 메모리에 대한 수요가 증가하게 되는 워크로드의 경우 **200%** 또는 **150%** 와 같은 더 높은 백분율을 선택합니다.
- 메모리 수요가 급증할수록 중요한 애플리케이션이나 워크로드의 경우 **150%** 또는 **None (0%)**과 같은 더 낮은 백분율을 선택합니다. **None** 을 선택하면 메모리 과다 할당을 방지할 수 있지만 **MoM**, 메모리 **balloon** 장치 및 **KSM**이 가상 메모리를 계속 최적화할 수 있습니다.



중요

구성을 프로덕션에 배포하기 전에 항상 광범위한 조건에서 테스트하여 메모리 최적화 설정을 테스트합니다.

메모리 최적화 설정을 구성하려면 새 클러스터 또는 클러스터 편집 창에서 최적화 탭을 클릭합니다.  
**8.2.3절. “최적화 설정 설명”** 을 참조하십시오.

추가 댓글:

- [호스트 ECDHE 보기](#)에는 오버 커밋 비율의 크기를 조정하는 데 유용한 기록 정보가 표시됩니다.
- 사용 가능한 실제 메모리는 **KSM** 및 메모리 볼루닝 변경에 의해 달성되는 메모리 최적화의 양이 지속적으로 변경되기 때문에 실시간으로 확인할 수 없습니다.
- 가상 머신이 가상 메모리 제한에 도달하면 새 앱을 시작할 수 없습니다.
- 호스트에서 실행할 가상 머신 수를 계획하는 경우 최대 가상 메모리(실제 메모리 크기 및 메모리 최적화 설정)를 시작점으로 사용합니다. 메모리 볼루닝 및 **KSM**과 같은 메모리 최적화에 의해 달성되는 작은 가상 메모리를 고려하지 마십시오.

### 8.2.15.2. 스왑 공간 및 메모리 과다 할당

**Red Hat**은 스왑 공간 구성을 위한 권장 사항을 제공합니다.

이러한 권장 사항을 적용할 때 지침에 따라 스왑 공간의 크기를 최악의 경우 "최종 작업 메모리"로 조정합니다. 총 가상 메모리 크기를 추정하기 위한 기준으로 실제 메모리 크기 및 메모리 최적화 설정을 사용합니다. **MoM**, 메모리 **ballooning** 및 **KSM** 최적화에서 가상 메모리 크기 감소를 제외합니다.



중요

**OOM** 상태를 방지하려면 가장 심각한 시나리오를 처리할 수 있을 만큼 스왑 공간을 크게 만들고 안전 여백을 사용할 수 있습니다. 프로덕션에 배포하기 전에 항상 다양한 조건에서 구성을 테스트합니다.

### 8.2.15.3. Memory Overcommit Manager (MoM)

**MoM(Memory Overcommit Manager)** 은 다음 두 가지 작업을 수행합니다.

- 이전 섹션에 설명된 대로 **Memory Optimization** 설정을 클러스터의 호스트에 적용하여 메모리 과다 할당을 제한합니다.
- 다음 섹션에 설명된 대로 메모리 **ballooning** 및 **KSM** 을 관리하여 메모리를 최적화합니다.

**MoM**을 활성화하거나 비활성화할 필요는 없습니다.

호스트의 사용 가능한 메모리가 **20%** 미만으로 떨어지면 **oboon .Controllers.Ballooning guest:half1**에서 **1096400**에서 **ECDHE580**까지의 **ballooning** 명령과 같은 명령은 **/var/log/vdsm/mdsm/mom.log** 에 기록됩니다.

#### 8.2.15.4. 메모리(Memory)

가상 머신은 할당된 전체 가상 메모리 양으로 시작합니다. 가상 메모리 사용량이 **RAM**을 초과하면 호스트는 스왑 공간에 더 많이 의존합니다. 활성화된 경우 메모리 **ballooning**을 사용하면 가상 머신에서 해당 메모리의 사용되지 않는 부분을 포기할 수 있습니다. 사용 가능한 메모리는 호스트의 다른 프로세스 및 가상 시스템에서 재사용할 수 있습니다. 메모리 공간을 줄이면 공간을 낮추고 성능을 향상시킬 수 있습니다.

메모리 **balloon** 장치 및 드라이버를 제공하는 **virtio-balloon** 패키지는 로드 가능한 커널 모듈(**LKM**)으로 제공됩니다. 기본적으로 자동으로 로드되도록 구성됩니다. 모듈을 블랙리스트로 지정하거나 언로드하면 볼링이 비활성화됩니다.

메모리 **balloon** 장치는 서로 직접 조정되지 않으므로 호스트의 **MoM(Memory Overcommit Manager)** 프로세스에 의존하여 각 가상 시스템의 요구 사항을 지속적으로 모니터링하고 가상 메모리를 늘리거나 줄이도록 **balloon** 장치에 지시합니다.

성능 고려 사항:

- **Red Hat**은 지속적인 고성능과 짧은 대기 시간이 필요한 워크로드에 대해 메모리 볼루닝 및 과다 할당을 권장하지 않습니다. **High-Performance** 가상 머신, 템플릿 및 풀 구성 을 참조하십시오.
- **Red Hat**은 가상 머신 밀도(**economy**)를 늘리는 경우 성능보다 메모리 증대를 권장합니다.
-

메모리 볼루는 **CPU** 사용률에 큰 영향을 미치지 않습니다. (**KSM**은 일부 **CPU** 리소스를 사용하지만 소비는 계속 부족합니다.)

메모리 증대를 활성화하려면 새 클러스터 또는 클러스터 편집 창에서 최적화 탭을 클릭합니다. 그런 다음 메모리 **group(메모리) Optimization(메모리 최적화 활성화)** 확인란을 선택합니다. 이 설정을 사용하면 이 클러스터의 호스트에서 실행 중인 가상 머신에서 메모리 과다 할당이 가능합니다. 이 확인란을 선택하면 **MoM**이 모든 가상 머신의 보장된 메모리 크기 제한으로 가능한 경우 볼루닝을 시작합니다. **8.2.3절. “최적화 설정 설명”**을 참조하십시오.

이 클러스터의 각 호스트는 상태가 **Up**으로 변경될 때 **balloon** 정책 업데이트를 수신합니다. 필요한 경우 상태를 변경하지 않고도 호스트에서 **balloon** 정책을 수동으로 업데이트할 수 있습니다. **8.2.9절. “클러스터의 호스트에서 MoM 정책 업데이트”**을 참조하십시오.

### 8.2.15.5. 커널 동일 페이지 병합(KSM)

가상 머신이 실행되면 공통 라이브러리 및 사용량이 많은 데이터와 같은 항목에 대해 중복 메모리 페이지를 생성하는 경우가 많습니다. 또한 유사한 게스트 운영 체제 및 애플리케이션을 실행하는 가상 머신은 가상 메모리에 중복 메모리 페이지를 생성합니다.

활성화된 경우 **KSM(커널 동일 페이지 병합)**은 호스트의 가상 메모리를 검사하고 중복 메모리 페이지를 제거하고 여러 애플리케이션 및 가상 머신에서 나머지 메모리 페이지를 공유합니다. 이러한 공유 메모리 페이지는 **COW(Copy-On-Write)**로 표시됩니다. 가상 시스템이 페이지에 변경 사항을 작성해야 하는 경우 해당 복사에 수정 사항을 쓰기 전에 복사를 먼저 수행합니다.

**KSM**이 활성화되는 동안 **MoM**은 **KSM**을 관리합니다. **KSM**을 수동으로 구성하거나 제어할 필요가 없습니다.

**KSM**은 두 가지 방법으로 가상 메모리 성능을 향상시킵니다. 공유 메모리 페이지가 더 자주 사용되므로 호스트는 캐시 또는 기본 메모리에 저장하여 메모리 액세스 속도가 향상됩니다. 또한 메모리 과다 할당을 통해 **KSM**은 가상 메모리 공간을 줄여 스왑 가능성을 줄이고 성능을 향상시킬 수 있습니다.

**KSM**은 메모리 **ballooning**보다 더 많은 **CPU** 리소스를 사용합니다. **CPU KSM** 소비의 양은 부족 시 일관되게 유지됩니다. 호스트에서 동일한 가상 머신 및 애플리케이션을 실행하면 **KSM**에 유사한 가상 머신 실행보다 메모리 페이지를 병합할 수 있는 기회가 더 많습니다. 대부분의 가상 머신 및 애플리케이션을 실행하는 경우 **KSM**을 사용하는 **CPU** 비용이 이점을 상쇄할 수 있습니다.

성능 고려 사항:

-

**KSM** 데몬이 많은 메모리를 병합한 후 커널 메모리 회계 통계는 결국 서로 모순될 수 있습니다. 시스템에 많은 양의 사용 가능한 메모리가 있는 경우 **KSM**을 비활성화하여 성능을 향상시킬 수 있습니다.

- **Red Hat**은 지속적인 고성능 및 짧은 대기 시간이 필요한 워크로드에 **KSM** 및 오버 커밋을 권장하지 않습니다. **High-Performance** 가상 머신, 템플릿 및 풀 구성 을 참조하십시오.
- **Red Hat**은 가상 머신 밀도(**Economy**)를 늘리는 경우 성능보다 더 중요한 **KSM**을 권장합니다.

**KSM**을 활성화하려면 새 클러스터 또는 클러스터 편집 창에서 최적화 탭을 클릭합니다. 그런 다음 **KSM** 사용 확인란을 선택합니다. 이 설정을 사용하면 **MoM**이 필요한 경우 **KSM**을 실행할 수 있으며 메모리 절약을 통해 **CPU** 비용보다 큰 이점을 얻을 수 있습니다. **8.2.3절. “최적화 설정 설명”**을 참조하십시오.

#### 8.2.16. 클러스터 호환성 버전 변경

**Red Hat Virtualization** 클러스터에는 호환성 버전이 있습니다. 클러스터 호환성 버전은 클러스터의 모든 호스트에서 지원하는 **Red Hat Virtualization**의 기능을 나타냅니다. 클러스터 호환성은 클러스터에서 최소 가능 호스트 운영 체제 버전에 따라 설정됩니다.



#### 중요

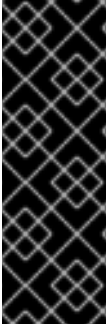
클러스터 호환성 버전을 변경하려면 먼저 클러스터의 모든 호스트를 원하는 호환성 수준을 지원하는 수준으로 업데이트해야 합니다. 업데이트가 사용 가능함을 나타내는 호스트 옆에 아이콘이 있는지 확인합니다.

#### 절차

1. 관리 포털에서 **Compute** → **Clusters** )를 클릭합니다.
2. 변경할 클러스터를 선택하고 편집 을 클릭합니다.
3. 일반 탭에서 호환성 버전을 원하는 값으로 변경합니다.
4. **OK**를 클릭합니다. 클러스터 호환성 버전 변경 확인 대화 상자가 열립니다.

5.

확인을 클릭하여 확인합니다.



중요

일부 가상 머신 및 템플릿이 잘못 구성되어 있음을 오류 메시지가 표시될 수 있습니다. 이 오류를 해결하려면 각 가상 시스템을 수동으로 편집합니다. **Edit Virtual Machine** (가상 머신 편집) 창에서 수정할 내용을 표시하는 추가 검증 및 경고를 제공합니다. 문제가 자동으로 수정되고 가상 머신의 구성을 다시 저장해야 하는 경우가 있습니다. 각 가상 머신을 편집한 후 클러스터 호환성 버전을 변경할 수 있습니다.

클러스터 호환성 버전을 업데이트한 후 관리 포털에서 재부팅하거나 게스트 운영 체제 내에서 **REST API**를 사용하여 실행 중이거나 일시 중지된 모든 가상 머신의 클러스터 호환성 버전을 업데이트해야 합니다. 재부팅이 필요한 가상 머신은 보류 중인 변경 사항 아이콘(



)로 표시됩니다. 프리뷰에 있는 가상 머신 스냅샷의 클러스터 호환성 버전을 변경할 수 없습니다. 먼저 프리뷰를 커밋하거나 실행 취소해야 합니다.

자체 호스팅 엔진 환경에서는 **Manager** 가상 머신을 재시작할 필요가 없습니다.

편리한 시간에 가상 시스템을 재부팅할 때까지 기다릴 수 있지만 가상 머신에서 최신 구성을 사용하도록 즉시 재부팅하는 것이 좋습니다. 이전 구성으로 업데이트되지 않은 가상 머신을 배포하고, 재부팅하기 전에 가상 머신을 다른 변경 사항을 적용하면 새 구성을 덮어쓸 수 있습니다.

데이터 센터의 모든 클러스터 및 가상 머신의 호환성 버전을 업데이트한 후 데이터 센터 자체의 호환성 버전을 변경할 수 있습니다.

## 9장. 논리 네트워크

### 9.1. 논리적 네트워크 작업

#### 9.1.1. 네트워킹 작업 수행

네트워크 → 네트워크는 사용자가 논리적 네트워크 관련 작업을 수행하고 각 네트워크의 속성 또는 다른 리소스와의 연결을 기반으로 논리 네트워크를 검색할 수 있는 중앙 위치를 제공합니다. 새, 편집 및 제거 버튼을 사용하면 데이터 센터 내에서 논리 네트워크를 생성, 변경 및 삭제할 수 있습니다.

각 네트워크 이름을 클릭하고 세부 정보 보기에서 탭을 사용하여 다음을 포함한 기능을 수행합니다.

- 네트워크 연결 또는 클러스터 및 호스트에 분리
- 가상 머신 및 템플릿에서 네트워크 인터페이스 제거
- 사용자가 네트워크 액세스 및 관리를 위한 권한 추가 및 제거

이러한 기능은 각 개별 리소스를 통해 액세스할 수도 있습니다.



#### 주의

호스트가 실행 중인 경우 호스트에 연결할 수 없는 위험이 있으므로 데이터 센터 또는 클러스터에서 네트워킹을 변경하지 마십시오.



중요

**Red Hat Virtualization** 노드를 사용하여 서비스를 제공하려는 경우 **Red Hat Virtualization** 환경이 작동을 중지하면 서비스가 중지됩니다.

이는 모든 서비스에 적용되지만 특히 **Red Hat Virtualization**에서 다음 실행을 인지하고 있어야 합니다.

- 디렉터리 서비스
- **DNS**
- 스토리지

**9.1.2. 데이터 센터 또는 클러스터에서 새 논리 네트워크 생성**

논리적 네트워크를 생성하고 데이터 센터 또는 데이터 센터의 클러스터에서 해당 사용을 정의합니다.

데이터 센터 또는 클러스터에서 새 논리 네트워크 생성

1. **Compute** → **Data Centers** 또는 **Compute** → **Clusters** 를 클릭합니다.
2. 데이터 센터 또는 클러스터 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **Logical Networks** 탭을 클릭합니다.
4. **New Logical Network** 창을 엽니다.
  - 데이터 센터 세부 정보 보기에서 새로 생성을 클릭합니다.
  - 클러스터 세부 정보 보기에서 네트워크 추가를 클릭합니다.



5. 논리 네트워크에 이름, 설명, 주석을 입력합니다.
6. 필요한 경우 **VLAN** 태그 지정 활성화를 활성화합니다.
7. 필요한 경우 **VM** 네트워크 비활성화.
8. 필요한 경우 외부 공급자 생성 확인란을 선택합니다. 이렇게 하면 네트워크 레이블, **VM** 네트워크 및 **MTU** 옵션이 비활성화됩니다. 자세한 내용은 [14장. 외부 공급자](#) 을 참조하십시오.
9. 외부 공급자를 선택합니다. 외부 공급자 목록에는 읽기 전용 모드에 있는 외부 공급자가 포함되어 있지 않습니다.  
  
외부 공급자 목록에서 **ovirt-provider-ovn** 을 선택하고 물리적 네트워크에 연결을 선택하지 않은 상태로 유지하여 내부 격리된 네트워크를 생성할 수 있습니다.
10. 새 레이블을 입력하거나 **Network Label** (네트워크 라벨) 텍스트 필드에 논리 네트워크에 대한 기존 레이블을 선택합니다.
11. **MTU** 값을 **Default (1500)** 또는 **Custom** 로 설정합니다.
12. 외부 공급자 드롭다운 목록에서 **ovirt-provider-ovn** 을 선택한 경우 네트워크에서 보안 그룹을 구현해야 하는지 여부를 정의합니다. 자세한 내용은 [9.1.7절. “설명되는 논리적 네트워크 일반 설정”](#) 을 참조하십시오.
13. **Cluster** 탭에서 네트워크를 할당할 클러스터를 선택합니다. 논리 네트워크가 필수 네트워크가 될지 여부를 지정할 수도 있습니다.
14. 외부 공급자에서 **Create**를 선택하면 **Subnet** 탭이 표시됩니다. **Subnet** 탭에서 **Create subnet** 을 선택하고 **Name, CIDR, Gateway** 주소를 입력하고 논리 네트워크에서 제공할 서브넷의 **IP** 버전을 선택합니다. 필요에 따라 **DNS** 서버를 추가할 수도 있습니다.
15. **vNIC Profiles** 탭에서 필요에 따라 논리 네트워크에 **vNIC** 프로필을 추가합니다.

16.

**OK**를 클릭합니다.

논리 네트워크의 레이블을 입력하면 해당 레이블이 있는 모든 호스트 네트워크 인터페이스에 자동으로 추가됩니다.



참고

새 논리적 네트워크를 생성하거나 디스플레이 네트워크로 사용되는 기존 논리적 네트워크를 변경할 때 네트워크를 사용하거나 변경 사항을 적용하기 전에 해당 네트워크를 사용하는 실행 중인 가상 머신을 재부팅해야 합니다.

### 9.1.3. 논리 네트워크 편집



중요

논리적 네트워크는 호스트의 네트워크 구성과 동기화되지 않은 경우 편집하거나 다른 인터페이스로 이동할 수 없습니다. 네트워크를 동기화하는 방법은 [9.4.2절. “호스트 네트워크 인터페이스 편집 및 호스트에 논리 네트워크 할당”](#)을 참조하십시오.

#### 논리 네트워크 편집

1. **Compute** → **Data Centers** (데이터 센터)를 클릭합니다.
2. 데이터 센터의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **Logical Networks** 탭을 클릭하고 논리 네트워크를 선택합니다.
4. 편집 을 클릭합니다.
5. 필요한 설정을 편집합니다.



## 참고

가상 머신을 중지하지 않고도 기본 네트워크를 제외하고 새 네트워크 또는 기존 네트워크의 이름을 편집할 수 있습니다.

6.

**OK**를 클릭합니다.



## 참고

다중 호스트 네트워크 구성은 네트워크가 할당된 데이터 센터 내의 모든 호스트에 업데이트된 네트워크 설정을 자동으로 적용합니다. 변경 사항은 네트워크를 사용하는 가상 머신이 다운된 경우에만 적용할 수 있습니다. 호스트에 이미 구성된 논리 네트워크의 이름을 변경할 수 없습니다. 해당 네트워크를 사용하는 가상 머신 또는 템플릿이 실행되는 동안 **VM** 네트워크 옵션을 비활성화할 수 없습니다.

#### 9.1.4. 논리 네트워크 제거

네트워크 네트워크 또는 컴퓨팅 → 데이터 센터에서 논리적 네트워크를 제거할 수 있습니다. 다음 절차에서는 데이터 센터와 연결된 논리 네트워크를 제거하는 방법을 보여줍니다. **Red Hat Virtualization** 환경의 경우 **ovirtmgmt** 관리 네트워크로 사용되는 논리 네트워크가 하나 이상 있어야 합니다.

##### 논리 네트워크 제거

1. **Compute → Data Centers** (데이터 센터)를 클릭합니다.
2. 데이터 센터의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 논리 네트워크 탭을 클릭하여 데이터 센터의 논리 네트워크를 나열합니다.
4. 논리 네트워크를 선택하고 제거를 클릭합니다.
5. 필요한 경우 외부 공급자가 네트워크를 제공하는 경우 공급자에서 외부 네트워크 제거 확인란을 선택하여 **Manager** 및 외부 공급자 모두에서 논리 네트워크를 제거합니다. 외부 공급자가 읽기 전용 모드이면 확인란이 회색으로 표시됩니다.

6.

**OK**를 클릭합니다.

논리 네트워크는 **Manager**에서 제거되며 더 이상 사용할 수 없습니다.

### 9.1.5. 기본 경로로 비관리 논리 네트워크 구성

클러스터의 호스트에서 사용하는 기본 경로는 관리 네트워크(**ovirtmgmt**)입니다. 다음 절차에서는 비관리 논리 네트워크를 기본 경로로 구성하는 지침을 제공합니다.

사전 요구 사항:

- **default\_route** 사용자 지정 속성을 사용하는 경우 연결된 모든 호스트에서 사용자 지정 속성을 지우고 다음 절차를 따라야 합니다.

기본 경로 역할 구성

1.

**Network** → **Networks** 를 클릭합니다.

2.

비관리 논리 네트워크의 이름을 클릭하여 세부 정보에 액세스할 기본 경로로 구성합니다.

3.

클러스터 탭을 클릭합니다.

4.

네트워크 관리를 클릭하여 네트워크 관리 창을 엽니다.

5.

적절한 클러스터의 **Default Route** 확인란을 선택합니다.

6.

**OK**를 클릭합니다.

네트워크가 호스트에 연결되면 선택한 네트워크에 호스트의 기본 경로가 설정됩니다. 호스트가 클러스터에 추가되기 전에 기본 경로 역할을 구성하는 것이 좋습니다. 클러스터에 호스트가 이미 포함된 경우

변경 사항을 동기화할 때까지 동기화되지 않을 수 있습니다.

### IPv6의 중요한 제한 사항

- IPv6의 경우 **Red Hat Virtualization**은 정적 주소 지정만 지원합니다.
- 두 네트워크 모두 단일 게이트웨이(동일한 서브넷에 있음)를 공유하는 경우 기본 경로 역할을 관리 네트워크(**ovirtmgmt**)에서 다른 논리 네트워크로 이동할 수 있습니다.
- 호스트와 **Manager**가 동일한 서브넷에 있지 않은 경우 **IPv6** 게이트웨이가 제거되었기 때문에 **Manager**의 연결이 끊어집니다.
- 기본 경로 역할을 비관리 네트워크로 이동하면 네트워크 인터페이스에서 **IPv6** 게이트웨이가 제거되고 경고가 생성됩니다. '기본 경로 역할' 네트워크가 더 이상 네트워크 **ovirtmgmt**가 아닙니다. 이 네트워크에서 **IPv6** 게이트웨이가 제거되었습니다."

#### 9.1.6. 논리 네트워크의 게이트웨이 보기 또는 편집

사용자는 논리 네트워크에 대해 **IP** 주소 및 서브넷 마스크와 함께 게이트웨이를 정의할 수 있습니다. 이는 호스트에 여러 네트워크가 있고 트래픽을 기본 게이트웨이가 아닌 지정된 네트워크를 통해 라우팅해야 하는 경우 필요합니다.

호스트에 여러 네트워크가 있고 게이트웨이가 정의되지 않은 경우 반환 트래픽은 기본 게이트웨이를 통해 라우팅되며 이는 의도된 대상에 도달하지 못할 수 있습니다. 이로 인해 사용자가 호스트를 **ping** 할 수 없습니다.

**Red Hat Virtualization**은 인터페이스가 위 또는 중단될 때마다 여러 게이트웨이를 자동으로 처리합니다.

#### 논리 네트워크의 게이트웨이 보기 또는 편집

1. 컴퓨팅 → 호스트.
2. 호스트 이름을 클릭하여 세부 정보 보기를 엽니다.

3. 네트워크 인터페이스 탭을 클릭하여 호스트에 연결된 네트워크 인터페이스와 해당 구성을 나열합니다.
4. **Setup Host Networks (호스트 네트워크 설정)**를 클릭합니다.
5. 할당된 논리적 네트워크 위에 커서를 올려 놓은 후 연필 아이콘을 클릭하여 관리 네트워크 편집 창을 엽니다.

**Edit Management Network (관리 네트워크 편집)** 창에 네트워크 이름, 부팅 프로토콜, IP, 서브넷 마스크 및 게이트웨이 주소가 표시됩니다. 정적 부팅 프로토콜을 선택하여 주소 정보를 수동으로 편집할 수 있습니다.

### 9.1.7. 설명되는 논리적 네트워크 일반 설정

아래 표에서는 **New Logical Network** 및 **Edit Logical Network** 창의 **General** 탭에 대한 설정을 설명합니다.

표 9.1. 새로운 논리 네트워크 및 편집 논리 네트워크 설정

필드 이름	설명
이름	논리 네트워크의 이름입니다. 이 텍스트 필드는 대문자 및 소문자, 숫자, 하이픈, 밑줄을 조합한 고유한 이름이어야 합니다.  논리 네트워크의 이름은 15자를 초과할 수 있지만 ASCII가 아닌 문자를 포함할 수 있지만 호스트 식별자 ( <i>vdsm_name</i> )는 정의된 이름과 다릅니다. 이러한 이름의 매핑을 표시하는 방법은 <a href="#">VDSM 이름을 논리 네트워크 이름에 매핑 매핑을 참조하십시오</a> .
설명	논리 네트워크에 대한 설명입니다. 이 텍스트 필드에는 40자 제한이 있습니다.
주석	논리 네트워크에 대한 일반 텍스트, 사람이 읽을 수 있는 주석을 추가하는 필드입니다.
외부 공급자에서 생성	Manager에 외부 공급자로 추가된 OpenStack Networking 인스턴스에 논리적 네트워크를 생성할 수 있습니다.  외부 공급자 - 논리 네트워크가 생성될 외부 공급자를 선택할 수 있습니다.

필드 이름	설명
VLAN 태그 지정 활성화	VLAN 태그 지정은 논리 네트워크에서 모든 네트워크 트래픽에 특정 특성을 제공하는 보안 기능입니다. VLAN 태그된 트래픽은 해당 특성이 없는 인터페이스에서 읽을 수 없습니다. 또한 논리 네트워크에서 VLAN을 사용하면 단일 네트워크 인터페이스를 태그된 여러 VLAN과 연결할 수 있습니다. VLAN 태그 지정이 활성화된 경우 텍스트 항목 필드에 숫자 값을 입력합니다.
VM 네트워크	가상 머신만 이 네트워크를 사용하는 경우 이 옵션을 선택합니다. 스토리지 통신과 같이 가상 머신이 포함되지 않은 트래픽에 네트워크를 사용하는 경우 이 확인란을 선택하지 마십시오.
MTU	최대 전송 단위(MTU)를 괄호()에 지정된 값으로 설정하는 <b>Default</b> , 또는 <b>Custom</b> 을 선택하여 논리 네트워크에 대한 사용자 지정 MTU를 설정합니다. 이를 사용하여 새 논리 네트워크에서 지원하는 MTU를 인터페이스하는 하드웨어에서 지원하는 MTU와 일치시킬 수 있습니다. <b>Custom</b> 이 선택된 경우 텍스트 항목 필드에 숫자 값을 입력합니다.
네트워크 라벨	네트워크에 새 레이블을 지정하거나 이미 호스트 네트워크 인터페이스에 연결된 기존 라벨에서 선택할 수 있습니다. 기존 레이블을 선택하면 논리 네트워크가 해당 레이블이 있는 모든 호스트 네트워크 인터페이스에 자동으로 할당됩니다.
보안 그룹	이 논리 네트워크의 포트에 보안 그룹을 할당할 수 있습니다. <b>disabled</b> 는 보안 그룹 기능을 비활성화합니다. <b>enabled</b> 를 사용하면 기능을 활성화합니다. 포트를 만들고 이 네트워크에 연결하면 포트 보안이 활성화된 상태로 정의됩니다. 즉, 가상 머신에 대한 액세스 권한에는 현재 프로비저닝 중인 보안 그룹이 적용됩니다. <b>Configuration</b> 에서 상속 하면 포트가 모든 네트워크에 대해 정의된 구성 파일에서 동작을 상속할 수 있습니다. 기본적으로 파일은 보안 그룹을 비활성화합니다. 자세한 내용은 9.3.6절. "논리 네트워크 및 포트에 보안 그룹 할당" 을 참조하십시오.

### 9.1.8. 설명되는 논리적 네트워크 클러스터 설정

아래 표에서는 **New Logical Network** 창의 **Cluster** 탭에 대한 설정을 설명합니다.

#### 표 9.2. 새로운 논리 네트워크 설정

필드 이름	설명
클러스터 연결/대상 네트워크 연결/클러스터에서	<p>데이터 센터의 클러스터에서 논리 네트워크를 연결하거나 분리하고 논리 네트워크가 개별 클러스터에 필요한 네트워크인지 여부를 지정할 수 있습니다.</p> <p><b>name</b> - 설정이 적용할 클러스터의 이름입니다. 이 값은 편집할 수 없습니다.</p> <p><b>모두 연결</b> - 논리 네트워크를 데이터 센터의 모든 클러스터로 연결하거나 분리할 수 있습니다. 또는 지정된 클러스터에서 논리 네트워크를 연결하거나 분리하기 위해 각 클러스터의 이름 옆에 있는 <b>Attach</b> (연결) 확인란을 선택하거나 지웁니다.</p> <p><b>필수 모두</b> - 논리 네트워크가 모든 클러스터에서 필수 네트워크인지 여부를 지정할 수 있습니다. 또는 각 클러스터의 이름 옆에 있는 <b>필수</b> 확인란을 선택하거나 지우고 논리 네트워크가 지정된 클러스터에 필요한 네트워크인지 여부를 지정합니다.</p>

**9.1.9. 설명된 논리 네트워크 vNIC 프로필 설정**

아래 표에서는 **New Logical Network** 창의 **vNIC Profiles** 탭에 대한 설정을 설명합니다.

표 9.3. 새로운 논리 네트워크 설정

필드 이름	설명
vNIC 프로필	<p>논리 네트워크에 대해 하나 이상의 vNIC 프로필을 지정할 수 있습니다. vNIC 프로필 옆에 있는 더하기 또는 - 버튼을 클릭하여 논리 네트워크에 vNIC 프로필을 추가하거나 제거할 수 있습니다. 첫 번째 필드는 vNIC 프로필의 이름을 입력하는 것입니다.</p> <p><b>Public</b> - 모든 사용자가 프로필을 사용할 수 있는지 여부를 지정할 수 있습니다.</p> <p><b>QoS</b> - 네트워크 서비스 품질(QoS) 프로필을 vNIC 프로필에 지정할 수 있습니다.</p>

**9.1.10. Manage Networks(네트워크 관리 창)를 사용하여 논리 네트워크의 특정 트래픽 유형 지정**

네트워크 트래픽 흐름을 최적화하기 위해 논리 네트워크의 트래픽 유형을 지정합니다.

논리 네트워크의 트래픽 유형 지정



1. **Compute** → **Clusters** 를 클릭합니다.
2. 클러스터 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **Logical Networks** 탭을 클릭합니다.
4. **Manage Networks** 를 클릭합니다.
5. 적절한 확인란을 선택하고 라디오 버튼을 선택합니다.
6. **OK**를 클릭합니다.



## 참고

외부 공급자가 제공하는 논리적 네트워크는 가상 시스템 네트워크로 사용해야 합니다. 디스플레이 또는 마이그레이션과 같은 특수 클러스터 역할을 할당할 수 없습니다.

### 9.1.11. 네트워크 관리 창에서 설정에 대한 설명

아래 표에서는 **Manage Networks (네트워크 관리)** 창에 대한 설정을 설명합니다.

표 9.4. 네트워크 설정 관리

필드	설명/Action
assign	논리적 네트워크를 클러스터의 모든 호스트에 할당합니다.
필수 항목	연결된 호스트가 제대로 작동하려면 "필수"로 표시된 네트워크는 제대로 작동해야 합니다. 필요한 네트워크가 작동을 중단하면 해당 네트워크와 관련된 모든 호스트가 작동하지 않게 됩니다.
VM 네트워크	"VM 네트워크"라고 표시된 논리 네트워크는 가상 시스템 네트워크와 관련된 네트워크 트래픽을 전달합니다.

필드	설명/Action
디스플레이 네트워크	"ECDHE Network"라고 표시된 논리 네트워크는 SPICE 및 가상 네트워크 컨트롤러와 관련된 네트워크 트래픽을 전달합니다.
Migration Network	"Migration Network"로 표시된 논리 네트워크는 가상 머신 및 스토리지 마이그레이션 트래픽을 전달합니다. 이 네트워크에서 중단이 발생하면 관리 네트워크(기본적으로 <code>ovirtmgmt</code> )가 대신 사용됩니다.

**9.1.12. NIC에서 가상 기능 구성 편집**



참고

이는 **Red Hat Virtualization**에서 **SR-IOV**를 설정하고 구성하는 방법을 보여주는 일련의 주제 중 하나입니다. 자세한 내용은 [SR-IOV 설정 및 구성](#)을 참조하십시오.

**SR-IOV(Single Root I/O Virtualization)**를 사용하면 단일 **PCIe** 끝점을 여러 개의 개별 장치로 사용할 수 있습니다. 이는 **PF**(물리적 기능) 및 **VF**(가상 기능)의 두 가지 **PCIe** 기능을 도입하여 수행됩니다. **PCIe** 카드는 **1~8개의 PF**를 가질 수 있지만 각 **PF**는 장치에 따라 더 많은 **VF**를 지원할 수 있습니다.

각 **NIC**의 **VF** 수를 포함하여 **Red Hat Virtualization Manager**를 통해 **SR-IOV** 가능 네트워크 인터페이스 컨트롤러(**NIC**) 구성을 편집하고 **VF**에 액세스할 수 있는 가상 네트워크를 지정할 수 있습니다.

**VF**가 생성되면 각각 독립 실행형 **NIC**로 취급할 수 있습니다. 여기에는 하나 이상의 논리 네트워크가 할당되고, 결합된 인터페이스를 생성하고, 직접 장치 패스스루를 위해 **vNIC**를 직접 할당하는 작업이 포함됩니다.

**VF**에 직접 연결하려면 **vNIC**에 **passthrough** 속성이 활성화되어 있어야 합니다. **9.2.4절. "vNIC 프로필에서 패스스루 활성화"**을 참조하십시오.

**NIC에서 가상 기능 구성 편집**

1. 컴퓨팅 → 호스트.

2. **SR-IOV** 가능 호스트의 이름을 클릭하여 세부 정보 보기를 엽니다.

3. 네트워크 인터페이스 탭을 클릭합니다.

4. **Setup Host Networks** (호스트 네트워크 설정)를 클릭합니다.

5.



로 표시된 **SR-IOV** 가능 **NIC**를 선택하고 연필 아이콘을 클릭합니다.

6.

가상 기능 수를 편집하려면 **Number of VFs** 설정 드롭다운 버튼을 클릭하고 **VF** 수 텍스트 필드를 편집합니다.



중요

**VF** 수를 변경하면 새 **VF**를 생성하기 전에 네트워크 인터페이스의 이전 **VF**가 모두 삭제됩니다. 여기에는 가상 머신이 직접 연결된 모든 **VF**가 포함됩니다.

7.

**All Networks** 확인란은 기본적으로 선택되어 모든 네트워크가 가상 기능에 액세스할 수 있습니다. 가상 기능에 액세스할 수 있는 가상 네트워크를 지정하려면 특정 네트워크 라디오 버튼을 선택하여 모든 네트워크를 나열합니다. 그런 다음 원하는 네트워크에 대한 확인란을 선택하거나 라벨 텍스트 필드를 사용하여 하나 이상의 네트워크 레이블을 기반으로 네트워크를 자동으로 선택할 수 있습니다.

8.

**OK**를 클릭합니다.

9.

**Setup Host Networks** (호스트 네트워크 설정) 창에서 **OK** 를 클릭합니다.

## 9.2. 가상 네트워크 인터페이스 카드

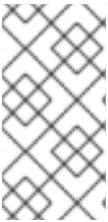
### 9.2.1. vNIC 프로파일 개요

가상 네트워크 인터페이스 카드(**vNIC**) 프로파일은 **Manager**의 개별 가상 네트워크 인터페이스 카드에 적용할 수 있는 설정 컬렉션입니다. **vNIC** 프로파일을 사용하면 네트워크 **QoS** 프로파일을 **vNIC**에 적용하고, 포트 미러링을 활성화 또는 비활성화하고, 사용자 지정 속성을 추가하거나 제거할 수 있습니다. **vNIC** 프

로필은 또한 이러한 프로필을 특정 사용자에게 부여할 수 있는 사용 권한(사용 권한)에 추가 관리 유연성 계층을 제공합니다. 이러한 방식으로 다른 사용자가 지정된 네트워크에서 수신하는 서비스 품질을 제어할 수 있습니다.

### 9.2.2. vNIC 프로파일 생성 또는 편집

가상 네트워크 인터페이스 컨트롤러(vNIC) 프로필을 생성하거나 편집하여 사용자 및 그룹의 네트워크 대역폭을 규제합니다.



참고

포트 미러링을 활성화하거나 비활성화하는 경우, 편집하기 전에 연결된 프로필을 사용하는 모든 가상 머신이 **down** 상태여야 합니다.

#### vNIC 프로파일 생성 또는 편집

1. **Network** → **Networks** 를 클릭합니다.
2. 논리 네트워크의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **vNIC** 프로파일 탭을 클릭합니다.
4. 새로 생성 또는 편집 을 클릭합니다.
5. 프로필의 이름 및 설명을 입력합니다.
6. **QoS** 목록에서 관련 서비스 품질 정책을 선택합니다.
7. 드롭다운 목록에서 네트워크 필터 를 선택하여 가상 머신 내 네트워크 패킷 트래픽을 관리합니다. 네트워크 필터에 대한 자세한 내용은 **Red Hat Enterprise Linux Virtualization** 배포 및 관리 가이드의 **네트워크 필터링 적용**을 참조하십시오.
8. **Passthrough** 확인란을 선택하여 vNIC의 패스스루를 활성화하고 가상 함수의 직접 장치 할

당을 허용합니다. **passthrough** 속성을 활성화하면 **QoS**, 네트워크 필터링, 포트 미러링은 호환되지 않습니다. **passthrough**에 대한 자세한 내용은 **9.2.4절. “vNIC 프로파일에서 패스스루 활성화”**을 참조하십시오.

9.

**Passthrough** 를 선택한 경우 이 프로필을 사용하여 **vNIC**에 대한 마이그레이션을 비활성화하려면 필요한 경우 **Migratable** 확인란을 선택 취소합니다. 이 확인란을 선택한 경우 가상 머신 관리 가이드에서 **SR-IOV-Enabled vNICs**를 사용하여 가상 머신에 대한 추가 사전 요구 사항을 참조하십시오.

10.

**Port Mirroring** 및 **Allow all users to use this Profile** (모든 사용자가 이 프로필을 사용하도록 허용) 확인란을 사용하여 이러한 옵션을 전환합니다.

11.

사용자 지정 속성 목록에서 사용자 지정 속성을 선택합니다. 이 목록에는 기본적으로 **Please select a key...** 가 표시됩니다. + 및 - 버튼을 사용하여 사용자 지정 속성을 추가하거나 제거합니다.

12.


**OK**를 클릭합니다.

이 프로필을 사용자 및 그룹에 적용하여 네트워크 대역폭을 규제합니다. **vNIC** 프로필을 편집한 경우 게스트 운영 체제가 **vNIC** 핫 플러그 및 핫 플러그를 지원하는 경우 가상 머신을 재시작하거나 핫 플러그를 연결 해제한 다음 **vNIC**를 핫 플러그로 연결해야 합니다.

### 9.2.3. VM 인터페이스 프로필 창에서 설정 설명

표 9.5. VM 인터페이스 프로필 창

필드 이름	설명
네트워크	vNIC 프로필을 적용할 사용 가능한 네트워크의 드롭다운 목록입니다.
이름	vNIC 프로필의 이름입니다. 대문자와 소문자, 숫자, 하이픈 및 밑줄 1~50자의 조합이 있는 고유한 이름이어야 합니다.
설명	vNIC 프로필에 대한 설명입니다. 이 필드는 권장되지만 필수는 아닙니다.
QoS	vNIC 프로필에 적용할 사용 가능한 Network Quality of Service 정책의 드롭다운 목록입니다. QoS 정책은 vNIC의 인바운드 및 아웃바운드 네트워크 트래픽을 규제합니다.

필드 이름	설명
<p>네트워크 필터</p>	<p>vNIC 프로필에 적용할 사용 가능한 네트워크 필터의 드롭다운 목록입니다. 네트워크 필터는 가상 머신으로 전송하거나 가상 머신에서 전송할 수 있는 패킷 유형을 필터링하여 네트워크 보안을 향상시킵니다. 기본 필터는 <b>no-mac-spoofing</b> 및 <b>no-arp-mac-spoofing</b> 의 조합인 <b>vds-m-no-mac-spoofing</b> 입니다. libvirt에서 제공하는 네트워크 필터에 대한 자세한 내용은 <i>Red Hat Enterprise Linux Virtualization Deployment and Administration Guide</i> 의 <a href="#">기존 네트워크 필터</a> 섹션을 참조하십시오.</p> <p>가상 머신 VLAN 및 본딩에는 <b>&amp;lt;No Network Filter</b>&gt;를 사용합니다. 신뢰할 수 있는 가상 머신에서 네트워크 필터를 사용하지 않도록 선택하면 성능이 향상됩니다.</p> <div style="display: flex; align-items: flex-start;">  <div data-bbox="1002 801 1433 1081"> <p><b>참고</b></p> <p>Red Hat은 <b>engine-config</b> 도구를 사용하여 <b>EnableMACAntiSpoofingFilterRules</b> 매개변수를 <b>false</b> 로 설정하여 필터 비활성화를 더 이상 지원하지 않습니다. 대신 <b>&amp;lt;No Network Filter</b>&gt; 옵션을 사용합니다.</p> </div> </div>
<p>Passthrough</p>	<p>passthrough 속성을 전환할 수 있는 확인란입니다. 통과를 통해 vNIC가 호스트 NIC의 가상 기능에 직접 연결할 수 있습니다. vNIC 프로필이 가상 머신에 연결된 경우 passthrough 속성을 편집할 수 없습니다.</p> <p>passthrough가 활성화된 경우 vNIC 프로필에서 QoS, 네트워크 필터 및 포트 미러링이 비활성화됩니다.</p>
<p>migratable</p>	<p>이 프로필을 사용하여 vNICs를 마이그레이션할 수 있는지 여부를 전환할 수 있는 확인란을 선택합니다. 마이그레이션은 일반 vNIC 프로필에서 기본적으로 활성화되어 있습니다. 확인란을 선택하면 변경할 수 없습니다. <b>Passthrough</b> 확인란을 선택하면 <b>Migratable</b> 을 사용할 수 있으며 필요한 경우 패스스루 vNIC의 마이그레이션을 비활성화하기 위해 선택 해제할 수 있습니다.</p>
<p>포트 미러링</p>	<p>포트 미러링을 전환하기 위한 확인란입니다. 포트 미러링은 논리 네트워크의 계층 3 네트워크 트래픽을 가상 시스템의 가상 인터페이스에 복사합니다. 이는 기본적으로 선택되지 않습니다. 자세한 내용은 <a href="#">기술 참조의 포트 미러링</a> 을 참조하십시오.</p>
<p>장치 사용자 정의 속성</p>	<p>vNIC 프로필에 적용할 사용 가능한 사용자 지정 속성을 선택하는 드롭다운 메뉴입니다. + 및 - 버튼을 사용하여 각각 속성을 추가하고 제거합니다.</p>

필드 이름	설명
모든 사용자가 이 프로필을 사용하도록 허용	프로필의 가용성을 환경의 모든 사용자로 전환하는 확인란을 선택합니다. 기본적으로 선택됩니다.

#### 9.2.4. vNIC 프로필에서 패스스루 활성화



참고

이는 **Red Hat Virtualization**에서 **SR-IOV**를 설정하고 구성하는 방법을 보여주는 일련의 주제 중 하나입니다. 자세한 내용은 [SR-IOV 설정 및 구성](#)을 참조하십시오.

**vNIC** 프로필의 **passthrough** 속성을 사용하면 **vNIC**를 **SR-IOV** 지원 **NIC**의 **VF**(가상 기능)에 직접 연결할 수 있습니다. 그런 다음 **vNIC**는 소프트웨어 네트워크 가상화를 무시하고 직접 장치 할당을 위해 **VF**에 직접 연결됩니다.

**vNIC** 프로필이 이미 **vNIC**에 연결된 경우 **passthrough** 속성을 활성화할 수 없습니다. 이 절차에서는 이를 방지하기 위해 새 프로필을 생성합니다. **vNIC** 프로필이 패스스루가 활성화된 경우 **QoS**, 네트워크 필터 및 포트 미러링을 동일한 프로필에서 활성화할 수 없습니다.

**SR-IOV**, 직접 장치 할당 및 **Red Hat Virtualization**에서 이러한 구현을 위한 하드웨어 고려 사항에 대한 자세한 내용은 [SR-IOV 구현을 위한 하드웨어 고려 사항](#)을 참조하십시오.

#### **Passthrough** 활성화

1. **Network** → **Networks** 를 클릭합니다.
2. 논리 네트워크의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **vNIC Profiles** 탭을 클릭하여 해당 논리 네트워크의 모든 **vNIC** 프로필을 나열합니다.
4. 새로 생성을 클릭합니다.

5.

프로필의 이름 및 설명을 입력합니다.

6.

**Passthrough** 확인란을 선택합니다.

7.

선택적으로 **Migratable** 확인란을 선택 취소하여 이 프로필을 사용하여 **vNIC**에 대한 마이그레이션을 비활성화합니다. 이 확인란을 선택한 경우 가상 머신 관리 가이드에서 **SR-IOV-Enabled vNICs**를 사용하여 가상 머신에 대한 추가 사전 요구 사항을 참조하십시오.

8.

필요한 경우 사용자 지정 속성 목록에서 사용자 지정 속성을 선택합니다. 이 목록에는 기본적으로 **Please select a key...**가 표시됩니다. + 및 - 버튼을 사용하여 사용자 지정 속성을 추가하거나 제거합니다.

9.

**OK**를 클릭합니다.

이제 **vNIC** 프로필이 **passthrough-enable**이 됩니다. 이 프로필을 사용하여 가상 머신을 **NIC** 또는 **PCI VF**에 직접 연결하려면 논리 네트워크를 **NIC**에 연결하고 **passthrough vNIC** 프로필을 사용하는 원하는 가상 머신에 새 **PCI** 패스스루 **vNIC**를 생성합니다. 이러한 절차에 대한 자세한 내용은 **9.4.2절. “호스트 네트워크 인터페이스 편집 및 호스트에 논리 네트워크 할당”** 및 가상 머신 관리 가이드에서 새 네트워크 인터페이스 추가를 참조하십시오.

### 9.2.5. vNIC 프로필 제거

**vNIC** 프로필을 제거하여 가상화 환경에서 삭제합니다.

#### vNIC 프로필 제거

1.

**Network** → **Networks** 를 클릭합니다.

2.

논리 네트워크의 이름을 클릭하여 세부 정보 보기를 엽니다.

3.

**vNIC Profiles** 탭을 클릭하여 사용 가능한 **vNIC** 프로필을 표시합니다.



4. 하나 이상의 프로필을 선택하고 제거를 클릭합니다.
5. **OK**를 클릭합니다.

### 9.2.6. vNIC 프로필에 보안 그룹 할당



참고

이 기능은 **OpenStack Networking(neutron)**이 외부 네트워크 공급자로 추가된 경우에만 사용할 수 있습니다. 보안 그룹은 **Red Hat Virtualization Manager**를 통해 생성할 수 없습니다. **OpenStack**을 통해 보안 그룹을 생성해야 합니다. 자세한 내용은 **Red Hat OpenStack Platform 사용자 및 ID 관리 가이드의 프로젝트 보안 관리**를 참조하십시오.

**OpenStack Networking** 인스턴스에서 가져온 네트워크의 **vNIC** 프로필에 보안 그룹을 할당하고 **Open vSwitch** 플러그인을 사용할 수 있습니다. 보안 그룹은 네트워크 인터페이스를 통해 인바운드 및 아웃바운드 트래픽을 필터링할 수 있는 엄격하게 적용되는 규칙의 컬렉션입니다. 다음 절차에서는 **vNIC** 프로필에 보안 그룹을 연결하는 방법을 간략하게 설명합니다.



참고

보안 그룹은 **OpenStack Networking** 인스턴스에 등록된 보안 그룹의 **ID**를 사용하여 식별됩니다. **OpenStack Networking**이 설치된 시스템에서 다음 명령을 실행하여 지정된 테넌트의 보안 그룹 **ID**를 찾을 수 있습니다.

```
# neutron security-group-list
```

### vNIC 프로필에 보안 그룹 할당

1. **Network** → **Networks** 를 클릭합니다.
2. 논리 네트워크의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **vNIC** 프로필 탭을 클릭합니다.

4. 새를 클릭하거나 기존 **vNIC** 프로필을 선택하고 편집 을 클릭합니다.
5. 사용자 지정 속성 드롭다운 목록에서 **SecurityGroups** 를 선택합니다. 사용자 지정 속성 드롭다운을 비워 두면 모든 아웃바운드 트래픽과 상호 통신을 허용하지만 기본 보안 그룹 외부에서는 인바운드 트래픽을 거부합니다. 나중에 **SecurityGroups** 속성을 제거하면 적용된 보안 그룹에는 영향을 미치지 않습니다.
6. 텍스트 필드에 **vNIC** 프로필에 연결할 보안 그룹의 **ID**를 입력합니다.
7. **OK**를 클릭합니다.

**vNIC** 프로필에 보안 그룹을 연결했습니다. 해당 프로필이 연결된 논리 네트워크를 통한 모든 트래픽은 해당 보안 그룹에 정의된 규칙에 따라 필터링됩니다.

### 9.2.7. vNIC 프로필에 대한 사용자 권한

사용자를 특정 **vNIC** 프로필에 할당하도록 사용자 권한을 구성합니다. 사용자에게 **VnicProfileUser** 역할을 할당하여 프로필을 사용할 수 있습니다. 해당 프로필에 대한 권한을 제거하여 특정 프로필에서 사용자를 제한합니다.

#### vNIC 프로필에 대한 사용자 권한

1. 네트워크 → **vNIC** 프로필을 클릭합니다.
2. **vNIC** 프로필의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 권한 탭을 클릭하여 프로필에 대한 현재 사용자 권한을 표시합니다.
4. 추가 또는 제거를 클릭하여 **vNIC** 프로필에 대한 사용자 권한을 변경합니다.
5. 사용자에게 권한 추가 창에서 내 그룹을 클릭하여 사용자 그룹을 표시합니다. 이 옵션을 사용하여 그룹의 다른 사용자에게 권한을 부여할 수 있습니다.

**vNIC** 프로필에 대한 사용자 권한이 구성되어 있습니다.

### 9.2.8. UCS Integration을 위한 vNIC 프로필 구성

**Cisco UCS(Unified Computing System)**는 컴퓨팅, 네트워킹 및 스토리지 리소스와 같은 데이터 센터 측면을 관리하는 데 사용됩니다.

**vdsms-hook-vmfex-dev** 후크를 사용하면 가상 머신이 **vNIC** 프로필을 구성하여 **Cisco**의 **UCS** 정의 포트 프로필에 연결할 수 있습니다. **UCS** 정의 포트 프로파일에는 **UCS**에서 가상 인터페이스를 구성하는 데 사용되는 속성 및 설정이 포함되어 있습니다. **vdsms-hook-vmfex-dev** 후크는 기본적으로 **VDSM**을 사용하여 설치됩니다. 자세한 내용은 [부록 A. VDSM 및 후크](#)를 참조하십시오.

**vNIC** 프로필을 사용하는 가상 머신이 생성되면 **Cisco vNIC**를 사용합니다.

**UCS** 통합을 위해 **vNIC** 프로필을 구성하는 절차는 먼저 사용자 지정 장치 속성을 구성하는 절차입니다. 사용자 지정 장치 속성을 구성할 때 포함된 기존 값을 덮어씁니다. 새 사용자 지정 속성과 기존 사용자 지정 속성을 결합하는 경우 키 값을 설정하는 데 사용되는 명령에 모든 사용자 지정 속성을 포함합니다. 여러 사용자 지정 속성은 **Semi-colon**으로 구분됩니다.



참고

**vNIC** 프로필을 구성하기 전에 **UCS** 포트 프로필을 **Cisco UCS**에서 구성해야 합니다.

사용자 지정 장치 속성 구성

1. **Red Hat Virtualization Manager**에서 **vmfex** 사용자 지정 속성을 구성하고 **--cver** 를 사용하여 클러스터 호환성 수준을 설정합니다.

```
# engine-config -s CustomDeviceProperties='{type=interface;prop={vmfex='^[a-zA-Z0-9_-.]{2,32}$}}' --cver=3.6
```

2. **vmfex** 사용자 지정 장치 속성이 추가되었는지 확인합니다.

```
# engine-config -g CustomDeviceProperties
```

3. **ovirt-engine** 서비스를 다시 시작합니다.

```
# systemctl restart ovirt-engine.service
```

구성할 **vNIC** 프로필은 새 또는 기존 논리적 네트워크에 속할 수 있습니다. 새 논리적 네트워크를 구성하는 방법은 **9.1.2절. “데이터 센터 또는 클러스터에서 새 논리 네트워크 생성”**을 참조하십시오.

### UCS Integration을 위한 vNIC 프로필 구성

1. **Network** → **Networks** 를 클릭합니다.
2. 논리 네트워크의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **vNIC** 프로필 탭을 클릭합니다.
4. 새로 생성을 클릭하거나 **vNIC** 프로필을 선택하고 편집 을 클릭합니다.
5. 프로필의 이름 및 설명을 입력합니다.
6. 사용자 지정 속성 목록에서 **vmfex** 사용자 지정 속성을 선택하고 **UCS** 포트 프로필 이름을 입력합니다.
7. **OK**를 클릭합니다.

## 9.3. 외부 공급자 네트워크

### 9.3.1. 외부 공급자에서 네트워크 가져오기

외부 네트워크 공급자(**OpenStack Networking** 또는 **OpenStack Neutron REST API**를 구현하는 타사 공급자)의 네트워크를 사용하려면 공급자를 **Manager**에 등록합니다. 자세한 내용은 네트워크 **프로비저닝**을 위해 **OpenStack Network Service Neutron** 추가 또는 **외부 네트워크 공급자** 추가를 참조하십시오. 그런 다음 다음 절차에 따라 가상 머신에서 네트워크를 사용할 수 있도록 해당 공급자가 제공하는 네트워크를 **Manager**로 가져옵니다.

## 외부 공급자에서 네트워크 가져오기

1. **Network** → **Networks** 를 클릭합니다.
2. 가져오기 를 클릭합니다.
3. 네트워크 공급자 드롭다운 목록에서 외부 공급자를 선택합니다. 해당 공급자가 제공하는 네트워크는 **Provider Networks** 목록에 자동으로 검색 및 나열됩니다.
4. 확인란을 사용하여 **Provider Networks (프로바이더 네트워크)** 목록에서 가져올 네트워크를 선택하고 아래로 화살표를 클릭하여 해당 네트워크를 **Networks to Import** 목록으로 이동합니다.
5. 가져올 네트워크의 이름을 사용자 지정할 수 있습니다. 이름을 사용자 지정하려면 이름 옆에서 네트워크 이름을 클릭하고 텍스트를 변경합니다.
6. 데이터 센터 드롭다운 목록에서 네트워크를 가져올 데이터 센터를 선택합니다.
7. 선택 사항: **Allow All** 확인란을 지워 모든 사용자가 네트워크를 사용할 수 없도록 합니다.
8. 가져오기 를 클릭합니다.

선택한 네트워크는 대상 데이터 센터로 가져오고 가상 머신에 연결할 수 있습니다. 자세한 내용은 가상 머신 관리 가이드에 새 네트워크 인터페이스 추가를 참조하십시오.

### 9.3.2. 외부 공급자 네트워크 사용 제한 사항

다음 제한 사항은 **Red Hat Virtualization** 환경에서 외부 공급자에서 가져온 논리 네트워크 사용에 적용됩니다.

- 외부 공급자가 제공하는 논리적 네트워크는 가상 시스템 네트워크로 사용해야 하며 디스플레이 네트워크로 사용할 수 없습니다.

- 동일한 논리 네트워크를 두 번 이상 가져올 수 있지만 다른 데이터 센터로만 가져올 수 있습니다.
- **Manager**의 외부 공급자가 제공하는 논리적 네트워크는 편집할 수 없습니다. 외부 공급자가 제공하는 논리 네트워크의 세부 정보를 편집하려면 해당 논리 네트워크를 제공하는 외부 공급자에서 직접 논리 네트워크를 편집해야 합니다.
- 포트 미러링은 외부 공급자가 제공하는 논리적 네트워크에 연결된 가상 네트워크 인터페이스 카드에는 사용할 수 없습니다.
- 가상 시스템이 외부 공급자가 제공하는 논리 네트워크를 사용하는 경우, 해당 공급자는 **Manager**에서 삭제할 수 없으며, 논리 네트워크는 가상 시스템에서 계속 사용됩니다.
- 외부 공급자가 제공하는 네트워크는 필수가 아닙니다. 따라서 이러한 논리 네트워크를 가져온 클러스터의 예약은 호스트 선택 중에 논리 네트워크를 고려하지 않습니다. 또한 사용자는 이러한 논리 네트워크를 가져온 클러스터의 호스트에서 논리 네트워크를 사용할 수 있는지 확인해야 합니다.

### 9.3.3. 외부 공급자 논리 네트워크에서 서브넷 구성

외부 공급자가 제공하는 논리 네트워크는 해당 논리 네트워크에 하나 이상의 서브넷이 정의된 경우에만 가상 시스템에 IP 주소를 할당할 수 있습니다. 서브넷이 정의되지 않은 경우 가상 시스템에는 IP 주소가 할당되지 않습니다. 서브넷이 한 개 있는 경우 가상 시스템에는 해당 서브넷의 IP 주소가 할당되고 여러 서브넷이 있는 경우 가상 머신에 사용 가능한 서브넷의 IP 주소가 할당됩니다. 논리 네트워크가 호스팅되는 외부 네트워크 공급자가 제공하는 DHCP 서비스는 이러한 IP 주소를 할당합니다.

**Red Hat Virtualization Manager**는 가져온 논리 네트워크에서 사전 정의된 서브넷을 자동으로 검색하지만 **Manager** 내에서 또는 논리 네트워크에서 서브넷을 추가하거나 제거할 수도 있습니다.

**OVN(Open Virtual Network)(ovirt-provider-ovn)**을 외부 네트워크 공급자로 추가하는 경우 라우터를 통해 여러 서브넷을 서로 연결할 수 있습니다. 이러한 라우터를 관리하려면 **OpenStack Networking API v2.0**을 사용할 수 있습니다. 그러나 **ovirt-provider-ovn**에는 제한이 있습니다. 즉, **OpenStack API**에서 **Source NAT(enable\_snat)**는 구현되지 않습니다.

### 9.3.4. 외부 공급자 논리 네트워크에 서브넷 추가

외부 공급자가 제공하는 논리 네트워크에서 서브넷을 생성합니다.

## 외부 공급자 논리 네트워크에 서브넷 추가

1. **Network** → **Networks** 를 클릭합니다.
2. 논리 네트워크의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **Subnets** 탭을 클릭합니다.
4. 새로 생성을 클릭합니다.
5. 새 서브넷의 이름 및 **CIDR** 을 입력합니다.
6. **IP** 버전 드롭다운 목록에서 **IPv4** 또는 **IPv6** 을 선택합니다.
7. **OK**를 클릭합니다.



참고

**IPv6**의 경우 **Red Hat Virtualization**은 정적 주소 지정만 지원합니다.

## 9.3.5. 외부 공급자 논리 네트워크에서 서브넷 제거

외부 공급자가 제공하는 논리 네트워크에서 서브넷을 제거합니다.

## 외부 공급자 논리 네트워크에서 서브넷 제거

1. **Network** → **Networks** 를 클릭합니다.
2. 논리 네트워크의 이름을 클릭하여 세부 정보 보기를 엽니다.

3. **Subnets** 탭을 클릭합니다.
4. 서브넷을 선택하고 제거를 클릭합니다.
5. **OK**를 클릭합니다.

### 9.3.6. 논리 네트워크 및 포트에 보안 그룹 할당



#### 참고

이 기능은 **OVN(Open Virtual Network)**이 외부 네트워크 공급자(**ovirt-provider-ovn**)로 추가된 경우에만 사용할 수 있습니다. 보안 그룹은 **Red Hat Virtualization Manager**를 통해 생성할 수 없습니다. **OpenStack Networking API v2.0** 또는 **Ansible**을 통해 보안 그룹을 생성해야 합니다.

보안 그룹은 네트워크를 통해 인바운드 및 아웃바운드 트래픽을 필터링할 수 있는 엄격하게 적용되는 규칙의 컬렉션입니다. 보안 그룹을 사용하여 포트 수준에서 트래픽을 필터링할 수도 있습니다.

**Red Hat Virtualization 4.2.7**에서는 보안 그룹이 기본적으로 비활성화되어 있습니다.

#### 논리 네트워크에 보안 그룹 할당

1. **Compute** → **Clusters** 를 클릭합니다.
2. 클러스터 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **Logical Networks** 탭을 클릭합니다.
4. 네트워크 추가를 클릭하고 속성을 정의하여 외부 공급자 드롭다운 목록에서 **ovirt-provider-ovn** 을 선택하도록 합니다. 자세한 내용은 [9.1.2절. “데이터 센터 또는 클러스터에서 새 논리 네트워크 생성”](#)의 내용을 참조하십시오.
5. **Security Group** 드롭다운 목록에서 **Enabled** 를 선택합니다. 자세한 내용은 [9.1.7절. “설명되는 논리적 네트워크 일반 설정”](#)에서 참조하십시오.



6. **OK**를 클릭합니다.
7. **OpenStack Networking API v2.0** 또는 **Ansible** 을 사용하여 보안 그룹을 만듭니다.
8. **OpenStack Networking API v2.0** 또는 **Ansible** 을 사용하여 보안 그룹 규칙을 만듭니다.
9. **OpenStack Networking API v2.0** 또는 **Ansible** 을 사용하여 정의한 보안 그룹으로 포트를 업데이트합니다.
10. **선택 사항:** 포트 수준에서 보안 기능을 활성화할지 여부를 정의합니다. 현재는 **OpenStack Networking API** 를 사용하는 경우에만 사용할 수 있습니다. **port\_security\_enabled** 속성이 설정되지 않은 경우 기본적으로 해당 특성이 속한 네트워크에 지정된 값으로 설정됩니다.

## 9.4. 호스트 및 네트워킹

### 9.4.1. 호스트 기능 새로 고침

호스트에 네트워크 인터페이스 카드를 추가하면 **Manager**에 해당 네트워크 인터페이스 카드를 표시하려면 호스트의 기능을 새로 고쳐야 합니다.

#### 호스트 기능 새로 고침

1. **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.
2. **Management** → **Refresh Capabilities** 클릭합니다.

선택한 호스트의 네트워크 인터페이스 탭에 있는 네트워크 인터페이스 카드 목록이 업데이트됩니다. 이제 새로운 네트워크 인터페이스 카드를 **Manager**에서 사용할 수 있습니다.

### 9.4.2. 호스트 네트워크 인터페이스 편집 및 호스트에 논리 네트워크 할당

물리적 호스트 네트워크 인터페이스의 설정을 변경하고, 관리 네트워크를 한 물리적 호스트 네트워크

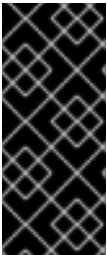
인터페이스에서 다른 인터페이스로 이동하고, 물리적 호스트 네트워크 인터페이스에 논리 네트워크를 할당할 수 있습니다. **bridge** 및 **ethtool** 사용자 지정 속성도 지원됩니다.



주의

**Red Hat Virtualization**에서 호스트의 **IP** 주소를 변경하는 유일한 방법은 호스트를 제거한 다음 다시 추가하는 것입니다.

호스트의 **VLAN** 설정을 변경하려면 **9.4.4절. “호스트의 VLAN 설정 편집”**을 참조하십시오.



중요

외부 공급자가 제공하는 논리적 네트워크를 물리적 호스트 네트워크 인터페이스에 할당할 수 없습니다. 이러한 네트워크는 가상 머신에 필요한 대로 호스트에 동적으로 할당됩니다.



참고

**LLDP(Link Layer Discovery Protocol)** 정보를 제공하도록 스위치가 구성된 경우 실제 네트워크 인터페이스 위에 커서를 올려서 스위치 포트의 현재 구성을 볼 수 있습니다. 이렇게 하면 잘못된 설정을 방지할 수 있습니다. 논리적 네트워크를 할당하기 전에 다음 정보를 확인하는 것이 좋습니다.

- 포트 설명(**TLV 유형 4**) 및 시스템 이름(**TLV 유형 5**)은 어떤 포트를 감지하여 호스트의 인터페이스를 패치하는 데 도움이 됩니다.
- 포트 **VLAN ID**는 태그되지 않은 이더넷 프레임의 스위치 포트에 구성된 기본 **VLAN ID**를 보여줍니다. 스위치 포트에 구성된 모든 **VLAN**은 **VLAN** 이름 및 **VLAN ID** 조합으로 표시됩니다.

호스트 네트워크 인터페이스 편집 및 호스트에 논리 네트워크 할당

1.       컴퓨팅 → 호스트.
2.       호스트 이름을 클릭하여 세부 정보 보기를 엽니다.
3.       네트워크 인터페이스 탭을 클릭합니다.
4.       **Setup Host Networks (호스트 네트워크 설정)**를 클릭합니다.
5.       선택적으로, 호스트 네트워크 인터페이스 위에 커서를 올려 놓으면 스위치에서 제공하는 구성 정보를 볼 수 있습니다.
6.       논리 네트워크를 선택하고 물리적 호스트 네트워크 인터페이스 옆에 할당된 논리 네트워크 영역으로 드래그하여 물리적 호스트 네트워크 인터페이스에 논리 네트워크를 연결합니다.



참고

**NIC**가 두 개 이상의 논리적 네트워크에 연결된 경우 네트워크 중 하나만 **VLAN**이 될 수 있습니다. 다른 모든 논리 네트워크는 고유한 **VLAN**이어야 합니다.

7.       논리 네트워크를 구성합니다.
  - a.       할당된 논리적 네트워크 위에 커서를 올려 놓은 후 연필 아이콘을 클릭하여 관리 네트워크 편집 창을 엽니다.
  - b.       **IPv4** 탭에서 **None, DHCP**, 또는 **static** 으로부터 **Boot Protocol** 을 선택합니다. 정적을 선택한 경우 **IP, Netmask / Routing Prefix**, 게이트웨이 를 입력합니다.



참고

**IPv6**의 경우 정적 **IPv6** 주소 지정만 지원됩니다. 논리 네트워크를 구성하려면 **IPv6** 탭을 선택하고 다음 항목을 만듭니다.

- 부팅 프로토콜 을 정적 으로 설정합니다.
- **Routing Prefix** 의 경우 슬래시와 **10**진수를 사용하여 접두사 길이를 입력합니다. 예: /48
- **IP:** 호스트 네트워크 인터페이스의 전체 **IPv6** 주소입니다. 예: 2001:db8::1:0:0:6
- **gateway:** 소스 라우터의 **IPv6** 주소입니다. 예: 2001:db8::1:0:0:1



참고

호스트의 관리 네트워크 **IP** 주소를 변경하는 경우 구성할 새 **IP** 주소에 대해 [호스트를 다시 설치해야](#) 합니다.

각 논리 네트워크는 관리 네트워크 게이트웨이와 별도의 게이트웨이를 정의할 수 있습니다. 이렇게 하면 논리 네트워크에 도달하는 트래픽이 관리 네트워크에서 사용하는 기본 게이트웨이 대신 논리 네트워크의 게이트웨이를 사용하여 전달됩니다.



중요

클러스터의 모든 호스트를 관리 네트워크에 동일한 **IP** 스택을 사용하도록 설정합니다. **IPv4** 또는 **IPv6**만 사용합니다. 듀얼 스택은 지원되지 않습니다.

c.

**QoS** 탭을 사용하여 기본 호스트 네트워크 품질을 재정의합니다. **QoS** 덮어쓰기를 선택하고 다음 필드에 원하는 값을 입력합니다.

- **weighted DestinationRule:** 동일한 논리 링크에 연결된 다른 네트워크를 기준으로 특정 네트워크를 할당해야 하는 논리 링크 용량의 양을 나타냅니다. 정확한 공유는 해

당 링크의 모든 네트워크 공유 합계에 따라 다릅니다. 기본적으로 이 값은 **1~100** 범위의 숫자입니다.

- 속도 제한 [**Mbps**]: 네트워크에서 사용할 최대 대역폭입니다.
- **committed Rate [Mbps]**: 네트워크에 필요한 최소 대역폭입니다. 요청된 **Rate**는 보장되지 않으며 동일한 논리 링크의 다른 네트워크에서 요청하는 네트워크 인프라 및 요청된 비율에 따라 다릅니다.

d.

네트워크 브리지를 구성하려면 **Custom Properties** 탭을 클릭하고 드롭다운 목록에서 **bridge\_opts** 를 선택합니다. 다음 구문을 사용하여 유효한 키와 값을 입력합니다. **key=value**. 공백 문자로 여러 항목을 구분합니다. 다음 키는 유효하며, 값이 예제로 제공됩니다. 이러한 매개변수에 대한 자세한 내용은 **B.1절. “bridge\_opts 매개변수 설명”** 을 참조하십시오.

```
forward_delay=1500
gc_timer=3765
group_addr=1:80:c2:0:0:0
group_fwd_mask=0x0
hash_elasticity=4
hash_max=512
hello_time=200
hello_timer=70
max_age=2000
multicast_last_member_count=2
multicast_last_member_interval=100
multicast_membership_interval=26000
multicast_querier=0
multicast_querier_interval=25500
multicast_query_interval=13000
multicast_query_response_interval=1000
multicast_query_use_ifaddr=0
multicast_router=1
multicast_snooping=1
multicast_startup_query_count=2
multicast_startup_query_interval=3125
```

e.

이더넷 속성을 구성하려면 **Custom Properties** 탭을 클릭하고 드롭다운 목록에서 **ethtool\_opts** 를 선택합니다. **ethtool**의 명령줄 인수 형식을 사용하여 유효한 값을 입력합니다. 예를 들면 다음과 같습니다.

```
--coalesce em1 rx-usecs 14 sample-interval 3 --offload em2 rx on lro on tso off --change
em1 speed 1000 duplex half
```

이 필드는 와일드카드를 허용할 수 있습니다. 예를 들어 이 모든 네트워크의 인터페이스에 동일한 옵션을 적용하려면 다음을 사용합니다.

```
--coalesce * rx-usecs 14 sample-interval 3
```

**ethtool\_opts** 옵션은 기본적으로 사용할 수 없습니다. 엔진 구성 도구를 사용하여 추가해야 합니다. 자세한 내용은 [B.2절. “Red Hat Virtualization Manager를 Ethtool을 사용하도록 설정하는 방법”](#)를 참조하십시오. **ethtool** 속성에 대한 자세한 내용은 명령줄에서 **man ethtool** 을 입력하여 도움말 페이지를 참조하십시오.

f.

**FCoE(Fibre Channel over Ethernet)**를 구성하려면 **Custom Properties** 탭을 클릭하고 드롭다운 목록에서 **fcoe** 을 선택합니다. 다음 구문을 사용하여 유효한 키와 값을 입력합니다. **key=value**. 최소한 **enable=yes** 가 필요합니다. **dcb=** 및 **auto\_vlan=[yes/no]** 를 추가할 수도 있습니다. 공백 문자로 여러 항목을 구분합니다. **fcoe** 옵션은 기본적으로 사용할 수 없습니다. 엔진 구성 도구를 사용하여 추가해야 합니다. 자세한 내용은 [B.3절. “FCoE를 사용하도록 Red Hat Virtualization Manager 설정 방법”](#)를 참조하십시오.



참고

**FCoE**와 함께 사용하기 위해 별도의 전용 논리적 네트워크를 사용하는 것이 좋습니다.

g.

호스트에서 사용하는 기본 네트워크를 관리 네트워크(**ovirtmgmt**)에서 비 관리 네트워크로 변경하려면 비관리 네트워크의 기본 경로를 구성합니다. 자세한 내용은 [9.1.5절. “기본 경로로 비관리 논리 네트워크 구성”](#)를 참조하십시오.

h.

논리 네트워크 정의가 호스트의 네트워크 구성과 동기화되지 않으면 **Sync network (네트워크 동기화)** 확인란을 선택합니다. 동기화되지 않은 호스트 및 동기화 방법에 대한 자세한 내용은 [9.4.3절. “호스트 네트워크 동기화”](#) 을 참조하십시오.

8.

**Verify connectivity between Host and Engine** 확인란을 선택하여 네트워크 연결을 확인합니다. 이 작업은 호스트가 유지 관리 모드에 있는 경우에만 작동합니다.

9.

**OK**를 클릭합니다.



참고

호스트의 모든 네트워크 인터페이스 카드가 표시되지 않는 경우 **Management** → **Refresh Capabilities** 를 클릭하여 해당 호스트에 사용 가능한 네트워크 인터페이스 카드 목록을 업데이트합니다.

### 9.4.3. 호스트 네트워크 동기화

**Manager**는 호스트의 인터페이스 정의가 **Manager**에 저장된 정의와 다른 경우 네트워크 인터페이스를 동기화 하지 않습니다. 비활성 네트워크는 호스트의 네트워크 인터페이스 탭에 있는 비동기(**Out-of-sync**) 아이콘



과 **Setup Host Networks (호스트 네트워크 설정)** 창에서 이 아이콘



로 표시됩니다.

호스트의 네트워크가 동기화되지 않은 경우 **Setup Host Networks (호스트 네트워크 설정)** 창의 동기화되지 않은 네트워크에서 수행할 수 있는 유일한 활동만 네트워크 인터페이스에서 논리 네트워크를 분리하거나 네트워크를 동기화합니다.

호스트가 동기화되지 않는 방법 이해

다음과 같은 경우 호스트가 동기화되지 않습니다.

- **Edit Logical Networks** 창을 사용하지 않고 호스트에서 구성을 변경합니다. 예를 들면 다음과 같습니다.
  - 물리적 호스트의 **VLAN** 식별자 변경.
  - 물리적 호스트의 사용자 지정 **MTU** 변경.
- 네트워크 이름이 동일하지만 값/**parameters**가 다른 데이터 센터로 호스트를 이동합니다.
- 호스트에서 브릿지를 수동으로 제거하여 네트워크의 **VM** 네트워크 속성을 변경합니다.

호스트가 **Unsynchronized**되지 않도록 방지

다음 모범 사례를 따라 호스트가 동기화되지 않도록 합니다.

1. 호스트에서 로컬로 변경하지 않고 관리 포털을 사용합니다.

2.

**9.4.4절. “호스트의 VLAN 설정 편집”**의 지침에 따라 **VLAN** 설정을 편집합니다.

#### 호스트 동기화

호스트의 네트워크 인터페이스 정의를 동기화하려면 **Manager**의 정의를 사용하여 호스트에 적용해야 합니다. 필요한 정의가 아닌 경우, 호스트를 동기화한 후 관리 포털에서 해당 정의를 업데이트합니다. 세 가지 수준에서 호스트의 네트워크를 동기화할 수 있습니다.

- 논리 네트워크당
- 호스트당
- 클러스터당

#### 논리 네트워크 수준에서 호스트 네트워크 동기화

1. 컴퓨팅 → 호스트.
2. 호스트 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 네트워크 인터페이스 탭을 클릭합니다.
4. **Setup Host Networks** (호스트 네트워크 설정)를 클릭합니다.
5. 동기화되지 않은 네트워크 위에 커서를 올려 놓은 후 연필 아이콘을 클릭하여 **Edit Network** 창을 엽니다.
6. **Sync network** 확인란을 선택합니다.
7. 확인을 클릭하여 네트워크 변경 사항을 저장합니다.



8.

**OK** 를 클릭하여 **Setup Host Networks** (호스트 네트워크 설정) 창을 닫습니다.

호스트 수준에서 호스트의 네트워크 동기화

•

호스트의 네트워크 인터페이스 탭에서 **Sync All Networks** 버튼을 클릭하여 모든 호스트의 동기화되지 않은 네트워크 인터페이스를 동기화합니다.

클러스터 수준에서 호스트의 네트워크 동기화

•

클러스터의 **Logical Networks** 탭에서 **Sync All Networks** 버튼을 클릭하여 전체 클러스터에 대해 동기화되지 않은 논리 네트워크 정의를 모두 동기화합니다.



참고

**REST API**를 통해 호스트 네트워크를 동기화할 수도 있습니다. **REST API** 가이드의 [syncallnetworks](#) 를 참조하십시오.

#### 9.4.4. 호스트의 VLAN 설정 편집

호스트의 **VLAN** 설정을 변경하려면 **Manager**에서 호스트를 제거하고 다시 **Manager**에 추가해야 합니다.

네트워킹을 동기화하려면 다음을 수행하십시오.

1.

호스트를 유지 관리 모드로 설정합니다.

2.

호스트에서 관리 네트워크를 수동으로 제거합니다. 그러면 새 **VLAN**을 통해 호스트에 연결할 수 있습니다.

3.

클러스터에 호스트를 추가합니다. 관리 네트워크에 직접 연결되지 않은 가상 시스템은 호스트 간에 안전하게 마이그레이션할 수 있습니다.

관리 네트워크의 **VLAN ID**가 변경되면 다음 경고 메시지가 표시됩니다.

Changing certain properties (e.g. VLAN, MTU) of the management network could lead to loss of connectivity to hosts in the data center, if its underlying network infrastructure isn't configured to accommodate the changes. Are you sure you want to proceed?

진행하면 데이터 센터의 모든 호스트가 **Manager**에 대한 연결이 끊어지며 호스트를 새 관리 네트워크로 마이그레이션할 수 없습니다. 관리 네트워크는 "out-of-sync"로 보고됩니다.



중요

관리 네트워크의 **VLAN ID**를 변경하는 경우 **호스트를 다시** 설치하여 새 **VLAN ID**를 적용해야 합니다.

### 9.4.5. 논리 네트워크를 사용하여 단일 네트워크 인터페이스에 여러 VLAN 추가

단일 네트워크 인터페이스에 여러 개의 **VLAN**을 추가하여 하나의 호스트에서 트래픽을 분리할 수 있습니다.



중요

논리 네트워크 또는 논리 네트워크 편집 창에서 선택한 모든 **VLAN** 태그 지정 활성화 확인란을 사용하여 두 개 이상의 논리 네트워크를 만들어야 합니다.

논리 네트워크를 사용하여 네트워크 인터페이스에 여러 **VLAN** 추가

1.       컴퓨팅 → 호스트.
2.       호스트 이름을 클릭하여 세부 정보 보기를 엽니다.
3.       네트워크 인터페이스 탭을 클릭합니다.
4.       **Setup Host Networks (호스트 네트워크 설정)**를 클릭합니다.
5.       **VLAN** 태그된 논리 네트워크를 물리적 네트워크 인터페이스 옆에 있는 할당된 논리 네트워크 영역으로 드래그합니다. 물리적 네트워크 인터페이스에는 **VLAN** 태그 지정으로 인해 여러 개

의 논리 네트워크가 할당될 수 있습니다.

6. 논리 네트워크를 편집합니다.
  - a. 할당된 논리적 네트워크 위에 커서를 올려 놓은 후 연필 아이콘을 클릭합니다.
  - b. 논리 네트워크 정의가 호스트의 네트워크 구성과 동기화되지 않으면 **Sync network (네트워크 동기화)** 확인란을 선택합니다.
  - c. 부팅 프로토콜 을 선택합니다.
    - 없음
    - **DHCP**
    - 고정
  - d. **IP 및 SubnetECDHE**를 제공합니다.
  - e. **OK**를 클릭합니다.
7. **Verify connectivity between Host and Engine** 확인란을 선택하여 네트워크 확인란을 실행합니다. 이는 호스트가 유지 관리 모드에 있는 경우에만 작동합니다.
8. **OK**를 클릭합니다.

클러스터의 각 호스트에서 **NIC**를 편집하여 클러스터의 각 호스트에 논리 네트워크를 추가합니다. 이 작업이 완료되면 네트워크가 작동합니다.

이 프로세스를 여러 번 반복하여 각 호스트에서 동일한 네트워크 인터페이스를 선택하여 단일 네트워크 인터페이스에 서로 다른 **VLAN** 태그가 있는 논리 네트워크를 추가할 수 있습니다.

#### 9.4.6. 호스트 네트워크에 추가 IPv4 주소 할당

**ovirtmgmt** 관리 네트워크와 같은 호스트 네트워크는 처음 설정 시 하나의 **IP** 주소만 사용하여 생성됩니다. 즉, **NIC** 구성 파일(예: `/etc/sysconfig/network-scripts/ifcfg-eth01`)이 여러 **IP** 주소로 구성된 경우 처음 나열된 **IP** 주소만 호스트 네트워크에 할당됩니다. 스토리지에 연결하거나 동일한 **NIC**를 사용하여 별도의 프라이빗 서브넷의 서버에 연결하는 경우 추가 **IP** 주소가 필요할 수 있습니다.

**vds-hook-extra-ipv4-addr** 후크를 사용하면 호스트 네트워크에 대한 추가 **IPv4** 주소를 구성할 수 있습니다. 후크에 대한 자세한 내용은 [부록 A. VDSM 및 후크](#) 을 참조하십시오.

다음 절차에서는 추가 **IP** 주소를 구성할 각 호스트에서 호스트별 작업을 수행해야 합니다.

##### 호스트 네트워크에 추가 IPv4 주소 할당

1. 추가 **IPv4** 주소를 구성할 호스트에서 **VDSM** 후크 패키지를 설치합니다. 패키지는 **Red Hat Virtualization** 호스트에서 기본적으로 사용할 수 있지만 **Red Hat Enterprise Linux** 호스트에 설치해야 합니다.
 

```
# yum install vds-hook-extra-ipv4-addr
```
2. **Manager**에서 다음 명령을 실행하여 키를 추가합니다.
 

```
# engine-config -s 'UserDefinedNetworkCustomProperties=ipv4_addr=.'
```
3. **ovirt-engine** 서비스를 다시 시작합니다.
 

```
# systemctl restart ovirt-engine.service
```
4. 관리 포털에서 **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭합니다.
5. 호스트 이름을 클릭하여 세부 정보 보기를 엽니다.
6. 네트워크 인터페이스 탭을 클릭하고 **Setup Host Networks** 를 클릭합니다.

7. 할당된 논리 네트워크 위로 커서를 마우스로 이동하고 연필 아이콘을 클릭하여 호스트 네트워크 인터페이스를 편집합니다.
8. **Custom Properties** 드롭다운 목록에서 **ipv4\_addr** 을 선택하고 추가 IP 주소 및 접두사(예: **5.5.5.5/24**)를 추가합니다. 여러 IP 주소는 쉼표로 구분해야 합니다.
9. **OK** 를 클릭하여 **Edit Network** 창을 닫습니다.
10. **OK** 를 클릭하여 **Setup Host Networks (호스트 네트워크 설정)** 창을 닫습니다.

추가 IP 주소는 **Manager**에 표시되지 않지만 호스트에서 **ip addr show** 명령을 실행하여 추가되었는지 확인할 수 있습니다.

#### 9.4.7. 호스트 네트워크 인터페이스에 네트워크 라벨 추가

네트워크 레이블을 사용하면 호스트 네트워크 인터페이스에 논리 네트워크를 할당하는 것과 관련된 관리 워크로드를 크게 단순화할 수 있습니다. 역할 네트워크(예: 마이그레이션 네트워크 또는 디스플레이 네트워크)에 레이블을 설정하면 모든 호스트에 해당 네트워크를 배포하게 됩니다. 이러한 네트워크 추가는 **DHCP**를 사용하여 수행됩니다. 이 대량 배포 방법은 많은 고정 IP 주소 입력 작업의 확장 불가능한 특성으로 인해 정적 주소 입력 방법을 통해 선택되었습니다.

호스트 네트워크 인터페이스에 레이블을 추가하는 방법은 다음 두 가지가 있습니다.

- 관리 포털에서 수동으로
- **LLDP** 라벨러 서비스와 함께 자동으로

관리 포털에 네트워크 라벨 추가

1. 컴퓨팅 → 호스트.
2. 호스트 이름을 클릭하여 세부 정보 보기를 엽니다.

3. 네트워크 인터페이스 탭을 클릭합니다.
4. **Setup Host Networks (호스트 네트워크 설정)**를 클릭합니다.
5. 레이블을 클릭하고 **[New Label]** 을 마우스 오른쪽 버튼으로 클릭합니다. 레이블을 지정할 물리적 네트워크 인터페이스를 선택합니다.
6. 라벨 텍스트 필드에 네트워크 레이블의 이름을 입력합니다.
7. **OK**를 클릭합니다.

#### LLDP 라벨러 서비스를 사용하여 네트워크 라벨 추가

**LLDP 라벨러** 서비스를 사용하여 구성된 클러스터 목록에서 호스트 네트워크 인터페이스에 레이블을 할당하는 프로세스를 자동화할 수 있습니다.

기본적으로 **LLDP** 라벨러는 시간별 서비스로 실행됩니다. 이 옵션은 하드웨어를 변경하는 경우(예: **NIC**, 스위치 또는 비지니스) 또는 스위치 구성을 변경할 때 유용합니다.

#### 사전 요구 사항

- 인터페이스는 **Juniper** 스위치에 연결되어 있어야 합니다.
- **Juniper** 스위치는 **LLDP**를 사용하여 **Port VLAN** 을 제공하도록 구성해야 합니다.

#### 절차

1. **/etc/ovirt-lldp-labeler/conf.d/ovirt-lldp-credentials.conf** 에서 사용자 이름과 암호 를 구성합니다.
  - 사용자 이름 - 관리자 관리자의 사용자 이름입니다. 기본값은 **admin@internal** 입니다.

- **password - Manager** 관리자의 암호입니다. 기본값은 **123456** 입니다.
- 2.
- etc/ovirt-lldp-labeler/conf.d/ovirt-lldp-credentials.conf** 에서 다음 값을 업데이트하여 **LLDP** 라벨러 서비스를 구성합니다.
- **Clusters** - 서비스를 실행해야 하는 클러스터의 쉼표로 구분된 목록입니다. 와일드카드가 지원됩니다. 예를 들어 **Cluster\*** 는 **Cluster** 로 시작하는 모든 클러스터에서 실행할 **LLDP Labeler**를 정의합니다. 데이터 센터의 모든 클러스터에서 서비스를 실행하려면 \* 를 입력합니다. 기본값은 **Def\*** 입니다.
  - **api\_url - Manager API**의 전체 **URL** 기본값은 **https://Manager\_FQDN/ovirt-engine/api**입니다.
  - **ca\_file** - 사용자 정의 **CA** 인증서 파일의 경로입니다. 사용자 정의 인증서를 사용하지 않는 경우 이 값을 비워 둡니다. 기본값은 비어 있습니다.
  - **auto\_bonding - LLDP** 라벨러의 본딩 기능을 활성화합니다. 기본값은 **true**입니다.
  - **auto\_labeling - LLDP** 라벨 작성 기능을 활성화합니다. 기본값은 **true**입니다.
- 3.
- 필요한 경우 **etc/ovirt-lldp-labeler/conf.d/ovirt-lldp-labeler.timer** 에서 **OnUnitActiveSec** 값을 변경하여 다른 시간 간격으로 실행되도록 서비스를 구성할 수 있습니다. 기본값은 **1h** 입니다.
- 4.
- 다음 명령을 입력하여 현재 및 부팅 시 서비스를 구성하십시오.

```
# systemctl enable --now ovirt-lldp-labeler
```

서비스를 수동으로 호출하려면 다음 명령을 입력합니다.

```
# /usr/bin/python /usr/share/ovirt-lldp-labeler/ovirt_lldp_labeler_cli.py
```

호스트 네트워크 인터페이스에 네트워크 레이블을 추가했습니다. 레이블이 동일한 새로 생성된 논리 네트워크는 해당 레이블이 있는 모든 호스트 네트워크 인터페이스에 자동으로 할당됩니다. 논리 네트워크

에서 레이블을 제거하면 해당 레이블이 있는 모든 호스트 네트워크 인터페이스에서 해당 논리 네트워크가 자동으로 제거됩니다.

#### 9.4.8. 호스트의 FQDN 변경

다음 절차에 따라 호스트의 정규화된 도메인 이름을 변경합니다.

##### 호스트의 FQDN 업데이트

1. 가상 시스템이 다른 호스트로 실시간 마이그레이션되도록 호스트를 유지 관리 모드에 배치합니다. 자세한 내용은 [10.5.15절. “호스트를 유지 관리 모드로 전환”](#)를 참조하십시오. 또는 모든 가상 시스템을 다른 호스트로 수동으로 종료하거나 마이그레이션합니다. 자세한 내용은 [가상 머신 관리 가이드](#)에서 가상 머신 수동 마이그레이션을 참조하십시오.
2. 제거를 클릭하고 **OK** 를 클릭하여 관리 포털에서 호스트를 제거합니다.
3. **hostnamectl** 툴을 사용하여 호스트 이름을 업데이트합니다. 자세한 내용은 [Red Hat Enterprise Linux 7 네트워킹 가이드의 호스트 이름 구성](#) 을 참조하십시오.
 

```
# hostnamectl set-hostname NEW_FQDN
```
4. 호스트를 재부팅합니다.
5. **Manager**로 호스트를 다시 등록합니다. 자세한 내용은 [10.5.1절. “Red Hat Virtualization Manager에 표준 호스트 추가”](#)를 참조하십시오.

#### 9.4.9. IPv6 네트워킹 지원

**Red Hat Virtualization**은 대부분의 컨텍스트에서 정적 **IPv6** 네트워킹을 지원합니다.



##### 참고

**Red Hat Virtualization**에서는 **Manager**를 실행하는 컴퓨터 또는 가상 머신에서 **IPv6**를 계속 활성화해야 합니다(관리자 시스템라고도 함). 시스템이 사용하지 않는 경우에도 **Manager** 시스템에서 **IPv6**를 비활성화하지 마십시오.



**IPv6 제한 사항**

- 정적 **IPv6** 주소만 지원됩니다. **DHCP** 또는 **Stateless Address Autoconfiguration** 을 사용한 동적 **IPv6** 주소는 지원되지 않습니다.
- 듀얼 스택 주소 지정, **IPv4** 및 **IPv6**은 지원되지 않습니다.
- **OVN** 네트워킹은 **IPv4** 또는 **IPv6**에서만 사용할 수 있습니다.
- 클러스터를 **IPv4**에서 **IPv6**로 전환할 수 없습니다.
- 호스트당 단일 게이트웨이만 **IPv6**에 대해 설정할 수 있습니다.
- 두 네트워크 모두 단일 게이트웨이(동일한 서브넷에 있음)를 공유하는 경우 기본 경로 역할을 관리 네트워크(**ovirtmgmt**)에서 다른 논리 네트워크로 이동할 수 있습니다. 호스트와 **Manager**에는 동일한 **IPv6** 게이트웨이가 있어야 합니다. 호스트와 **Manager**가 동일한 서브넷에 있지 않은 경우 **IPv6** 게이트웨이가 제거되었기 때문에 **Manager**의 연결이 끊어질 수 있습니다.
- **IPv6** 주소가 지정된 **gruster** 서버가 있는 **glusterfs** 스토리지 도메인 사용은 지원되지 않습니다.

**9.4.10. SR-IOV 설정 및 구성**

이 항목에는 **SR-IOV** 설정 및 구성 단계가 요약되어 있으며 각 단계를 자세히 다루는 항목에 대한 링크가 포함되어 있습니다.

**9.4.10.1. 사전 요구 사항**

**SR-IOV** 구현을 위한 하드웨어 고려사항에 따라 하드웨어 설정

**9.4.10.2. SR-IOV 설정 및 구성**

**SR-IOV**를 설정하고 구성하려면 다음 작업을 완료합니다.

1. **PCI** 패스스루를 위한 호스트 구성
2. **NIC**에서 가상 기능 구성 편집.
3. **vNIC** 프로필에서 패스스루 활성화.
4. **SR-IOV-Enabled vNICs**로 가상 머신을 구성하여 마이그레이션 중 네트워크 중단을 줄입니다.

참고

- **'passthrough' vNIC**의 수는 호스트에서 사용 가능한 **VF**(가상 기능) 수에 따라 달라집니다. 예를 들어 세 개의 **SR-IOV** 카드(**vNIC**)를 사용하여 **VM**(가상 머신)을 실행하려면 호스트에 **3**개 이상의 **VF**가 활성화되어 있어야 합니다.
- 핫플러그 및 **unplug**가 지원됩니다.
- **RHV** 버전 **4.1** 이후의 실시간 마이그레이션이 지원됩니다.
- **VM**을 마이그레이션하려면 대상 호스트에 **VM**을 수신할 수 있는 충분한 **VF**가 있어야 합니다. 마이그레이션 중에 **VM**은 소스 호스트에서 여러 **VF**를 릴리스하고 대상 호스트에서 동일한 **VF** 수를 차지합니다.
- 호스트에서 장치, 링크 또는 다른 인터페이스와 마찬가지로 **ifcae**가 표시됩니다. 해당 장치는 **VM**에 연결되면 사라졌으며 릴리스될 때 다시 나타납니다.
- **SR-IOV** 기능의 **VM**에 직접 호스트 장치를 연결하지 마십시오.
- **VF**를 여러 **VLAN**이 있는 트렁크 포트에 사용하고 게스트 내에서 **VLAN**을 구성하려면 가상 머신 내의 **SR-IOV VF** 인터페이스에서 **VLAN**을 구성할 수 없습니다.

다음은 인터페이스에 대한 **libvirt XML**의 예입니다.

```

----
<interface type='hostdev'>
  <mac address='00:1a:yy:xx:vv:xx'/>
  <driver name='vfio'/>
  <source>
    <address type='pci' domain='0x0000' bus='0x05' slot='0x10' function='0x0'/>
  </source>
  <alias name='ua-18400536-5688-4477-8471-be720e9efc68'/>
  <address type='pci' domain='0x0000' bus='0x00' slot='0x08' function='0x0'/>
</interface>
----

```

### 문제 해결

다음 예제에서는 인터페이스에 연결된 **VF**에 대한 진단 정보를 가져오는 방법을 보여줍니다.

```
# ip -s link show dev enp5s0f0
```

```

1: enp5s0f0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9000 qdisc mq state UP mode
DEFAULT qlen 1000
  link/ether 86:e2:ba:c2:50:f0 brd ff:ff:ff:ff:ff:ff
  RX: bytes  packets  errors  dropped  overrun  mcast
  30931671 218401  0      0      0      19165434
  TX: bytes  packets  errors  dropped  carrier  collsns
  997136   13661  0      0      0      0
  vf 0 MAC 02:00:00:00:00:01, spoof checking on, link-state auto, trust off, query_rss off
  vf 1 MAC 00:1a:4b:16:01:5e, spoof checking on, link-state auto, trust off, query_rss off
  vf 2 MAC 02:00:00:00:00:01, spoof checking on, link-state auto, trust off, query_rss off

```

#### 9.4.10.3. 추가 리소스

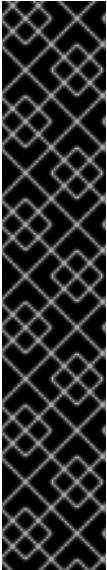
- **RHV VM**에 대한 **SR-IOV** 패스스루를 구성하는 방법은 무엇입니까?
- **RHV**에서 **SR-IOV VF(Virtual Function)**로 본딩을 구성하는 방법
- **RHV**의 가상 머신에 전용 가상 **NIC**를 할당하기 위해 호스트 장치 패스스루 및 **SR-IOV**를 활성화하는 방법

#### 9.5. 네트워크 본딩

네트워크 본딩은 다음과 같은 이점과 함께 여러 **NIC**를 본딩 장치에 결합합니다.

- 본딩된 **NIC**의 전송 속도가 단일 **NIC**의 전송 속도보다 큽니다.
- 네트워크 본딩은 모든 **NIC**가 실패하지 않는 한 본딩 장치가 실패하지 않기 때문에 내결함성을 제공합니다.

동일한 **make** 및 **model**의 **NIC**를 사용하면 동일한 본딩 옵션 및 모드를 지원할 수 있습니다.



중요

**Red Hat Virtualization**의 기본 본딩 모드 (**Mode 4**) 동적 링크 집계에는 **802.3ad**를 지원하는 스위치가 필요합니다.

본딩의 논리적 네트워크는 호환 가능해야 합니다. 본딩은 비**VLAN** 논리 네트워크를 하나만 지원할 수 있습니다. 나머지 논리 네트워크에는 고유한 **VLAN ID**가 있어야 합니다.

스위치 포트에 본딩을 활성화해야 합니다. 특정 지침은 스위치 공급 업체에서 제공하는 설명서를 참조하십시오.

다음 방법 중 하나를 사용하여 네트워크 본딩 장치를 생성할 수 있습니다.

- 특정 호스트에 대한 수동으로 [관리 포털에서](#)
- 클러스터 또는 데이터 센터의 모든 호스트에 대한 **LLDP 라벨러**를 자동으로 사용

환경에서 **iSCSI** 스토리지를 사용하고 중복성을 구현하려는 경우 **iSCSI 다중 경로 구성** 지침을 따르십시오.

### 9.5.1. 관리 포털에서 본딩 장치 생성

관리 포털에서 특정 호스트에 본딩 장치를 생성할 수 있습니다. 본딩 장치는 **VLAN** 태그 지정 및 태그되지 않은 트래픽을 모두 전송할 수 있습니다.

#### 절차

1.       컴퓨팅 → 호스트.
2.       호스트 이름을 클릭하여 세부 정보 보기를 엽니다.
3.       네트워크 인터페이스 탭을 클릭하여 호스트에 연결된 물리적 네트워크 인터페이스를 나열합니다.
4.       **Setup Host Networks (호스트 네트워크 설정)**를 클릭합니다.
5.       스위치 구성을 확인합니다. 스위치가**LLDP(Link Layer Discovery Protocol)** 정보를 제공하도록 구성된 경우 실제 **NIC** 위에 커서를 이동하여 스위치 포트의 집계 구성을 확인합니다.
6.       **NIC**를 다른 **NIC**로 드래그 앤 드롭하거나 본딩에 놓습니다.



참고

두 개의 **NIC**가 새로운 본딩을 형성합니다. **NIC**와 본딩은 **NIC**를 기존 본딩에 추가합니다.

논리 네트워크가 **호환되지 않는** 경우 본딩 작업이 차단됩니다.

7.       드롭다운 메뉴에서 본딩 이름 및 연결 모드를 선택합니다. 자세한 내용은 **9.5.3절. “본딩 모드”**을 참조하십시오.

사용자 정의 본딩 모드를 선택하는 경우 다음 예와 같이 텍스트 필드에 본딩 옵션을 입력할 수 있습니다.

- 환경에서 **ethtool**의 링크 상태를 보고하지 않는 경우 **mode=1 arp\_interval=1 arp\_ip\_target=192.168.0.2**를 입력하여 **ARP** 모니터링을 설정할 수 있습니다.
- **mode=1 primary=eth0**을 입력하여 처리량이 높은 **NIC**를 기본 인터페이스로 지정할 수 있습니다.

본딩 옵션 및 해당 설명의 포괄적인 목록은 [Kernel.org](http://Kernel.org)의 [Linux Ethernet Bonding Driver HOWTO](#) 를 참조하십시오.

8. **OK**를 클릭합니다.

9. 논리 네트워크를 새 본딩에 연결하고 구성합니다. 자세한 내용은 [9.4.2절. “호스트 네트워크 인터페이스 편집 및 호스트에 논리 네트워크 할당”](#) 를 참조하십시오.



참고

논리 네트워크를 본딩의 개별 **NIC**에 직접 연결할 수 없습니다.

10. 선택적으로 호스트가 유지 관리 모드에 있는 경우 호스트와 엔진 간 연결 확인을 선택할 수 있습니다.

11. **OK**를 클릭합니다.

### 9.5.2. LLDP 라벨러 서비스를 사용하여 본딩 장치 생성

**LLDP** 라벨러 서비스를 사용하면 하나 이상의 클러스터 또는 전체 데이터 센터에 있는 모든 호스트에 대해 통합되지 않은 **NIC**를 사용하여 자동으로 본딩 장치를 생성할 수 있습니다. 본딩 모드는 **(Mode 4) 동적 링크 집계 (802.3ad)** 입니다.

호환되지 않는 논리 네트워크가 있는 **NIC**는 결합할 수 없습니다.

기본적으로 **LLDP** 라벨러는 시간별 서비스로 실행됩니다. 이 옵션은 하드웨어를 변경하는 경우(예: **NIC**, 스위치 또는 비지니스) 또는 스위치 구성을 변경할 때 유용합니다.

#### 사전 요구 사항

- 인터페이스는 **Juniper** 스위치에 연결되어 있어야 합니다.
- **Juniper** 스위치는 **LLDP**를 사용하여 **LACP(Link Aggregation Control Protocol)**에 대해 구성해야 합니다.

## 절차

1. `/etc/ovirt-lldp-labeler/conf.d/ovirt-lldp-credentials.conf` 에서 사용자 이름과 암호 를 구성합니다.
  - 사용자 이름 - 관리자 관리자의 사용자 이름입니다. 기본값은 **admin@internal** 입니다.
  - **password - Manager** 관리자의 암호입니다. 기본값은 **123456** 입니다.
2. `etc/ovirt-lldp-labeler/conf.d/ovirt-lldp-credentials.conf` 에서 다음 값을 업데이트하여 **LLDP** 라벨러 서비스를 구성합니다.
  - **Clusters** - 서비스를 실행해야 하는 클러스터의 쉼표로 구분된 목록입니다. 와일드카드 가 지원됩니다. 예를 들어 **Cluster\*** 는 **Cluster** 로 시작하는 모든 클러스터에서 실행할 **LLDP Labeler**를 정의합니다. 데이터 센터의 모든 클러스터에서 서비스를 실행하려면 \* 를 입력합니다. 기본값은 **Def\*** 입니다.
  - **api\_url - Manager API**의 전체 **URL** 기본값은 **https://Manager\_FQDN/ovirt-engine/api**입니다.
  - **ca\_file** - 사용자 정의 **CA** 인증서 파일의 경로입니다. 사용자 정의 인증서를 사용하지 않는 경우 이 값을 비워 둡니다. 기본값은 비어 있습니다.
  - **auto\_bonding - LLDP** 라벨러의 본딩 기능을 활성화합니다. 기본값은 **true**입니다.
  - **auto\_labeling - LLDP** 라벨 작성 기능을 활성화합니다. 기본값은 **true**입니다.
3. 필요한 경우 `etc/ovirt-lldp-labeler/conf.d/ovirt-lldp-labeler.timer` 에서 **OnUnitActiveSec** 값을 변경하여 다른 시간 간격으로 실행되도록 서비스를 구성할 수 있습니다. 기본값은 **1h** 입니다.
4. 다음 명령을 입력하여 현재 및 부팅 시 서비스를 구성하십시오.

```
# systemctl enable --now ovirt-lldp-labeler
```

서비스를 수동으로 호출하려면 다음 명령을 입력합니다.

```
# /usr/bin/python /usr/share/ovirt-lldp-labeler/ovirt_lldp_labeler_cli.py
```

5.

논리 네트워크를 새 본딩에 연결하고 구성합니다. 자세한 내용은 [9.4.2절. “호스트 네트워크 인터페이스 편집 및 호스트에 논리 네트워크 할당”](#) 를 참조하십시오.



참고

논리 네트워크를 본딩의 개별 **NIC**에 직접 연결할 수 없습니다.

### 9.5.3. 본딩 모드

패킷 분산 알고리즘은 본딩 모드에 따라 결정됩니다. (자세한 내용은 [Linux 이더넷 본딩 드라이버 HOWTO](#) 를 참조하십시오.) **Red Hat Virtualization**의 기본 본딩 모드는 **(Mode 4) 동적 링크 집계 (802.3ad)** 입니다.

**Red Hat Virtualization**은 가상 머신(**bridged**) 네트워크에서 사용할 수 있으므로 다음과 같은 본딩 모드를 지원합니다.

#### (mode 1) Active-Backup

하나의 **NIC**가 활성화되어 있습니다. 활성 **NIC**가 실패하면 백업 **NIC** 중 하나가 본딩에서 유일한 활성 **NIC**로 교체됩니다. 이 본딩의 **MAC** 주소는 네트워크 어댑터 포트에서만 볼 수 있습니다. 이렇게 하면 본딩의 **MAC** 주소가 변경될 경우 새 활성 **NIC**의 **MAC** 주소를 반영하여 **MAC** 주소 혼동을 방지할 수 있습니다.

#### (mode 2) 로드 밸런싱(balance-xor)

패킷을 전송하는 **NIC**는 소스 **MAC** 주소 및 대상 **MAC** 주소에 **XOR** 작업을 수행하여 선택되며 총 **NIC** 수의 모듈도 곱해집니다. 이 알고리즘을 사용하면 각 대상 **MAC** 주소에 대해 동일한 **NIC**가 선택됩니다.

#### (mode 3) 브로드캐스트

패킷이 모든 **NIC**에 전송됩니다.

#### (mode 4) 동적 링크 집계 (802.3ad) (기본값)

**NIC**는 동일한 속도와 듀플렉스 설정을 공유하는 그룹으로 집계됩니다. 활성 집계 그룹의 모든 **NIC**가 사용됩니다.



## 참고

(mode 4) 동적 링크 집계 (802.3ad)에는 802.3ad를 지원하는 스위치가 필요합니다.

본딩된 NIC에는 동일한 집계기 ID가 있어야 합니다. 그렇지 않으면 Manager에 Network Interfaces 탭의 본딩에 경고 느낌표 아이콘이 표시되고 본딩의 ad\_partner\_mac 값은 00:00:00:00:00:00으로 보고됩니다. 다음 명령을 입력하여 집계기 ID를 확인할 수 있습니다.

```
# cat /proc/net/bonding/bond0
```

<https://access.redhat.com/solutions/67546> 을 참조하십시오.

Red Hat Virtualization은 브리지 네트워크에서 사용할 수 없으며 가상 머신 논리 네트워크와 호환되지 않으므로 다음과 같은 본딩 모드를 지원하지 않습니다.

### (mode 0) Round-Robin

NIC는 패킷을 순서대로 전송합니다. 패킷은 본딩에서 사용 가능한 첫 번째 NIC로 시작하고 본딩에서 마지막 사용 가능한 NIC로 끝나는 루프로 전송됩니다. 후속 루프는 사용 가능한 첫 번째 NIC로 시작됩니다.

### (mode 5) Balance-TLB, Transmit Load-Balance라고도 함

발신 트래픽은 본딩의 모든 NIC에 대한 부하를 기반으로 분산됩니다. 활성 NIC에서 들어오는 트래픽을 수신합니다. 들어오는 트래픽을 수신하는 NIC가 실패하면 다른 NIC가 할당됩니다.

### (mode 6) Balance-ALB, Adaptive Load-Balance라고도 함

(mode 5) Balance-TLB는 IPv4 트래픽의 수신 부하 분산과 결합됩니다. ARP 협상은 수신로드의 균형을 조정하는 데 사용됩니다.

## 9.6. 네트워크 연결 분석 및 모니터링

### 9.6.1. Skydive 소개

Skydive는 외부 네트워크 공급자로 정의된 OVN(Open Virtual Networks)을 포함하여 논리 네트워크를 모니터링하는 데 사용할 수 있습니다. Skydive는 네트워크 토폴로지, 종속 항목 및 흐름에 대한 실시간 보기를 제공하며 보고서를 생성하고 구성 감사를 수행합니다.

**Skydive**에서 제공하는 데이터를 사용하여 다음을 수행할 수 있습니다.

- 패킷 손실 감지
- 브리지 및 인터페이스를 포함하여 클러스터의 네트워크 토폴로지를 캡처하여 배포가 올바르게 작동하는지 확인합니다.
- 예상되는 **MTU** 설정이 올바르게 적용되었는지 확인합니다.
- 가상 머신 간 또는 가상 머신과 호스트 간 네트워크 트래픽 캡처

**Skydive**의 기능 세트에 대한 자세한 내용은 <http://skydive.network> 를 참조하십시오.



참고

**Skydive**는 기술 프리뷰 기능 전용입니다. 기술 프리뷰 기능은 **Red Hat** 프로덕션 서비스 수준 계약(**SLA**)에서 지원하지 않으며, 기능상 완전하지 않을 수 있어 프로덕션에 사용하지 않는 것이 좋습니다. 이러한 기능을 사용하면 향후 제품 기능을 조기에 이용할 수 있어 개발 과정에서 고객이 기능을 테스트하고 피드백을 제공할 수 있습니다.

**Red Hat** 기술 프리뷰 기능 지원 범위에 대한 자세한 내용은 <https://access.redhat.com/support/offerings/techpreview/>를 참조하십시오.

### 9.6.2. Skydive 설치

절차

1. **Manager** 머신에 **grdive-ansible** 을 설치합니다.

```
# yum --disablerepo="*" --enablerepo="rhel-7-server-rpms,rhel-7-server-extras-rpms,rhel-7-server-rh-common-rpms,rhel-7-server-openstack-14-rpms" install skydive-ansible
```

2. **/usr/share/ovirt-engine/playbooks/install-ECDHEdive.inventory.sample** 을 현재 디렉터리로 복사하고 이름을 인벤토리로 바꿉니다.

3.

**inventory/01\_hosts** 파일을 다음과 같이 수정합니다(전체 내용은 아래 참조).

a.

**Manager**의 **FQDN**을 사용하여 **geo\_dive\_os\_auth\_url** 을 업데이트합니다. 이는 **Manager**와 동일한 **FQDN**을 사용하는 **OVN**에서 사용됩니다.

b.

**OVN** 공급자에 사용되는 사용자 이름으로 **ovn\_provider\_username** 을 업데이트합니다. 기본값은 **/etc/ovirt-provider-ovn/ovirt-provider-ovn.conf** 에 정의되어 있습니다.

c.

**ovn\_provider\_password** 를 업데이트합니다.

d.

**[agents:ECDHE] <host\_group >**에서 **Skydive** 에이전트를 설치하는 호스트, 클러스터 또는 데이터 센터를 정의합니다.

다음을 실행하여 유효한 그룹 목록을 볼 수 있습니다.

```
/usr/share/ovirt-engine-metrics/bin/ovirt-engine-hosts-ansible-inventory | python -m json.tool
```



참고

각 호스트를 명시적으로 나열할 필요는 없습니다. 클러스터의 모든 호스트에 에이전트를 설치하려면 **ovirt\_cluster\_Default** 를 추가합니다. 또는 데이터 센터의 모든 호스트에 에이전트를 설치하려면 **ovirt\_datacenter\_Default** 를 추가합니다.

샘플 인벤토리 파일

```
[agents]
[analyzers]
[skydive:children]
  analyzers
  agents

[skydive:vars]
  skydive_listen_ip=0.0.0.0
  skydive_deployment_mode=package
  skydive_extra_config={'agent.topology.probes': ['ovsdb', 'neutron'],
    'agent.topology.neutron.ssl_insecure': true}

  skydive_fabric_default_interface=ovirtmgmt
```

```

skydive_os_auth_url=https://MANAGERS_FQDN:35357/v2.0
skydive_os_service_username=ovn_provider_username
skydive_os_service_password=ovn_provider_password
skydive_os_service_tenant_name=service
skydive_os_service_domain_name=Default
skydive_os_service_region_name=RegionOne

[agents:vars]
ansible_ssh_private_key_file=/etc/pki/ovirt-engine/keys/engine_id_rsa

[agents:children]
host_group

[analyzers]
localhost ansible_connection=local

```

4.

플레이북을 실행합니다.

```

# ansible-playbook -i inventory /usr/share/ovirt-engine/playbooks/install-skydive.yml
/usr/share/skydive-ansible/playbook.yml.sample

```

5.

**Skydive**가 **http://MANAGERS\_FQDN:8082**로 이동하여 가상 머신을 선택하고 캡처 탭의 메타데이터 섹션에 있는 다음 필드를 확인하여 가상 머신의 포트를 인식하는지 확인합니다.

- 관리자: **Neutron**
- **NetworkName: network\_name**
- **IPV4:** 서버넷이 사용되는 경우 **IP\_address**

**9.6.3절. “Skydive를 사용하여 네트워크 연결 테스트”**에서 **Skydive**를 사용하여 네트워크의 활동을 캡처하는 방법에 대한 예를 확인하십시오.

### 9.6.3. Skydive를 사용하여 네트워크 연결 테스트

이 예에서는 **IPv4** 주소가 있는 **NIC**가 있는 두 호스트 간의 연결을 테스트합니다. **NIC**는 **VLAN 4**로 태그된 논리적 네트워크에 연결됩니다. 논리 네트워크에 **IP** 주소를 할당하는 방법에 대한 자세한 내용은 **9.4.2절. “호스트 네트워크 인터페이스 편집 및 호스트에 논리 네트워크 할당”**을 참조하십시오.

절차

1. **Skydive**를 설치합니다.
2. **http://MANAGERS\_FQDN:8082** 에서 **Skydive**를 엽니다.
3. 네트워크 맵의 **rhv-host1** 에서 **network\_4** 를 선택합니다.
4. 캡처 탭에서 생성 을 클릭하고 시작을 클릭합니다.
5. **rhv-host0** 에서 **network\_4** 에 대해 이전 단계를 반복합니다.
6. 생성 탭을 클릭합니다.
7. **rhv-host0** 의 **eth0** 을 소스로, **rhv-host1** 의 **eth0** 을 **Destination** 으로 선택합니다.
8. 유형 드롭다운 목록에서 **ICMPv4/Echo Request** 를 선택합니다.
9. **Inject** 를 클릭하여 패킷을 삽입합니다.
10. 흐름 탭을 엽니다. **ping** 결과가 테이블에 표시됩니다. **ping**에 성공하면 **ICMPv4** 및 소스 및 대상 **IP** 주소가 포함된 행이 표시됩니다. 해당 행 위로 커서를 이동하면 **network\_4** 가 네트워크 맵의 노란색 원으로 강조 표시됩니다.

**Skydive** 사용에 대한 자세한 내용은 **Skydive** 설명서를 참조하십시오.

설치의 경우 : **testing!**

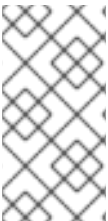
## 10장. 호스트

## 10.1. 호스트 소개

하이퍼바이저라고도 하는 호스트는 가상 머신이 실행되는 물리적 서버입니다. 전체 가상화는 **KVM(커널 기반 가상 시스템)**이라는 로드 가능한 **Linux** 커널 모듈을 사용하여 제공됩니다.

**KVM**은 **Windows** 또는 **Linux** 운영 체제를 실행하는 여러 가상 머신을 동시에 호스팅할 수 있습니다. 가상 머신은 호스트 머신에서 개별 **Linux** 프로세스 및 스레드로 실행되며 **Red Hat Virtualization Manager**에서 원격으로 관리합니다. **Red Hat Virtualization** 환경에는 하나 이상의 호스트가 연결되어 있습니다.

**Red Hat Virtualization**은 두 가지 호스트 설치 방법을 지원합니다. **RHVH(Red Hat Virtualization Host)** 설치 미디어를 사용하거나 표준 **Red Hat Enterprise Linux** 설치에 하이퍼바이저 패키지를 설치할 수 있습니다.



## 참고

호스트 이름을 선택하여 세부 정보 보기를 열고 소프트웨어에서 **OS** 설명을 확인하여 **Red Hat Virtualization Manager**에서 개별 호스트의 호스트 유형을 확인할 수 있습니다.

호스트는 가상화 최적화를 제공하는 **tuned** 프로필을 사용합니다. **tuned**에 대한 자세한 내용은 **Red Hat Enterprise Linux 7 성능 튜닝 가이드**를 참조하십시오.

**Red Hat Virtualization Host**에는 보안 기능이 활성화되어 있습니다. **SELinux(Security Enhanced Linux)** 및 방화벽은 기본적으로 완전히 구성되어 있습니다. 선택한 호스트의 **SELinux** 상태는 세부 정보 보기의 일반 탭에서 **SELinux** 모드에서 보고됩니다. **Manager**는 환경에 추가할 때 **Red Hat Enterprise Linux** 호스트에서 필수 포트를 열 수 있습니다.

호스트는 **Red Hat Enterprise Linux 7 AMD64/Intel 64** 버전을 실행하는 **Intel VT** 또는 **AMD-V** 확장 기능이 있는 물리적 **64비트** 서버입니다.

**Red Hat Virtualization** 플랫폼의 물리적 호스트:

- 시스템의 클러스터에만 속해야 합니다.

- **AMD-V 또는 Intel VT 하드웨어 가상화 확장을 지원하는 CPU가 있어야 합니다.**
- 클러스터 생성 시 선택한 가상 **CPU** 유형에서 노출하는 모든 기능을 지원하는 **CPU가 있어야 합니다.**
- **최소 2GB의 RAM이 있습니다.**
- **시스템 권한이 있는 할당된 시스템 관리자가 있을 수 있습니다.**

관리자는 **Red Hat Virtualization** 위키 목록에서 최신 보안 권고를 받을 수 있습니다. **Red Hat Virtualization** 위키 목록에 가입하여 **Red Hat Virtualization** 제품에 대한 새로운 보안 권고를 이메일로 받으십시오. 이 양식을 작성하여 구독하십시오.

<https://www.redhat.com/mailman/listinfo/rhsa-announce>

## 10.2. RED HAT VIRTUALIZATION HOST

**RHVH(Red Hat Virtualization Host)**는 가상 머신을 호스팅하는 데 필요한 패키지만 있는 **Red Hat Enterprise Linux**의 특수 빌드를 사용하여 설치됩니다. **Red Hat Enterprise Linux** 호스트에서 사용하는 항목을 기반으로 **Anaconda** 설치 인터페이스를 사용하며 **Red Hat Virtualization Manager** 또는 **yum** 을 통해 업데이트할 수 있습니다. **yum** 명령을 사용하는 것은 추가 패키지를 설치할 수 있는 유일한 방법이며 업그레이드 후에도 지속됩니다.

**RHVH**에는 호스트 리소스를 모니터링하고 관리 작업을 수행하는 데 필요한 **Cockpit** 웹 인터페이스가 있습니다. **SSH** 또는 콘솔을 통해 **RHVH**에 직접 액세스할 수 없으므로 **Cockpit** 웹 인터페이스에서는 호스트가 **Red Hat Virtualization Manager**에 추가되기 전에 수행되는 작업에 대한 그래픽 사용자 인터페이스를 제공합니다(예: 네트워킹 구성 및 호스트 엔진 배포, 터미널 하위 탭을 통해 터미널 명령을 실행하는 데 사용할 수도 있습니다).

웹 브라우저의 <https://HostFQDNorIP:9090>에서 **Cockpit** 웹 인터페이스에 액세스합니다. **RHVH**용 **Cockpit**에는 호스트의 상태, **SSH** 호스트 키, 셸프 호스트 엔진 상태, 가상 시스템 및 가상 시스템 통계를 표시하는 사용자 지정 가상화 대시보드가 포함되어 있습니다.

**RHVH**에서는 **ABRT(Automatic Bug Reporting Tool)**를 사용하여 애플리케이션 충돌에 대한 의미 있는 디버그 정보를 수집합니다. 자세한 내용은 **Red Hat Enterprise Linux** 시스템 관리자 가이드를 참조하십시오.



## 참고

**grubby** 툴을 사용하여 사용자 지정 부팅 커널 인수를 **Red Hat Virtualization** 호스트에 추가할 수 있습니다. **grubby** 툴은 **grub.cfg** 파일을 영구적으로 변경합니다. 호스트 **Cockpit** 웹 인터페이스의 **Terminal** 하위 탭으로 이동하여 **grubby** 명령을 사용합니다. 자세한 내용은 [Red Hat Enterprise Linux System Administrator's Guide](#) 를 참조하십시오.



## 주의

**Red Hat**은 **RHVH**에서 신뢰할 수 없는 사용자를 생성하지 않는 것이 좋습니다. 이는 로컬 보안 취약점이 악용될 수 있기 때문입니다.

### 10.3. RED HAT ENTERPRISE LINUX 호스트

가능 하드웨어에 **Red Hat Enterprise Linux 7** 설치를 호스트로 사용할 수 있습니다. **Red Hat Virtualization**은 **Intel VT** 또는 **AMD-V** 확장을 사용하여 **Red Hat Enterprise Linux 7 Server AMD64/Intel 64** 버전을 실행하는 호스트를 지원합니다. **Red Hat Enterprise Linux** 머신을 호스트로 사용하려면 **Red Hat Enterprise Linux Server** 및 **Red Hat Virtualization** 서브스크립션을 연결해야 합니다.

호스트 추가에는 가상화 검사, 패키지 설치, 브리지 생성 등 플랫폼에서 다음 단계를 완료하므로 시간이 다소 걸릴 수 있습니다. 세부 정보 보기를 사용하여 호스트 및 관리 시스템에서 연결을 설정하는 프로세스를 모니터링합니다.

필요한 경우 **Cockpit** 웹 인터페이스를 설치하여 호스트의 리소스를 모니터링하고 관리 작업을 수행할 수 있습니다. **Cockpit** 웹 인터페이스는 네트워킹 구성 및 셸프 호스트 엔진 배포와 같이 호스트를 **Red Hat Virtualization Manager**에 추가하기 전에 수행되는 작업에 대한 그래픽 사용자 인터페이스를 제공하며, 터미널 하위 탭을 통해 터미널 명령을 실행하는 데 사용할 수도 있습니다.



## 중요

**VDSM**에서 제공하는 위치독 데몬을 방해할 수 있으므로 **Red Hat Enterprise Linux** 호스트에 타사 위치독을 설치하면 안 됩니다.

### 10.4. SATELLITE 호스트 공급자 호스트



**Satellite** 호스트 공급자가 제공하는 호스트는 **Red Hat Virtualization Manager**에서 가상화 호스트로 사용할 수도 있습니다. **Satellite** 호스트 공급자가 외부 공급자로 **Manager**에 추가되면 **RHVH(Red Hat Virtualization Host)** 및 **Red Hat Enterprise Linux** 호스트와 동일한 방식으로 **Red Hat Virtualization**에서 제공하는 모든 호스트를 추가하여 사용할 수 있습니다.

## 10.5. 호스트 작업

### 10.5.1. Red Hat Virtualization Manager에 표준 호스트 추가

**Red Hat Virtualization** 환경에 호스트를 추가하려면 가상화 검사, 패키지 설치 및 브리지 생성과 같은 플랫폼에서 다음 단계를 완료하므로 시간이 다소 걸릴 수 있습니다.



#### 중요

정적 **IPv6** 주소를 사용하는 관리 브릿지를 생성할 때 호스트를 추가하기 전에 인터페이스 구성(**ifcfg**) 파일에서 네트워크 관리자 제어를 비활성화합니다. 자세한 내용은 <https://access.redhat.com/solutions/3981311> 을 참조하십시오.

#### 절차

1. 관리 포털에서 **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭합니다.
2. 새로 생성을 클릭합니다.
3. 드롭다운 목록을 사용하여 새 호스트의 **Data Center** (데이터 센터) 및 **Host Cluster** 를 선택합니다.
4. 새 호스트의 이름 및 주소를 입력합니다. 표준 **SSH** 포트인 포트 **22**는 **SSH** 포트 필드에 자동으로 채워집니다.
5. **Manager**에서 호스트에 액세스하는 데 사용할 인증 방법을 선택합니다.
  - **root** 사용자 암호를 입력하여 암호 인증을 사용합니다.
  - 또는 **SSH PublicKey** 필드에 표시되는 키를 호스트의 **/root/.ssh/authorized\_keys** 에 복사하여 공개 키 인증을 사용합니다.

6.

필요한 경우 고급 매개 변수 버튼을 클릭하여 다음 고급 호스트 설정을 변경합니다.

•

자동 방화벽 구성을 비활성화합니다.

•

호스트 **SSH** 지문을 추가하여 보안을 강화합니다. 수동으로 추가하거나 자동으로 가져올 수 있습니다.

7.

선택적으로 호스트에 지원되는 전원 관리 카드가 있는 전원 관리를 구성합니다. 전원 관리 구성에 대한 자세한 내용은 관리 가이드에 설명된 [호스트 전원 관리 설정](#)을 참조하십시오.

8.

**OK**를 클릭합니다.

새 호스트가 **Installing**( 설치 중) 상태의 호스트 목록에 표시되고 알림 **Drawer** (



)의 **Events** 섹션에서 설치 진행 상황을 볼 수 있습니다. 잠시 후 호스트 상태가 **Up** 으로 변경됩니다.

### 10.5.2. Satellite 호스트 공급자 호스트 추가

**Satellite** 호스트 공급자 호스트를 추가하는 프로세스는 **Manager**에서 호스트를 식별하는 방법을 제외하고 **Red Hat Enterprise Linux** 호스트를 추가하는 프로세스와 거의 동일합니다. 다음 절차에서는 **Satellite** 호스트 공급자가 제공하는 호스트를 추가하는 방법을 간략하게 설명합니다.

#### **Satellite** 호스트 공급자 호스트 추가

1.

컴퓨팅 → 호스트.

2.

새로 생성을 클릭합니다.

3.

드롭다운 메뉴를 사용하여 새 호스트의 **Host Cluster** 를 선택합니다.

4.

**FlexVolume /Satellite** 확인란을 선택하여 **Satellite** 호스트 공급자 호스트를 추가하기 위한

옵션을 표시하고 호스트를 추가할 공급자를 선택합니다.

5.

**Discovered Hosts** 또는 **Provisioned Hosts** 를 선택합니다.

- 

검색된 호스트 (기본 옵션): 드롭다운 목록에서 호스트, 호스트 그룹 및 컴퓨팅 리소스를 선택합니다.

- 

프로비저닝된 호스트: 공급자 호스트 드롭다운 목록에서 호스트를 선택합니다.

외부 공급자에서 검색할 수 있는 호스트에 대한 모든 세부 정보는 자동으로 설정되며 원하는 대로 편집할 수 있습니다.

6.

새 호스트의 이름 및 **SSH** 포트 (프로비저닝 호스트만 해당)를 입력합니다.

7.

호스트와 함께 사용할 인증 방법을 선택합니다.

- 

**root** 사용자 암호를 입력하여 암호 인증을 사용합니다.

- 

**SSH PublicKey** 필드에 표시되는 키를 호스트의 `/root/.ssh/authorized_hosts` 에 복사하여 공개 키 인증(프로비저닝 호스트만 해당)을 사용합니다.

8.

이제 **Red Hat Enterprise Linux** 호스트를 추가하기 위한 필수 단계를 완료했습니다. 고급 호스트 설정을 표시하려면 고급 매개 변수 드롭다운 버튼을 클릭합니다.

a.

선택적으로 자동 방화벽 구성을 비활성화합니다.

b.

선택적으로 호스트 **SSH** 지문을 추가하여 보안을 강화합니다. 수동으로 추가하거나 자동으로 가져올 수 있습니다.

9.

이제 해당 탭을 사용하여 **Power Management, ECDHE, Console, Network Provider** 를 구성할 수 있지만 **Red Hat Enterprise Linux** 호스트를 추가하는 데는 기본이 아니기 때문에 이 절차에서는 다루지 않습니다.

10.

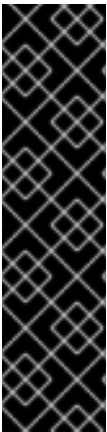
확인을 클릭하여 호스트를 추가하고 창을 닫습니다.

새 호스트는 설치 상태가 있는 호스트 목록에 표시되고 세부 정보 보기에서 설치 진행 상황을 확인할 수 있습니다. 설치가 완료되면 상태가 **Reboot** 로 업데이트됩니다. 상태를 **Up** 으로 변경하려면 호스트를 활성화해야 합니다.

### 10.5.3. 호스트에 대한 **Satellite** 에라타 관리 구성

**Red Hat Virtualization**은 **Red Hat Satellite**에서 에라타를 볼 수 있도록 구성할 수 있습니다. 이를 통해 호스트 관리자는 호스트 구성을 관리하는 데 사용되는 것과 동일한 대시보드에서 사용 가능한 에라타에 대한 업데이트를 받을 수 있습니다. **Red Hat Satellite**에 대한 자세한 내용은 [Red Hat Satellite 문서](#)를 참조하십시오.

**Red Hat Virtualization 4.3**에서는 **Red Hat Satellite 6.5**를 통한 에라타 관리를 지원합니다.



#### 중요

호스트는 **Satellite** 서버에서 **FQDN**으로 식별됩니다. **IP** 주소를 사용하여 추가된 호스트는 에라타를 보고할 수 없습니다. 따라서 **Red Hat Virtualization**에서 외부 콘텐츠 호스트 **ID**를 유지 관리할 필요가 없습니다.

호스트 관리에 사용되는 **Satellite** 계정에는 관리자 권한과 기본 조직이 설정되어 있어야 합니다.

#### 호스트에 대한 **Satellite** 에라타 관리 구성

1.

**Satellite** 서버를 외부 공급자로 추가합니다. 자세한 내용은 [14.2.1 절. “호스트 프로비저닝을 위한 \*\*Red Hat Satellite\*\* 인스턴스 추가”](#)를 참조하십시오.

2.

필요한 호스트를 **Satellite** 서버와 연결합니다.



## 참고

호스트는 **Satellite** 서버에 등록하고 **katello-agent** 패키지가 설치되어 있어야 합니다.

호스트 등록을 구성하는 방법과 **katello-agent** 패키지를 설치하고 설치하는 방법에 대한 자세한 내용은 **Red Hat Satellite** 문서 관리 문서의 호스트 등록을 참조하십시오.

- a. **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.
- b. 편집 을 클릭합니다.
- c. **UseECDHE/Satellite** 확인란을 선택합니다.
- d. 드롭다운 목록에서 필요한 **Satellite** 서버를 선택합니다.
- e. **OK**를 클릭합니다.

호스트는 호스트 구성을 관리하는 데 사용되는 것과 동일한 대시보드에 사용 가능한 에러타를 표시하도록 구성되어 있습니다.

#### 10.5.4. 새 호스트 및 **Edit Host Windows**의 설정 및 제어 설명

#### 10.5.5. 호스트 일반 설정 설명

이러한 설정은 호스트의 세부 정보를 편집하거나 새 **Red Hat Enterprise Linux** 호스트 및 **Satellite** 호스트 공급자 호스트를 추가할 때 적용됩니다.

일반 설정 표에는 새 호스트 또는 호스트 편집 창의 일반 탭에 필요한 정보가 포함되어 있습니다.

#### 표 10.1. 일반 설정

필드 이름	설명
호스트 클러스터	호스트가 속한 클러스터 및 데이터 센터입니다.
useECDHE/Satellite	<p>Satellite 호스트 공급자가 제공하는 호스트를 추가하기 위한 옵션을 보거나 숨기려면 이 확인란을 선택하거나 지웁니다. 다음 옵션도 사용할 수 있습니다.</p> <p><b>검색된 호스트</b></p> <ul style="list-style-type: none"> <li>● <b>검색된 호스트</b> - 엔진에서 검색한 Satellite 호스트의 이름으로 채워진 드롭다운 목록입니다.</li> <li>● <b>호스트 그룹</b> - 사용 가능한 호스트 그룹의 드롭다운 목록입니다.</li> <li>● <b>컴퓨팅 리소스</b> - 컴퓨팅 리소스를 제공하는 하이퍼바이저 드롭다운 목록입니다.</li> </ul> <p><b>프로비저닝된 호스트</b></p> <ul style="list-style-type: none"> <li>● <b>공급자 호스트</b> - 선택한 외부 공급자가 제공하는 호스트의 이름으로 채워진 드롭다운 목록입니다. 이 목록의 항목은 <b>공급자 검색 필터에 입력된 검색 쿼리에 따라 필터링</b> 됩니다.</li> <li>● <b>공급자 검색 필터</b> - 선택한 외부 공급자가 제공하는 호스트를 검색할 수 있는 텍스트 필드입니다. 이 옵션은 공급자마다 다릅니다. 특정 공급자의 검색 쿼리 작성에 대한 자세한 내용은 공급자 설명서를 참조하십시오. 사용할 수 있는 모든 호스트를 보려면 이 필드를 비워 둡니다.</li> </ul>
이름	호스트 이름입니다. 이 텍스트 필드에는 40자 제한이 있으며 대문자 및 소문자, 숫자, 하이픈 및 밑줄이 조합된 고유한 이름이어야 합니다.
주석	호스트와 관련된 일반 텍스트, 사람이 읽을 수 있는 주석을 추가하는 필드입니다.
호스트 이름	호스트의 IP 주소 또는 확인 가능한 호스트 이름입니다. 확인 가능한 호스트 이름을 사용하는 경우 호스트 이름이 호스트의 관리 네트워크에서 사용하는 IP 주소(IPv4 및 IPv6)와 일치하도록 확인되었는지 확인해야 합니다.
암호	호스트 root 사용자의 암호입니다. 호스트를 추가할 때만 지정할 수 있으며 나중에 편집할 수 없습니다.
SSH 공개 키	텍스트 상자의 콘텐츠를 호스트의 <code>/root/.ssh/authorized_hosts</code> 파일에 복사하여 암호 대신 Manager의 SSH 키를 사용하여 호스트로 인증합니다.

필드 이름	설명
호스트 방화벽 자동 설정	새 호스트를 추가할 때 Manager는 호스트의 방화벽에서 필요한 포트를 열 수 있습니다. 이는 기본적으로 활성화되어 있습니다. 이는 <b>고급 매개 변수</b> 입니다.
SSH#1gerprint	호스트의 SSH 지문을 가져와서 호스트가 반환할 것으로 예상되는 지문과 비교하여 일치 여부를 확인할 수 있습니다. 이는 <b>고급 매개 변수</b> 입니다.

### 10.5.6. 호스트 전원 관리 설정 설명

**Power Management** 설정 표에는 새 호스트 또는 호스트 편집 창의 **Power Management** 탭에 필요한 정보가 포함되어 있습니다. 호스트에 지원되는 전원 관리 카드가 있는 경우 전원을 구성할 수 있습니다.

표 10.2. 전원 관리 설정

필드 이름	설명
전원 관리 활성화	호스트에서 전원을 활성화합니다. <b>Power Management</b> (전원 관리) 탭에서 나머지 필드를 활성화하려면 이 확인란을 선택합니다.
kdump 통합	크래시 덤프가 중단되지 않도록 커널 크래시 덤프를 수행하는 동안 호스트가 펜싱되지 않도록 합니다. Red Hat Enterprise Linux 7.1 이상에서는 kdump를 기본적으로 사용할 수 있습니다. 호스트에서 kdump를 사용할 수 있지만 구성이 유효하지 않은 경우 (kdump 서비스를 시작할 수 없음) <b>Kdump 통합</b> 을 사용하면 호스트 설치에 실패합니다. 이 경우 10.6.4절. " <a href="#">fence_kdump 고급 구성</a> "을 참조하십시오.
전원 관리에 대한 정책 제어 비활성화	전원 관리는 호스트 클러스터의 스케줄링 정책에 의해 제어됩니다. 전원 관리가 활성화되고 낮은 사용률에 도달한 경우 Manager는 호스트 시스템의 전원을 끄고 부하 분산이 필요하거나 클러스터에 사용 가능한 호스트가 충분하지 않은 경우 이를 다시 시작합니다. 정책 제어를 비활성화하려면 이 확인란을 선택합니다.

필드 이름	설명
Sequential Order의 에이전트	<p>호스트의 차단 에이전트를 나열합니다. 차단 에이전트는 순차적, 동시 또는 둘 다 혼합할 수 있습니다.</p> <ul style="list-style-type: none"> <li>차단 에이전트를 순차적으로 사용하면 먼저 기본 에이전트를 사용하여 호스트를 중지하거나 시작하며 실패하는 경우 보조 에이전트가 사용됩니다.</li> <li>차단 에이전트를 동시에 사용하는 경우 두 차단 에이전트가 모두 중지되도록 호스트에 대한 Stop 명령에 응답해야 합니다. 하나의 에이전트가 Start 명령에 응답하는 경우 호스트는 가동됩니다.</li> </ul> <p>펜싱 에이전트는 기본적으로 순차적입니다. 위쪽 및 아래쪽 버튼을 사용하여 차단 에이전트가 사용되는 순서를 변경합니다.</p> <p>두 개의 차단 에이전트를 동시에 만들려면 <b>Concurrent</b>에서 다른 차단 에이전트 옆에 있는 드롭다운 목록을 사용하여 하나의 차단 에이전트를 선택합니다. 추가 차단 에이전트 옆에 있는 드롭다운 목록에서 그룹을 선택하여 동시 차단 에이전트 그룹에 추가될 수 있습니다.</p>
Fence Agent 추가	<p>+ 버튼을 클릭하여 새 차단 에이전트를 추가합니다. <b>Edit fence agent</b> 창이 열립니다. 이 창의 필드에 대한 자세한 내용은 아래 표를 참조하십시오.</p>
전원 관리 프록시 기본 설정	<p>기본적으로 Manager가 호스트와 동일한 클러스터 내에서 펜싱 프록시를 검색하도록 지정하고 펜싱 프록시를 찾을 수 없는 경우 Manager는 동일한 <b>dc</b> (데이터 센터)에서 검색합니다. 위쪽 및 아래쪽 버튼을 사용하여 이러한 리소스가 사용되는 순서를 변경합니다. 이 필드는 <b>Advanced Parameters</b>에서 사용할 수 있습니다.</p>

다음 표에는 **Edit fence agent** 창에 필요한 정보가 나와 있습니다.

표 10.3. 펜싱 에이전트 설정 편집

필드 이름	설명
address	호스트의 전원 관리 장치에 액세스하는 주소입니다. 확인 가능한 호스트 이름 또는 IP 주소 중 하나입니다.
사용자 이름	전원 관리 장치에 액세스할 사용자 계정입니다. 장치에서 사용자를 설정하거나 default 사용자를 사용할 수 있습니다.
암호	전원 관리 장치에 액세스하는 사용자의 암호입니다.



필드 이름	설명
<b>유형</b>	<p>호스트의 전원 관리 장치 유형입니다. 다음 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> <li>● APC - APC MasterSwitch 네트워크 전원 스위치. APC 5.x 전원 스위치 장치에서는 사용하지 않습니다.</li> <li>● <b>apc_snmp</b> - APC 5.x 전원 스위치 장치와 함께 사용하십시오.</li> <li>● <b>BladeCenter</b> - IBM Bladecenter Remote Supervisor Adapter.</li> <li>● <b>cisco_ucs</b> - Cisco Unified Computing System.</li> <li>● <b>hugeac5</b> - Dell 컴퓨터용 Dell 원격 액세스 컨트롤러</li> <li>● <b>completelyac7</b> - Dell 컴퓨터의 Dell 원격 액세스 컨트롤러.</li> <li>● <b>ECDHE</b> - ePowerSwitch 8M+ 네트워크 전원 스위치.</li> <li>● <b>hpblade</b> - HP BladeSystem.</li> <li>● <b>ILO,ilo2,ilo3,ilo4</b> - HP Integrated Lights-Out.</li> <li>● <b>ipmilan</b> - Intelligent Platform Management Interface 및 Sun Integrated Lights Out Management 장치입니다.</li> <li>● <b>RSA</b> - IBM Remote Supervisor Adapter</li> <li>● <b>RSB</b> - Fujitsu-Siemens RSB 관리 인터페이스.</li> <li>● <b>WTI</b> - WTI 네트워크 전원 스위치.</li> </ul> <p>전원 관리 장치에 대한 자세한 내용은 <a href="#">기술 참조의 전원 관리</a>를 참조하십시오.</p>
<b>포트</b>	<p>전원 관리 장치에서 호스트와 통신하는 데 사용하는 포트 번호입니다.</p>
<b>슬롯</b>	<p>전원 관리 장치의 분리를 식별하는 데 사용되는 번호입니다.</p>
<b>서비스 프로필</b>	<p>전원 관리 장치의 분리를 식별하는 데 사용되는 서비스 프로필 이름입니다. 이 필드는 장치 유형이 <b>cisco_ucs</b> 인 경우 <b>Slot</b> 대신 표시됩니다.</p>

필드 이름	설명
옵션	<p>전원 관리 장치 특정 옵션 이를 'key=value'로 입력합니다. 사용 가능한 옵션은 호스트의 전원 관리 장치 설명서를 참조하십시오.</p> <p>Red Hat Enterprise Linux 7 호스트의 경우 전원 관리 장치로 cisco_ucs를 사용하는 경우 <b>옵션 필드</b>에 <b>ssl_insecure=1</b> 을 추가해야 합니다.</p>
보안	<p>전원 관리 장치가 호스트에 안전하게 연결할 수 있도록 하려면 이 확인란을 선택합니다. 이 작업은 전원 관리 에이전트에 따라 ssh, ssl 또는 기타 인증 프로토콜을 통해 수행할 수 있습니다.</p>

**10.5.7. Restic 우선 순위 설정 설명**

**setup table** 은 새 호스트 또는 호스트 편집 창의 **ECDHE** 탭에 필요한 정보를 자세히 설명합니다.

표 10.4. ECDHE 설정

필드 이름	설명
ECDHE 우선 순위	<p>호스트에 스토리지 풀 관리자(SPM)의 역할이 부여될 가능성을 정의합니다. 옵션은 <b>낮음</b>, <b>정상</b> 및 <b>높은</b> 우선 순위입니다. 낮은 우선 순위는 호스트에 역할이 할당될 가능성이 감소하고 높은 우선 순위는 가능성이 높아질 수 있음을 의미합니다. 기본 설정은 Normal입니다.</p>

**10.5.8. 호스트 콘솔 설정 설명**

콘솔 설정 표에 새 호스트 또는 호스트 편집 창의 콘솔 탭에 필요한 정보가 자세히 나와 있습니다.

표 10.5. 콘솔 설정

필드 이름	설명
표시 주소 덮어쓰기	<p>이 확인란을 선택하여 호스트의 표시 주소를 덮어씁니다. 이 기능은 호스트가 내부 IP로 정의되고 NAT 방화벽 뒤에 있는 경우에 유용합니다. 사용자가 내부 네트워크 외부에서 가상 머신에 연결하는 경우 가상 시스템이 실행 중인 호스트의 개인 주소를 반환하지 않고 시스템에서 공용 IP 또는 FQDN(외부 네트워크에서 공용 IP로 확인됨)을 반환합니다.</p>

필드 이름	설명
표시 주소	여기에 지정된 표시 주소는 이 호스트에서 실행되는 모든 가상 머신에 사용됩니다. 주소는 정규화된 도메인 이름 또는 IP의 형식이어야 합니다.

### 10.5.9. 설명된 네트워크 공급자 설정

네트워크 공급자 설정 표에서 새 호스트 또는 호스트 편집 창의 네트워크 공급자 탭에 필요한 정보를 자세히 설명합니다.

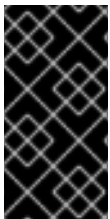
표 10.6. 네트워크 공급자 설정

필드 이름	설명
외부 네트워크 공급자	외부 네트워크 공급자를 추가하고 외부 네트워크 공급자가 호스트의 네트워크를 프로비저닝하려면 목록에서 하나를 선택합니다.

### 10.5.10. 커널 설정 설명

커널 설정 표에는 새 호스트 또는 호스트 편집 창의 커널 탭에 필요한 정보가 자세히 설명되어 있습니다. 일반적인 커널 부팅 매개변수 옵션은 확인란을 선택하여 쉽게 선택할 수 있습니다.

보다 복잡한 변경 사항은 커널 명령줄 옆에 있는 자유 텍스트 항목 필드를 사용하여 필요한 추가 매개변수를 추가합니다. 커널 명령줄 매개변수를 변경하는 경우 [호스트를 다시 설치해야 합니다](#).



중요

호스트가 **Manager**에 연결된 경우 변경하기 전에 호스트를 유지 관리 모드로 설정해야 합니다. 변경한 후 [호스트를 다시 설치](#) 하여 변경 사항을 적용합니다.

표 10.7. 커널 설정

필드 이름	설명
-------	----

필드 이름	설명
Hostdev Passthrough & SR-IOV	커널의 IOMMU 플래그를 활성화하여 장치가 가상 머신 자체에 직접 연결된 장치인 것처럼 가상 머신에서 호스트 장치를 사용할 수 있도록 합니다. 호스트 하드웨어 및 펌웨어도 IOMMU를 지원해야 합니다. 가상화 확장 및 IOMMU 확장은 하드웨어에서 활성화되어야 합니다. <a href="#">PCI 패스스루용 호스트 구성</a> 을 참조하십시오. IBM POWER8에는 기본적으로 IOMMU가 활성화되어 있습니다.
중첩 가상화	vmx 또는 svm 플래그를 활성화하여 가상 머신 내에서 가상 머신을 실행할 수 있습니다. 이 옵션은 평가 목적으로만 사용되며 프로덕션 용도로는 지원되지 않습니다. <b>vdsms-hook-nestedvt</b> 후크가 호스트에 설치되어 있어야 합니다.
안전하지 않은 인터럽트	IOMMU가 활성화되어 있지만 하드웨어에서 인터럽트 재조정을 지원하지 않기 때문에 패스스루가 실패하는 경우 이 옵션을 활성화하는 것을 고려할 수 있습니다. 호스트의 가상 머신을 신뢰할 수 있는 경우에만 이 옵션을 활성화해야 합니다. 옵션을 활성화하면 가상 머신의 ECDHE 공격을 통해 호스트를 노출할 수 있습니다. 이 옵션은 평가를 위해 인증되지 않은 하드웨어를 사용할 때만 해결 방법으로 사용하기 위한 것입니다.
PCI 실제 할당	SR-IOV NIC에서 메모리 문제로 인해 가상 기능을 할당할 수 없는 경우 이 옵션을 활성화하는 것이 좋습니다. 호스트 하드웨어 및 펌웨어도 PCI 재할당을 지원해야 합니다. 이 옵션은 평가를 위해 인증되지 않은 하드웨어를 사용할 때만 해결 방법으로 사용하기 위한 것입니다.
커널 명령줄	이 필드를 사용하면 기본 매개변수에 커널 매개변수를 더 추가할 수 있습니다.



참고

커널 부팅 매개변수가 회색으로 표시되면 재설정 버튼을 클릭하면 옵션을 사용할 수 있습니다.

10.5.11. 호스트 엔진 설정 설명

호스팅된 엔진 설정 테이블에는 새 호스트 또는 호스트 편집 창의 호스팅 엔진 탭에 필요한 정보가 자세히 설명되어 있습니다.

표 10.8. 호스트 엔진 설정

필드 이름	설명
호스팅 엔진 배포 작업 선택	<p>세 가지 옵션을 사용할 수 있습니다.</p> <ul style="list-style-type: none"> <li>● <b>none</b> - 작업이 필요하지 않습니다.</li> <li>● <b>배포</b> - 호스트를 셀프 호스팅 엔진 노드로 배포하려면 이 옵션을 선택합니다.</li> <li>● <b>배포 취소</b> - 셀프 호스팅 엔진 노드의 경우 이 옵션을 선택하여 호스트 배포를 취소하고 셀프 호스팅 엔진 관련 구성을 삭제할 수 있습니다.</li> </ul>

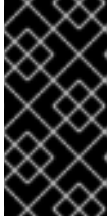
### 10.5.12. 호스트 전원 관리 설정 구성

관리 포털에서 호스트 라이프사이클 작업(중지, 시작, 재시작)을 수행하도록 호스트 전원 관리 장치 설정을 구성합니다.

호스트 고가용성 및 가상 머신 고가용성을 활용하려면 호스트 전원을 구성해야 합니다. 전원 관리 장치에 대한 자세한 내용은 기술 참조의 [전원 관리](#)를 참조하십시오.

#### 전원 관리 설정 구성

1. **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.
2. **Management** → **Maintenance** (유지 관리)를 클릭하고 **OK (확인)**를 클릭하여 확인합니다.
3. 호스트가 유지 관리 모드인 경우 편집을 클릭합니다.
4. **Power Management** 탭을 클릭합니다.
5. **Enable Power Management (전원 관리 활성화)** 확인란을 선택하여 필드를 활성화합니다.
6. 커널 크래시 덤프를 수행하는 동안 호스트가 펜싱되지 않도록 **Kdump** 통합 확인란을 선택합니다.



중요

기존 호스트에서 **Kdump** 통합을 활성화하거나 비활성화하는 경우 **kdump**를 구성할 호스트를 다시 설치해야 합니다.

7. 선택적으로 호스트 클러스터의 스케줄링 정책에 의해 호스트의 전원 관리를 제어하지 않도록 하려면 전원 관리 정책 비활성화 확인란을 선택합니다.
8. 더하기(+) 버튼을 클릭하여 새 전원 관리 장치를 추가합니다. **Edit fence agent** 창이 열립니다.
9. 해당 필드에 전원 관리 장치의 **User Name** 및 **Password** 를 입력합니다.
10. 드롭다운 목록에서 전원 관리 장치 유형을 선택합니다.
11. 주소 필드에 **IP** 주소를 입력합니다.
12. 전원 관리 장치에서 호스트와 통신하는 데 사용하는 **SSH** 포트 번호를 입력합니다.
13. 전원 관리 장치의 분리를 식별하는 데 사용되는 슬롯 번호를 입력합니다.
14. 전원 관리 장치의 옵션을 입력합니다. 쉼표로 구분된 **'key=value'** 항목을 사용합니다.
  - **IPv4** 및 **IPv6 IP** 주소를 모두 사용할 수 있는 경우 (기본값) **Options** 필드를 비워 둡니다.
  - **IPv4 IP** 주소만 사용할 수 있는 경우 **inet4\_only=1** 을 입력합니다.
  - **IPv6 IP** 주소만 사용할 수 있는 경우 **inet6\_only=1** 을 입력합니다.
15. 보안 확인란을 선택하여 전원 관리 장치가 호스트에 안전하게 연결할 수 있도록 합니다.

16. 테스트를 클릭하여 설정이 올바른지 확인합니다. **Test Succeeded, Host Status is: on** 은 확인에 성공하면 표시됩니다.
17. 확인 을 클릭하여 **Edit fence agent** 창을 닫습니다.
18. **Power Management** 탭에서 고급 매개 변수 를 확장하고 **up** 및 **down** 버튼을 사용하여 **Manager**에서 호스트의 클러스터 를 검색하고 **dc** (데이터 센터)에서 펜싱 프로세스를 검색할 순서를 지정합니다.
19. **OK**를 클릭합니다.



참고

- **IPv6**의 경우 **Red Hat Virtualization**은 정적 주소 지정만 지원합니다.
- 듀얼 스택 **IPv4** 및 **IPv6** 주소는 지원되지 않습니다.

관리 포털에서 **Management** → **Power Management** 드롭다운 메뉴가 활성화되어 있습니다.

### 10.5.13. 호스트 스토리지 풀 관리자 설정 구성

스토리지 풀 관리자(**SPM**)는 스토리지 도메인에 대한 액세스 제어를 유지하기 위해 데이터 센터의 호스트 중 하나에 제공되는 관리 역할입니다. **BOOM**을 항상 사용할 수 있어야 하며, **host**를 사용할 수 없게 되면 다른 호스트에 역할이 할당됩니다. **ECDHE** 역할은 호스트의 사용 가능한 리소스 중 일부를 사용하여 리소스를 부여할 수 있는 호스트에 우선 순위를 지정하는 것이 중요합니다.

호스트의 스토리지 풀 관리자(**SPM**) 우선 순위 설정은 **host**에 **DestinationRule** 역할이 할당될 가능성을 변경합니다. 우선 순위가 높은 호스트에는 우선 순위가 낮은 호스트에 보다 먼저 **ECDHE** 역할이 할당됩니다.

#### **ECDHE** 설정 구성

1. 컴퓨팅 → 호스트.

2. 편집 을 클릭합니다.
3. **ECDHE** 탭을 클릭합니다.
4. 라디오 버튼을 사용하여 호스트의 적절한 **DestinationRule** 우선 순위를 선택합니다.
5. **OK**를 클릭합니다.

### 10.5.14. PCI 패스스루용 호스트 구성



참고

이는 **Red Hat Virtualization**에서 **SR-IOV**를 설정하고 구성하는 방법을 보여주는 일련의 주제 중 하나입니다. 자세한 내용은 **SR-IOV 설정 및 구성**을 참조하십시오.

**PCI** 패스스루를 사용하면 가상 머신에서 장치가 가상 머신에 직접 연결된 것처럼 호스트 장치를 사용할 수 있습니다. **PCI** 패스스루 기능을 활성화하려면 가상화 확장 및 **IOMMU** 기능을 활성화해야 합니다. 다음 절차에 따라 호스트를 재부팅해야 합니다. 호스트가 이미 **Manager**에 연결된 경우 호스트를 유지보수 모드에 먼저 배치해야 합니다.

#### 사전 요구 사항

- 호스트 하드웨어가 **PCI** 장치 패스스루 및 할당 요구사항을 충족하는지 확인합니다. 자세한 내용은 **PCI 장치 요구 사항**을 참조하십시오.

#### PCI 패스스루용 호스트 구성

1. **BIOS**에서 가상화 확장 및 **IOMMU** 확장을 활성화합니다. 자세한 내용은 **Red Hat Enterprise Linux Virtualization 배포 및 관리 가이드의 BIOS에서 Intel VT-x 및 AMD-V 가상화 하드웨어 확장 활성화**를 참조하십시오.
2. **Manager**에 호스트를 추가하거나 **grub** 구성 파일을 수동으로 편집하여 커널에서 **IOMMU** 플래그를 활성화합니다.
  - 관리 포털에서 **IOMMU** 플래그를 활성화하려면 **Red Hat Virtualization Manager**에 표준 호스트 추가 및 설명되는 커널 설정을 참조하십시오.



- **grub** 설정 파일을 수동으로 편집하려면 **IOMMU 수동 활성화**를 참조하십시오.
3. **GPU** 패스스루의 경우 호스트와 게스트 시스템 모두에서 추가 구성 단계를 실행해야 합니다. 자세한 내용은 **GPU 장치 패스스루: Red Hat Virtualization**에서 가상 머신에 **NVIDIA GPU** 설정의 단일 가상 머신에 호스트 **GPU** 할당을 참조하십시오.

### IOMMU 수동 활성화

1. **grub** 구성 파일을 편집하여 **IOMMU**를 활성화합니다.



참고

**IBM POWER8** 하드웨어를 사용하는 경우 기본적으로 **IOMMU**가 활성화되어 있으므로 이 단계를 건너뛰니다.

- **Intel**의 경우 시스템을 부팅하고 **grub** 설정 파일의 **GRUB\_CMDLINE\_LINUX** 줄 끝에 **intel\_iommu=on** 을 추가합니다.

```
# vi /etc/default/grub
...
GRUB_CMDLINE_LINUX="nofb splash=quiet console=tty0 ... intel_iommu=on
...
```

- **AMD**의 경우 시스템을 부팅하고 **grub** 설정 파일의 **GRUB\_CMDLINE\_LINUX** 줄 끝에 **amd\_iommu=on** 을 추가합니다.

```
# vi /etc/default/grub
...
GRUB_CMDLINE_LINUX="nofb splash=quiet console=tty0 ... amd_iommu=on
...
```



## 참고

**intel\_iommu=on** 또는 **amd\_iommu=on** 이 작동하는 경우 **iommu=pt** 또는 **amd\_iommu=pt** 를 추가할 수 있습니다. **pt** 옵션은 패스스루에 사용되는 장치에 대해서만 **IOMMU**를 활성화하고 더 나은 호스트 성능을 제공합니다. 그러나 옵션은 모든 하드웨어에서 지원되지 않을 수 있습니다. **pt** 옵션이 호스트에 대해 작동하지 않는 경우 이전 옵션으로 되돌립니다.

하드웨어에서 인터럽트 재조정을 지원하지 않기 때문에 패스스루가 실패하면 가상 머신이 신뢰할 수 있는 경우 **allow\_unsafe\_interrupts** 옵션을 활성화할 수 있습니다. **allow\_unsafe\_interrupts** 는 가상 머신의 **host**를 **ECDHE** 공격에 노출시킬 수 있기 때문에 기본적으로 활성화되지 않습니다. 옵션을 활성화하려면 다음을 수행합니다.

```
# vi /etc/modprobe.d
options vfio_iommu_type1 allow_unsafe_interrupts=1
```

2.

**grub.cfg** 파일을 새로고침하고 이러한 변경 사항을 적용하려면 호스트를 재부팅합니다.

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

```
# reboot
```

**SR-IOV**를 활성화하고 전용 가상 **NIC**를 가상 머신에 할당하려면 <https://access.redhat.com/articles/2335291> 을 참조하십시오.

### 10.5.15. 호스트를 유지 관리 모드로 전환

네트워크 구성 및 소프트웨어 업데이트 배포를 포함한 일반적인 여러 유지 관리 작업을 수행하려면 호스트를 유지 관리 모드로 전환해야 합니다. **VDSM**이 재부팅 또는 네트워크 또는 스토리지 문제와 같이 제대로 작동하지 않도록 하려면 호스트를 유지 관리 모드로 전환해야 합니다.

호스트가 유지 관리 모드로 전환되면 **Red Hat Virtualization Manager**는 실행 중인 모든 가상 머신을 대체 호스트로 마이그레이션합니다. 실시간 마이그레이션을 위한 표준 사전 요구 사항이 적용됩니다. 특히 마이그레이션된 가상 머신을 실행할 수 있는 용량이 있는 클러스터에 활성 호스트가 하나 이상 있어야 합니다.

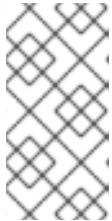


## 참고

호스트에 고정되어 있고 마이그레이션할 수 없는 가상 머신은 종료됩니다. 호스트 세부 정보 보기의 가상 머신 탭에서 **Pinned to Host** 를 클릭하여 호스트에 고정되는 가상 머신을 확인할 수 있습니다.

## 호스트를 유지 관리 모드로 배치

1. **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 원하는 호스트를 선택합니다.
2. **Management** → **Maintenance** (유지 관리)를 클릭하여 **Maintenance Host(s)** 확인 창을 엽니다.
3. 선택적으로, 호스트를 유지 관리 모드로 이동하는 이유를 입력합니다. 이 모드는 로그에 표시되고 호스트가 다시 활성화될 때입니다.



## 참고

호스트 유지 관리 이유 필드는 클러스터 설정에서 활성화된 경우에만 표시됩니다. 자세한 내용은 [8.2.2절](#). “일반 클러스터 설정 설명”를 참조하십시오.

4. 선택적으로 **Gluster**를 지원하는 호스트에 필요한 옵션을 선택합니다.

기본 점검을 방지하려면 **Ignore Gluster Quorum** 및 **Self-HECDHE Validations** 옵션을 선택합니다. 기본적으로 **Manager**는 호스트를 유지 관리 모드로 이동할 때 **Gluster** 퀴럼이 손실되지 않았는지 확인합니다. 또한 관리자는 호스트를 유지 관리 모드로 이동하여 영향을 받는 자동 복구 작업이 없는지 확인합니다. **Gluster** 퀴럼이 손실되거나 영향을 받을 자동 복구 작업이 있는 경우 **Manager**에서 호스트를 유지 관리 모드로 배치하지 않습니다. 호스트를 유지 관리 모드로 배치할 수 있는 다른 방법이 없는 경우에만 이 옵션을 사용합니다.

호스트를 유지 관리 모드로 이동하는 동안 모든 **Gluster** 서비스를 중지하려면 **Gluster Service** 중지 옵션을 선택합니다.



## 참고

이러한 필드는 선택한 호스트에서 **Gluster**를 지원하는 경우에만 호스트 유지 관리 기간에 표시됩니다. 자세한 내용은 **Red Hat Hyperconverged Infrastructure**에서 기본 **Gluster Storage** 노드 교체 를 참조하십시오.

5.

확인 을 클릭하여 유지보수 모드를 시작합니다.

실행 중인 모든 가상 시스템이 대체 호스트로 마이그레이션됩니다. 호스트가 **SPM(Storage Pool Manager)**인 경우 **ECDHE** 역할이 다른 호스트로 마이그레이션됩니다. 호스트의 **Status** 필드는 **Preparing for Maintenance**, 마지막으로 작업이 성공적으로 완료되면 **Maintenance** 로 변경됩니다. 호스트가 유지 관리 모드에 있는 동안 **VDSM**은 중지되지 않습니다.



## 참고

가상 머신에서 마이그레이션이 실패하면 호스트에서 **Management** → **Activate** (활성화)를 클릭하여 작업을 중지한 다음 가상 머신에서 마이그레이션 취소 를 클릭하여 마이그레이션을 중지합니다.

### 10.5.16. 유지 관리 모드에서 호스트 활성화

유지 관리 모드에 배치되었거나 최근에 환경에 추가된 호스트는 사용하기 전에 활성화해야 합니다. 호스트가 준비되지 않은 경우 활성화가 실패할 수 있습니다. 호스트 활성화를 시도하기 전에 모든 작업이 완료되었는지 확인합니다.

유지 관리 모드에서 호스트 활성화

1.

**Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.

2.

**Management** → **Activate** (활성화)를 클릭합니다.

호스트 상태가 **Unassigned** 으로 변경되고 마지막으로 작업이 완료되면 업그레이드됩니다. 이제 호스트에서 가상 머신을 실행할 수 있습니다. 유지 관리 모드로 전환될 때 호스트에서 마이그레이션된 가상 머신은 활성화될 때 호스트로 자동으로 마이그레이션되지 않지만 수동으로 마이그레이션할 수 있습니다. 유지 관리 모드로 전환되기 전에 호스트가 스토리지 풀 관리자(**SPM**)인 경우, 호스트가 활성화될 때 자동으로 반환되지 않습니다.

### 10.5.17. 호스트 방화벽 규칙 구성

**Ansible**을 사용하여 호스트 방화벽 규칙을 영구적으로 구성할 수 있습니다. 클러스터는 **iptables** 가 아닌 **firewalld** 를 사용하도록 구성해야 합니다.



참고

**iptables** 는 더 이상 사용되지 않습니다.

호스트에 대한 방화벽 규칙 구성

1.

**Manager** 시스템에서 **ovirt-host-deploy-post-tasks.example** 을 편집하여 사용자 지정 방화벽 포트를 추가합니다.

```
# vi /etc/ovirt-engine/ansible/ovirt-host-deploy-post-tasks.yml.example
---
#
# Any additional tasks required to be executing during host deploy process can
# be added below
#
- name: Enable additional port on firewalld
  firewalld:
    port: "12345/tcp"
    permanent: yes
    immediate: yes
    state: enabled
```

2.

파일을 **ovirt-host-deploy-post-tasks.yml** 으로 다른 위치에 저장합니다.

새로 설치하거나 다시 설치한 호스트는 업데이트된 방화벽 규칙을 사용하여 구성됩니다.

**Installation** → **Reinstall** 을 클릭하고 **Automatically configure host firewall** 을 선택하여 기존 호스트를 다시 설치해야 합니다.

### 10.5.18. 호스트 제거

가상화 환경에서 호스트를 제거합니다.

호스트 제거

1. **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.
2. **Management** → **Maintenance** 를 클릭합니다.
3. 호스트가 유지 관리 모드인 경우 제거를 클릭하여 호스트 제거 확인 창을 엽니다.
4. 호스트가 **Red Hat Gluster Storage** 클러스터의 일부인 경우 강제 제거 확인란을 선택합니다.
5. **OK**를 클릭합니다.

### 10.5.19. 마이너 릴리스 간 호스트 업데이트

클러스터의 모든 호스트를 업데이트하거나 개별 호스트를 업데이트할 수 있습니다.

#### 10.5.19.1. 클러스터의 모든 호스트 업데이트

호스트를 개별적으로 업데이트하는 대신 클러스터의 모든 호스트를 업데이트할 수 있습니다. 이는 **Red Hat Virtualization**의 새 버전으로 업그레이드하는 동안 특히 유용합니다. 업데이트를 자동화하는 데 사용되는 **Ansible** 역할에 대한 자세한 내용은 <https://github.com/oVirt/ovirt-ansible-cluster-upgrade/blob/master/README.md> 를 참조하십시오.

**Red Hat**은 클러스터를 한 번에 하나씩 업데이트할 것을 권장합니다.

#### 제한 사항

- **RHVH**에서 업데이트는 **/etc** 및 **/var** 디렉터리의 수정된 콘텐츠만 유지합니다. 다른 경로에서 수정된 데이터를 업데이트 중에 덮어씁니다.
- 클러스터 수준에서 마이그레이션이 활성화되면 가상 머신이 클러스터의 다른 호스트로 자동 마이그레이션됩니다.
- 자체 호스팅 엔진 환경에서 **Manager** 가상 머신은 동일한 클러스터에 있는 자체 호스팅 엔진 노드 간에만 마이그레이션할 수 있습니다. 표준 호스트로 마이그레이션할 수 없습니다.

- 클러스터에는 호스트가 유지보수를 수행하기 위해 충분한 메모리가 예약되어 있어야 합니다. 그렇지 않으면 가상 머신 마이그레이션이 중단되고 실패합니다. 호스트를 업데이트하기 전에 일부 또는 모든 가상 머신을 종료하여 호스트 업데이트의 메모리 사용량을 줄일 수 있습니다.

- 고정된 가상 머신(예: vGPU를 사용하는 가상 머신)을 다른 호스트로 마이그레이션할 수 없습니다. 대신 해당 호스트를 건너뛰는 경우를 제외하고 업데이트 중에 고정된 가상 머신이 종료됩니다.

## 절차

1. 관리 포털에서 **Compute** → **Clusters (클러스터)**를 클릭하고 클러스터를 선택합니다.
2. 업그레이드 를 클릭합니다.
3. 업데이트할 호스트를 선택한 다음 다음을 클릭합니다.
4. 옵션을 구성합니다.


- 고정 VM을 중지 하면 클러스터의 호스트에 고정된 가상 머신이 종료되고 기본적으로 선택됩니다. 고정 가상 머신이 중요한 서비스 또는 프로세스를 실행하는 경우와 같이 고정된 가상 머신이 실행 상태를 유지하고 업데이트 중에 알 수 없는 시간에 종료하지 않도록 이 확인란을 건너뛸 수 있습니다.

- Upgrade Timeout (Minutes)** 은 클러스터 업그레이드에 시간 초과로 실패하기 전에 개별 호스트를 업데이트할 때까지 대기하는 시간을 설정합니다. 기본값은 **60**입니다. **60**분이 충분하지 않은 대규모 클러스터의 경우 늘리거나 호스트가 빠르게 업데이트되는 소규모 클러스터의 경우 줄일 수 있습니다.

- 업그레이드 프로세스를 실행하기 전에 업그레이드에서 사용 가능한 업데이트가 있는지 확인합니다. 기본적으로 선택되지 않지만 최근 업데이트가 포함되어 있는지 확인해야 하는 경우 (예: **Manager**에서 호스트 업데이트를 기본값보다 자주 확인하도록 구성한 경우) 선택할 수 있습니다.

- 업그레이드 후 재부팅 하면 각 호스트가 업데이트된 후 재부팅되며 기본적으로 선택됩니다. 호스트를 재부팅해야 하는 보류 중인 업데이트가 없는지 확인하려면 이 확인란을 지우면 프로세스의 속도를 높일 수 있습니다.

- 유지 관리 정책을 사용하면 업데이트 중에 클러스터의 스케줄링 정책을 **cluster\_maintenance** 로 설정합니다. 기본적으로 선택되므로 작업이 제한되며고가용성이 아닌 경우 가상 머신을 시작할 수 없습니다. 업데이트 중에 계속 사용하려는 사용자 정의 스케줄링 정책이 있는 경우 이 확인란을 지우면 알 수 없는 결과가 발생할 수 있습니다. 이 옵션을 비활성화하기 전에 사용자 정의 정책이 클러스터 업그레이드 활동과 호환되는지 확인하십시오.
5. 다음을 클릭합니다.
  6. 영향을 받을 호스트 및 가상 시스템의 요약 검토합니다.
  7. 업그레이드 를 클릭합니다.

**Compute** → **Hosts** 보기에서 호스트 업데이트 진행 상황을 추적하고 **Notification Drawer** (  )의 **Events** 섹션에서 확인할 수 있습니다.

**Compute** → **Virtual Machines** 보기의 상태 열에서 개별 가상 머신 마이그레이션의 진행 상황을 추적할 수 있습니다. 대규모 환경에서는 결과를 필터링하여 특정 가상 시스템 그룹을 표시해야 할 수 있습니다.

### 10.5.19.2. 개별 호스트 업데이트

호스트 업그레이드 관리자를 사용하여 관리 포털에서 직접 개별 호스트를 업데이트합니다.



#### 참고

업그레이드 관리자는 호스트에 **Up** 또는 **Non-operational** 상태만 확인하지만 유지 관리 상태는 확인하지 않습니다.

#### 제한 사항

- RHVH**에서 업데이트는 **/etc** 및 **/var** 디렉터리의 수정된 콘텐츠만 유지합니다. 다른 경로에서 수정된 데이터를 업데이트 중에 덮어씁니다.
- 클러스터 수준에서 마이그레이션이 활성화되면 가상 머신이 클러스터의 다른 호스트로 자



동 마이그레이션됩니다. 사용량이 상대적으로 낮으면 호스트를 업데이트합니다.

- 자체 호스팅 엔진 환경에서 **Manager** 가상 머신은 동일한 클러스터에 있는 자체 호스팅 엔진 노드 간에만 마이그레이션할 수 있습니다. 표준 호스트로 마이그레이션할 수 없습니다.
- 클러스터에는 호스트가 유지보수를 수행하기 위해 충분한 메모리가 예약되어 있어야 합니다. 그렇지 않으면 가상 머신 마이그레이션이 중단되고 실패합니다. 호스트를 업데이트하기 전에 일부 또는 모든 가상 머신을 종료하여 호스트 업데이트의 메모리 사용량을 줄일 수 있습니다.
- 스토리지 풀 관리자(**SPM**) 작업을 수행하기 위해 하나의 호스트를 사용할 수 있어야 하므로 모든 호스트를 동시에 업데이트하지 마십시오.
- 고정된 가상 머신(예: **vGPU**를 사용하는 가상 머신)을 다른 호스트로 마이그레이션할 수 없습니다. 호스트를 업데이트하기 전에 고정된 가상 머신을 종료해야 합니다.

## 절차

1. 올바른 리포지토리가 활성화되어 있는지 확인합니다. 현재 활성화된 리포지토리 목록을 보려면 **yum repolist** 를 실행합니다.

- **Red Hat Virtualization** 호스트의 경우:

```
# subscription-manager repos --enable=rhel-7-server-rhvh-4-rpms
```

- **Red Hat Enterprise Linux** 호스트의 경우:

```
# subscription-manager repos \
  --enable=rhel-7-server-rpms \
  --enable=rhel-7-server-rhv-4-mgmt-agent-rpms \
  --enable=rhel-7-server-ansible-2.9-rpms
```

2. 관리 포털에서 **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 업데이트할 호스트를 선택합니다.

3. **Upgrade**를 위한 **Installation** → **Check for Upgrade** 를 클릭하고 **OK** 를 클릭합니다.

**Notification Drawer** (



)을 열고 이벤트 섹션을 확장하여 결과를 확인합니다.

4.

업데이트를 사용할 수 있는 경우 설치 → 업그레이드 를 클릭합니다.

5.

**OK** 를 클릭하여 호스트를 업데이트합니다. 마이그레이션 정책에 따라 가상 머신 실행이 마이그레이션됩니다. 가상 머신에 대해 마이그레이션이 비활성화되면 종료하라는 메시지가 표시됩니다.

호스트의 세부 정보는 **Compute** → **Hosts** 에서 업데이트되고 상태가 다음 단계를 통해 전환됩니다.

유지 관리 > 설치 > 재부팅 > Up



참고

업데이트에 실패하면 호스트의 상태가 **Install Failed** 로 변경됩니다. **Install Failed** 에서 설치 → 업그레이드 를 다시 클릭할 수 있습니다.

**Red Hat Virtualization** 환경의 각 호스트에 대해 이 절차를 반복합니다.

**Red Hat** 은 관리 포털에서 호스트를 업데이트할 것을 권장합니다. 그러나 **yum update** 를 사용하여 호스트를 업데이트할 수 있습니다.

### 10.5.19.3. 호스트 수동 업데이트

**yum** 명령을 사용하여 호스트를 업데이트할 수 있습니다. 시스템을 정기적으로 업데이트하여 보안 및 버그 수정을 적시에 적용할 수 있도록 합니다.

#### 제한 사항

- 

**RHVH**에서 업데이트는 **/etc** 및 **/var** 디렉터리의 수정된 콘텐츠만 유지합니다. 다른 경로에서 수정된 데이터를 업데이트 중에 덮어씁니다.

- 

클러스터 수준에서 마이그레이션이 활성화되면 가상 머신이 클러스터의 다른 호스트로 자동 마이그레이션됩니다. 사용량이 상대적으로 낮으면 호스트를 업데이트합니다.

- 자체 호스팅 엔진 환경에서 **Manager** 가상 머신은 동일한 클러스터에 있는 자체 호스팅 엔진 노드 간에만 마이그레이션할 수 있습니다. 표준 호스트로 마이그레이션할 수 없습니다.
- 클러스터에는 호스트가 유지보수를 수행하기 위해 충분한 메모리가 예약되어 있어야 합니다. 그렇지 않으면 가상 머신 마이그레이션이 중단되고 실패합니다. 호스트를 업데이트하기 전에 일부 또는 모든 가상 머신을 종료하여 호스트 업데이트의 메모리 사용량을 줄일 수 있습니다.
- 스토리지 풀 관리자(**SPM**) 작업을 수행하기 위해 하나의 호스트를 사용할 수 있어야 하므로 모든 호스트를 동시에 업데이트하지 마십시오.
- 고정된 가상 머신(예: **vGPU**를 사용하는 가상 머신)을 다른 호스트로 마이그레이션할 수 없습니다. 호스트를 업데이트하기 전에 고정된 가상 머신을 종료해야 합니다.

## 절차

1. 올바른 리포지토리가 활성화되어 있는지 확인합니다. **yum repolist** 를 실행하여 현재 활성화되어 있는 리포지토리를 확인할 수 있습니다.
  - **Red Hat Virtualization** 호스트의 경우:
 

```
# subscription-manager repos --enable=rhel-7-server-rhvh-4-rpms
```
  - **Red Hat Enterprise Linux** 호스트의 경우:
 

```
# subscription-manager repos \
  --enable=rhel-7-server-rpms \
  --enable=rhel-7-server-rhv-4-mgmt-agent-rpms \
  --enable=rhel-7-server-ansible-2.9-rpms
```
2. 관리 포털에서 **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 업데이트할 호스트를 선택합니다.
3. **Management** → **Maintenance** 를 클릭합니다.

4. 호스트를 업데이트합니다.

```
# yum update
```

5. 호스트를 재부팅하여 모든 업데이트가 올바르게 적용되었는지 확인합니다.



참고

**img** 기반 로그를 확인하여 **Red Hat Virtualization Host**에서 추가 패키지 업데이트가 실패한지 확인합니다. 업데이트 후 일부 패키지가 다시 설치되지 않은 경우 패키지가 `/var/imagbased/persisted-rpms`에 나열되어 있는지 확인합니다. 누락된 패키지를 추가한 다음 `rpm -Uvh /var/imagbased/persisted-rpms/*`를 실행합니다.

**Red Hat Virtualization** 환경의 각 호스트에 대해 이 프로세스를 반복합니다.

#### 10.5.20. 호스트 재설치

관리 포털에서 **RHVH(Red Hat Virtualization Host)** 및 **Red Hat Enterprise Linux** 호스트를 다시 설치합니다. 이 절차에는 호스트를 중지하고 다시 시작하는 작업이 포함됩니다.

##### 사전 요구 사항

- 클러스터 수준에서 마이그레이션을 활성화하면 가상 머신이 클러스터의 다른 호스트로 자동 마이그레이션됩니다. 따라서 호스트 사용량이 상대적으로 낮은 시점에 호스트 재설치를 수행하는 것이 좋습니다.
- 호스트가 유지보수를 수행하기 위해 클러스터에 충분한 메모리 예약이 있는지 확인합니다. 클러스터에 충분한 메모리가 없으면 가상 머신 마이그레이션 작업이 중단되고 실패합니다. 호스트를 유지 관리로 이동하기 전에 일부 또는 모든 가상 머신을 종료하여 이 작업의 메모리 사용량을 줄일 수 있습니다.
- 다시 설치를 수행하기 전에 클러스터에 두 개 이상의 호스트가 포함되어 있는지 확인합니다. 하나의 호스트를 스토리지 풀 관리자(**SPM**) 작업을 수행하는 데 사용할 수 있어야 하므로 모든 호스트를 동시에 다시 설치하지 마십시오.

##### 절차

1. **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.
2. **Management** → **Maintenance** 를 클릭합니다.
3. **Installation** → **Reinstall** 을 클릭하여 호스트 설치 창을 엽니다.
4. **OK** 를 클릭하여 호스트를 다시 설치합니다.

성공적으로 다시 설치되면 호스트에 **Up** 상태가 표시됩니다. 호스트를 마이그레이션한 모든 가상 머신을 이제 다시 마이그레이션할 수 있습니다.



중요

**Red Hat Virtualization Host**를 **Red Hat Virtualization Manager**에 성공적으로 등록한 후 다시 설치한 후 **Install Failed** (설치 실패) 상태의 관리 포털에 잘못 표시될 수 있습니다. **Management** → **Activate** (활성화) 를 클릭하면 호스트는 **Up** 상태로 변경되고 사용할 준비가 됩니다.

#### 10.5.21. 호스트 에라타 보기


**Red Hat Satellite** 서버에서 에라타 정보를 수신하도록 호스트가 구성된 후 각 호스트의 에라타를 볼 수 있습니다. 에라타 정보를 수신하도록 호스트 구성에 대한 자세한 내용은 다음을 참조하십시오. **10.5.3 절**. “호스트에 대한 **Satellite** 에라타 관리 구성”

##### 호스트 에라타 보기

1. **컴퓨팅** → **호스트**.
2. 호스트 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 에라타 탭을 클릭합니다.

#### 10.5.22. 호스트의 상태 보기

호스트에는 일반 상태 외에도 외부 상태가 있습니다. 외부 상태는 플러그인 또는 외부 시스템에 의해 보고되거나 관리자가 설정하며 호스트 이름 왼쪽에 다음 아이콘 중 하나로 표시됩니다.

- **OK:** 아이콘 없음
- **info:** 
- **경고:** 
- **오류:** 
- **실패:** 

호스트 상태에 대한 세부 정보를 보려면 호스트 이름을 클릭하여 세부 정보 보기를 열고 이벤트 탭을 클릭합니다.

호스트의 상태는 **REST API**를 사용하여 볼 수도 있습니다. 호스트의 **GET** 요청에는 상태가 포함된 **external\_status** 요소가 포함됩니다.

이벤트 컬렉션을 통해 **REST API**에서 호스트 상태를 설정할 수 있습니다. 자세한 내용은 **REST API** 가이드에 **이벤트 추가**를 참조하십시오.

### 10.5.23. 호스트 장치 보기

세부 정보 보기의 호스트 장치 탭에서 각 호스트의 호스트 장치를 볼 수 있습니다. 호스트가 직접 장치 할당을 위해 구성된 경우 성능 향상을 위해 이러한 장치를 가상 머신에 직접 연결할 수 있습니다.

직접 장치 할당에 대한 하드웨어 요구 사항에 대한 자세한 내용은 **SR-IOV** 구현을 위한 하드웨어 할당 고려 사항 사용에 대한 추가 하드웨어 고려 사항을 참조하십시오.

직접 장치 할당을 위해 호스트 구성에 대한 자세한 내용은 **10.5.14절. “PCI 패스스루용 호스트 구성”**을 참조하십시오.

호스트 장치를 가상 머신에 연결하는 방법에 대한 자세한 내용은 가상 머신 관리 가이드의 **호스트 장치를 참조하십시오.**

#### 호스트 장치 보기

1.       컴퓨팅 → 호스트.
2.       호스트 이름을 클릭하여 세부 정보 보기를 엽니다.
3.       호스트 장치 탭을 클릭합니다.

이 탭에는 장치가 가상 머신에 연결되었는지 여부와 해당 가상 머신에서 현재 사용 중인지 여부를 포함하여 호스트 장치의 세부 정보가 나열됩니다.

#### 10.5.24. 관리 포털에서 Cockpit에 액세스

**Cockpit**은 기본적으로 **RHVH(Red Hat Virtualization Host)** 및 **Red Hat Enterprise Linux** 호스트에서 사용할 수 있습니다. 주소를 브라우저에 입력하거나 관리 포털을 통해 **Cockpit** 웹 인터페이스에 액세스할 수 있습니다.

#### 관리 포털에서 Cockpit에 액세스

1.       관리 포털에서 **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.
2.       호스트 콘솔을 클릭합니다.

**Cockpit** 로그인 페이지가 새 브라우저 창에서 열립니다.

### 10.5.25. 레거시 SPICE Cipher 설정

**SPICE** 콘솔은 기본적으로 암호화 문자열과 함께 **FIPS** 호환 암호화를 사용합니다. 기본 **SPICE** 암호화 문자열: **kECDHE+FIPS:kDHE+FIPS:kRSA+FIPS:!eNULL:!aNULL**

이 문자열은 일반적으로 충분합니다. 그러나 이전 운영 체제 또는 **SPICE** 클라이언트가 있는 가상 머신이 있는 경우 하나 또는 다른 클라이언트가 **FIPS** 호환 암호화를 지원하지 않는 경우 약한 암호화 문자열을 사용해야 합니다. 그렇지 않으면 기존 클러스터에 새 클러스터 또는 새 호스트를 설치하고 해당 가상 머신에 연결을 시도하는 경우 연결 보안 오류가 발생할 수 있습니다.

**Ansible** 플레이북을 사용하여 암호화 문자열을 변경할 수 있습니다.

암호화 문자열 변경

1.

**Manager** 시스템에서 **/usr/share/ovirt-engine/playbooks** 디렉토리에 파일을 만듭니다. 예를 들면 다음과 같습니다.

```
# vim /usr/share/ovirt-engine/playbooks/change-spice-cipher.yml
```

2.

파일에 다음을 입력하고 저장합니다.

```
name: oVirt - setup weaker SPICE encryption for old clients
hosts: hostname
vars:
  host_deploy_spice_cipher_string: 'DEFAULT:-RC4:-3DES:-DES'
roles:
  - ovirt-host-deploy-spice-encryption
```

3.

방금 만든 파일을 실행합니다.

```
# ansible-playbook -I hostname /usr/share/ovirt-engine/playbooks/change-spice-cipher.yml
```

또는 다음과 같이 **host\_deploy\_spice\_cipher\_string** 과 함께 **--extra-vars** 옵션을 사용하여 **Ansible** 플레이북 **ovirt-host-deploy** 를 사용하여 호스트를 재구성할 수 있습니다.



```
# ansible-playbook -l hostname \
--extra-vars host_deploy_spice_cipher_string="DEFAULT:-RC4:-3DES:-DES" \
/usr/share/ovirt-engine/playbooks/ovirt-host-deploy.yml
```

## 10.6. 호스트 복원

### 10.6.1. 호스트 고가용성

**Red Hat Virtualization Manager**는 펜싱을 사용하여 클러스터에 호스트를 신속하게 유지합니다. 무응답 호스트는 비작동 호스트와 다릅니다. 비운영 호스트는 **Manager**와 통신할 수 있지만 잘못된 구성이 있습니다(예: 누락된 논리 네트워크). 응답하지 않는 호스트는 **Manager**와 통신할 수 없습니다.

펜싱을 사용하면 클러스터가 예기치 않은 호스트 장애에 대응하고 절전, 로드 밸런싱, 가상 시스템 가용성 정책을 적용할 수 있습니다. 호스트의 전원 관리 장치에 대한 펜싱 매개 변수를 구성하고 때때로 정확성을 테스트해야 합니다. 펜싱 작업에서는 무응답 호스트가 재부팅되고 호스트가 지정된 시간 내에 활성 상태로 돌아가지 않으면 보류 중인 수동 개입 및 문제 해결로 남아 있습니다.

#### 참고

펜싱 매개 변수를 자동으로 확인하려면 **PMHealthCheckEnabled** (기본적으로 **false**) 및 **PMHealthCheckIntervalInSec** (**3600 sec by default**)을 구성할 수 있습니다.

**true**로 설정하면 **PMHealthCheckEnabled** 는 **PMHealthCheckIntervalInSec** 에서 지정한 간격으로 모든 호스트 에이전트를 확인하고 문제가 탐지되면 경고를 발행합니다. **engine-config** 옵션 구성에 대한 자세한 내용은 [22.2.2절. “engine-config 명령의 구문”](#) 를 참조하십시오.

전원 관리 작업은 **Red Hat Virtualization Manager**를 재부팅한 후 프록시 호스트에서 수행하거나 관리 포털에서 수동으로 수행할 수 있습니다. 무응답 호스트에서 실행 중인 모든 가상 머신이 중지되고 고가용성 가상 머신이 다른 호스트에서 시작됩니다. 전원 관리 작업에는 두 개 이상의 호스트가 필요합니다.

**Manager**가 시작되면 음소거 시간(5분)이 경과한 후 전원 관리가 활성화된 무응답 호스트를 자동으로 펜싱합니다. 자동 시간은 **DisableFenceAtStartupInSec engine-config** 옵션을 업데이트하여 구성할 수 있습니다.



## 참고

**DisableFenceAtStartupInSec engine-config** 옵션을 사용하면 **Manager**에서 호스트를 부팅할 때 호스트를 펜싱하려고 하는 시나리오를 방지할 수 있습니다. 이는 호스트의 부팅 프로세스가 관리자 부팅 프로세스보다 길기 때문에 데이터 센터 중단 후에 발생할 수 있습니다.

전원 관리 매개 변수를 사용하여 프록시 호스트에서 자동으로 또는 호스트를 마우스 오른쪽 버튼으로 클릭하고 메뉴에서 옵션을 사용하여 수동으로 호스트를 펜싱할 수 있습니다.



## 중요

호스트가 가용성이 높은 가상 머신을 실행하는 경우 전원 관리를 활성화하고 구성해야 합니다.

### 10.6.2. Red Hat Virtualization의 Proxy에 의한 전원 관리

**Red Hat Virtualization Manager**는 차단 에이전트와 직접 통신하지 않습니다. 대신 **Manager**는 프록시를 사용하여 호스트 전원 관리 장치에 전원 관리 명령을 보냅니다. **Manager**는 **VDSM**을 사용하여 전원 관리 장치 작업을 실행하므로 환경의 다른 호스트가 펜싱 프록시로 사용됩니다.

다음 중 하나를 선택할 수 있습니다.

- 펜싱이 필요한 호스트와 동일한 클러스터에 있는 호스트입니다.
- 펜싱이 필요한 호스트와 동일한 데이터 센터에 있는 모든 호스트입니다.

실행 가능한 펜싱 프록시 호스트의 상태는 **UP** 또는 **Maintenance** 중 하나입니다.

### 10.6.3. 호스트에서 펜싱 매개 변수 설정

호스트 펜싱의 매개 변수는 새 호스트 또는 호스트 편집 창에서 **Power Management** 필드를 사용하여 설정됩니다. 전원 관리를 사용하면 시스템이 **RAC(Remote Access Card)**와 같은 추가 인터페이스를 사용하여 문제가 있는 호스트를 펜싱할 수 있습니다.

모든 전원 관리 작업은 **Red Hat Virtualization Manager**에서 직접 사용하는 대신 프록시 호스트를 사

용하여 수행됩니다. 전원 관리 작업에는 두 개 이상의 호스트가 필요합니다.

### 호스트에서 펜싱 매개 변수 설정

1. **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.
2. **편집** 을 클릭합니다.
3. **Power Management** 탭을 클릭합니다.
4. **Enable Power Management** (전원 관리 활성화) 확인란을 선택하여 필드를 활성화합니다.
5. 커널 크래시 덤프를 수행하는 동안 호스트가 펜싱되지 않도록 **Kdump** 통합 확인란을 선택합니다.



#### 중요

기존 호스트에서 **Kdump** 통합을 활성화하거나 비활성화하는 경우 호스트를 다시 설치해야 합니다.

6. 선택적으로 호스트 클러스터의 스케줄링 정책에 의해 호스트의 전원 관리를 제어하지 않도록 하려면 전원 관리 정책 비활성화 확인란을 선택합니다.
7. **+** 버튼을 클릭하여 새 전원 관리 장치를 추가합니다. **Edit fence agent** 창이 열립니다.
8. 전원 관리 장치의 주소, 사용자 이름, 암호를 입력합니다.
9. 드롭다운 목록에서 전원 관리 장치 유형을 선택합니다.



참고

사용자 지정 전원 관리 장치 설정 방법에 대한 자세한 내용은 <https://access.redhat.com/articles/1238743> 를 참조하십시오.

- 10. 전원 관리 장치에서 호스트와 통신하는 데 사용하는 **SSH** 포트 번호를 입력합니다.
- 11. 전원 관리 장치의 분리를 식별하는 데 사용되는 슬롯 번호를 입력합니다.
- 12. 전원 관리 장치의 옵션을 입력합니다. 쉼표로 구분된 **'key=value'** 항목을 사용합니다.
- 13. 보안 확인란을 선택하여 전원 관리 장치가 호스트에 안전하게 연결할 수 있도록 합니다.
- 14. 테스트 버튼을 클릭하여 설정이 올바른지 확인합니다. **Test Succeeded, Host Status is: on** 은 확인에 성공하면 표시됩니다.



주의

전원 관리 매개변수(**userid, password, options** 등)는 **Red Hat Virtualization Manager**에서 테스트한 후 설치 후 수동으로 테스트합니다. 잘못된 매개변수에 대한 경고를 무시하거나 **Red Hat Virtualization Manager**에서 해당 변경없이 전원 관리 하드웨어에서 매개변수가 변경되면 펜싱이 실패하는 경우 가장 필요할 때 펜싱이 실패할 가능성이 높습니다.

- 15. 확인 을 클릭하여 **Edit fence agent** 창을 닫습니다.
- 16. **Power Management** 탭에서 고급 매개 변수 를 확장하고 **up** 및 **down** 버튼을 사용하여 **Manager**에서 호스트의 클러스터 를 검색하고 **dc** (데이터 센터)에서 펜싱 프록시를 검색할 순서를 지정합니다.
- 17. **OK**를 클릭합니다.

호스트 목록으로 반환됩니다. 이제 호스트 이름 옆에 있는 느낌표가 사라졌으므로 전원 관리가 성공적으로 구성되었음을 나타냅니다.

#### 10.6.4. fence\_kdump 고급 구성

##### kdump

호스트 이름을 클릭하여 세부 정보 보기의 일반 탭에서 **kdump** 서비스 상태를 확인합니다.

- **enabled:** kdump가 올바르게 구성되어 **kdump** 서비스가 실행 중입니다.
- **disabled:** kdump 서비스가 실행되지 않습니다 (이 경우 **kdump** 통합이 제대로 작동하지 않습니다).
- **Unknown:** kdump 상태를 보고하지 않는 이전 **VDSM** 버전의 호스트에서만 발생합니다.

**kdump** 설치 및 사용에 대한 자세한 내용은 [Red Hat Enterprise Linux 7 Kernel Crash Dump Guide](#)를 참조하십시오.

##### fence\_kdump

**New Host** 또는 **Edit Host** 창의 **Power Management** 탭에서 **Kdump** 통합을 활성화하면 표준 **fence\_kdump** 설정이 구성됩니다. 환경의 네트워크 구성이 간단하며 **Manager**의 **FQDN**을 모든 호스트에서 확인할 수 있는 경우 기본 **fence\_kdump** 설정으로 충분합니다.

그러나 **fence\_kdump**의 고급 구성이 필요한 경우도 있습니다. 더 복잡한 네트워킹이 있는 환경에서는 **Manager**, **fence\_kdump** 리스너 또는 둘 다의 구성을 수동으로 변경해야 할 수 있습니다. 예를 들어 **Kdump** 통합이 활성화된 모든 호스트에서 관리자의 **FQDN**을 확인할 수 없는 경우 **engine-config**를 사용하여 적절한 호스트 이름 또는 **IP** 주소를 설정할 수 있습니다.

```
engine-config -s FenceKdumpDestinationAddress=A.B.C.D
```

다음 예제 사례에는 구성을 변경해야 할 수도 있습니다.

- **Manager**에는 두 개의 **NIC**가 있습니다. 여기서 이 중 하나는 공용이며, 두 번째는 **fence\_kdump** 메시지의 기본 대상입니다.
- 다른 **IP** 또는 포트에서 **fence\_kdump** 리스너를 실행해야 합니다.
- 가능한 패킷 손실을 방지하려면 **fence\_kdump** 알림 메시지의 사용자 지정 간격을 설정해야 합니다.

기본 구성 변경은 더 복잡한 네트워킹 설정에서만 필요하므로 사용자 지정된 **fence\_kdump** 탐지 설정을 사용하는 것이 좋습니다. **fence\_kdump** 리스너에 대한 구성 옵션의 경우 **fence\_kdump** 리스너 구성을 참조하십시오. 관리자의 **kdump** 구성은 **Manager**에서 **fence\_kdump** 구성을 참조하십시오.

### 10.6.4.1. fence\_kdump 리스너 구성

**fence\_kdump** 리스너의 구성을 편집합니다. 이는 기본 구성이 충분하지 않은 경우에만 필요합니다.

#### **fence\_kdump Listener** 수동 구성

1. **/etc/ovirt-engine/ovirt-ECDHE-kdump-listener.conf.d/**에서 새 파일(예: **my-delete-kdump.conf**)을 만듭니다.
2. 구문으로 사용자 지정 값을 입력하고 파일을 저장합니다.



중요

**10.6.4.2절. “관리자에서 fence\_kdump 구성”**의 **fence\_kdump Listener** 구성 옵션 테이블에 설명된 대로 편집된 값도 **engine-config** 에서 변경해야 합니다.

3. **fence\_kdump** 리스너를 다시 시작합니다.

```
# systemctl restart ovirt-fence-kdump-listener.service
```

필요한 경우 다음 옵션을 사용자 정의할 수 있습니다.

표 10.9. *fence\_kdump Listener* 구성 옵션

변수	설명	기본값	참고
LISTENER_ADDRESS	fence_kdump 메시지를 수신할 IP 주소를 정의합니다.	0.0.0.0	이 매개변수의 값이 변경되면 <b>engine-config</b> 의 <b>FenceKdumpDestinationAddress</b> 값과 일치해야 합니다.
LISTENER_PORT	fence_kdump 메시지를 수신할 포트를 정의합니다.	7410	이 매개변수의 값이 변경되면 <b>engine-config</b> 의 <b>FenceKdumpDestinationPort</b> 값과 일치해야 합니다.
HEARTBEAT_INTERVAL	리스너의 하트비트 업데이트 간격(초)을 정의합니다.	30	이 매개변수의 값이 변경되면 <b>engine-config</b> 에서 <b>FenceKdumpListenerTimeout</b> 값보다 절반 또는 작아야 합니다.
SESSION_SYNC_INTERVAL	메모리의 리스너 호스트 kdumping 세션을 데이터베이스에 동기화하는 간격(초)을 정의합니다.	5	이 매개변수의 값이 변경되면 <b>engine-config</b> 의 <b>KdumpStartedTimeout</b> 값보다 크기가 절반이거나 작아야 합니다.
REOPEN_DB_CONNECTION_INTERVAL	이전에 사용할 수 없는 데이터베이스 연결을 여는 간격(초)을 정의합니다.	30	-
KDUMP_FINISHED_TIMEOUT	호스트 kdump 흐름이 FINISHED로 표시된 kdumping 호스트에서 마지막으로 수신된 메시지 후 최대 시간 초과(초)를 정의합니다.	60	이 매개변수의 값이 변경되면 <b>engine-config</b> 에서 <b>FenceKdumpMessageInterval</b> 값보다 크기가 두 배로 커야 합니다.

#### 10.6.4.2. 관리자에서 *fence\_kdump* 구성

**Manager**의 **kdump** 설정을 편집합니다. 이는 기본 구성이 충분하지 않은 경우에만 필요합니다. 현재 구성 값은 다음을 사용하여 찾을 수 있습니다.

```
# engine-config -g OPTION
```

**engine-config**를 사용하여 **Kdump** 수동 구성

1. **engine-config** 명령을 사용하여 **kdump**의 설정을 편집합니다.

```
# engine-config -s OPTION=value
```



중요

**Kdump** 구성 옵션 표에 설명된 대로 편집된 값도 **fence\_kdump** 리스너 구성 파일에서 변경해야 합니다. **10.6.4.1절. “fence\_kdump 리스너 구성”**을 참조하십시오.

2. **ovirt-engine** 서비스를 다시 시작합니다.

```
# systemctl restart ovirt-engine.service
```

3. 필요한 경우 **Kdump** 통합이 활성화된 모든 호스트를 다시 설치합니다(아래 표 참조).

**engine-config** 를 사용하여 다음 옵션을 구성할 수 있습니다.

표 10.10. **kdump** 설정 옵션

변수	설명	기본값	참고
FenceKdumpDestinationAddress	fence_kdump 메시지를 보낼 호스트 이름 또는 IP 주소를 정의합니다. 비어 있는 경우 관리자의 FQDN이 사용됩니다.	빈 문자열 (관리자 FQDN 사용)	이 매개변수 값이 변경되면 fence_kdump 리스너 구성 파일에서 <b>LISTENER_ADDRESS</b> 값과 일치해야 하며 <b>Kdump 통합</b> 이 활성화된 모든 호스트를 다시 설치해야 합니다.
FenceKdumpDestinationPort	fence_kdump 메시지를 보낼 포트를 정의합니다.	7410	이 매개변수 값이 변경된 경우 fence_kdump 리스너 구성 파일의 <b>LISTENER_PORT</b> 값과 일치해야 하며 <b>Kdump 통합</b> 이 활성화된 모든 호스트를 다시 설치해야 합니다.



변수	설명	기본값	참고
FenceKdumpMessageInterval	fence_kdump에서 보낸 메시지 간 간격(초)을 정의합니다.	5	이 매개 변수의 값이 변경되면 fence_kdump 리스너 구성 파일에서 <b>KDUMP_FINISHED_TIMEOUT</b> 값보다 크기가 절반이거나 작아야 하며 <b>Kdump 통합이</b> 활성화된 모든 호스트를 다시 설치해야 합니다.
FenceKdumpListenerTimeout	마지막 하트비트에서 fence_kdump 리스너를 활성 상태로 간주하기 때문에 최대 시간 초과를 초단위로 정의합니다.	90	이 매개 변수의 값이 변경되면 fence_kdump 리스너 구성 파일에서 <b>HEARTBEAT_INTERVAL</b> 값보다 크기가 두 배로 커야 합니다.
KdumpStartedTimeout	kdump 호스트의 첫 번째 메시지가 수신될 때까지 대기하는 최대 제한 시간(초)을 정의합니다(호스트 kdump 흐름이 시작되었음을 감지).	30	이 매개 변수의 값이 변경되면 fence_kdump 리스너 구성 파일에서 <b>SESSION_SYNC_INTERVAL</b> 값보다 두 배로 커야 하며 <b>FenceKdumpMessageInterval</b> .

### 10.6.5. 소프트 펜싱 호스트

예기치 않은 문제로 인해 호스트가 무응답이 될 수 있으며 **VDSM**에 요청에 응답할 수 없는 경우에도 **VDSM**에 의존하는 가상 머신은 활성 상태로 유지되어 액세스할 수 있습니다. 이러한 경우 **VDSM** 재시작은 **VDSM**을 응답 상태로 반환하고 이 문제를 해결합니다.

**"SSHECDHE Fencing"**은 관리자가 무응답 호스트에서 **SSH**를 통해 **VDSM**을 다시 시작하려고 시도하는 프로세스입니다. **Manager**가 **SSH**를 통해 **VDSM**을 재시작하지 못하는 경우 외부 펜싱 에이전트가 구성된 경우 펜싱에 대한 책임은 외부 펜싱 에이전트에 종속됩니다.

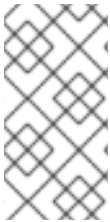
**SSH**를 통한 소프트 펜싱은 다음과 같이 작동합니다. 호스트에서 펜싱을 구성하고 활성화해야 하며 유효한 프록시 호스트(데이터 센터의 **UP** 상태의 두 번째 호스트)가 있어야 합니다. 관리자와 호스트 간의 연결이 시간 초과되면 다음이 발생합니다.

1. 첫 번째 네트워크 오류에서 호스트 상태가 "연결"으로 변경됩니다.

- 2.

그러면 **Manager**에서 **VDSM**에 상태를 요청하거나 호스트의 부하에 따라 결정되는 간격을 기다립니다. 간격의 길이를 결정하는 공식은 **TimeoutToResetVdsInSeconds** (기본값은 60초) + **[DelayResetPerVmInSeconds] \* (호스트에서 가상 머신 실행 횟수) + [DelayResetForSpmlnSeconds] \* [DelayResetForSpmlnSeconds (기본값) \* (기본값은 20초) \* (기본값은 0초) \* (초) \* (기본값: 0초) \* (초)**에 의해 구성됩니다. **VDSM**에 응답 시간을 최대로 제공하기 위해 **Manager**는 위에서 언급한 두 가지 옵션 중 더 긴 기간(양식 3개 또는 위 공식에 따라 결정된 간격)을 선택합니다.

3. 해당 간격이 경과할 때 호스트가 응답하지 않으면 **SSH**를 통해 **vdsm restart** 가 실행됩니다.
4. **vdsm** 재시작 이 호스트와 **Manager** 간 연결을 다시 설정하지 않으면 호스트의 상태가 **Non Responsive** 로 변경되고 전원 관리가 구성된 경우 펜싱이 외부 펜싱 에이전트로 전달됩니다.



참고

**SSH**를 통한 소프트 펜싱은 전원 관리가 구성되지 않은 호스트에서 실행할 수 있습니다. 이는 "연결"과 다릅니다. 펜싱은 전원 관리가 구성된 호스트에서만 실행할 수 있습니다.

10.6.6. 호스트 전원 관리 기능 사용

전원 관리가 호스트에 대해 구성된 경우 관리 포털 인터페이스에서 다양한 옵션에 액세스할 수 있습니다. 각 전원 관리 장치에는 사용자 정의 옵션이 있지만 모두 호스트를 시작, 중지 및 다시 시작하는 기본 옵션을 지원합니다.

호스트 전원 관리 기능 사용

1. **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.
2. **Management** 드롭다운 메뉴를 클릭하고 다음 **Power Management** 옵션 중 하나를 선택합니다.
  - **restart:** 이 옵션은 호스트를 중지하고 호스트의 상태가 **Down** 으로 변경될 때까지 기다립니다. 에이전트에서 호스트가 다운된 것을 확인하면 클러스터의 다른 호스트에서 고가용성 가상 시스템을 다시 시작합니다. 그런 다음 에이전트가 이 호스트를 다시 시작합니다. 호스트가 사용할 준비가 되면 상태가 **Up** 으로 표시됩니다.
  - **Start:** 이 옵션은 호스트를 시작하고 클러스터에 참여할 수 있게 합니다. 사용할 준비가 되면 **Up** 으로 표시됩니다.

**stop:** 이 옵션은 호스트의 전원을 끕니다. 이 옵션을 사용하기 전에 호스트에서 실행 중인 가상 머신이 클러스터의 다른 호스트로 마이그레이션되었는지 확인합니다. 그렇지 않으면 가상 머신이 충돌하고 고가용성 가상 머신만 다른 호스트에서 다시 시작됩니다. 호스트가 중지된 경우 해당 상태가 **Non-Operational** 로 표시됩니다.



#### 참고

**Power Management**가 활성화되지 않은 경우 호스트를 선택하거나 관리 드롭다운 메뉴를 클릭하고 **SSH** 관리 옵션, 재시작 또는 중지를 선택하여 호스트를 재시작 하거나 중지 할 수 있습니다.



#### 중요

호스트에 두 개의 펜싱 에이전트가 정의되면 동시에 또는 순차적으로 사용할 수 있습니다. 동시 에이전트의 경우 두 에이전트 모두 중지될 호스트에 대한 **Stop** 명령에 응답해야 하며, 하나의 에이전트가 **Start** 명령에 응답할 때 호스트는 가동됩니다. 순차적 에이전트의 경우 호스트를 시작하거나 중지하기 위해 기본 에이전트가 먼저 사용됩니다. 실패하는 경우 보조 에이전트가 사용됩니다.

3.

**OK**를 클릭합니다.

### 10.6.7. 비 응답 호스트 수동 펜싱 또는 분리


예를 들어 하드웨어 오류로 인해 호스트가 무응답 상태가 되면 환경의 성능에 큰 영향을 미칠 수 있습니다. 전원 관리 장치가 없거나 잘못 구성된 경우 호스트를 수동으로 재부팅할 수 있습니다.



#### 주의

호스트를 수동으로 재부팅하지 않으면 **Confirm host has been rebooted** 옵션을 사용하지 마십시오. 호스트가 계속 실행되고 있는 동안 이 옵션을 사용하면 가상 머신 이미지가 손상될 수 있습니다.

수동으로 펜싱하거나 무응답 호스트 분리

1. 관리 포털에서 **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트의 상태가 **Non Responsive** 인지 확인합니다.
2. 호스트를 수동으로 재부팅합니다. 즉, 물리적으로 랩에 들어가 호스트를 재부팅할 수 있습니다.
3. 관리 포털에서 호스트를 선택하고 **More Actions** (  )을 클릭한 다음 **Confirm 'Host has been Rebooted'** 를 클릭합니다.
4. **Approve Operation** 확인란을 선택하고 **OK** 를 클릭합니다.
5. 호스트가 비정상적으로 부팅하는 데 시간이 걸리는 경우 **ServerRebootTimeout** 을 설정하여 호스트가 무응답 인지 확인하기 전에 대기하는 시간(초)을 지정할 수 있습니다.

```
# engine-config --set ServerRebootTimeout=integer
```

## 11장. 스토리지

**Red Hat Virtualization**은 가상 디스크, **ISO** 파일 및 스냅샷에 중앙 집중식 스토리지 시스템을 사용합니다. 스토리지 네트워킹은 다음을 사용하여 구현할 수 있습니다.

- 네트워크 파일 시스템(**NFS**)
- **GlusterFS** 내보내기
- 기타 **POSIX** 호환 파일 시스템
- **iSCSI**(**I**nternet **S**mall **C**omputer **S**ystem **I**nterface)
- 가상화 호스트에 직접 연결된 로컬 스토리지
- 파이버 채널 프로토콜(**FCP**)
- 병렬 **NFS**(**pNFS**)

스토리지 도메인을 연결하고 활성화하지 않는 한 데이터 센터를 초기화할 수 없기 때문에 스토리지 설정은 새 데이터 센터의 사전 요구 사항입니다.

**Red Hat Virtualization** 시스템 관리자는 가상화된 엔터프라이즈용 스토리지를 생성, 구성, 연결 및 유지 관리해야 합니다. 스토리지 유형 및 용도에 대해 잘 알고 있어야 합니다. 스토리지 어레이 공급 업체의 가이드를 읽고 개념, 프로토콜, 요구 사항 또는 스토리지 일반적인 사용에 대한 자세한 내용은 [Red Hat Enterprise Linux Storage 관리 가이드](#)를 참조하십시오.

스토리지 도메인을 추가하려면 관리 포털에 성공적으로 액세스할 수 있어야 하며 **Up** 상태로 연결된 호스트가 하나 이상 있어야 합니다.

**Red Hat Virtualization**에는 세 가지 유형의 스토리지 도메인이 있습니다.



**데이터 도메인:** 데이터 도메인에는 데이터 센터에 있는 모든 가상 머신 및 템플릿의 가상 하드 디스크와 **OVF** 파일이 있습니다. 또한 가상 머신의 스냅샷도 데이터 도메인에 저장됩니다.

데이터 도메인은 데이터 센터 간에 공유할 수 없습니다. 여러 유형의 데이터 도메인(**iSCSI, NFS, FC, POSIX, Gluster**)은 모두 로컬, 도메인이 아니라 공유되는 경우 동일한 데이터 센터에 추가할 수 있습니다.

다른 유형의 도메인을 연결하기 전에 데이터 도메인을 데이터 센터에 연결해야 합니다.



**ISO 도메인:** **ISO** 도메인은 가상 시스템의 운영 체제 및 애플리케이션을 설치하고 부팅하는 데 사용되는 **ISO** 파일(또는 논리 **CD**)을 저장합니다. **ISO** 도메인에서는 물리적 미디어에 대한 데이터 센터의 필요성을 제거합니다. **ISO** 도메인은 다른 데이터 센터 간에 공유할 수 있습니다. **ISO** 도메인은 **NFS** 기반일 수 있습니다. 하나의 **ISO** 도메인만 데이터 센터에 추가할 수 있습니다.



**내보내기 도메인:** 데이터 센터와 **Red Hat Virtualization** 환경 간에 이미지를 복사하고 이동하는 데 사용되는 임시 스토리지 리포지토리입니다. 내보내기 도메인을 사용하여 가상 머신을 백업할 수 있습니다. 내보내기 도메인은 데이터 센터 간에 이동할 수 있지만 한 번에 하나의 데이터 센터에서만 활성화할 수 있습니다. 내보내기 도메인은 **NFS** 기반만 할 수 있습니다. 하나의 내보내기 도메인만 데이터 센터에 추가할 수 있습니다.



참고

내보내기 스토리지 도메인은 더 이상 사용되지 않습니다. 스토리지 데이터 도메인은 데이터 센터에서 연결 해제되어 동일한 환경 또는 다른 환경의 다른 데이터 센터로 가져올 수 있습니다. 그런 다음 가져온 스토리지 도메인에서 연결된 데이터 센터로 가상 머신, 유동 가상 디스크 및 템플릿을 업로드할 수 있습니다. 스토리지 도메인 가져오기에 대한 자세한 내용은 **11.7절. “기존 스토리지 도메인 가져오기”**를 참조하십시오.



중요

데이터센터의 스토리지 요구 사항을 확인한 후 **Red Hat Virtualization** 환경에 대한 스토리지 구성 및 연결을 시작하십시오.

11.1. 스토리지 도메인 이해

스토리지 도메인은 공통 스토리지 인터페이스가 있는 이미지 컬렉션입니다. 스토리지 도메인에는 템플릿 및 가상 머신(스냅샷 포함) 또는 **ISO** 파일의 전체 이미지가 포함되어 있습니다. 스토리지 도메인은 블록 장치(**SAN - iSCSI** 또는 **FCP**) 또는 파일 시스템(**NAS - NFS, GlusterFS** 또는 기타 **POSIX** 호환 파일 시스템)으로 만들 수 있습니다.

기본적으로 **GlusterFS** 도메인 및 로컬 스토리지 도메인은 **4K** 블록 크기를 지원합니다. **4K** 블록 크기는 특히 큰 파일을 사용할 때 더 나은 성능을 제공할 수 있으며 **VDO**와 같은 **4K** 호환성이 필요한 경우에도 필요합니다.

**NFS**에서 모든 가상 디스크, 템플릿 및 스냅샷은 파일입니다.

**SAN(iSCSI/FCP)**에서 각 가상 디스크, 템플릿 또는 스냅샷은 논리 볼륨입니다. 블록 장치는 볼륨 그룹이라는 논리 엔티티로 집계된 다음 **LVM(Logical Volume Manager)**으로 분할되어 가상 하드 디스크로 사용됩니다. **LVM**에 대한 자세한 내용은 **Red Hat Enterprise Linux Logical Volume Manager** 관리 가이드를 참조하십시오.

가상 디스크는 **QCOW2** 또는 **raw**의 두 가지 형식 중 하나를 사용할 수 있습니다. 스토리지 유형은 스파스이거나 사전 할당될 수 있습니다. 스냅샷은 항상 스파스이지만 두 형식의 디스크에 사용할 수 있습니다.

동일한 스토리지 도메인을 공유하는 가상 머신은 동일한 클러스터에 속한 호스트 간에 마이그레이션할 수 있습니다.

## 11.2. NFS 스토리지 준비 및 추가

### 11.2.1. NFS 스토리지 준비

**Red Hat Enterprise Virtualization Host** 시스템에서 스토리지 도메인 역할을 하도록 파일 스토리지 또는 원격 서버에 **NFS** 공유를 설정합니다. 원격 스토리지에서 공유를 내보내고 **Red Hat Virtualization Manager**에 구성하면 **Red Hat Virtualization** 호스트에서 공유가 자동으로 가져옵니다.

**NFS** 설정 및 구성에 대한 자세한 내용은 **Red Hat Enterprise Linux 7** 스토리지 관리 가이드의 **NFS(Network File System)** 를 참조하십시오.

'**NFS**' 공유를 내보내는 방법에 대한 자세한 내용은 **Red Hat Virtualization**의 **NetApp Storage / EMC SAN**에서 '**NFS**' 공유를 내보내는 방법을 참조하십시오.

**Red Hat Virtualization**에는 특정 시스템 사용자 계정 및 시스템 사용자 그룹이 필요하므로 **Manager**가 내보낸 디렉토리로 표시되는 스토리지 도메인에 데이터를 저장할 수 있습니다. 다음 절차에서는 한 디렉토리에 대한 권한을 설정합니다. **Red Hat Virtualization**에서 스토리지 도메인으로 사용하려는 모든 디렉토리에 대해 **chown** 및 **ECDHE** 단계를 반복해야 합니다.

절차

1.

**kvm** 그룹을 생성합니다.

```
# groupadd kvm -g 36
```

2.

**kvm** 그룹에 **vdsm** 을 생성합니다.

```
# useradd vdsm -u 36 -g 36
```

3.

내보낸 디렉터리의 소유권을 **36:36**으로 설정하여 **vdsm:kvm** 소유권을 부여합니다.

```
# chown -R 36:36 /exports/data
```

4.

읽기 및 쓰기 권한이 소유자에 부여되고 읽기 및 실행 액세스 권한이 그룹 및 기타 사용자에게 부여되도록 디렉터리의 모드를 변경합니다.

```
# chmod 0755 /exports/data
```

### 11.2.2. NFS 스토리지 추가

다음 절차에서는 기존 **NFS** 스토리지를 **Red Hat Virtualization** 환경에 데이터 도메인으로 연결하는 방법을 설명합니다.

**ISO** 또는 내보내기 도메인이 필요한 경우 다음 절차를 사용하지만 **Domain Function** 목록에서 **ISO** 또는 **Export** 를 선택합니다.

#### 절차

1.

관리 포털에서 스토리지 도메인을 클릭합니다.

2.

**New Domain** 을 클릭합니다.

3.

스토리지 도메인 의 이름을 입력합니다.

4.

데이터 센터,도메인 기능,스토리지 유형,형식, 호스트 목록의 기본값을 수락합니다.



5. 스토리지 도메인에 사용할 내보내기 경로를 입력합니다. 내보내기 경로는 **0:0.123.0.10:/data (IPv4)**, **[2001:0:0:0:5db1]:/data (IPv6)** 또는 **domain.example.com:/data** 형식이어야 합니다.
6. 필요한 경우 고급 매개변수를 구성할 수 있습니다.
  - a. 고급 매개 변수 를 클릭합니다.
  - b. **Warning Low Space Indicator** 필드에 백분율 값을 입력합니다. 스토리지 도메인에서 사용 가능한 공간이 이 백분율 미만이면 사용자에게 경고 메시지가 표시됩니다.
  - c. **Critical Space Action Blocker** 필드에 **GB** 값을 입력합니다. 스토리지 도메인에서 사용 가능한 공간이 이 값 아래에 있는 경우 사용자에게 오류 메시지가 표시되고, 공간을 사용하는 새 작업이 일시적으로 차단됩니다.
  - d. **delete** 옵션 이후에 초기화를 활성화하려면 **shape After Delete** 확인란을 선택합니다. 이 옵션은 도메인을 생성한 후 편집할 수 있지만, 이렇게 하면 이미 존재하는 디스크의 삭제 속성 이후 초기화를 변경하지 않습니다.
7. **OK**를 클릭합니다.

디스크가 준비될 때까지 새 **NFS** 데이터 도메인의 상태가 **Locked** 입니다. 그러면 데이터 도메인이 데이터 센터에 자동으로 연결됩니다.

### 11.2.3. NFS 스토리지 증가

**NFS** 스토리지의 양을 늘리려면 새 스토리지 도메인을 생성하고 기존 데이터 센터에 추가하거나 **NFS** 서버에서 사용 가능한 여유 공간을 늘릴 수 있습니다. 이전 옵션은 **11.2.2절. “NFS 스토리지 추가”**에서 참조하십시오. 다음 절차에서는 기존 **NFS** 서버에서 사용 가능한 여유 공간을 늘리는 방법을 설명합니다.

#### 기존 **NFS** 스토리지 도메인 증가

1. 스토리지 도메인을 클릭합니다.

2. **NFS** 스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **Data Center** 탭을 클릭하고 **Maintenance** 를 클릭하여 스토리지 도메인을 유지 관리 모드로 배치합니다. 이렇게 하면 기존 공유를 마운트 해제하고 스토리지 도메인의 크기를 조정할 수 있습니다.
4. **NFS** 서버에서 스토리지의 크기를 조정합니다. **Red Hat Enterprise Linux 6** 시스템의 경우 **Red Hat Enterprise Linux 6** 스토리지 관리 가이드를 참조하십시오. **Red Hat Enterprise Linux 7** 시스템의 경우 **Red Hat Enterprise Linux 7** 스토리지 관리 가이드를 참조하십시오.
5. 세부 정보 보기에서 데이터 센터 탭을 클릭하고 **Activate** 를 클릭하여 스토리지 도메인을 마운트합니다.

### 11.3. 로컬 스토리지 준비 및 추가

#### 11.3.1. 로컬 스토리지 준비

호스트에서 로컬 스토리지 도메인을 설정할 수 있습니다. 로컬 스토리지를 사용하도록 호스트를 설정하면 호스트는 새 데이터 센터에 자동으로 추가되고 다른 호스트를 추가할 수 없는 클러스터에 호스트가 자동으로 추가됩니다. 다중 호스트 클러스터를 사용하려면 모든 호스트가 로컬 스토리지에서 사용할 수 없는 모든 스토리지 도메인에 액세스할 수 있어야 합니다. 단일 호스트 클러스터에서 생성된 가상 시스템은 마이그레이션, 차단 또는 예약할 수 없습니다.



#### 중요

**RHVH(Red Hat Virtualization Host)**에서는 **/ (root)**와 별도로 로컬 스토리지를 파일 시스템에 정의해야 합니다. 업그레이드 중 데이터 손실을 방지하기 위해 별도의 논리 볼륨 또는 디스크를 사용하는 것이 좋습니다.

#### Red Hat Enterprise Linux 호스트용 로컬 스토리지 준비

1. 호스트에서 로컬 스토리지에 사용할 디렉터리를 만듭니다.

```
# mkdir -p /data/images
```

2. 디렉터리에 **vds** 사용자(**UID 36**) 및 **kvm** 그룹(**GID 36**)에 대한 읽기/쓰기 액세스를 허용하는 권한이 있는지 확인합니다.

```
# chown 36:36 /data /data/images
# chmod 0755 /data /data/images
```

### Red Hat Virtualization 호스트를 위한 로컬 스토리지 준비

다음과 같이 논리 볼륨에 로컬 스토리지를 생성하는 것이 좋습니다.

1.

로컬 스토리지 디렉터리를 생성합니다.

```
# mkdir /data
# lvcreate -L $SIZE rhvh -n data
# mkfs.ext4 /dev/mapper/rhvh-data
# echo "/dev/mapper/rhvh-data /data ext4 defaults,discard 1 2" >> /etc/fstab
# mount /data
```

2.

새 로컬 스토리지를 마운트한 다음 권한 및 소유권을 수정합니다.

```
# mount -a
# chown 36:36 /data /rhvh-data
# chmod 0755 /data /rhvh-data
```

#### 11.3.2. 로컬 스토리지 추가

호스트에 로컬 스토리지를 추가하면 호스트가 새 데이터 센터와 클러스터에 배치됩니다. 로컬 스토리지 구성 창은 데이터 센터, 클러스터 및 스토리지를 단일 프로세스로 결합합니다.

##### 절차

1.

**Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.

2.

**Management** → **Maintenance** 를 클릭하고 **OK** 를 클릭합니다.

3.

**Management** → **Configure Local Storage** 를 클릭합니다.

4.

**Data Center, Cluster, Storage** 필드 옆에 있는 편집 버튼을 클릭하여 로컬 스토리지 도메인을 구성하고 이름을 지정합니다.

5. 텍스트 항목 필드에서 로컬 스토리지의 경로를 설정합니다.
6. 해당하는 경우 최적화 탭을 클릭하여 새 로컬 스토리지 클러스터에 대한 메모리 최적화 정책을 구성합니다.
7. **OK**를 클릭합니다.

호스트는 자체 데이터 센터에서 온라인 상태가 됩니다.

### 11.4. POSIX 호환 파일 시스템 스토리지 준비 및 추가

#### 11.4.1. POSIX 호환 파일 시스템 스토리지 준비

**POSIX** 파일 시스템 지원을 통해 명령줄에서 수동으로 마운트할 때 일반적으로 사용하는 것과 동일한 마운트 옵션을 사용하여 파일 시스템을 마운트할 수 있습니다. 이 기능은 **NFS**, **iSCSI** 또는 **FCP**를 사용하여 노출되지 않는 스토리지에 대한 액세스를 허용하기 위한 것입니다.

**Red Hat Virtualization**에서 스토리지 도메인으로 사용되는 **POSIX** 호환 파일 시스템은 **GFS2(Global File System 2)**와 같은 클러스터 파일 시스템이어야 하며 스파스 파일 및 직접 **I/O**를 지원해야 합니다. 예를 들어 **CIFS(Common Internet File System)**는 직접 **I/O**를 지원하지 않으므로 **Red Hat Virtualization**과 호환되지 않습니다.

**POSIX** 호환 파일 시스템 스토리지 설정 및 구성에 대한 자세한 내용은 [Red Hat Enterprise Linux Global File System 2](#)를 참조하십시오.



#### 중요

**POSIX** 호환 파일 시스템 스토리지 도메인을 생성하여 **NFS** 스토리지를 마운트 하지 마십시오. 대신 항상 **NFS** 스토리지 도메인을 생성합니다.

#### 11.4.2. POSIX 호환 파일 시스템 스토리지 추가

다음 절차에서는 기존 **POSIX** 호환 파일 시스템 스토리지를 **Red Hat Virtualization** 환경에 데이터 도메인으로 연결하는 방법을 설명합니다.

#### 절차

1. 스토리지 도메인을 클릭합니다.
2. **New Domain** 을 클릭합니다.
3. 스토리지 도메인 의 이름을 입력합니다.
4. 스토리지 도메인과 연결할 데이터 센터를 선택합니다. 선택한 데이터 센터 유형은 **POSIX (POSIX 호환 FS)** 여야 합니다. 또는 **(none)** 를 선택합니다.
5. **Domain Function** 드롭다운 목록에서 **Data** 를 선택하고 **Storage Type** 드롭다운 목록에서 **POSIX 호환 FS** 를 선택합니다.  
  
해당하는 경우 드롭다운 메뉴에서 형식을 선택합니다.
6. 호스트 드롭다운 목록에서 호스트를 선택합니다.
7. 일반적으로 **mount** 명령에 제공하는 것처럼 **POSIX** 파일 시스템의 경로를 입력합니다.
8. 일반적으로 **-t** 인수를 사용하여 **mount** 명령에 제공하기 때문에 **VFS** 유형 을 입력합니다. 유효한 **VFS** 유형 목록은 **man mount** 를 참조하십시오.
9. 일반적으로 **-o** 인수를 사용하여 **mount** 명령에 제공하는 추가 마운트 옵션 을 입력합니다. 마운트 옵션은 쉼표로 구분된 목록으로 제공해야 합니다. 유효한 마운트 옵션 목록은 **man mount** 를 참조하십시오.
10. 필요한 경우 고급 매개 변수를 구성할 수 있습니다.
  - a. 고급 매개 변수 를 클릭합니다.
  - b. **Warning Low Space Indicator** 필드에 백분율 값을 입력합니다. 스토리지 도메인에서 사용 가능한 공간이 이 백분율 미만이면 사용자에게 경고 메시지가 표시됩니다.

- c. **Critical Space Action Blocker** 필드에 **GB** 값을 입력합니다. 스토리지 도메인에서 사용 가능한 공간이 이 값 아래에 있는 경우 사용자에게 오류 메시지가 표시되고, 공간을 사용하는 새 작업이 일시적으로 차단됩니다.
- d. **delete** 옵션 이후에 초기화를 활성화하려면 **shape After Delete** 확인란을 선택합니다. 이 옵션은 도메인을 생성한 후 편집할 수 있지만, 이렇게 하면 이미 존재하는 디스크의 삭제 속성 이후 초기화를 변경하지 않습니다.
11. **OK**를 클릭합니다.

## 11.5. 블록 스토리지 준비 및 추가

### 11.5.1. iSCSI 스토리지 준비

**Red Hat Virtualization**은 **LUN**으로 구성된 볼륨 그룹에서 생성된 스토리지 도메인인 **iSCSI** 스토리지를 지원합니다. 볼륨 그룹과 **LUN**을 한 번에 두 개 이상의 스토리지 도메인에 연결할 수 없습니다.

**iSCSI** 스토리지 설정 및 구성에 대한 자세한 내용은 **Red Hat Enterprise Linux 7** 스토리지 관리 가이드의 **온라인** 스토리지 관리를 참조하십시오.



#### 중요

블록 스토리지를 사용하고 있고 원시 장치 또는 직접 **LUN**에 가상 머신을 배포하고 논리 볼륨 관리자를 사용하여 가상 머신을 관리하려면 게스트 논리 볼륨을 숨기는 필터를 생성해야 합니다. 이렇게 하면 호스트가 부팅될 때 게스트 논리 볼륨이 활성화되지 않습니다. 이 경우 오래된 논리 볼륨이 활성화되고 데이터가 손상될 수 있습니다. 자세한 내용은 <https://access.redhat.com/solutions/2662261> 을 참조하십시오.



#### 중요

**Red Hat Virtualization**은 현재 블록 크기가 **4K**인 블록 스토리지를 지원하지 않습니다. 레거시(**512b** 블록) 모드로 블록 스토리지를 구성해야 합니다.

## 중요

호스트가 **SAN** 스토리지에서 부팅되고 스토리지 연결이 끊어지면 스토리지 파일 시스템이 읽기 전용으로 되어 연결이 복원된 후에도 이 상태로 유지됩니다.

이러한 상황을 방지하기 위해 **Red Hat**은 부팅 **LUN**에 대해 **SAN**의 루트 파일 시스템에 드롭인 다중 경로 구성 파일을 추가하여 연결이 있을 때 대기열에 있는지 확인하는 것이 좋습니다.

```
# cat /etc/multipath/conf.d/host.conf
multipaths {
  multipath {
    wwid boot_LUN_wwid
    no_path_retry queue
  }
}
```

### 11.5.2. iSCSI 스토리지 추가

다음 절차에서는 기존 **iSCSI** 스토리지를 **Red Hat Virtualization** 환경에 데이터 도메인으로 연결하는 방법을 설명합니다.

#### 절차

1. 스토리지 도메인을 클릭합니다.
2. **New Domain** 을 클릭합니다.
3. 새 스토리지 도메인의 이름을 입력합니다.
4. 드롭다운 목록에서 데이터 센터를 선택합니다.
5. **Domain Function** (도메인 기능)로 **Data** 를 선택하고 **Storage Type** (스토리지 유형)으로 **iSCSI** 를 선택합니다.
6. 활성 호스트를 호스트로 선택합니다.



중요

스토리지 도메인과의 통신은 **Manager**가 아닌 선택한 호스트에서 가져온 것입니다. 따라서 스토리지 도메인을 구성하기 전에 모든 호스트에 스토리지 장치에 액세스할 수 있어야 합니다.

7.

**Manager**는 **iSCSI** 대상을 **iSCSI** 대상에 **LUN** 또는 **LUN**에 매핑할 수 있습니다. **iSCSI** 스토리지 유형을 선택하면 **New Domain** (새 도메인) 창에는 사용되지 않은 **LUN**으로 알려진 대상이 자동으로 표시됩니다. 스토리지를 추가하는 데 사용하는 대상이 나타나지 않으면 대상 검색을 사용하여 찾을 수 있습니다. 그렇지 않으면 다음 단계로 진행하십시오.

a.

대상 검색을 클릭하여 대상 검색 옵션을 활성화합니다. 대상을 찾아서 로그인하면 **New Domain** (새 도메인) 창에서 환경에서 사용하지 않는 **LUN**으로 대상을 자동으로 표시합니다.



참고

외부에서 환경에 사용되는 **LUN**도 표시됩니다.

대상 검색 옵션을 사용하여 여러 대상 또는 여러 경로의 **LUN**을 동일한 **LUN**에 추가할 수 있습니다.

b.

**Address** 필드에 **iSCSI** 호스트의 **FQDN** 또는 **IP** 주소를 입력합니다.

c.

**Port**(포트) 필드에서 대상을 검색할 때 호스트에 연결할 포트를 입력합니다. 기본값은 **3260**입니다.

d.

**CHAP**를 사용하여 스토리지를 보호하는 경우 사용자 인증 확인란을 선택합니다. **CHAP** 사용자 이름 및 **CHAP** 암호를 입력합니다.



참고

**REST API**를 사용하여 특정 호스트에 대한 **iSCSI** 대상의 자격 증명을 정의할 수 있습니다. 자세한 내용은 **StorageServerConnectionExtensions: add in the REST API Guide** 를 참조하십시오.



e. **Discover** 를 클릭합니다.

f. 검색 결과에서 하나 이상의 대상을 선택하고 하나의 대상에 대해 로그인 또는 여러 대상에 대해 모두 로그인 을 클릭합니다.



중요

둘 이상의 경로 액세스가 필요한 경우 필요한 모든 경로를 통해 대상을 검색하고 로그인해야 합니다. 추가 경로를 추가하도록 스토리지 도메인을 수정하는 것은 현재 지원되지 않습니다.

8. 원하는 대상 옆에 있는 **+** 버튼을 클릭합니다. 이렇게 하면 항목이 확장되고 대상에 연결된 사용되지 않는 모든 **LUN**이 표시됩니다.

9. 사용 중인 각 **LUN**에 대한 확인란을 선택하여 스토리지 도메인을 생성합니다.

10. 필요한 경우 고급 매개변수를 구성할 수 있습니다.

a. 고급 매개 변수 를 클릭합니다.

b. **Warning Low Space Indicator** 필드에 백분율 값을 입력합니다. 스토리지 도메인에서 사용 가능한 공간이 이 백분율 미만이면 사용자에게 경고 메시지가 표시됩니다.

c. **Critical Space Action Blocker** 필드에 **GB** 값을 입력합니다. 스토리지 도메인에서 사용 가능한 공간이 이 값 아래에 있는 경우 사용자에게 오류 메시지가 표시되고, 공간을 사용하는 새 작업이 일시적으로 차단됩니다.

d. **delete** 옵션 이후에 초기화를 활성화하려면 **shape After Delete** 확인란을 선택합니다. 이 옵션은 도메인을 생성한 후 편집할 수 있지만, 이렇게 하면 이미 존재하는 디스크의 삭제 속성 이후 초기화를 변경하지 않습니다.

e. 삭제 후 삭제 옵션을 활성화하려면 **Discard After Delete** 확인란을 선택합니다. 이 옵션은 도메인을 생성한 후 편집할 수 있습니다. 이 옵션은 블록 스토리지 도메인에서만 사용할 수 있습니다.

11.

**OK**를 클릭합니다.

동일한 대상에 여러 스토리지 연결 경로를 구성한 경우 **iSCSI** 다중 경로 구성의 절차에 따라 **iSCSI** 본딩을 완료합니다.

현재 스토리지 네트워크를 **iSCSI** 본딩으로 마이그레이션하려면 **Logical Network to an iSCSI Bond**를 참조하십시오.

### 11.5.3. iSCSI 다중 경로 구성

**iSCSI** 멀티패스를 사용하면 논리 네트워크 및 **iSCSI** 스토리지 연결 그룹을 생성하고 관리할 수 있습니다. 호스트와 **iSCSI** 스토리지 간의 여러 네트워크 경로는 네트워크 경로 오류로 인한 호스트 다운타임을 방지합니다.

**Manager**는 **iSCSI** 본딩의 논리 네트워크에 할당된 **NIC** 또는 **VLAN**을 사용하여 데이터 센터의 각 호스트를 각 대상에 연결합니다.

중복성을 위해 여러 대상 및 논리 네트워크를 사용하여 **iSCSI** 본딩을 생성할 수 있습니다.

사전 요구 사항

- 하나 이상의 **iSCSI** 대상
- 다음 요구 사항을 충족하는 하나 이상의 논리적 네트워크:
  - 필수 또는 **VM**네트워크로 정의되지 않음
  - 호스트 인터페이스에 할당
  - **iSCSI** 본딩의 다른 논리 네트워크와 동일한 **VLAN** 및 서브넷의 고정 **IP** 주소가 할당 됨

절차

1. **Compute** → **Data Centers** (데이터 센터)를 클릭합니다.
2. 데이터 센터 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **iSCSI** 다중 경로 탭에서 추가를 클릭합니다.
4. **iSCSI** 본딩 추가 창에서 이름과 설명을 입력합니다.
5. **Logical Networks** 에서 논리적 네트워크를 선택하고 **Storage Targets** 에서 스토리지 도메인을 선택합니다. 동일한 대상의 모든 경로를 선택해야 합니다.
6. **OK**를 클릭합니다.

데이터 센터의 호스트는 **iSCSI** 본딩의 논리 네트워크를 통해 **iSCSI** 대상에 연결됩니다.

#### 11.5.4. 논리 네트워크를 **iSCSI** 본딩으로 마이그레이션

**iSCSI** 트래픽에 대해 생성되어 기존 네트워크 본딩 위에 구성된 논리 네트워크가 있는 경우, 중단이나 다운타임 없이 동일한 서브넷의 **iSCSI** 본딩으로 마이그레이션할 수 있습니다.

#### 절차

1. 필수 가 아닌 현재 논리 네트워크를 수정합니다.
  - a. **Compute** → **Clusters** 를 클릭합니다.
  - b. 클러스터 이름을 클릭하여 세부 정보 보기를 엽니다.
  - c. **Logical Networks** 탭에서 현재 논리 네트워크(**net-1**)를 선택하고 **Manage Networks** 를 클릭합니다.

- d. **Require** 확인란을 지우고 확인을 클릭합니다.
2. **VM** 네트워크가 필요하지 않고 필수 가 아닌 새 논리적 네트워크를 생성합니다.
    - a. **Add Network** 를 클릭하여 **New Logical Network** 창을 엽니다.
    - b. 일반 탭에서 이름 (**net-2**)을 입력하고 **VM** 네트워크 확인란을 지웁니다.
    - c. **Cluster** (클러스터) 탭에서 **Require** (필수) 확인란을 지우고 **OK** 를 클릭합니다.
  3. 현재 네트워크 본딩을 제거하고 논리 네트워크를 다시 할당합니다.
    - a. 컴퓨팅 → 호스트.
    - b. 호스트 이름을 클릭하여 세부 정보 보기를 엽니다.
    - c. 네트워크 인터페이스 탭에서 **Setup Host Networks** 를 클릭합니다.
    - d. **net-1** 을 오른쪽으로 드래그하여 할당을 해제합니다.
    - e. 현재 본딩을 오른쪽으로 끌어서 제거합니다.
    - f. **net-1** 및 **net-2** 를 왼쪽으로 드래그하여 물리적 인터페이스에 할당합니다.
    - g. **net-2** 의 언필 아이콘을 클릭하여 **Edit Network** 창을 엽니다.
    - h. **IPV4** 탭에서 고정을 선택합니다.

- i. 서버넷의 **IP** 및 **Netmask/Routing Prefix** 를 입력하고 **OK** 를 클릭합니다.
4. **iSCSI** 본딩을 만듭니다.
    - a. **Compute** → **Data Centers** (데이터 센터)를 클릭합니다.
    - b. 데이터 센터 이름을 클릭하여 세부 정보 보기를 엽니다.
    - c. **iSCSI** 다중 경로 탭에서 추가를 클릭합니다.
    - d. **iSCSI** 본딩 추가 창에서 이름을 입력하고 네트워크 **net-1** 및 **net-2** 를 선택한 다음 확인을 클릭합니다.

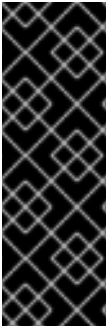
데이터 센터에는 이전 논리 네트워크와 새 논리 네트워크가 포함된 **iSCSI** 본딩이 있습니다.

#### 11.5.5. FCP 스토리지 준비

**Red Hat Virtualization**은 기존 **LUN**으로 구성된 볼륨 그룹에서 스토리지 도메인을 생성하여 **SAN** 스토리지를 지원합니다. 볼륨 그룹이나 **LUN**을 한 번에 두 개 이상의 스토리지 도메인에 연결할 수 없습니다.

**Red Hat Virtualization** 시스템 관리자는 **SAN(Storage Area Networks)** 개념에 대한 작업 지식이 필요합니다. **SAN**은 일반적으로 호스트 및 공유 외부 스토리지 간의 트래픽에 파이버 채널 프로토콜(**FCP**)을 사용합니다. 이러한 이유로 **SAN**은 때때로 **FCP** 스토리지라고 할 수 있습니다.

**Red Hat Enterprise Linux**에서 **FCP** 또는 멀티패스 설정 및 구성에 대한 자세한 내용은 [스토리지 관리 가이드](#) 및 [DM Multipath 가이드](#)를 참조하십시오.



중요

블록 스토리지를 사용하고 있고 원시 장치 또는 직접 LUN에 가상 머신을 배포하고 논리 볼륨 관리자를 사용하여 가상 머신을 관리하려면 게스트 논리 볼륨을 숨기는 필터를 생성해야 합니다. 이렇게 하면 호스트가 부팅될 때 게스트 논리 볼륨이 활성화되지 않습니다. 이 경우 오래된 논리 볼륨이 활성화되고 데이터가 손상될 수 있습니다. 자세한 내용은 <https://access.redhat.com/solutions/2662261> 을 참조하십시오.



중요

**Red Hat Virtualization**은 현재 블록 크기가 **4K**인 블록 스토리지를 지원하지 않습니다. 레저시(**512b** 블록) 모드로 블록 스토리지를 구성해야 합니다.



중요

호스트가 **SAN** 스토리지에서 부팅되고 스토리지 연결이 끊어지면 스토리지 파일 시스템이 읽기 전용으로 되어 연결이 복원된 후에도 이 상태로 유지됩니다.

이러한 상황을 방지하기 위해 **Red Hat**은 부팅 **LUN**에 대해 **SAN**의 루트 파일 시스템에 드롭인 다중 경로 구성 파일을 추가하여 연결이 있을 때 대기열에 있는지 확인하는 것이 좋습니다.

```
# cat /etc/multipath/conf.d/host.conf
multipaths {
  multipath {
    wwid boot_LUN_wwid
    no_path_retry queue
  }
}
```

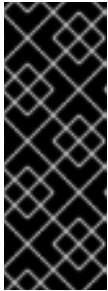
### 11.5.6. FCP 스토리지 추가

다음 절차에서는 기존 **FCP** 스토리지를 **Red Hat Virtualization** 환경에 데이터 도메인으로 연결하는 방법을 설명합니다.

절차

1. 스토리지 도메인을 클릭합니다.
2. **New Domain** 을 클릭합니다.

3. 스토리지 도메인 의 이름을 입력합니다.
4. 드롭다운 목록에서 **FCP** 데이터 센터를 선택합니다.  
  
아직 적절한 **FCP** 데이터 센터가 없는 경우 (없음) 을 선택합니다.
5. 드롭다운 목록에서 도메인 기능 및 스토리지 유형을 선택합니다. 선택한 데이터 센터와 호환되지 않는 스토리지 도메인 유형은 사용할 수 없습니다.
6. **Host (호스트)** 필드에서 활성 호스트를 선택합니다. 데이터 센터의 첫 번째 데이터 도메인이 아닌 경우 데이터 센터의 **host**를 선택해야 합니다.



#### 중요

스토리지 도메인과의 모든 통신은 **Red Hat Virtualization Manager**가 아닌 선택한 호스트를 통해 이루어집니다. 시스템에 하나 이상의 활성 호스트가 있어야 하며 선택한 데이터 센터에 연결해야 합니다. 스토리지 도메인을 구성하려면 모든 호스트에 스토리지 장치에 액세스할 수 있어야 합니다.

7. **New Domain** 창에서 파이버 채널을 스토리지 유형으로 선택할 때 사용되지 않는 **LUN**으로 알려진 대상을 자동으로 표시합니다. 사용 가능한 모든 **LUN**을 선택하려면 **LUN ID** 확인란을 선택합니다.
8. 필요한 경우 고급 매개 변수를 구성할 수 있습니다.
  - a. 고급 매개 변수 를 클릭합니다.
  - b. **Warning Low Space Indicator** 필드에 백분율 값을 입력합니다. 스토리지 도메인에서 사용 가능한 공간이 이 백분율 미만이면 사용자에게 경고 메시지가 표시됩니다.
  - c. **Critical Space Action Blocker** 필드에 **GB** 값을 입력합니다. 스토리지 도메인에서 사용 가능한 공간이 이 값 아래에 있는 경우 사용자에게 오류 메시지가 표시되고, 공간을 사용하는 새 작업이 일시적으로 차단됩니다.
  - d.

**delete** 옵션 이후에 초기화를 활성화하려면 **shape After Delete** 확인란을 선택합니다. 이 옵션은 도메인을 생성한 후 편집할 수 있지만, 이렇게 하면 이미 존재하는 디스크의 삭제 속성 이후 초기화를 변경하지 않습니다.

e.

삭제 후 삭제 옵션을 활성화하려면 **Discard After Delete** 확인란을 선택합니다. 이 옵션은 도메인을 생성한 후 편집할 수 있습니다. 이 옵션은 블록 스토리지 도메인에서만 사용할 수 있습니다.

9.

**OK**를 클릭합니다.

새로운 **FCP** 데이터 도메인은 사용 준비 중인 동안 **Locked** 상태로 유지됩니다. 준비가 되면 자동으로 데이터 센터에 연결됩니다.

### 11.5.7. iSCSI 또는 FCP 스토리지 증가

**iSCSI** 또는 **FCP** 스토리지 크기를 늘리는 방법에는 여러 가지가 있습니다.

- 기존 **LUN**을 현재 스토리지 도메인에 추가합니다.
- 새 **LUN**을 사용하여 새 스토리지 도메인을 생성하고 기존 데이터 센터에 추가합니다. **11.5.2 절. “iSCSI 스토리지 추가”** 을 참조하십시오.
- 기본 **LUN**의 크기를 조정하여 스토리지 도메인을 확장합니다.

**Red Hat Enterprise Linux 7** 시스템에서 **iSCSI** 스토리지 생성, 구성 또는 크기 조정에 대한 자세한 내용은 **Red Hat Enterprise Linux 7 Storage** 관리 가이드를 참조하십시오.

다음 절차에서는 기존 스토리지 도메인에 새 **LUN**을 추가하여 **SAN**(스토리지 영역 네트워크) 스토리지를 확장하는 방법을 설명합니다.

사전 요구 사항

- 스토리지 도메인의 상태는 **UP** 여야 합니다.



- 상태가 **UP** 인 모든 호스트에서 **LUN**에 액세스할 수 있어야 합니다. 그러지 않으면 작업이 실패하고 **LUN**이 도메인에 추가되지 않습니다. 그러나 호스트 자체는 영향을 받지 않습니다. 새로 추가된 호스트 또는 유지 관리가 중단되거나 비작동 상태가 되는 호스트가 **LUN**에 액세스할 수 없는 경우 호스트의 상태가 비작동 상태가 됩니다.

#### 기존 **iSCSI** 또는 **FCP** 스토리지 도메인 증가

1. **Storage** → **Domains** 를 클릭하고 **iSCSI** 또는 **FCP** 도메인을 선택합니다.
2. 도메인 관리를 클릭합니다.
3. **Targets > LUN**을 클릭하고 **Discover Targets** 확장 버튼을 클릭합니다.
4. 스토리지 서버의 연결 정보를 입력하고 **Discover** 를 클릭하여 연결을 시작합니다.
5. **LUN >** 대상을 클릭하고 새로 사용 가능한 **LUN**의 확인란을 선택합니다.
6. **OK** 를 클릭하여 선택한 스토리지 도메인에 **LUN**을 추가합니다.

이렇게 하면 추가된 **LUN** 크기에 따라 스토리지 도메인이 증가합니다.

기본 **LUN** 크기를 조정하여 스토리지 도메인을 확장할 때 관리 포털에서 **LUN**도 새로 고쳐야 합니다.

#### **LUN** 크기 새로 고침

1. **Storage** → **Domains** 를 클릭하고 **iSCSI** 또는 **FCP** 도메인을 선택합니다.
2. 도메인 관리를 클릭합니다.

3. **LUN >** 대상을 클릭합니다.
4. 추가 크기 열에서 **LUN**의 추가\_Storage\_Size 버튼을 클릭하여 새로 고칩니다.
5. 확인을 클릭하여 새 스토리지 크기를 나타내기 위해 **LUN**을 새로 고칩니다.

### 11.5.8. LUN 재사용

스토리지 도메인 또는 가상 디스크를 생성하기 위해 **LUN**을 다시 사용할 수 없습니다. **LUN**을 재사용하려고 하면 관리 포털에 다음과 같은 오류 메시지가 표시됩니다.

*Physical device initialization failed. Please check that the device is empty and accessible by the host.*

셀프 호스트 엔진은 설치 중에 다음 오류를 표시합니다.

```
[ ERROR ] Error creating Volume Group: Failed to initialize physical device: ("
[u'/dev/mapper/00000000000000000000000000000000']",)
[ ERROR ] Failed to execute stage 'Misc configuration': Failed to initialize physical device: ("
[u'/dev/mapper/00000000000000000000000000000000']",)
```

**LUN**을 재사용하려면 먼저 이전 파티션 테이블을 지워야 합니다.

#### LUN에서 파티션 테이블 지우기



중요

실수로 데이터를 삭제하지 않도록 올바른 **LUN**에서 이 절차를 실행해야 합니다.

재사용하려는 **LUN ID**, 한 번에 읽고 쓸 최대 바이트 수, 복사할 입력 블록 수와 함께 **dd** 명령을 실행합니다.

```
# dd if=/dev/zero of=/dev/mapper/LUN_ID bs=1M count=200 oflag=direct
```

### 11.6. RED HAT GLUSTER STORAGE 준비 및 추가

### 11.6.1. Red Hat Gluster Storage 준비

**Red Hat Gluster Storage** 설정 및 구성에 대한 자세한 내용은 [Red Hat Gluster Storage 설치 가이드](#)를 참조하십시오.

**Red Hat Virtualization**에서 지원되는 **Red Hat Gluster Storage** 버전은 <https://access.redhat.com/articles/2356261> 을 참조하십시오.

### 11.6.2. Red Hat Gluster Storage 추가

**Red Hat Virtualization**에서 **Red Hat Gluster Storage**를 사용하려면 **Red Hat Gluster Storage**를 사용한 **Red Hat Virtualization** 구성 을 참조하십시오.

**Red Hat Virtualization**에서 지원되는 **Red Hat Gluster Storage** 버전은 <https://access.redhat.com/articles/2356261> 을 참조하십시오.

## 11.7. 기존 스토리지 도메인 가져오기

### 11.7.1. 기존 스토리지 도메인 가져오기 개요

데이터가 없는 새 스토리지 도메인을 추가하는 것 외에도 기존 스토리지 도메인을 가져와서 포함된 데이터에 액세스할 수 있습니다. 스토리지 도메인을 가져오면 **Manager** 데이터베이스에서 오류가 발생할 경우 데이터를 복구하고 한 데이터 센터 또는 환경에서 다른 데이터 센터로 데이터를 마이그레이션할 수 있습니다.

다음은 각 스토리지 도메인 유형을 가져오는 개요입니다.

#### **data**

기존 데이터 스토리지 도메인을 가져오면 데이터 스토리지 도메인에 포함된 모든 가상 머신 및 템플릿에 액세스할 수 있습니다. 스토리지 도메인을 가져온 후 가상 머신, 부동 디스크 이미지 및 템플릿을 대상 데이터 센터로 수동으로 가져와야 합니다. 데이터 스토리지 도메인에 포함된 가상 머신 및 템플릿을 가져오는 프로세스는 내보내기 스토리지 도메인의 프로세스와 유사합니다. 그러나 데이터 스토리지 도메인에는 지정된 데이터 센터의 모든 가상 머신과 템플릿이 포함되어 있으므로 데이터 스토리지 도메인을 가져오는 것이 좋습니다. 데이터 스토리지 도메인은 데이터 센터 또는 환경 간 가상 머신의 데이터 복구 또는 대규모 마이그레이션에 권장됩니다.



중요

지원되는 올바른 호환성 수준으로 데이터 센터에 연결된 기존 데이터 스토리지 도메인을 가져올 수 있습니다. 자세한 내용은 이전 **RHV** 버전에서 스토리지 도메인 및 가상 머신 가져오기와 관련된 지원 기능 및 제약 조건을 참조하십시오.

ISO

기존 **ISO** 스토리지 도메인을 가져오면 **ISO** 스토리지 도메인에 포함된 모든 **ISO** 파일 및 가상 디스크였으므로 액세스할 수 있습니다. 이러한 리소스에 액세스하기 위해 스토리지 도메인을 가져온 후에는 추가 작업이 필요하지 않습니다. 필요에 따라 가상 머신에 연결할 수 있습니다.

내보내기

기존 내보내기 스토리지 도메인을 가져오면 내보내기 스토리지 도메인에 포함된 모든 가상 머신 이미지 및 템플릿에 액세스할 수 있습니다. 내보내기 도메인은 가상 머신 이미지 및 템플릿을 내보내고 가져오기 위해 설계되었으므로 내보내기 스토리지 도메인을 가져오는 것이 환경 내에서 또는 환경 간에 소수의 가상 머신과 템플릿을 마이그레이션하는 것이 좋습니다. 내보내기 스토리지 도메인에서 가상 머신 및 템플릿을 내보내고 가져오는 방법에 대한 자세한 내용은 가상 머신 관리 가이드의 **가상 머신 및 템플릿 내보내기 및 가져오기**를 참조하십시오.



참고

내보내기 스토리지 도메인은 더 이상 사용되지 않습니다. 스토리지 데이터 도메인은 데이터 센터에서 연결 해제되어 동일한 환경 또는 다른 환경의 다른 데이터 센터로 가져올 수 있습니다. 그런 다음 가져온 스토리지 도메인에서 연결된 데이터 센터로 가상 머신, 유동 가상 디스크 및 템플릿을 업로드할 수 있습니다.



주의

스토리지 도메인을 대상 **Data-Center**에 연결하면 최신 스토리지 도메인 형식으로 업그레이드할 수 있으며 소스 **Data-Center**에 다시 연결할 수 없습니다. 이렇게 하면 내보내기 도메인 대신 데이터 도메인을 사용하지 않습니다.

11.7.2. 스토리지 도메인 가져오기

이전에 동일한 환경 또는 다른 환경에서 데이터 센터에 연결된 스토리지 도메인을 가져옵니다. 이 절차에서는 데이터 손상을 방지하기 위해 스토리지 도메인이 모든 환경의 데이터 센터에 더 이상 연결되지

않은 것으로 가정합니다. 기존 데이터 스토리지 도메인을 데이터 센터에 가져와서 연결하려면 대상 데이터 센터를 초기화해야 합니다.

#### 절차

1. 스토리지 도메인을 클릭합니다.
2. **Import Domain** 을 클릭합니다.
3. 스토리지 도메인을 가져올 데이터 센터를 선택합니다.
4. 스토리지 도메인 의 이름을 입력합니다.
5. 드롭다운 목록에서 **Domain Function and Storage Type** 을 선택합니다.
6. 호스트 드롭다운 목록에서 호스트를 선택합니다.



#### 중요

스토리지 도메인과의 모든 통신은 **Red Hat Virtualization Manager**가 아닌 선택한 호스트를 통해 이루어집니다. 시스템에 하나 이상의 활성 호스트가 있어야 하며 선택한 데이터 센터에 연결해야 합니다. 스토리지 도메인을 구성하려면 모든 호스트에 스토리지 장치에 액세스할 수 있어야 합니다.

7. 스토리지 도메인의 세부 정보를 입력합니다.



#### 참고

**Domain Function and Storage Type** 목록에서 선택한 값에 따라 스토리지 도메인 변경의 세부 정보를 지정하는 필드입니다. 이러한 필드는 새 스토리지 도메인을 추가하는 데 사용할 수 있는 필드와 동일합니다.

8. **Activate Domain in Data Center** (데이터 센터에서 도메인 활성화) 확인란을 선택하여 선택한 데이터 센터에 스토리지 도메인을 연결한 후 스토리지 도메인을 활성화합니다.

9.

**OK**를 클릭합니다.

이제 스토리지 도메인에서 데이터 센터로 가상 머신 및 템플릿을 가져올 수 있습니다.



주의

스토리지 도메인을 대상 **Data-Center**에 연결하면 최신 스토리지 도메인 형식으로 업그레이드할 수 있으며 소스 **Data-Center**에 다시 연결할 수 없습니다. 이렇게 하면 내보내기 도메인 대신 데이터 도메인을 사용하지 않습니다.

#### 관련 정보

- [11.7.5절. “가져온 데이터 스토리지 도메인에서 가상 머신 가져오기”](#)
- [11.7.6절. “가져온 데이터 스토리지 도메인에서 템플릿 가져오기”](#)

#### 11.7.3. 동일한 환경에서 데이터 센터 간 스토리지 도메인 마이그레이션

동일한 **Red Hat Virtualization** 환경에서 스토리지 도메인을 한 데이터 센터에서 다른 데이터 센터로 마이그레이션하여 대상 데이터 센터에서 스토리지 도메인에 포함된 데이터에 액세스할 수 있습니다. 이 절차에서는 스토리지 도메인을 하나의 데이터 센터에서 분리하고 다른 데이터 센터에 연결하는 작업이 포함됩니다.

#### 절차

1. 필수 스토리지 도메인에서 실행 중인 모든 가상 머신을 종료합니다.
2. 스토리지 도메인을 클릭합니다.
3. 스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.

4. 데이터 센터 탭을 클릭합니다.
5. 유지 관리를 클릭한 다음 확인을 클릭합니다.
6. 분리 를 클릭한 다음 확인을 클릭합니다.
7. **Attach** 를 클릭합니다.
8. 대상 데이터 센터를 선택하고 확인을 클릭합니다.

스토리지 도메인은 대상 데이터 센터에 연결되고 자동으로 활성화됩니다. 이제 스토리지 도메인에서 대상 데이터 센터로 가상 머신 및 템플릿을 가져올 수 있습니다.



#### 주의

스토리지 도메인을 대상 **Data-Center**에 연결하면 최신 스토리지 도메인 형식으로 업그레이드할 수 있으며 소스 **Data-Center**에 다시 연결할 수 없습니다. 이렇게 하면 내보내기 도메인 대신 데이터 도메인을 사용하지 않습니다.

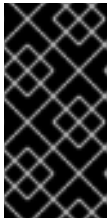
#### 11.7.4. 다른 환경에서 데이터 센터 간 스토리지 도메인 마이그레이션

대상 환경이 스토리지 도메인에 포함된 데이터에 액세스할 수 있도록 한 **Red Hat Virtualization** 환경에서 다른 **Red Hat Virtualization** 환경으로 스토리지 도메인을 마이그레이션합니다. 다음 절차에서는 하나의 **Red Hat Virtualization** 환경에서 스토리지 도메인을 제거하고 다른 환경으로 가져오는 작업을 수행합니다. 기존 데이터 스토리지 도메인을 **Red Hat Virtualization** 데이터 센터에 가져와서 연결하려면 스토리지 도메인의 소스 데이터 센터에 올바른 호환성 수준이 있어야 합니다. 자세한 내용은 [이전 RHV 버전](#)에서 스토리지 도메인 및 가상 머신 가져오기와 관련된 지원 기능 및 제약 조건을 참조하십시오.

#### 절차

1. 소스 환경의 관리 포털에 로그인합니다.

2. 필수 스토리지 도메인에서 실행 중인 모든 가상 머신을 종료합니다.
3. 스토리지 도메인을 클릭합니다.
4. 스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.
5. 데이터 센터 탭을 클릭합니다.
6. 유지 관리를 클릭합니다.



중요

**Ignore OVF** 업데이트 실패 확인란을 확인하지 마십시오. 스토리지 도메인의 유지 관리 작업에서 **OVF**를 업데이트해야 합니다.

**OK**를 클릭합니다.

7. 분리 를 클릭한 다음 확인을 클릭합니다.
8. 제거를 클릭합니다.
9. 스토리지 제거 창에서 형식 도메인( 예: **Storage Content**이 손실됨!) 확인란이 선택되지 않았는지 확인합니다. 이 단계에서는 나중에 사용할 수 있도록 스토리지 도메인의 데이터를 유지합니다.
10. **OK** 를 클릭하여 소스 환경에서 스토리지 도메인을 제거합니다.
11. 대상 환경의 관리 포털에 로그인합니다.
12. 스토리지 도메인을 클릭합니다.



13. **Import Domain** 을 클릭합니다.
14. 데이터 센터 드롭다운 목록에서 대상 데이터 센터를 선택합니다.
15. 스토리지 도메인의 이름을 입력합니다.
16. 적절한 드롭다운 목록에서 **Domain Function and Storage Type** 을 선택합니다.
17. 호스트 드롭다운 목록에서 호스트를 선택합니다.
18. 스토리지 도메인의 세부 정보를 입력합니다.



## 참고

스토리지 유형 드롭다운 목록에서 선택한 값에 따라 스토리지 도메인 변경의 세부 정보를 지정하는 필드입니다. 이러한 필드는 새 스토리지 도메인을 추가하는 데 사용할 수 있는 필드와 동일합니다.

19. **Activate Domain in Data Center** (데이터 센터에서 도메인 활성화) 확인란을 선택하여 스토리지 도메인을 자동으로 활성화합니다.
20. **OK**를 클릭합니다.

스토리지 도메인은 새 **Red Hat Virtualization** 환경의 대상 데이터 센터에 연결되며 자동으로 활성화됩니다. 가져온 스토리지 도메인에서 대상 데이터 센터로 가상 머신 및 템플릿을 가져올 수 있습니다.



## 주의

스토리지 도메인을 대상 **Data-Center**에 연결하면 최신 스토리지 도메인 형식으로 업그레이드할 수 있으며 소스 **Data-Center**에 다시 연결할 수 없습니다. 이렇게 하면 내보내기 도메인 대신 데이터 도메인을 사용하지 않습니다.

### 11.7.5. 가져온 데이터 스토리지 도메인에서 가상 머신 가져오기

**Red Hat Virtualization** 환경으로 가져온 데이터 스토리지 도메인에서 하나 이상의 클러스터로 가상 머신을 가져옵니다. 이 절차에서는 가져온 데이터 스토리지 도메인이 데이터 센터에 연결되었으며 활성화 되었다고 가정합니다.

#### 절차

1. 스토리지 도메인을 클릭합니다.
2. 가져온 스토리지 도메인 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **VM Import** 탭을 클릭합니다.
4. 가져올 하나 이상의 가상 머신을 선택합니다.
5. 가져오기를 클릭합니다.
6. **Import Virtual Machine(s) (가상 머신 가져오기)** 창의 각 가상 머신에 대해 클러스터 목록에서 올바른 대상 클러스터가 선택되어 있는지 확인합니다.
7. 외부 가상 머신 **vNIC** 프로필을 대상 클러스터에 있는 프로필에 매핑합니다.
  - a. **vNic Profiles** 매핑을 클릭합니다.

- b. 대상 **vNic Profile** 드롭다운 목록에서 사용할 **vNIC** 프로필을 선택합니다.
- c. **Import Virtual Machine(s)** 창에서 여러 대상 클러스터를 선택한 경우 대상 클러스터 드롭다운 목록에서 각 대상 클러스터를 선택하고 매핑이 올바른지 확인합니다.
- d. **OK**를 클릭합니다.

8. **MAC** 주소 충돌이 감지되면 가상 머신 이름 옆에 느낌표가 표시됩니다. 아이콘 위로 마우스를 이동하여 발생한 오류 유형을 표시하는 툴팁을 봅니다.

**ReassignECDHE MACs** 확인란을 선택하여 새 **MAC** 주소를 모든 문제가 있는 가상 머신에 다시 할당합니다. 또는 가상 시스템당 **Reassign (재할당)** 확인란을 선택할 수 있습니다.



참고

할당할 수 있는 주소가 없으면 가져오기 작업이 실패합니다. 그러나 클러스터의 **MAC** 주소 풀 범위를 벗어나는 **MAC** 주소의 경우 새 **MAC** 주소를 다시 할당하지 않고 가상 머신을 가져올 수 있습니다.

9. **OK**를 클릭합니다.

가져온 가상 머신이 **VM** 가져오기 탭의 목록에 더 이상 표시되지 않습니다.

#### 11.7.6. 가져온 데이터 스토리지 도메인에서 템플릿 가져오기

**Red Hat Virtualization** 환경으로 가져온 데이터 스토리지 도메인에서 템플릿을 가져옵니다. 이 절차에서는 가져온 데이터 스토리지 도메인이 데이터 센터에 연결되었으며 활성화되었다고 가정합니다.

절차

1. 스토리지 도메인을 클릭합니다.
2. 가져온 스토리지 도메인 이름을 클릭하여 세부 정보 보기를 엽니다.

3. **템플릿 가져오기** 탭을 클릭합니다.
4. 가져올 하나 이상의 템플릿을 선택합니다.
5. **가져오기** 를 클릭합니다.
6. **Import Templates(s)** 창의 각 템플릿에 대해 **Cluster** 목록에서 올바른 대상 클러스터가 선택되어 있는지 확인합니다.
7. 외부 가상 머신 **vNIC** 프로필을 대상 클러스터에 있는 프로필에 매핑합니다.
  - a. **vNic Profiles** 매핑 을 클릭합니다.
  - b. 대상 **vNic Profile** 드롭다운 목록에서 사용할 **vNIC** 프로필을 선택합니다.
  - c. **Import Templates** 창에서 여러 대상 클러스터를 선택한 경우 대상 클러스터 드롭다운 목록에서 각 대상 클러스터를 선택하고 매핑이 올바른지 확인합니다.
  - d. **OK**를 클릭합니다.
8. **OK**를 클릭합니다.

가져온 템플릿은 **Template Import (템플릿 가져오기)** 탭의 목록에 더 이상 표시되지 않습니다.

## 11.8. 스토리지 작업

### 11.8.1. 데이터 스토리지 도메인에 이미지 업로드

관리 포털 또는 **REST API**를 사용하여 가상 디스크 이미지와 **ISO** 이미지를 데이터 스토리지 도메인에 업로드할 수 있습니다.



## 참고

**REST API**를 사용하여 이미지를 업로드하려면 **REST API 가이드**의 **IMAGETRANSFERS** 및 **IMAGETRANSFER** 를 참조하십시오.

**QEMU** 호환 가상 디스크를 가상 머신에 연결할 수 있습니다. 가상 디스크 유형은 **QCOW2** 또는 **raw**여야 합니다. **QCOW2** 가상 디스크에서 생성된 디스크는 공유할 수 없으며 **QCOW2** 가상 디스크 파일에 백업 파일이 없어야 합니다.

**ISO** 이미지는 가상 머신에 **CDROM**으로 연결하거나 가상 머신을 부팅하는 데 사용할 수 있습니다.

## 사전 요구 사항

업로드 함수는 **HTML 5 API**를 사용하며, 이를 사용하려면 환경에 다음이 필요합니다.

- **engine-setup** 으로 구성된 이미지 I/O 프록시( **ovirt-imageio-proxy** ). 자세한 내용은 **Red Hat Virtualization Manager** 구성 을 참조하십시오.
- 관리 포털에 액세스하는 데 사용되는 웹 브라우저로 가져온 인증 기관.  
  
인증 기관을 가져오려면 **https://engine\_address/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA** 를 찾아 모든 신뢰 설정을 활성화합니다. **Firefox, Internet Explorer** 또는 **Google Chrome** 에 인증 기관을 설치하려면 지침을 참조하십시오.
- **Firefox 35, Internet Explorer 10, Chrome 13** 이상과 같은 **HTML 5**를 지원하는 브라우저입니다.

## 데이터 스토리지 도메인에 이미지 업로드

1. 스토리지 디스크를 클릭합니다.
2. 업로드 메뉴에서 **Start** 를 선택합니다.

3. **Deploy File (파일 이름)**을 클릭하고 업로드할 이미지를 선택합니다.

4. 디스크 옵션 필드를 작성합니다. 관련 필드에 대한 자세한 내용은 [13.6.2절](#) “**새 가상 디스크 창에서 설정에 대한 설명**”을 참조하십시오.

5. **OK**를 클릭합니다.

진행률 표시줄은 업로드 상태를 나타냅니다. 업로드 메뉴에서 업로드를 일시 중지, 취소 또는 재개할 수 있습니다.

### 업로드 시간 제한 값 증가

1. 업로드 시간이 초과되고 메시지가 표시되면 전송 비활성으로 인한 이유: 시간 초과를 늘리면 시간 초과 값을 늘립니다.

```
# engine-config -s TransferImageClientInactivityTimeoutInSeconds=6000
```

2. **ovirt-engine** 서비스를 다시 시작합니다.

```
# systemctl restart ovirt-engine
```

### 11.8.2. 스토리지 도메인을 유지 관리 모드로 이동

스토리지 도메인을 분리하고 제거하려면 스토리지 도메인을 유지보수 모드에 있어야 합니다. 이는 다른 데이터 도메인을 마스터 데이터 도메인으로 다시 설계하는 데 필요합니다.



#### 중요

가상 머신에 스토리지 도메인에 임대(lease)가 있는 경우 스토리지 도메인을 유지 관리 모드로 이동할 수 없습니다. 가상 머신을 종료하거나 리스를 먼저 제거하거나 다른 스토리지 도메인으로 이동해야 합니다. 가상 머신 리스에 대한 자세한 내용은 [가상 머신 관리 가이드](#)를 참조하십시오.

더 많은 **LUN**을 추가하여 **iSCSI** 도메인을 확장하는 것은 도메인이 활성화된 경우에만 수행할 수 있습니다.

### 스토리지 도메인을 유지 관리 모드로 이동

1. 스토리지 도메인에서 실행 중인 모든 가상 머신을 종료합니다.
2. 스토리지 도메인을 클릭합니다.
3. 스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.
4. 데이터 센터 탭을 클릭합니다.
5. 유지 관리를 클릭합니다.



참고

**Ignore OVF** 업데이트 실패 확인란을 사용하면 **OVF** 업데이트가 실패하더라도 스토리지 도메인을 유지 관리 모드로 전환할 수 있습니다.

6. **OK**를 클릭합니다.

스토리지 도메인이 비활성화되고 결과 목록에 비활성 상태가 있습니다. 데이터 센터에서 비활성 스토리지 도메인을 편집, 분리, 제거 또는 다시 활성화할 수 있습니다.



참고

연결된 데이터 센터의 세부 정보 보기에서 **Storage** 탭을 사용하여 도메인을 유지 관리 모드로 활성화, 분리 및 배치할 수도 있습니다.

#### 11.8.3. 스토리지 도메인 편집

관리 포털을 통해 스토리지 도메인 매개변수를 편집할 수 있습니다. 스토리지 도메인 상태(활성 또는 비활성)에 따라 다른 필드를 편집할 수 있습니다. **Data Center, Domain Function, Storage Type, Format** 과 같은 필드는 변경할 수 없습니다.

- 활성 상태:** 스토리지 도메인이 활성 상태인 경우 이름, 설명 설명, 주석, 경고 낮음 공간 표시 (%), 중요 공간 작업 블록 기 (GB), 삭제 후 삭제 후 삭제 후 카드 필드 삭제 후 삭제 취소 필드가 편집할 수 있습니다. **Name** 필드는 스토리지 도메인이 활성 상태인 경우에만 편집할 수 있습니다. 스토리지 도메인이 비활성화된 동안 다른 모든 필드를 편집할 수도 있습니다.
- 비활성:** 스토리지 도메인이 유지 관리 모드 또는 연결 해제 중이면 비활성 상태인 경우 이름, 데이터 센터, 도메인 기능, 스토리지 유형 및 형식을 제외한 모든 필드를 편집할 수 있습니다. 스토리지 연결, 마운트 옵션 및 기타 고급 매개 변수를 편집하려면 스토리지 도메인을 비활성화해야 합니다. 이는 **NFS, POSIX** 및 로컬 스토리지 유형에서만 지원됩니다.



참고

**iSCSI** 스토리지 연결은 관리 포털을 통해 편집할 수 없지만 **REST API**를 통해 편집할 수 있습니다. **REST API** 가이드의 [스토리지 연결](#) 업데이트를 참조하십시오.

활성 스토리지 도메인 편집

- Storage** → **Domains** 를 클릭하고 스토리지 도메인을 선택합니다.
- 도메인 관리를 클릭합니다.
- 필요에 따라 사용 가능한 필드를 편집합니다.
- OK**를 클릭합니다.

비활성 스토리지 도메인 편집

- 스토리지 도메인을 클릭합니다.
- 스토리지 도메인이 활성화된 경우 유지보수 모드로 이동합니다.

  - 스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.




- b. 데이터 센터 탭을 클릭합니다.
  - c. 유지 관리를 클릭합니다.
  - d. **OK**를 클릭합니다.
3. 도메인 관리를 클릭합니다.
  4. 필요에 따라 스토리지 경로 및 기타 세부 정보를 편집합니다. 새 연결 세부 정보는 원래 연결과 동일한 스토리지 유형이어야 합니다.
  5. **OK**를 클릭합니다.
  6. 스토리지 도메인을 활성화합니다.
    - a. 스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.
    - b. 데이터 센터 탭을 클릭합니다.
    - c. 활성화 를 클릭합니다.

#### 11.8.4. OVF 업데이트

기본적으로 **OVF**는 60분마다 업데이트됩니다. 그러나 중요한 가상 머신을 가져오거나 중요한 업데이트를 수행한 경우 **OVF**를 수동으로 업데이트할 수 있습니다.

##### **OVF** 업데이트

1. 스토리지 도메인을 클릭합니다.
- 2.

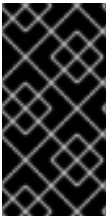
스토리지 도메인을 선택하고 **More Actions** (  )을 클릭한 다음 **OVF** 업데이트를 클릭합니다.

**OVFs**가 업데이트되고 이벤트가 이벤트에 표시됩니다.

### 11.8.5. 유지 관리 모드에서 스토리지 도메인 활성화

데이터 센터의 스토리지를 변경한 경우 스토리지 도메인을 유지 관리 모드로 설정해야 합니다. 스토리지 도메인을 활성화하여 사용을 재개합니다.

1. 스토리지 도메인을 클릭합니다.
2. 비활성 스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 데이터 센터 탭을 클릭합니다.
4. 활성화 를 클릭합니다.



#### 중요

데이터 도메인을 활성화하기 전에 **ISO** 도메인 활성화를 시도하는 경우 오류 메시지가 표시되고 도메인이 활성화되지 않습니다.

### 11.8.6. 데이터 센터에서 스토리지 도메인 분리

스토리지 도메인을 한 데이터 센터에서 분리하여 다른 데이터 센터로 마이그레이션합니다.

데이터 센터에서 스토리지 도메인 분리

1. 스토리지 도메인을 클릭합니다.
2. 스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.

3. 데이터 센터 탭을 클릭합니다.
4. 유지 관리를 클릭합니다.
5. 확인 을 클릭하여 유지보수 모드를 시작합니다.
6. 분리 를 클릭합니다.
7. **OK** 를 클릭하여 스토리지 도메인을 분리합니다.

스토리지 도메인이 데이터 센터에서 분리되었으며 다른 데이터 센터에 연결할 준비가 되었습니다.

#### 11.8.7. 데이터 센터에 스토리지 도메인 연결

스토리지 도메인을 데이터 센터에 연결합니다.

데이터 센터에 스토리지 도메인 연결

1. 스토리지 도메인을 클릭합니다.
2. 스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 데이터 센터 탭을 클릭합니다.
4. **Attach** 를 클릭합니다.
5. 적절한 데이터 센터를 선택합니다.

6.

**OK**를 클릭합니다.

스토리지 도메인은 데이터 센터에 연결되고 자동으로 활성화됩니다.

### 11.8.8. 스토리지 도메인 제거

가상화 환경에서 삭제할 데이터 센터에 스토리지 도메인이 있습니다.

#### 절차

1.

스토리지 도메인을 클릭합니다.

2.

스토리지 도메인을 유지 관리 모드로 이동하고 분리합니다.

a.

스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.

b.

데이터 센터 탭을 클릭합니다.

c.

유지 관리를 클릭한 다음 확인을 클릭합니다.

d.

분리 를 클릭한 다음 확인을 클릭합니다.

3.

제거를 클릭합니다.

4.

선택적으로 형식 도메인 (예: **Storage Content will be lost!**) 확인란을 선택하여 도메인의 콘텐츠를 해제합니다.

5.


**OK**를 클릭합니다.

스토리지 도메인은 환경에서 영구적으로 제거됩니다.

### 11.8.9. 스토리지 도메인 삭제

정상적인 절차를 통해 오류가 발생하는 스토리지 도메인을 제거하지 못할 수 있습니다. 스토리지 도메인을 강제로 삭제하면 가상화된 환경에서 스토리지 도메인이 제거됩니다.

#### 스토리지 도메인 삭제

1. 스토리지 도메인을 클릭합니다.
2. 스토리지 도메인을 선택하고 **More Actions** (  )를 클릭한 다음 **Destroy** 를 클릭합니다.
3. 승인 작업 확인란을 선택합니다.
4. **OK**를 클릭합니다.

### 11.8.10. 디스크 프로필 생성

디스크 프로필은 스토리지 도메인의 가상 디스크에 대한 최대 처리량 수준과 최대 수준의 입력 및 출력 작업을 정의합니다. 디스크 프로필은 데이터 센터에 정의된 스토리지 프로필을 기반으로 생성되며 프로필을 적용하려면 개별 가상 디스크에 수동으로 할당해야 합니다.

이 절차에서는 스토리지 도메인이 속한 데이터 센터 아래에 하나 이상의 스토리지 품질의 서비스 항목을 이미 정의했다고 가정합니다.

#### 디스크 프로필 생성

1. 스토리지 도메인을 클릭합니다.
2. 데이터 스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.

3. 디스크 프로필 탭을 클릭합니다.
4. 새로 생성을 클릭합니다.
5. 디스크 프로필에 대한 이름 및 설명을 입력합니다.
6. **QoS** 목록에서 디스크 프로필에 적용할 서비스 품질을 선택합니다.
7. **OK**를 클릭합니다.

### 11.8.11. 디스크 프로필 제거

**Red Hat Virtualization** 환경에서 기존 디스크 프로필을 제거합니다.

#### 디스크 프로필 제거

1. 스토리지 도메인을 클릭합니다.
2. 데이터 스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 디스크 프로필 탭을 클릭합니다.
4. 제거할 디스크 프로필을 선택합니다.
5. 제거를 클릭합니다.
6. **OK**를 클릭합니다.

디스크 프로필이 가상 디스크에 할당되면 해당 가상 디스크에서 디스크 프로필이 제거됩니다.


### 11.8.12. 스토리지 도메인의 상태 보기


스토리지 도메인에는 일반 상태 외에도 외부 상태가 있습니다. 외부 상태는 플러그인 또는 외부 시스템에 의해 보고되거나 관리자가 설정하며 스토리지 도메인 이름 왼쪽에 다음 아이콘 중 하나로 표시됩니다.

- **OK:** 아이콘 없음

- **info:**  


- **경고:**  


- **오류:**  


- **실패:**  


스토리지 도메인의 상태에 대한 세부 정보를 보려면 스토리지 도메인 이름을 클릭하여 세부 정보 보기를 열고 이벤트 탭을 클릭합니다.

스토리지 도메인의 상태도 **REST API**를 사용하여 볼 수 있습니다. 스토리지 도메인의 **GET** 요청에는 상태가 포함된 **external\_status** 요소가 포함됩니다.

이벤트 컬렉션을 통해 **REST API**에서 스토리지 도메인의 상태를 설정할 수 있습니다. 자세한 내용은 **REST API 가이드**에 [이벤트 추가](#)를 참조하십시오.

### 11.8.13. 스토리지 도메인에 대한 삭제 후 카드 설정

**Discard After Delete** 확인란을 선택하면 논리 볼륨에서 **blkdiscard** 명령이 호출되고 블록이 사용 가

능하도록 기본 스토리지에 알림이 표시됩니다. 스토리지 어레이는 해제된 공간을 사용하고 요청 시 할당할 수 있습니다. 삭제 후 삭제는 블록 스토리지에서만 작동합니다. **Red Hat Virtualization Manager**에서는 파일 스토리지(예: **NFS**)에서 플래그를 사용할 수 없습니다.

제한 사항:

- 삭제 후 삭제는 **iSCSI** 또는 파이버 채널과 같은 블록 스토리지 도메인에서만 사용할 수 있습니다.
- 기본 스토리지에서 **Discard** 를 지원해야 합니다.

삭제 후 블록 스토리지 도메인을 생성하거나 블록 스토리지 도메인을 편집할 때 모두 활성화할 수 있습니다. **블록 스토리지 준비 및 추가**를 참조하십시오. 스토리지 **도메인 편집**.

#### 11.8.14. 250개 이상의 호스트가 있는 환경에서 **4K** 지원 활성화

기본적으로 **GlusterFS** 도메인 및 로컬 스토리지 도메인은 **Red Hat Virtualization** 환경에서 최대 **250**개의 호스트가 있는 **4K** 블록 크기를 지원합니다. **4K** 블록 크기는 특히 큰 파일을 사용할 때 더 나은 성능을 제공할 수 있으며 **VDO**와 같은 **4K** 호환성이 필요한 경우에도 필요합니다.

**Sanlock**이 할당하는 잠금 공간은 최대 호스트 수가 **250**인 경우 **1MB**입니다. **4K** 스토리지를 사용할 때 최대 호스트 수를 늘리면 잠금 공간 영역이 커집니다. 예를 들어, **2000** 호스트를 사용할 때 잠금 공간 영역은 **8MB**만큼 클 수 있습니다.

엔진 구성 매개변수 **MaxNumberOfHostsInStoragePool** 을 설정하여 **250**개 이상의 호스트가 있는 환경에서 **4K** 블록 지원을 활성화할 수 있습니다.

절차

1. **Manager** 시스템에서 필요한 최대 호스트 수를 활성화합니다.

```
# engine-config -s MaxNumberOfHostsInStoragePool=NUMBER_OF_HOSTS
```

2. **JBoss Application Server**를 다시 시작하십시오.

```
# service jboss-as restart
```



예를 들어 호스트가 **300**개인 클러스터가 있는 경우 다음을 입력합니다.

```
# engine-config -s MaxNumberOfHostsInStoragePool=300
# service jboss-as restart
```

검증

**Manager**에서 **MaxNumberOfHostsInStoragePool** 매개변수 값을 확인합니다.

```
# engine-config --get=MaxNumberOfHostsInStoragePool
MaxNumberOfHostsInStoragePool: 250 version: general
```

### 11.8.15. 4K 지원 비활성화

기본적으로 **GlusterFS** 도메인 및 로컬 스토리지 도메인은 **4K** 블록 크기를 지원합니다. **4K** 블록 크기는 특히 큰 파일을 사용할 때 더 나은 성능을 제공할 수 있으며 **VDO**와 같은 **4K** 호환성이 필요한 경우에도 필요합니다.

**4K** 블록 지원을 비활성화할 수 있습니다.

절차

1. **4K** 블록 지원이 활성화되어 있는지 확인합니다.

```
$ vdsm-client Host getCapabilities
...
{
  "GLUSTERFS" : [
    0,
    512,
    4096,
  ]
  ...
}
```

2. **/etc/vdsm/vdsm.conf.d/gluster.conf** 를 편집하고 **enable\_4k\_storage** 를 **false** 로 설정합니다. 예를 들면 다음과 같습니다.

```
$ vi /etc/vdsm/vdsm.conf.d/gluster.conf

[gluster]
```

```
# Use to disable 4k support  
# if needed.  
enable_4k_storage = false
```

## 12장. 풀

### 12.1. 가상 머신 풀 소개

가상 머신 풀은 동일한 템플릿의 모든 복제본이며 지정된 그룹의 모든 사용자가 필요에 따라 사용할 수 있는 가상 머신 그룹입니다. 관리자는 가상 머신 풀을 통해 사용자를 위한 일반 가상 머신 세트를 빠르게 구성할 수 있습니다.

사용자는 풀에서 가상 머신을 가져와서 가상 머신 풀에 액세스합니다. 사용자가 풀에서 가상 머신을 가져올 때 사용 가능한 경우 풀의 가상 머신 중 하나가 제공됩니다. 해당 가상 머신은 풀이 기반으로 하는 템플릿의 운영 체제 및 구성과 동일하지만 가상 머신을 사용할 때마다 풀의 동일한 멤버를 받지 못할 수 있습니다. 또한 사용자는 해당 풀의 구성에 따라 동일한 가상 머신 풀에서 여러 가상 머신을 가져올 수 있습니다.

가상 머신 풀은 기본적으로 상태 비저장이므로 가상 머신 데이터 및 구성 변경 사항은 재부팅해도 유지되지 않습니다. 그러나 풀을 상태 저장으로 구성할 수 있으므로 이전 사용자가 변경한 사항을 유지할 수 있습니다. 그러나 사용자가 가상 머신 풀에서 가져온 가상 머신에 대한 콘솔 옵션을 구성하는 경우 해당 옵션은 해당 가상 머신 풀에 대해 해당 사용자의 기본값으로 설정됩니다.



#### 참고

풀에서 가져온 가상 머신은 관리 포털에서 액세스할 때 상태 비저장이 아닙니다. 필요한 경우 관리자가 디스크에 변경 사항을 쓸 수 있어야 하기 때문입니다.

기본적으로 풀의 가상 머신은 사용자가 수행할 때 시작되고 사용자가 완료되면 종료됩니다. 그러나 가상 머신 풀에는 사전 시작 가상 머신이 포함될 수도 있습니다. 사전 시작된 가상 머신은 **up** 상태로 유지되며 사용자가 수행할 때까지 유휴 상태로 유지됩니다. 이를 통해 사용자는 이러한 가상 머신을 즉시 사용할 수 있지만 이러한 가상 머신은 유휴 상태이므로 사용하지 않는 경우에도 시스템 리소스를 사용할 수 있습니다.

### 12.2. 가상 머신 풀 생성

공통 템플릿을 기반으로 여러 가상 시스템이 포함된 가상 머신 풀을 생성할 수 있습니다. 가상 머신 노출 및 **템플릿** 생성에 대한 자세한 내용은 가상 머신 관리 가이드의 템플릿을 참조하십시오.

**Windows** 가상 머신용 **sysprep** 파일 구성 옵션

요구 사항에 따라 여러 **sysprep** 파일 구성 옵션을 사용할 수 있습니다.

풀에서 도메인에 연결할 필요가 없는 경우 `/usr/share/ovirt-engine/conf/sysprep/`에 있는 기본 **sysprep** 파일을 사용할 수 있습니다.

풀이 도메인에 참여해야 하는 경우 각 **Windows** 운영 체제에 대해 사용자 지정 **sysprep** 을 생성할 수 있습니다.

1. `/usr/share/ovirt-engine/conf/osinfo-defaults.properties`의 각 운영 체제에 대한 관련 섹션을 새 파일로 복사하고 **99-defaults.properties**로 저장합니다.
2. **99-defaults.properties**에서 **Windows** 제품 활성화 키와 새로운 사용자 지정 **sysprep** 파일의 경로를 지정하십시오.

```
os.operating_system.productKey.value=Windows_product_activation_key
...
os.operating_system.sysprepPath.value =
${ENGINE_USR}/conf/sysprep/sysprep.operating_system
```

3. 도메인, 도메인 암호 및 도메인 관리자를 지정하여 새 **sysprep** 파일을 생성합니다.

```
<Credentials>
  <Domain>AD_Domain</Domain>
  <Password>Domain_Password</Password>
  <Username>Domain_Administrator</Username>
</Credentials>
```

다른 **Windows** 가상 머신 풀에 대해 다른 **sysprep** 설정을 구성해야 하는 경우 관리 포털에서 사용자 지정 **sysprep** 파일을 만들 수 있습니다(아래 가상 머신 풀 생성 참조). 자세한 내용은 [Using Sysprep to Automate the Configuration of Virtual Machines in the Virtual Machine Guide](#)에서 참조하십시오.

가상 머신 풀 생성

1. **Compute** → **Pools** 를 클릭합니다.
2. 새로 생성을 클릭합니다.

3. 드롭다운 목록에서 클러스터 를 선택합니다.
4. 드롭다운 메뉴에서 템플릿 및 버전을 선택합니다. 템플릿은 풀의 모든 가상 시스템에 대한 표준 설정을 제공합니다.
5. 드롭다운 목록에서 운영 체제 를 선택합니다.
6. **Desktop** 또는 **Server** 에 대해 가상 머신을 최적화하려면 **drop-down** 목록을 사용합니다.



#### 참고

고성능 가상 머신이 단일 호스트 및 구체적인 리소스에 고정되므로 풀에 고성능 최적화는 권장되지 않습니다. 이러한 구성으로 여러 가상 머신이 포함된 풀은 제대로 실행되지 않습니다.

7. 이름 및 설명 (선택 사항)을 입력합니다.

풀의 이름은 숫자 접미사와 함께 풀의 각 가상 머신에 적용됩니다. 자리 표시자로 **?** 을 사용하여 가상 머신의 번호를 사용자 지정할 수 있습니다.

#### 예 12.1. 풀 이름 및 가상 머신 번호 지정 예

- **Pool: MyPool**  
가상 머신: **MyPool-1, MyPool -2, ... MyPool-10**
- **풀: MyPool-?**  
가상 머신: **MyPool-001, MyPool-002, ... MyPool-010**

8. 풀의 **VM** 수를 입력합니다.

9. **Prestarted** 필드에 사전 시작할 가상 머신 수를 입력합니다.
10. 단일 사용자가 세션에서 실행할 수 있는 사용자당 최대 **VM** 수를 선택합니다. 최소 값은 **1** 입니다.
11. 삭제 보호 확인란을 선택하여 삭제 보호를 활성화합니다.
12. **Windows**가 아닌 가상 머신 풀을 생성하거나 기본 **sysprep** 을 사용하는 경우 이 단계를 건너 뛴니다. **Windows** 가상 머신 풀에 대한 사용자 지정 **sysprep** 파일을 생성하는 경우:
  - a. **Show Advanced Options** 버튼을 클릭합니다.
  - b. **Initial Run (최초 실행)** 탭을 클릭하고 **Use Cloud-Init/Sysprep (Cloud-Init/Sysprep 사용)** 확인란을 선택합니다.
  - c. 인증 화살표를 클릭하고 **User Name and Password** 를 입력하거나 이미 구성된 암호 사용을 선택합니다.



참고

이 사용자 이름은 로컬 관리자의 이름입니다. 인증 섹션 또는 사용자 지정 **sysprep** 파일의 기본값(사용자)에서 해당 값을 변경할 수 있습니다.

- d. **Custom Script (사용자 지정 스크립트)** 화살표를 클릭하고 **/usr/share/ovirt-engine/conf/ sysprep /** 에 있는 기본 **sysprep** 파일의 내용을 텍스트 상자에 붙여넣습니다.
- e. **sysprep** 파일의 다음 값을 수정할 수 있습니다.
  - 키. 미리 정의된 **Windows** 활성화 제품 키를 사용하지 않으려면 **<![CDATA[\$ProductKey\$]>**를 유효한 제품 키로 교체합니다.

```
<ProductKey>
  <Key><![CDATA[$ProductKey$]]></Key>
</ProductKey>
```

예 12.2. Windows 제품 키 예

```
<ProductKey>
  <Key>0000-000-000-000</Key>
</ProductKey>
```

Windows 가상 머신이 참여할 도메인, 도메인의 암호, 도메인 관리자의 사용자 이름:

```
<Credentials>
  <Domain>AD_Domain</Domain>
  <Password>Domain_Password</Password>
  <Username>Domain_Administrator</Username>
</Credentials>
```

예 12.3. 도메인 자격 증명 예

```
<Credentials>
  <Domain>addomain.local</Domain>
  <Password>12345678</Password>
  <Username>Sarah_Smith</Username>
</Credentials>
```

참고

도메인에 가입하려면 도메인, 암호 및 사용자 이름이 필요합니다. 둘 다 필요하지 않습니다.

**Initial Run** 탭에서 도메인 및 자격 증명을 수정할 수 없습니다.

로컬 관리자의 전체 이름입니다.

```
<UserData>
...
  <FullName>Local_Administrator</FullName>
...
</UserData>
```

**displayName** 및 **Name** 로컬 관리자의 이름입니다.

```

<LocalAccounts>
  <LocalAccount wcm:action="add">
    <Password>
      <Value><![CDATA[$AdminPassword$]]></Value>
      <PlainText>>true</PlainText>
    </Password>
    <DisplayName>Local_Administrator</DisplayName>
    <Group>administrators</Group>
    <Name>Local_Administrator</Name>
  </LocalAccount>
</LocalAccounts>

```

**sysprep** 파일의 나머지 변수는 **Initial Run** 탭에서 채울 수 있습니다.

13.

선택 사항: 풀 유형 설정:

a.

유형 탭을 클릭하고 풀 유형을 선택합니다.

- 

**Manual** - 관리자가 가상 머신을 풀로 명시적으로 반환해야 합니다.

- 

**Automatic** - 가상 머신이 가상 머신 풀로 자동 반환됩니다.

b.

**Stateful Pool** 확인란을 선택하여 가상 머신이 상태 저장 모드에서 시작되도록 합니다. 이렇게 하면 이전 사용자가 변경한 내용이 가상 시스템에서 유지됩니다.

c.

**OK**를 클릭합니다.

14.

선택 사항: **SPICE** 프록시를 재정의합니다.

a.

콘솔 탭에서 **SPICE** 프록시 덮어쓰기 확인란을 선택합니다.

b.

**Overridden SPICE** 프록시 주소 텍스트 필드에서 글로벌 **SPICE** 프록시를 재정의할 **SPICE** 프록시 주소를 지정합니다.

c.

**OK**를 클릭합니다.



15.

**Windows** 가상 머신 풀의 경우 **Compute** → **Virtual Machines** (가상 머신)를 클릭하고 풀에서 각 가상 머신을 선택한 다음 **Run ECDHE**(한 번)를 클릭합니다.

참고

가상 머신이 시작되지 않고 **Info [windeploy.exe] found가 %ECDHEDIR%\panther\unattend GC\setupact.log**에 표시되지 않는 경우 풀에 대한 템플릿을 생성하는 데 사용된 **Windows** 가상 머신의 레지스트리에 **UnattendFile** 키를 추가합니다.

1. **Windows** 가상 머신에 연결되지 않은 파일(예: **A:\Unattend.xml**)이 있는 연결된 **sidepy** 장치가 있는지 확인합니다.
2. 시작을 클릭하고 실행 을 클릭하고 열기 텍스트 상자에 **regedit** 를 입력한 다음 확인을 클릭합니다.
3. 왼쪽 창에서 **HKEY\_LOCAL\_MACHINE** → **SYSTEM** → 이동합니다.
4. 오른쪽 창을 마우스 오른쪽 버튼으로 클릭하고 새 → 문자열 값을 선택합니다.
5. 키 이름으로 **UnattendFile** 을 입력합니다.
6. 새 키를 두 번 클릭하고 연결 해제 파일 이름과 경로를 입력합니다(예: **A:\Unattend.xml**, 키의 값으로).
7. 레지스트리를 저장하고 **Windows** 가상 머신을 배포한 다음 새 템플릿을 생성합니다. 자세한 내용은 가상 머신 관리 가이드의 템플릿을 참조하십시오.

지정된 수의 동일한 가상 머신으로 가상 머신 풀을 생성하고 구성했습니다. **Compute** → **Virtual Machines** 에서 가상 머신을 보거나 풀 이름을 클릭하여 세부 정보 보기를 열 수 있습니다. 풀의 가상 머신은 독립 가상 머신과 해당 아이콘으로 구분됩니다.

### 12.3. 새 풀 및 편집 풀 **WINDOWS**의 설정 및 제어 설명

### 12.3.1. 새 풀 및 편집 풀 일반 설정 설명

다음 표에서는 새 풀의 일반 탭에 필요한 정보와 가상 머신 풀 고유의 **Edit Pool** 창에서 자세히 설명합니다. 기타 모든 설정은 **New Virtual Machine** 창에서와 동일합니다.

표 12.1. 일반 설정

필드 이름	설명
템플릿	가상 머신 풀이 기반으로 하는 템플릿 및 템플릿 하위 버전입니다. 템플릿의 최신 하위 버전을 기반으로 풀을 생성하면 풀의 모든 가상 머신은 재부팅 시 자동으로 최신 템플릿 버전을 받습니다. 가상 머신의 템플릿 구성에 대한 자세한 내용은 <a href="#">가상 머신 관리 가이드의 새 템플릿 및 Edit Template Windows의 설정 설명 및 설정 설명</a> 을 참조하십시오.
설명	가상 머신 풀에 대한 의미 있는 설명입니다.
주석	가상 머신 풀과 관련하여 사용자가 읽을 수 있는 일반 텍스트를 추가하는 필드입니다.
사전 시작된 VM	시작하기 전에 가상 머신 풀의 가상 머신 수를 지정하고 사용자가 사용할 해당 상태로 유지할 수 있습니다. 이 필드의 값은 0 과 가상 머신 풀의 총 가상 머신 수 사이여야 합니다.
VM 수/풀에 따른 VM 수 증가	생성할 가상 머신 수를 지정하고 가상 머신 풀에서 사용 가능하게 만들 수 있습니다. 편집 창에서 가상 머신 풀의 가상 머신 수를 지정된 수만큼 늘릴 수 있습니다. 기본적으로 풀에서 생성할 수 있는 최대 가상 머신 수는 1000입니다. 이 값은 <b>engine-config</b> 명령의 <b>MaxVmsInPool</b> 키를 사용하여 구성할 수 있습니다.
사용자당 최대 VM 수	단일 사용자가 한 번에 가상 머신 풀에서 수행할 수 있는 최대 가상 머신 수를 지정할 수 있습니다. 이 필드의 값은 1 에서 32,767 사이여야 합니다.
삭제 보호	풀의 가상 머신이 삭제되지 않도록 할 수 있습니다.

### 12.3.2. 새로운 풀 및 편집 풀 유형 설정 설명

다음 표에서는 **New Pool** (새 풀) 및 **Edit Pool** (풀 편집) 창의 유형 탭에 필요한 정보를 자세히 설명합니다.

표 12.2. 유형 설정

필드 이름	설명
풀 유형	이 드롭다운 메뉴를 사용하면 가상 머신 풀 유형을 지정할 수 있습니다. 다음 옵션을 사용할 수 있습니다. <ul style="list-style-type: none"> <li>● <b>Automatic:</b> 사용자가 가상 머신 풀에서 가져온 가상 머신 사용을 완료하면 해당 가상 머신이 가상 머신 풀로 자동 반환됩니다.</li> <li>● <b>수동:</b> 사용자가 가상 머신 풀에서 가져온 가상 머신 사용을 완료하면 관리자가 가상 머신을 수동으로 반환하는 경우에만 가상 머신 풀로 돌아갑니다.</li> </ul>
상태 저장 풀	가상 머신을 다른 사용자에게 전달할 때 풀에 있는 가상 머신의 상태를 보존할지 여부를 지정합니다. 즉, 이전 사용자가 변경한 내용은 가상 시스템에서 유지됩니다.

### 12.3.3. 새 풀 및 **Edit Pool Console** 설정 설명

다음 표에서는 새 풀의 콘솔 탭에 필요한 정보를 자세히 설명하거나 가상 머신 풀과 관련된 **Edit Pool** (풀 편집) 창에서 자세히 설명합니다. 다른 모든 설정은 새 가상 머신 및 가상 머신 편집 창의 설정과 동일합니다.

표 12.3. 콘솔 설정

필드 이름	설명
SPICE 프록시 재정의	글로벌 구성에 정의된 SPICE 프록시를 재정의하려면 이 확인란을 선택합니다. 이 기능은 사용자(예: VM 포털을 통해 연결)가 호스트가 상주하는 네트워크 외부에 있는 경우 유용합니다.
재정의된 SPICE 프록시 주소	SPICE 클라이언트가 가상 머신에 연결하는 프록시입니다. 이 프록시는 Red Hat Virtualization 환경에 대해 정의된 글로벌 SPICE 프록시와 가상 머신 풀이 속하는 클러스터에 정의된 SPICE 프록시를 모두 덮어씁니다. 주소는 다음 형식이어야 합니다. <div style="border: 1px solid black; padding: 2px; margin-top: 5px;"> <code>protocol://host:port</code> </div>

### 12.3.4. 가상 머신 풀 호스트 설정 설명

다음 표에서는 **New Pool** (새 풀) 및 **Edit Pool** (풀 편집) 창의 **Host** 탭에서 사용할 수 있는 옵션에 대해 자세히 설명합니다.

표 12.4. 가상 머신 풀: 호스트 설정

필드 이름	하위 요소	설명
<p>실행 시작</p>		<p>가상 머신을 실행할 기본 호스트를 정의합니다. 다음 중 하나를 선택합니다.</p> <ul style="list-style-type: none"> <li>● <b>any Host in Cluster</b> - 가상 머신은 클러스터의 사용 가능한 모든 호스트에서 시작하고 실행할 수 있습니다.</li> <li>● <b>특정 호스트 - 가상 머신이 클러스터의 특정 호스트에서 실행되기 시작합니다.</b> 그러나 관리자 또는 관리자는 가상 시스템의 마이그레이션 및 고가용성 설정에 따라 가상 머신을 클러스터의 다른 호스트로 마이그레이션할 수 있습니다. 사용 가능한 호스트 목록에서 특정 호스트 또는 호스트 그룹을 선택합니다.</li> </ul>
<p>마이그레이션 옵션</p>	<p>마이그레이션 모드</p>	<p>가상 머신을 실행하고 마이그레이션할 옵션을 정의합니다. 여기에 있는 옵션을 사용하지 않는 경우 가상 머신은 클러스터 정책에 따라 실행 또는 마이그레이션됩니다.</p> <ul style="list-style-type: none"> <li>● <b>수동 및 자동 마이그레이션 허용</b> - 환경 상태에 따라 호스트 간에 가상 머신을 자동으로 마이그레이션하거나 관리자가 수동으로 마이그레이션할 수 있습니다.</li> <li>● <b>수동 마이그레이션만 허용</b> - 가상 머신은 관리자가 수동으로 한 호스트에서 다른 호스트로만 마이그레이션할 수 있습니다.</li> <li>● <b>마이그레이션을 허용하지 않음</b> - 가상 머신을 자동으로 또는 수동으로 마이그레이션할 수 없습니다.</li> </ul>

필드 이름	하위 요소	설명
	사용자 정의 마이그레이션 정책 사용	<p>마이그레이션 통합 정책을 정의합니다. 확인란을 선택하지 않은 상태로 두면 호스트는 정책을 결정합니다.</p> <ul style="list-style-type: none"> <li>● <b>legacy</b> - 3.6 버전의 레거시 동작. <b>vdsm.conf</b>의 덮어쓰기는 계속 적용됩니다. 게스트 에이전트 후크 메커니즘이 비활성화되어 있습니다.</li> <li>● <b>다운타임 최소화</b> - 가상 머신을 일반적인 상황에서 마이그레이션할 수 있습니다. 가상 머신에 심각한 다운타임이 발생하지 않아야 합니다. 가상 머신 마이그레이션이 긴 시간(QEMU 반복에 따라 최대 500밀리초) 후에 통합되지 않으면 마이그레이션이 중단됩니다. 게스트 에이전트 후크 메커니즘이 활성화되어 있습니다.</li> <li>● <b>필요한 경우 워크로드 일시 중단</b> - 가상 머신이 많은 워크로드를 실행하는 시기를 포함하여 대부분의 상황에서 가상 머신을 마이그레이션할 수 있습니다. 이로 인해 가상 머신에 다른 설정보다 다운타임이 발생할 수 있습니다. 극단적인 워크로드에서 마이그레이션이 중단될 수 있습니다. 게스트 에이전트 후크 메커니즘이 활성화되어 있습니다.</li> </ul>
	사용자 정의 마이그레이션 다운타임 사용	<p>이 확인란을 사용하면 실시간 마이그레이션 중에 가상 머신을 중단할 수 있는 최대 밀리초 수를 지정할 수 있습니다. 워크로드 및 SLA 요구 사항에 따라 각 가상 시스템에 대해서도 다른 최대 다운타임을 구성합니다. VDSM 기본값을 사용하려면 <b>0</b>을 입력합니다.</p>

필드 이름	하위 요소	설명
	<p>자동 컨버지 마이그레이션</p>	<p><b>레거시</b> 마이그레이션 정책으로만 활성화됩니다. 가상 머신의 실시간 마이그레이션 중에 자동 통신 사용 여부를 설정할 수 있습니다. 워크로드가 많은 대규모 가상 머신은 실시간 마이그레이션 중에 달성된 전송 속도보다 더 빨리 메모리를 더 빠르게 수행할 수 있으며 마이그레이션이 중단되지 않을 수 있습니다.</p> <p>QEMU의 자동 일관성 기능을 사용하면 가상 머신 마이그레이션을 강제 통합할 수 있습니다. QEMU는 가상 머신의 vCPU가 제한되는 것과 트리거가 통합되지 않음을 자동으로 감지합니다. Auto-convergence는 기본적으로 전역에서 비활성화되어 있습니다.</p> <ul style="list-style-type: none"> <li>● <b>클러스터 수준에서 설정된 자동 일치 설정을 사용하려면 클러스터 설정에서 Inherit</b> 를 선택합니다. 이 옵션은 기본적으로 선택됩니다.</li> <li>● <b>Auto Converge</b> 를 선택하여 클러스터 설정 또는 글로벌 설정을 재정의하고 가상 머신에 자동 일관성을 허용합니다.</li> <li>● <b>don't Auto Converge</b> 를 선택하여 클러스터 설정 또는 글로벌 설정을 재정의하고 가상 머신의 자동 연결을 방지합니다.</li> </ul>

필드 이름	하위 요소	설명
	마이그레이션 압축 사용	<p><b>레거시</b> 마이그레이션 정책으로만 활성화됩니다. 옵션을 사용하면 가상 머신의 실시간 마이그레이션 중에 마이그레이션 압축을 사용할 수 있습니다. 이 기능은 Xor Binary Zero Run-Length-Encoding을 사용하여 메모리 쓰기 집약적인 워크로드 또는 스파스 메모리 업데이트 패턴을 실행하는 가상 머신의 다운타임과 총 실시간 마이그레이션 시간을 줄입니다. 마이그레이션 압축은 기본적으로 전역적으로 비활성화되어 있습니다.</p> <ul style="list-style-type: none"> <li>● <b>클러스터 수준에서 설정된 압축 설정을 사용하려면 클러스터 설정에서 Inherit</b> 를 선택합니다. 이 옵션은 기본적으로 선택됩니다.</li> <li>● <b>압축</b> 을 선택하여 클러스터 설정 또는 전역 설정을 재정의하고 가상 머신에 대해 압축을 허용합니다.</li> <li>● <b>don't compress</b> 를 선택하여 클러스터 설정 또는 글로벌 설정을 재정의하고 가상 머신의 압축을 방지합니다.</li> </ul>
	pass-Through Host CPU	이 확인란을 사용하면 가상 머신에서 해당 머신이 있는 호스트의 물리적 CPU 기능을 활용할 수 있습니다.
NUMA 구성	NUMA 노드 수	가상 머신에 할당할 가상 NUMA 노드 수입니다. <b>Tune Mode</b> 가 <b>Preferred</b> 이면 이 값을 <b>1</b> 로 설정해야 합니다.

필드 이름	하위 요소	설명
	조정 모드	<p>메모리를 할당하는 데 사용되는 방법입니다.</p> <ul style="list-style-type: none"> <li>● <b>Strict:</b> 대상 노드에 메모리를 할당할 수 없는 경우 메모리 할당이 실패합니다.</li> <li>● <b>preferred:</b> 단일 기본 노드에서 메모리가 할당됩니다. 충분한 메모리를 사용할 수 없는 경우 다른 노드에서 메모리를 할당할 수 있습니다.</li> <li>● <b>interleave:</b> 메모리는 라운드 로빈 알고리즘의 여러 노드에 할당됩니다.</li> </ul>
	NUMA 고정	<p>NUMA 토폴로지 창을 엽니다. 이 창에는 호스트의 총 CPU, 메모리 및 NUMA 노드, 가상 시스템의 가상 NUMA 노드가 표시됩니다. 각 vNUMA를 클릭하여 NUMA 노드를 호스팅할 가상 NUMA 노드를 고정하고 오른쪽 상자에서 왼쪽의 NUMA 노드로 드래그합니다.</p>

**12.3.5. 새로운 풀 및 편집 풀 리소스 설정 설명**

다음 표에서는 새 풀의 리소스 및 가상 머신 풀 고유의 **Edit Pool (풀 편집)** 탭에 필요한 정보를 자세히 설명합니다. 기타 모든 설정은 **New Virtual Machine** 창에서와 동일합니다. 자세한 내용은 [가상 머신 관리 가이드](#)에 설명된 가상 머신 리소스 **DestinationRule** 설정을 참조하십시오.

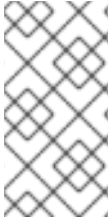
**표 12.5. resourceECDHE 설정**

필드 이름	하위 요소	설명
diskECDHE	자동 선택 대상	이 확인란을 선택하여 가장 많은 여유 공간이 있는 스토리지 도메인을 자동으로 선택합니다. <b>Target</b> 및 <b>Disk Profile</b> 필드는 비활성화되어 있습니다.
	형식	이 필드는 읽기 전용이며 스토리지 도메인 유형이 OpenStack Volume(Cinder)인 경우 <b>QCOW2</b> 를 항상 표시합니다. 이 경우 형식은 <b>Raw</b> 입니다.



## 12.4. 가상 머신 풀 편집

가상 머신 풀을 생성한 후 해당 속성을 편집할 수 있습니다. 가상 머신 풀을 편집할 때 사용할 수 있는 속성은 **Number of VMs** 속성이 풀의 **VM** 수 증가로 교체되는 경우를 제외하고 새 가상 머신 풀을 생성할 때 사용 가능한 속성과 동일합니다.



### 참고

가상 머신 풀을 편집할 때 도입된 변경 사항은 새 가상 머신에만 영향을 미칩니다. 도입된 변경 시점에 이미 존재하는 가상 머신은 영향을 받지 않습니다.

### 가상 머신 풀 편집

1. 컴퓨팅 → 풀을 클릭하고 가상 머신 풀을 선택합니다.
2. 편집 을 클릭합니다.
3. 가상 머신 풀의 속성을 편집합니다.
4. 확인을 클릭합니다.

## 12.5. 풀에서 가상 머신 사전 시작

가상 머신 풀의 가상 머신은 기본적으로 전원이 꺼집니다. 사용자가 풀에서 가상 시스템을 요청하면 시스템의 전원이 켜지고 사용자에게 할당됩니다. 반면 사전 시작된 가상 시스템이 이미 실행 중이며 사용자에게 할당되기를 기다리는 동안 대기 중인 시간이 단축되어 시스템에 액세스할 수 있습니다. 사전 시작된 가상 머신이 종료되면 풀로 반환되고 원래 상태로 복원됩니다. 사전 시작된 가상 머신의 최대 수는 풀에 있는 가상 머신의 수입니다.

사전 시작된 가상 머신은 사용자에게 구체적으로 할당되지 않은 가상 머신에 즉시 액세스해야 하는 환경에 적합합니다. 자동 풀만 사전 시작할 수 있는 가상 머신입니다.

### 풀에서 가상 머신 사전 시작

1. **Compute** → **Pools (풀)**를 클릭하고 가상 머신 풀을 선택합니다.
2. 편집 을 클릭합니다.
3. 사전 시작 **VM** 필드에 사전 시작할 가상 머신 수를 입력합니다.
4. 유형 탭을 클릭합니다. **Pool Type** 이 **Automatic** 으로 설정되어 있는지 확인합니다.
5. **OK**를 클릭합니다.

## 12.6. 가상 머신 풀에 가상 머신 추가

가상 시스템 풀에서 원래 프로비저닝된 것보다 더 많은 가상 머신이 필요한 경우 풀에 더 많은 머신을 추가합니다.

### 가상 머신 풀에 가상 머신 추가

1. **Compute** → **Pools (풀)**를 클릭하고 가상 머신 풀을 선택합니다.
2. 편집 을 클릭합니다.
3. 풀별 **VM** 수 증가 필드에 추가 가상 머신 수를 입력합니다.
4. **OK**를 클릭합니다.

## 12.7. 가상 머신 풀에서 가상 머신 분리

가상 머신 풀에서 가상 머신을 분리할 수 있습니다. 가상 머신을 분리하면 풀에서 가상 머신이 제거되어 독립 가상 머신이 됩니다.

## 가상 머신 풀에서 가상 머신 분리

1. **Compute** → **Pools** 를 클릭합니다.
2. 풀 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 가상 머신 탭을 클릭하여 풀의 가상 머신을 나열합니다.
4. 가상 머신의 상태가 **Down** 인지 확인합니다. 실행 중인 가상 머신을 분리할 수 없습니다.
5. 하나 이상의 가상 머신을 선택하고 **분리** 를 클릭합니다.
6. **OK**를 클릭합니다.



## 참고

가상 머신은 아직 환경에 존재하며 **Compute** → **Virtual Machines** 에서 보고 액세스할 수 있습니다. 분리된 가상 머신이 독립 가상 머신임을 나타내기 위해 아이콘이 변경됩니다.

## 12.8. 가상 머신 풀 제거

데이터 센터에서 가상 머신 풀을 제거할 수 있습니다. 먼저 풀의 모든 가상 머신을 삭제하거나 분리해야 합니다. 풀에서 가상 머신을 분리하면 독립 가상 머신으로 유지됩니다.

## 가상 머신 풀 제거

1. **Compute** → **Pools (풀)**를 클릭하고 가상 머신 풀을 선택합니다.
2. **제거**를 클릭합니다.

3. **OK**를 클릭합니다.

### 12.9. 신뢰할 수 있는 컴퓨팅 풀

신뢰할 수 있는 컴퓨팅 풀은 **Intel TXT(Trusted Execution Technology)**를 기반으로 하는 보안 클러스터입니다. 신뢰할 수 있는 클러스터는 **Intel OpenAttestation**에서 확인한 호스트만 허용하여 호스트 하드웨어 및 소프트웨어의 무결성을 화이트 **List** 데이터베이스에 대해 측정합니다. 신뢰할 수 있는 호스트 및 실행 중인 가상 시스템에는 더 높은 보안이 필요한 작업을 할당할 수 있습니다. **Intel TXT**, 신뢰할 수 있는 시스템 및 증명에 대한 자세한 내용은 <https://software.intel.com/en-us/articles/intel-trusted-execution-technology-intel-txt-enabling-guide>의 내용을 참조하십시오.

신뢰할 수 있는 컴퓨팅 풀을 생성하려면 다음 단계가 포함됩니다.

- **OpenAttestation** 서버와 통신하도록 **Manager** 구성.
- 신뢰할 수 있는 호스트만 실행할 수 있는 신뢰할 수 있는 클러스터 생성
- 신뢰할 수 있는 클러스터에 신뢰할 수 있는 호스트 추가. **OpenAttestation sever**에서 신뢰할 수 있는 것으로 확인하려면 **OpenAttestation** 에이전트를 실행해야 합니다.

#### 12.9.1. OpenAttestation Server를 Manager에 연결

신뢰할 수 있는 클러스터를 만들려면 먼저 **OpenAttestation** 서버를 인식하도록 **Red Hat Virtualization Manager**를 구성해야 합니다. **engine-config**를 사용하여 **OpenAttestation** 서버의 **FQDN** 또는 **IP** 주소를 추가합니다.

```
# engine-config -s AttestationServer=attestationserver.example.com
```

필요한 경우 다음 설정도 변경할 수 있습니다.

표 12.6. engine-config의 OpenAttestation 설정

옵션	기본값	설명
AttestationServer	oat-server	OpenAttestation 서버의 FQDN 또는 IP 주소입니다. Manager가 OpenAttestation 서버와 통신할 수 있도록 설정해야 합니다.

옵션	기본값	설명
AttestationPort	8443	OpenAttestation 서버에서 Manager와 통신하는 데 사용하는 포트입니다.
AttestationTruststore	TrustStore.jks	OpenAttestation 서버와의 통신을 보호하는 데 사용되는 신뢰 저장소입니다.
AttestationTruststorePass	암호	신뢰 저장소에 액세스하는 데 사용되는 암호입니다.
AttestationFirstStageSize	10	빠른 초기화에 사용됩니다. 좋은 이유 없이 이 값을 변경하는 것은 권장되지 않습니다.
SecureConnectionWithOATServers	true	OpenAttestation 서버로 보안 통신을 활성화하거나 비활성화합니다.
PollUri	AttestationService/resources/PollHosts	OpenAttestation 서비스에 액세스하는 데 사용되는 URI입니다.

### 12.9.2. 신뢰할 수 있는 클러스터 생성

신뢰할 수 있는 클러스터는 **OpenAttestation** 서버와 통신하여 호스트의 보안을 평가합니다. 호스트가 신뢰할 수 있는 클러스터에 추가되면 **OpenAttestation** 서버는 호스트의 하드웨어 및 소프트웨어를 화이트 목록 데이터베이스와 비교합니다. 신뢰할 수 있는 클러스터의 신뢰할 수 있는 호스트 간에 가상 머신을 마이그레이션하여 보안 환경에서 고가용성을 허용할 수 있습니다.

#### 신뢰할 수 있는 클러스터 생성

1. **Compute** → **Clusters** 를 클릭합니다.
2. 새로 생성을 클릭합니다.
3. 클러스터의 이름을 입력합니다.
4. **Enable Virt Service** 확인란을 선택합니다.

5. **Scheduling Policy** 탭을 클릭하고 신뢰할 수 있는 서비스 활성화 확인란을 선택합니다.
6. **OK**를 클릭합니다.

### 12.9.3. 신뢰할 수 있는 호스트 추가

**Red Hat Enterprise Linux** 호스트는 신뢰할 수 있는 클러스터에 추가할 수 있으며 **OpenAttestation** 서버에서 **White List** 데이터베이스에 대해 측정할 수 있습니다. 호스트는 **OpenAttestation** 서버에서 신뢰하려면 다음 요구 사항을 충족해야 합니다.

- **BIOS**에서 **Intel TXT**가 활성화되어 있습니다.
- **OpenAttestation** 에이전트가 설치되어 실행되고 있습니다.
- 호스트에서 실행되는 소프트웨어는 **OpenAttestation** 서버의 **White List** 데이터베이스와 일치합니다.

#### 신뢰할 수 있는 호스트 추가

1. 컴퓨팅 → 호스트.
2. 새로 생성을 클릭합니다.
3. **Host Cluster** 드롭다운 목록에서 신뢰할 수 있는 클러스터를 선택합니다.
4. 호스트 이름을 입력합니다.
5. 호스트의 호스트 이름을 입력합니다.
6. 호스트의 루트 암호를 입력합니다.

---

7.

**OK**를 클릭합니다.

호스트가 신뢰할 수 있는 클러스터에 추가되면 **OpenAttestation** 서버에서 평가합니다.

**OpenAttestation** 서버에서 호스트를 신뢰하지 않으면 비작동 상태로 전환되며 신뢰할 수 있는 클러스터에서 제거해야 합니다.

## 13장. 가상 디스크

### 13.1. 가상 머신 스토리지 이해

**Red Hat Virtualization**은 **NFS**, **iSCSI** 및 **FCP**의 세 가지 스토리지 유형을 지원합니다.

각 유형에서 **SPM(Storage Pool Manager)**이라는 호스트는 호스트와 스토리지 간의 액세스를 관리합니다. **DestinationRule** 호스트는 스토리지 풀 내에서 전체 액세스 권한이 있는 유일한 노드이며, **DestinationRule**은 스토리지 도메인 메타데이터와 풀의 메타데이터를 수정할 수 있습니다. 다른 모든 호스트는 가상 머신 하드 디스크 이미지 데이터에만 액세스할 수 있습니다.

기본적으로 **NFS**, 로컬 또는 **POSIX** 호환 데이터 센터의 경우 **EC2DHE**은 파일 시스템의 파일로 씬 프로 비저닝된 형식을 사용하여 가상 디스크를 생성합니다.

**iSCSI** 및 기타 블록 기반 데이터 센터에서 **BOOM**은 제공된 논리 단위 번호(**LUN**) 위에 볼륨 그룹을 만들고 가상 디스크로 논리 볼륨을 만듭니다. 블록 기반 스토리지의 가상 디스크는 기본적으로 사전 할당됩니다.

가상 디스크가 사전 할당되면 지정된 크기(**GB**)의 논리 볼륨이 생성됩니다. 가상 머신은 **kpartx**, **scan**, **EC2DHEchange** 또는 **mount** 를 사용하여 **Red Hat Enterprise Linux** 서버에 마운트하여 가상 머신의 프로세스 또는 문제를 조사할 수 있습니다.

가상 디스크가 씬 프로비저닝되면 **1GB** 논리 볼륨이 생성됩니다. 논리 볼륨은 가상 머신이 실행 중인 호스트에서 지속적으로 모니터링됩니다. 사용량이 임계값에 도달하는 즉시 호스트에서 **EC2DHE**에 알립니다. 그러면 논리 볼륨을 **1GB**만큼 확장합니다. 호스트는 논리 볼륨이 확장된 후 가상 시스템을 다시 시작합니다. 가상 머신이 일시 정지 상태가 되면 **EC2DHE**가 시간 내에 디스크를 확장할 수 없음을 의미합니다. 이 문제는 **EC2DHE**이 너무 바쁘거나 저장 공간이 충분하지 않은 경우 발생합니다.

사전 할당된(**raw**) 형식의 가상 디스크는 씬 프로비저닝(**QCOW2**) 형식의 가상 디스크보다 쓰기 속도가 훨씬 빠릅니다. 씬 프로비저닝은 가상 디스크를 생성하는 데 상당한 시간이 걸립니다. 씬 프로비저닝 형식은 **I/O** 집약적인 가상 머신에 적합합니다. 사전 할당된 형식은 **I/O** 쓰기가 높은 가상 머신에 권장됩니다. 가상 머신에서 **4초**마다 **1GB** 이상을 쓸 수 있는 경우 가능한 경우 사전 할당된 디스크를 사용합니다.

### 13.2. 가상 디스크 이해

**Red Hat Virtualization** 기능은 사전 할당된 (프로비저닝) 및 **Sparse** (프로비저닝) 스토리지 옵션.



## ● 사전 할당

사전 할당된 가상 디스크는 가상 머신에 필요한 모든 스토리지를 할당합니다. 예를 들어 가상 시스템의 데이터 파티션에 대해 생성된 **20GB**의 사전 할당된 논리 볼륨은 생성 시 즉시 **20GB**의 스토리지 공간을 차지합니다.

## ● 스파스

스파스 할당을 사용하면 관리자가 가상 머신에 할당할 총 스토리지를 정의할 수 있지만 필요한 경우에만 스토리지는 할당됩니다.

예를 들어, **20GB**의 썸 프로비저닝 논리 볼륨은 처음 생성될 때 **0GB**의 스토리지 공간을 사용합니다. 운영 체제가 설치되면 설치된 파일의 크기가 증가할 수 있으며 최대 **20GB** 크기까지 데이터가 추가됨에 따라 계속 증가할 수 있습니다.

스토리지 디스크에서 가상 디스크의 **ID**를 볼 수 → 있습니다. **ID**는 장치 이름(예: **/dev/vda0**)이 변경되어 디스크 손상을 유발할 수 있으므로 가상 디스크를 식별하는 데 사용됩니다. **/dev/disk/by-id**에서 가상 디스크의 **ID**를 볼 수도 있습니다.

스토리지 도메인, 가상 머신 및 템플릿에 대한 세부 정보 보기의 스토리지 디스크 및 디스크 탭에서 디스크의 가상 크기를 볼 수 있습니다. 가상 크기는 가상 머신에서 사용할 수 있는 총 디스크 공간입니다. 가상 디스크를 생성하거나 편집할 때 크기(**GB**) 필드에 입력하는 번호입니다.

스토리지 도메인 및 템플릿에 대한 세부 정보 보기의 디스크 탭에서 디스크의 실제 크기를 볼 수 있습니다. 이는 지금까지 가상 머신에 할당된 디스크 공간입니다. 사전 할당된 디스크에는 가상 크기 및 실제 크기에 대해 동일한 값이 표시됩니다. 스파스 디스크에는 할당된 디스크 공간 크기에 따라 다른 값이 표시될 수 있습니다.

## 참고

**Cinder** 가상 디스크를 생성할 때 **Cinder**에서 디스크 형식과 유형을 내부적으로 처리하며 **Red Hat Virtualization**에서 관리하지 않습니다.

스토리지 유형 및 형식의 가능한 조합은 다음 표에 설명되어 있습니다.

표 13.1. 허용된 스토리지 조합

스토리지	형식	유형	참고
NFS	원시	사전 할당	가상 디스크에 정의된 스토리지 양과 같고 포맷이 없는 초기 크기가 있는 파일입니다.
NFS	원시	스파스	0에 가까운 초기 크기가 있는 파일이며 포맷이 없습니다.
NFS	QCOW2	스파스	초기 크기가 0이고 QCOW2 형식이 있는 파일입니다. 후속 계층의 형식은 QCOW2입니다.
SAN	원시	사전 할당	초기 크기가 가상 디스크에 정의된 스토리지 양과 같고 포맷이 없는 블록 장치입니다.
SAN	QCOW2	스파스	초기 크기가 가상 디스크에 정의된 크기(현재 1GB)보다 작고, 필요에 따라 공간을 할당하는 QCOW2 포맷이 있습니다(현재 1GB 단위로).

### 13.3. 삭제 후 가상 디스크 설정

관리 포털에서 **deleting**으로 표시되는 **delete\_after\_delete** 플래그는 가상 디스크가 삭제 될 때 사용된 데이터를 **0**으로 대체합니다. 기본값인 **false**로 설정된 경우 디스크를 삭제하면 재사용을 위해 해당 블록이 열리지만 데이터가 지워지지 않습니다. 따라서 블록이 **0**으로 반환되지 않았기 때문에 이 데이터가 복구될 수 있습니다.

**delete\_after\_delete** 플래그는 블록 스토리지에서만 작동합니다. 파일 스토리지(예: **NFS**)에서 옵션은 파일 시스템에서 데이터가 없는지 확인하기 때문에 아무 작업도 수행하지 않습니다.

가상 디스크에 대해 **delete\_after\_delete** 를 활성화하면 더 안전합니다. 가상 디스크에 중요한 데이터가 포함되어 있는 경우 사용하는 것이 좋습니다. 이는 보다 집약적인 작업이며 사용자는 성능 저하 및 긴 삭제 시간이 발생할 수 있습니다.



## 참고

삭제 기능이 보안 삭제와 동일하지 않으며, 동일한 스토리지에 생성된 새 디스크가 이전 디스크의 데이터를 노출하지 않는 것만으로도 데이터가 스토리지에서 제거되었음을 보장할 수 없습니다.

**delete\_after\_delete** 플래그 기본값은 설정 프로세스 중에 **true** 로 변경하거나([Red Hat Virtualization Manager 구성 참조](#)) 또는 [Red Hat Virtualization Manager](#)에서 **engine-config** 툴을 사용하여 변경할 수 있습니다. 설정 변경 사항을 적용하려면 **ovirt-engine** 서비스를 다시 시작합니다.



## 참고

**delete\_after\_delete** 플래그의 기본 설정을 변경하면 이미 존재하는 디스크의 **Delete** 속성에는 영향을 미치지 않습니다.

엔진 구성 도구를 사용하여 **SANWipeAfterDelete**를 **Default**로 설정

1.

**--set** 작업과 함께 **engine-config** 툴을 실행합니다.

```
# engine-config --set SANWipeAfterDelete=true
```

2.

변경 사항을 적용하려면 **ovirt-engine** 서비스를 다시 시작합니다.

```
# systemctl restart ovirt-engine.service
```

호스트에 있는 **/var/log/vdsm/vdsm.log** 파일을 확인하여 가상 디스크를 성공적으로 지우고 삭제했는지 확인할 수 있습니다.

성공적으로 지워지려면 로그 파일에 **storage\_domain\_id/volume\_id** 가 **0**이 되어 삭제되는 항목이 포함됩니다. 예를 들면 다음과 같습니다.

```
a9cb0625-d5dc-49ab-8ad1-72722e82b0bf/a49351a7-15d8-4932-8d67-512a369f9d61 was zeroed and will be deleted
```

성공적으로 삭제하기 위해 로그 파일에 **VG:storage\_domain\_id LVs: list\_of\_volume\_ids, img: image\_id** 항목이 포함됩니다. 예를 들면 다음과 같습니다.

```
finished with VG:a9cb0625-d5dc-49ab-8ad1-72722e82b0bf LVs: {'a49351a7-15d8-4932-8d67-512a369f9d61': lmgPar(imgs=['11f8b3be-fa96-4f6a-bb83-14c9b12b6e0d'], parent='00000000-0000-0000-0000-000000000000')}, img: 11f8b3be-fa96-4f6a-bb83-14c9b12b6e0d
```

성공하지 못하면 로그 메시지 **zeroing storage\_domain\_id/volume\_id failed**가 표시됩니다. 이 볼륨 수동 를 제거하고 실패한 삭제는 일부 **VG: storage\_domain\_id 0ed** 볼륨( **list\_of\_volume\_ids**)에서 **Remove failed** 가 표시됩니다.

### 13.4. RED HAT VIRTUALIZATION의 공유 가능 디스크

일부 애플리케이션에서는 서버 간에 스토리지를 공유해야 합니다. **Red Hat Virtualization**을 사용하면 가상 머신 하드 디스크를 **shareable** 로 표시하고 해당 디스크를 가상 머신에 연결할 수 있습니다. 이렇게 하면 여러 클러스터 인식 게스트에서 단일 가상 디스크를 사용할 수 있습니다.

공유 디스크는 모든 상황에서 사용되지 않습니다. 클러스터된 데이터베이스 서버 및 기타 고가용성 서비스와 같은 애플리케이션의 경우 공유 디스크가 적합합니다. 클러스터에 인식되지 않는 여러 게스트에 공유 디스크를 연결하면 디스크에 대한 읽기 및 쓰기가 조정되지 않기 때문에 데이터 손상이 발생할 수 있습니다.

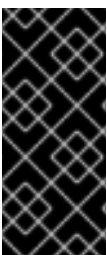
공유 디스크의 스냅샷을 가져올 수 없습니다. 스냅샷이 적용된 가상 디스크는 나중에 공유 가능한 것으로 표시할 수 없습니다.

디스크 공유 기능을 표시하거나 나중에 디스크를 편집하여 표시할 수 있습니다.

### 13.5. READ ONLY DISKS IN RED HAT VIRTUALIZATION

일부 애플리케이션에서는 관리자가 읽기 전용 권한으로 데이터를 공유해야 합니다. 가상 머신의 세부 정보 보기에 있는 디스크 탭을 통해 가상 머신에 연결된 디스크를 생성하거나 편집할 때 이 작업을 수행할 수 있습니다. 이렇게 하면 관리자가 쓰기 권한을 유지하면서 여러 클러스터 인식 게스트가 단일 디스크를 읽을 수 있습니다.

가상 머신이 실행되는 동안 디스크의 읽기 전용 상태를 변경할 수 없습니다.



#### 중요

저널링된 파일 시스템을 마운트하려면 읽기-쓰기 액세스가 필요합니다. **Read Only** 옵션을 사용하면 이러한 파일 시스템이 포함된 가상 디스크에 적합하지 않습니다(예: **EXT3,EXT4** 또는 **XFS**).

## 13.6. 가상 디스크 작업

### 13.6.1. 가상 디스크 생성

이미지 디스크 생성은 전적으로 **Manager**에 의해 관리됩니다. 직접 **LUN** 디스크에는 이미 존재하는 외부 준비된 대상이 필요합니다. **Cinder** 디스크는 외부 공급자 창을 사용하여 **Red Hat Virtualization** 환경에 추가된 **OpenStack** 볼륨 인스턴스에 액세스해야 합니다. 자세한 내용은 **14.2.4절. “스토리지 관리를 위해 OpenStack Block Storage(Cinder) 인스턴스 추가”**을 참조하십시오.

특정 가상 머신에 연결된 가상 디스크를 생성할 수 있습니다. **13.6.2절. “새 가상 디스크 창에서 설정에 대한 설명”**에 지정된 대로 연결된 가상 디스크를 생성할 때 추가 옵션을 사용할 수 있습니다.

가상 머신에 연결된 가상 디스크 생성

1. **Compute → Virtual Machines** 를 클릭합니다.
2. 가상 머신의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 디스크 탭을 클릭합니다.
4. 새로 생성을 클릭합니다.
5. 적절한 버튼을 클릭하여 가상 디스크가 이미지, 직접 **LUN** 또는 **Cinder** 디스크가 될지 여부를 지정합니다.
6. 가상 디스크에 필요한 옵션을 선택합니다. 선택한 디스크 유형에 따라 옵션이 변경됩니다. 각 디스크 유형의 각 옵션에 대한 자세한 내용은 **13.6.2절. “새 가상 디스크 창에서 설정에 대한 설명”**를 참조하십시오.
7. **OK**를 클릭합니다.

가상 머신에 속하지 않는 유동 가상 디스크를 생성할 수도 있습니다. 디스크를 공유할 수 있는 경우 이 디스크를 단일 가상 머신에 연결하거나 여러 가상 머신에 연결할 수 있습니다. **13.6.2절. “새 가상 디스크 창에서 설정에 대한 설명”**에 지정된 대로 가상 디스크를 생성할 때 일부 옵션을 사용할 수 없습니다.

유동 가상 디스크 생성



중요

부동 가상 디스크 생성은 기술 프리뷰 기능 전용입니다. 기술 프리뷰 기능은 **Red Hat** 프로덕션 서비스 수준 계약(SLA)에서 지원하지 않으며, 기능상 완전하지 않을 수 있어 프로덕션에 사용하지 않는 것이 좋습니다. 이러한 기능을 사용하면 향후 제품 기능을 조기에 이용할 수 있어 개발 과정에서 고객이 기능을 테스트하고 피드백을 제공할 수 있습니다.

**Red Hat** 기술 프리뷰 기능 지원 범위에 대한 자세한 내용은 <https://access.redhat.com/support/offerings/techpreview/>를 참조하십시오.

1. 스토리지 디스크를 클릭합니다.
2. 새로 생성을 클릭합니다.
3. 적절한 버튼을 클릭하여 가상 디스크가 이미지, 직접 **LUN** 또는 **Cinder** 디스크가 될지 여부를 지정합니다.
4. 가상 디스크에 필요한 옵션을 선택합니다. 선택한 디스크 유형에 따라 옵션이 변경됩니다. 각 디스크 유형의 각 옵션에 대한 자세한 내용은 **13.6.2절**. “새 가상 디스크 창에서 설정에 대한 설명”을 참조하십시오.
5. **OK**를 클릭합니다.

**13.6.2. 새 가상 디스크 창에서 설정에 대한 설명**

유동 및 연결된 가상 디스크를 만들기 위한 **New Virtual Disk** 창이 매우 유사하기 때문에 설정이 단일 섹션에 설명되어 있습니다.

표 13.2. 새로운 가상 디스크 및 가상 디스크 설정 편집: 이미지

필드 이름	설명
크기(GB)	새 가상 디스크의 크기(GB)입니다.

필드 이름	설명
별칭	40자로 제한되는 가상 디스크의 이름입니다.
설명	가상 디스크에 대한 설명입니다. 이 필드는 권장되지만 필수는 아닙니다.
인터페이스	<p>이 필드는 연결된 디스크를 생성할 때만 표시됩니다.</p> <p>디스크가 가상 머신에 제공하는 가상 인터페이스입니다. <b>virtio</b> 는 더 빠르지만 드라이버가 필요합니다. <b>Red Hat Enterprise Linux 5</b> 이상에는 이러한 드라이버가 포함되어 있습니다. <b>Windows</b>에는 이러한 드라이버가 포함되어 있지 않지만 게스트 도구 ISO 또는 가상 플로피 디스크에서 설치할 수 있습니다. <b>IDE</b> 장치에는 특수 드라이버가 필요하지 않습니다.</p> <p>디스크가 연결된 모든 가상 머신을 중지한 후 인터페이스 유형을 업데이트할 수 있습니다.</p>
데이터 센터	<p>이 필드는 부동 디스크를 생성할 때만 표시됩니다.</p> <p>가상 디스크를 사용할 수 있는 데이터 센터입니다.</p>
스토리지 도메인	가상 디스크를 저장할 스토리지 도메인입니다. 드롭다운 목록에는 지정된 데이터 센터에서 사용 가능한 모든 스토리지 도메인이 표시되며 스토리지 도메인의 총 공간과 현재 사용 가능한 공간도 표시됩니다.
할당 정책	<p>새 가상 디스크에 대한 프로비저닝 정책입니다.</p> <ul style="list-style-type: none"> <li>● <b>사전 할당</b> 은 가상 디스크가 생성될 때 스토리지 도메인에 디스크 전체 크기를 할당합니다. 가상 크기와 사전 할당된 디스크의 실제 크기는 동일합니다. 사전 할당된 가상 디스크는 썬 프로비저닝된 가상 디스크보다 생성하는 데 더 많은 시간이 소요되지만 읽기 및 쓰기 성능이 향상됩니다. 서버 및 기타 I/O 집약적 가상 머신에 사전 할당된 가상 디스크가 권장됩니다. 가상 머신에서 4초마다 1GB 이상을 쓸 수 있는 경우 가능한 경우 사전 할당된 디스크를 사용합니다.</li> <li>● <b>thin Provision</b> 은 가상 디스크가 생성될 때 1GB를 할당하고 디스크가 증가할 수 있는 크기에 최대 제한을 설정합니다. 디스크의 가상 크기는 최대 제한입니다. 디스크의 실제 크기는 지금까지 할당된 공간입니다. 썬 프로비저닝된 디스크는 사전 할당된 디스크보다 더 빠르게 생성되며 스토리지 과다 할당을 허용합니다. 썬 프로비저닝된 가상 디스크는 해당 데스크탑에 권장됩니다.</li> </ul>

필드 이름	설명
디스크 프로필	가상 디스크에 할당된 디스크 프로파일입니다. 디스크 프로파일은 스토리지 도메인의 가상 디스크에 대한 최대 처리량과 최대 수준의 입력 및 출력 작업을 정의합니다. 디스크 프로파일은 데이터 센터에 대해 생성된 서비스 항목의 스토리지 품질에 따라 스토리지 도메인 수준에서 정의됩니다.
디스크 활성화	이 필드는 연결된 디스크를 생성할 때만 표시됩니다.  생성 후 즉시 가상 디스크를 활성화합니다.
삭제 후 지우기	가상 디스크가 삭제될 때 중요한 자료 삭제를 위해 강화된 보안을 활성화할 수 있습니다.
부팅 가능	이 필드는 연결된 디스크를 생성할 때만 표시됩니다.  가상 디스크에서 부팅 가능한 플래그를 활성화할 수 있습니다.
공유 가능	가상 디스크를 한 번에 두 개 이상의 가상 머신에 연결할 수 있습니다.
읽기 전용	이 필드는 연결된 디스크를 생성할 때만 표시됩니다.  디스크를 읽기 전용으로 설정할 수 있습니다. 동일한 디스크를 하나의 가상 머신에 읽기 전용으로 연결하고 다른 가상 머신에 다시 연결할 수 있습니다.
Discard 활성화	이 필드는 연결된 디스크를 생성할 때만 표시됩니다.  가상 머신이 가동되는 동안 썸 프로비저닝 디스크를 줄일 수 있습니다. 블록 스토리지의 경우 기본 스토리지 장치에서는 삭제 호출을 지원해야 하며 기본 스토리지에서 discard_zeroes_data 속성을 지원하지 않는 한 옵션은 deleting과 함께 사용할 수 없습니다. 파일 스토리지의 경우 기본 파일 시스템 및 블록 장치는 삭제 호출을 지원해야 합니다. 모든 요구 사항이 충족되면 QEMU에서 실행되는 SCSI UNMAP 명령을 QEMU에서 기본 스토리지로 전달하여 사용하지 않는 공간을 확보합니다.

직접 **LUN** 설정은 대상 > **LUN** 또는 **LUN** > 대상에 표시할 수 있습니다. 대상 > **LUN**은 검색된 호스트에 따라 사용 가능한 **LUN**을 정렬하지만 **LUN** > 대상은 단일 **LUN** 목록을 표시합니다.

**Discover Targets (대상 검색)** 섹션에 필드를 입력하고 **Discover (검색)**를 클릭하여 대상 서버를 검색합니다. 그런 다음 **Login All** 버튼을 클릭하여 대상 서버에서 사용 가능한 **LUN**을 나열하고 각 **LUN** 옆



에 있는 라디오 버튼을 사용하여 추가할 **LUN**을 선택할 수 있습니다.

**LUN**을 가상 머신 하드 디스크 이미지로 직접 사용하면 가상 머신과 해당 데이터 간의 추상화 계층을 제거합니다.

직접 **LUN**을 가상 머신 하드 디스크 이미지로 사용하는 경우 다음 사항을 고려해야 합니다.

- 직접 **LUN** 하드 디스크 이미지의 실시간 스토리지 마이그레이션은 지원되지 않습니다.
- 직접 **LUN** 디스크는 가상 머신 내보내기에 포함되어 있지 않습니다.
- 직접 **LUN** 디스크는 가상 머신 스냅샷에 포함되어 있지 않습니다.

표 13.3. 새로운 가상 디스크 및 가상 디스크 설정 편집: 직접 **LUN**

필드 이름	설명
별칭	40자로 제한되는 가상 디스크의 이름입니다.
설명	가상 디스크에 대한 설명입니다. 이 필드는 권장되지만 필수는 아닙니다. 기본적으로 LUN ID의 마지막 4자가 필드에 삽입됩니다.  <b>PopulateECDHELUNDiskDescriptionWithLUNid</b> 구성 키를 <b>engine-config</b> 명령을 사용하여 적절한 값으로 설정하여 기본 동작을 구성할 수 있습니다. 전체 LUN ID를 사용할 경우 구성 키를 <b>-1</b> 로 설정하거나 이 기능을 무시하려면 <b>0</b> 을 설정할 수 있습니다. 양수 정수는 설명을 LUN ID의 해당 문자 수로 채웁니다.
인터페이스	이 필드는 연결된 디스크를 생성할 때만 표시됩니다.  디스크가 가상 머신에 제공하는 가상 인터페이스입니다. <b>virtio</b> 는 더 빠르지만 드라이버가 필요합니다. Red Hat Enterprise Linux 5 이상에는 이러한 드라이버가 포함되어 있습니다. Windows에는 이러한 드라이버가 포함되어 있지 않지만 게스트 도구 ISO 또는 가상 플로피 디스크에서 설치할 수 있습니다. <b>IDE</b> 장치에는 특수 드라이버가 필요하지 않습니다.  디스크가 연결된 모든 가상 머신을 중지한 후 인터페이스 유형을 업데이트할 수 있습니다.

필드 이름	설명
데이터 센터	<p>이 필드는 부동 디스크를 생성할 때만 표시됩니다.</p> <p>가상 디스크를 사용할 수 있는 데이터 센터입니다.</p>
호스트	<p>LUN이 마운트될 호스트입니다. 데이터 센터의 모든 호스트를 선택할 수 있습니다.</p>
스토리지 유형	<p>추가할 외부 LUN 유형입니다. <b>iSCSI</b> 또는 <b>Fibre 채널</b>에서 선택할 수 있습니다.</p>
대상 검색	<p>이 섹션은 iSCSI 외부 LUN 및 <b>대상</b>을 사용할 때 확장할 수 있습니다.</p> <p><b>address</b> - 대상 서버의 호스트 이름 또는 IP 주소입니다.</p> <p><b>포트</b> - 대상 서버에 대한 연결을 시도하는 포트입니다. 기본 포트는 3260입니다.</p> <p><b>사용자 인증</b> - iSCSI 서버에는 사용자 인증이 필요합니다. <b>사용자 인증</b> 필드는 iSCSI 외부 LUN을 사용할 때 표시됩니다.</p> <p><b>CHAP 사용자 이름</b> - LUN에 로그인할 수 있는 권한이 있는 사용자의 사용자 이름입니다. 이 필드는 <b>User Authentication</b> (사용자 인증) 확인란을 선택하면 액세스할 수 있습니다.</p> <p><b>CHAP 암호</b> - LUN에 로그인할 수 있는 권한이 있는 사용자의 암호입니다. 이 필드는 <b>User Authentication</b> (사용자 인증) 확인란을 선택하면 액세스할 수 있습니다.</p>
디스크 활성화	<p>이 필드는 연결된 디스크를 생성할 때만 표시됩니다.</p> <p>생성 후 즉시 가상 디스크를 활성화합니다.</p>
부팅 가능	<p>이 필드는 연결된 디스크를 생성할 때만 표시됩니다.</p> <p>가상 디스크에서 부팅 가능한 플래그를 활성화할 수 있습니다.</p>
공유 가능	<p>가상 디스크를 한 번에 두 개 이상의 가상 머신에 연결할 수 있습니다.</p>
읽기 전용	<p>이 필드는 연결된 디스크를 생성할 때만 표시됩니다.</p> <p>디스크를 읽기 전용으로 설정할 수 있습니다. 동일한 디스크를 하나의 가상 머신에 읽기 전용으로 연결하고 다른 가상 머신에 다시 연결할 수 있습니다.</p>

필드 이름	설명
Discard 활성화	이 필드는 연결된 디스크를 생성할 때만 표시됩니다.  가상 머신이 가동되는 동안 썬 프로비저닝 디스크를 줄일 수 있습니다. 이 옵션을 활성화하면 QEMU에서 발급한 SCSI UNMAP 명령을 기본 스토리지로 전달하여 사용하지 않는 공간을 확보합니다.
SCSI Pass-Through 활성화	이 필드는 연결된 디스크를 생성할 때만 표시됩니다.  인터페이스가 VirtIO-SCSI 로 설정된 경우 사용할 수 있습니다. 이 확인란을 선택하면 물리적 SCSI 장치를 가상 디스크로 통과할 수 있습니다. SCSI 패스투루가 활성화된 VirtIO-SCSI 인터페이스에는 SCSI 삭제 지원이 자동으로 포함됩니다. 이 확인란을 선택하면 읽기 전용 이 지원되지 않습니다.  이 확인란을 선택하지 않으면 가상 디스크는 에뮬레이션된 SCSI 장치를 사용합니다. 에뮬레이션된 VirtIO - SCSI 디스크에서 읽기 전용 이 지원됩니다.
권한이 있는 SCSI I/O 허용	이 필드는 연결된 디스크를 생성할 때만 표시됩니다.  SCSI Pass-Through 확인란을 선택할 때 사용할 수 있습니다. 이 확인란을 선택하면 필터링되지 않은 SCSI 일반 I/O(SG_IO) 액세스가 활성화되므로 디스크에서 권한이 부여된 SG_IO 명령이 허용됩니다. 이는 지속적인 예약에 필요합니다.
SCSI 사용	이 필드는 연결된 디스크를 생성할 때만 표시됩니다.  SCSI Pass-Through 및 Allow Privileged SCSI I/O 확인란을 선택하면 사용할 수 있습니다. 이 확인란을 선택하면 SCSI 예약을 사용하는 가상 머신이 디스크에 대한 액세스가 손실되지 않도록 이 디스크를 사용하는 가상 머신의 마이그레이션이 비활성화됩니다.

관련 데이터 센터에서 디스크를 만들 수 있는 권한이 있는 사용 가능한 **OpenStack** 볼륨 스토리지 도메인이 없는 경우 **Cinder** 설정 양식이 비활성화됩니다. **Cinder** 디스크는 외부 공급자 창을 사용하여 **Red Hat Virtualization** 환경에 추가된 **OpenStack** 볼륨 인스턴스에 액세스해야 합니다. 자세한 내용은 **14.2.4 절**. “스토리지 관리를 위해 **OpenStack Block Storage(Cinder)** 인스턴스 추가” 을 참조하십시오.

표 13.4. 새로운 가상 디스크 및 가상 디스크 설정 편집: **Cinder**

필드 이름	설명
크기(GB)	새 가상 디스크의 크기(GB)입니다.
별칭	40자로 제한되는 가상 디스크의 이름입니다.

필드 이름	설명
-------	----

<p><b>설명</b></p>	<p>가상 디스크에 대한 설명입니다. 이 필드는 권장되지만 필수는 아닙니다.</p>
<p><b>인터페이스</b></p>	<p>이 필드는 연결된 디스크를 생성할 때만 표시됩니다.</p> <p>디스크가 가상 머신에 제공하는 가상 인터페이스입니다. <b>virtio</b> 는 더 빠르지만 드라이버가 필요합니다. Red Hat Enterprise Linux 5 이상에는 이러한 드라이버가 포함되어 있습니다. Windows에는 이러한 드라이버가 포함되어 있지 않지만 게스트 도구 ISO 또는 가상 플로피 디스크에서 설치할 수 있습니다. <b>IDE</b> 장치에는 특수 드라이버가 필요하지 않습니다.</p> <p>디스크가 연결된 모든 가상 머신을 중지한 후 인터페이스 유형을 업데이트할 수 있습니다.</p>
<p><b>데이터 센터</b></p>	<p>이 필드는 부동 디스크를 생성할 때만 표시됩니다.</p> <p>가상 디스크를 사용할 수 있는 데이터 센터입니다.</p>
<p><b>스토리지 도메인</b></p>	<p>가상 디스크를 저장할 스토리지 도메인입니다. 드롭다운 목록에는 지정된 데이터 센터에서 사용 가능한 모든 스토리지 도메인이 표시되며 스토리지 도메인의 총 공간과 현재 사용 가능한 공간도 표시됩니다.</p>
<p><b>블록 유형</b></p>	<p>가상 디스크의 블록 유형입니다. 드롭다운 목록에는 사용 가능한 모든 블록 유형이 표시됩니다. 블록 유형은 OpenStack Cinder에서 관리 및 구성됩니다.</p>
<p><b>디스크 활성화</b></p>	<p>이 필드는 연결된 디스크를 생성할 때만 표시됩니다.</p> <p>생성 후 즉시 가상 디스크를 활성화합니다.</p>
<p><b>부팅 가능</b></p>	<p>이 필드는 연결된 디스크를 생성할 때만 표시됩니다.</p> <p>가상 디스크에서 부팅 가능한 플래그를 활성화할 수 있습니다.</p>
<p><b>공유 가능</b></p>	<p>가상 디스크를 한 번에 두 개 이상의 가상 머신에 연결할 수 있습니다.</p>
<p><b>읽기 전용</b></p>	<p>이 필드는 연결된 디스크를 생성할 때만 표시됩니다.</p> <p>디스크를 읽기 전용으로 설정할 수 있습니다. 동일한 디스크를 하나의 가상 머신에 읽기 전용으로 연결하고 다른 가상 머신에 다시 연결할 수 있습니다.</p>



## 중요

저널링된 파일 시스템을 마운트하려면 읽기-쓰기 액세스가 필요합니다. **Read-Only** 옵션을 사용하면 이러한 파일 시스템이 포함된 가상 디스크에 적합하지 않습니다(예: **EXT3,EXT4** 또는 **XFS**).

### 13.6.3. 실시간 스토리지 마이그레이션 개요

가상 디스크는 연결된 가상 머신이 실행되는 동안 하나의 스토리지 도메인에서 다른 스토리지 도메인으로 마이그레이션할 수 있습니다. 이를 실시간 스토리지 마이그레이션이라고 합니다. 실행 중인 가상 머신에 연결된 디스크가 마이그레이션되면 해당 디스크 이미지 체인의 스냅샷이 소스 스토리지 도메인에 생성되고 전체 이미지 체인이 대상 스토리지 도메인에 복제됩니다. 따라서 디스크 이미지 체인과 스냅샷을 모두 호스팅할 소스 스토리지 도메인과 대상 스토리지 도메인에 충분한 스토리지 공간이 있어야 합니다. 마이그레이션이 실패한 경우에도 각 실시간 스토리지 마이그레이션 시도에 새 스냅샷이 생성됩니다.

실시간 스토리지 마이그레이션을 사용할 때는 다음을 고려하십시오.

- 한 번에 여러 디스크를 실시간 마이그레이션할 수 있습니다.
- 동일한 가상 머신의 여러 디스크는 두 개 이상의 스토리지 도메인에 상주할 수 있지만 각 디스크의 이미지 체인은 단일 스토리지 도메인에 있어야 합니다.
- 동일한 데이터 센터의 두 스토리지 도메인 간에 디스크를 실시간 마이그레이션할 수 있습니다.
- 공유 가능으로 표시된 직접 **LUN** 하드 디스크 이미지 또는 디스크를 실시간으로 마이그레이션할 수 없습니다.

### 13.6.4. 가상 디스크 이동

가상 머신에 연결된 가상 디스크를 스토리지 도메인에서 다른 스토리지 도메인으로 이동합니다. 실행 중인 가상 머신에 연결된 가상 디스크를 이동할 수 있습니다. 이를 실시간 스토리지 마이그레이션이라고 합니다. 또는 계속하기 전에 가상 머신을 종료합니다.

디스크를 이동할 때는 다음을 고려하십시오.

- 여러 디스크를 동시에 이동할 수 있습니다.
- 동일한 데이터 센터의 두 스토리지 도메인 간에 디스크를 이동할 수 있습니다.
- 가상 디스크가 템플릿을 기반으로 생성된 가상 머신에 연결되어 썬 프로비저닝 스토리지 할당 옵션을 사용하는 경우 가상 시스템이 가상 디스크와 동일한 스토리지 도메인에 기반한 템플릿의 디스크를 복사해야 합니다.

### 가상 디스크 이동

1. **Storage** → **Disks** (스토리지 디스크)를 클릭하고 이동할 하나 이상의 가상 디스크를 선택합니다.
2. 이동을 클릭합니다.
3. 대상 목록에서 가상 디스크를 이동할 스토리지 도메인을 선택합니다.
4. 해당하는 경우 디스크 프로필 목록에서 디스크의 프로필을 선택합니다.
5. **OK**를 클릭합니다.

가상 디스크는 대상 스토리지 도메인으로 이동합니다. 이동 프로시저 중에 상태 열에 **Locked**가 표시되고 이동 작업의 진행 상황을 나타내는 진행률 표시줄이 표시됩니다. **During the move procedure, the Status column displays Locked and a progress bar indicating the progress of the move operation.**

#### 13.6.5. 디스크 인터페이스 유형 변경

사용자는 디스크를 생성한 후 디스크의 인터페이스 유형을 변경할 수 있습니다. 이를 통해 다른 인터페이스 유형이 필요한 가상 머신에 기존 디스크를 연결할 수 있습니다. 예를 들어 **VirtIO** 인터페이스를 사용하는 디스크를 **VirtIO-SCSI** 또는 **IDE** 인터페이스가 필요한 가상 머신에 연결할 수 있습니다. 이를 통해 백업 및 복원 또는 재해 복구를 위해 디스크를 마이그레이션할 수 있습니다. 공유 가능한 디스크의 디스크 인터페이스도 가상 머신별로 업데이트할 수 있습니다. 즉, 공유 디스크를 사용하는 각 가상 머신에서 다른 인터페이스 유형을 사용할 수 있습니다.

디스크 인터페이스 유형을 업데이트하려면 디스크를 사용하는 모든 가상 머신을 먼저 중지해야 합니다.

#### 디스크 인터페이스 유형 변경

1. **Compute → Virtual Machines (가상 머신)**를 클릭하고 적절한 가상 머신을 중지합니다.
2. 가상 머신의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 디스크 탭을 클릭하고 디스크를 선택합니다.
4. 편집 을 클릭합니다.
5. 인터페이스 목록에서 새 인터페이스 유형을 선택하고 확인을 클릭합니다.

다른 인터페이스 유형이 필요한 다른 가상 머신에 디스크를 연결할 수 있습니다.

#### 다른 인터페이스 유형을 사용하여 다른 가상 머신에 디스크 연결

1. **Compute → Virtual Machines (가상 머신)**를 클릭하고 적절한 가상 머신을 중지합니다.
2. 가상 머신의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 디스크 탭을 클릭하고 디스크를 선택합니다.
4. 제거를 클릭한 다음 확인을 클릭합니다.
5. 가상 머신으로 돌아가서 디스크가 연결될 새 가상 머신의 이름을 클릭합니다.

6. 디스크 탭을 클릭한 다음 연결 을 클릭합니다.
7. **Attach Virtual Disks** (가상 디스크 연결) 창에서 디스크를 선택하고 인터페이스 드롭다운에서 적절한 인터페이스를 선택합니다.
8. **OK**를 클릭합니다.

### 13.6.6. 가상 디스크 복사

한 스토리지 도메인에서 다른 스토리지 도메인으로 가상 디스크를 복사할 수 있습니다. 복사된 디스크는 가상 머신에 연결할 수 있습니다.

#### 가상 디스크 복사

1. **Storage** → **Disks** (스토리지 디스크)를 클릭하고 가상 디스크를 선택합니다.
2. 복사 를 클릭합니다.
3. 선택적으로 **Alias** 필드에 새 이름을 입력합니다.
4. 대상 목록에서 가상 디스크를 복사할 스토리지 도메인을 선택합니다.
5. 해당하는 경우 디스크 프로필 목록에서 디스크의 프로필을 선택합니다.
6. **OK**를 클릭합니다.

가상 디스크는 복사하는 동안 **Locked** 의 상태가 됩니다.

### 13.6.7. 데이터 스토리지 도메인에 이미지 업로드

관리 포털 또는 **REST API**를 사용하여 가상 디스크 이미지와 **ISO** 이미지를 데이터 스토리지 도메인에



업로드할 수 있습니다. **11.8.1절. “데이터 스토리지 도메인에 이미지 업로드”** 을 참조하십시오.

### 13.6.8. 가져오기 스토리지 도메인에서 디스크 이미지 가져오기

가져온 스토리지 도메인에서 유동 가상 디스크를 가져옵니다.



참고

**QEMU 호환 디스크만 Manager**로 가져올 수 있습니다.

#### 디스크 이미지 가져오기

1. 스토리지 도메인을 클릭합니다.
2. 가져온 스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 디스크 가져오기 탭을 클릭합니다.
4. 하나 이상의 디스크를 선택하고 가져오기 를 클릭합니다.
5. 각 디스크에 적절한 디스크 프로필을 선택합니다.
6. **OK**를 클릭합니다.

### 13.6.9. 가져온 스토리지 도메인에서 등록되지 않은 디스크 이미지 가져오기

스토리지 도메인에서 유동 가상 디스크를 가져옵니다. **Red Hat Virtualization** 환경 외부에서 생성된 부동 디스크는 **Manager**에 등록되지 않습니다. 스토리지 도메인을 스캔하여 가져올 등록되지 않은 유동 디스크를 식별합니다.



참고

**QEMU** 호환 디스크만 **Manager**로 가져올 수 있습니다.

#### 디스크 이미지 가져오기

1. 스토리지 도메인을 클릭합니다.
2. 스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. **More Actions (**  
⋮  
**)**를 클릭한 다음, **Manager**가 등록되지 않은 디스크를 식별할 수 있도록 디스크 스캔을 클릭합니다.
4. 디스크 가져오기 탭을 클릭합니다.
5. 하나 이상의 디스크 이미지를 선택하고 가져오기를 클릭합니다.
6. 각 디스크에 적절한 디스크 프로필을 선택합니다.
7. **OK**를 클릭합니다.

#### 13.6.10. OpenStack 이미지 서비스에서 가상 디스크 가져오기

**OpenStack Image Service**에서 관리하는 가상 디스크는 **OpenStack Image Service**를 외부 공급자로 **Manager**에 추가하는 경우 **Red Hat Virtualization Manager**로 가져올 수 있습니다.

1. 스토리지 도메인을 클릭합니다.
2. **OpenStack Image Service** 도메인 이름을 클릭하여 세부 정보 보기를 엽니다.

3. 이미지 탭을 클릭하고 이미지를 선택합니다.
4. 가져오기 를 클릭합니다.
5. 이미지를 가져올 데이터 센터를 선택합니다.
6. **Domain Name** 드롭다운 목록에서 이미지가 저장될 스토리지 도메인을 선택합니다.
7. 필요한 경우 **Quota (할당량)** 드롭다운 목록에서 이미지에 적용할 할당량을 선택합니다.
8. **OK**를 클릭합니다.

이제 가상 머신에 디스크를 연결할 수 있습니다.


### 13.6.11. OpenStack 이미지 서비스로 가상 디스크 내보내기

가상 디스크는 **Manager**에 외부 공급자로 추가된 **OpenStack** 이미지 서비스로 내보낼 수 있습니다.



#### 중요

가상 디스크는 볼륨이 여러 개 없고 썸 프로비저닝되지 않으며 스냅샷이 없는 경우에만 내보낼 수 있습니다.

1. **Storage** → **Disks** (스토리지 디스크)를 클릭하고 내보낼 디스크를 선택합니다.
2. 더 많은 작업 (  )을 클릭한 다음 내보내기 를 클릭합니다.
3. **Domain Name** 드롭다운 목록에서 디스크를 내보낼 **OpenStack Image** 서비스를 선택합니다.

4. 할당량을 적용하려면 **Quota (할당량)** 드롭다운 목록에서 디스크에 대한 할당량을 선택합니다.
5. **OK**를 클릭합니다.

### 13.6.12. 가상 디스크 공간 회수

썬 프로비저닝을 사용하는 가상 디스크는 파일을 삭제한 후 자동으로 축소되지 않습니다. 예를 들어 실제 디스크 크기가 **100GB**이고 **50GB**의 파일을 삭제하는 경우 할당된 디스크 크기는 **100GB**로 유지되고 나머지 **50GB**는 호스트에 반환되지 않으므로 다른 가상 머신에서 사용할 수 없습니다. 이 사용되지 않은 디스크 공간은 가상 머신의 디스크에서 **스arsify** 작업을 수행하여 호스트에서 회수할 수 있습니다. 이렇게 하면 디스크 이미지에서 호스트로 사용 가능한 공간이 전송됩니다. 여러 가상 디스크를 병렬로 정렬할 수 있습니다.


**Red Hat**은 가상 머신을 복제하거나, 가상 머신을 기반으로 템플릿을 생성하거나 스토리지 도메인의 디스크 공간을 정리하기 전에 이 작업을 수행할 것을 권장합니다.

#### 제한 사항

- **NFS** 스토리지 도메인은 **NFS** 버전 **4.2** 이상을 사용해야 합니다.
- 직접 **LUN** 또는 **Cinder**를 사용하는 디스크를 **스arsify**할 수 없습니다.
- 사전 할당된 할당 정책을 사용하는 디스크는 비정규화할 수 없습니다. 템플릿에서 가상 머신을 생성하는 경우 스토리지 필드에서 **Thin**을 선택하거나 **Clone**을 선택하는 경우 템플릿이 썬 프로비저닝이 있는 가상 머신을 기반으로 해야 합니다.
- 활성 스냅샷만 급증할 수 있습니다.

#### 디스크 구문 분석

1. 컴퓨팅 → 가상 머신을 클릭하고 필요한 가상 머신을 종료합니다.

2. 가상 머신의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 디스크 탭을 클릭합니다. 디스크 상태가 **OK** 인지 확인합니다.
4. 더 많은 작업 (  )을 클릭한 다음 분리를 클릭합니다.
5. **OK**를 클릭합니다.

이벤트 시작은 **sparsify** 작업 중에 이벤트 탭에 표시되고 디스크의 상태가 **Locked** 로 표시됩니다. 작업이 완료되면 **Sparsified successfully** 이벤트가 **Events** 탭에 표시되고 디스크의 상태가 **OK** 로 표시됩니다. 사용되지 않은 디스크 공간이 호스트로 반환되었으며 다른 가상 머신에서 사용할 수 있습니다.

## 14장. 외부 공급자

### 14.1. RED HAT VIRTUALIZATION의 외부 공급자 소개

**Red Hat Virtualization Manager** 자체에서 관리하는 리소스 외에도 **Red Hat Virtualization**은 외부 소스에서 관리하는 리소스를 활용할 수 있습니다. 외부 공급자라는 이러한 리소스의 공급자는 가상화 호스트, 가상 시스템 이미지 및 네트워크와 같은 리소스를 제공할 수 있습니다.

**Red Hat Virtualization**에서는 현재 다음과 같은 외부 공급자를 지원합니다.

#### Red Hat Satellite for Host Provisioning

**Satellite**는 물리적 호스트와 가상 호스트의 라이프사이클의 모든 측면을 관리하는 틀입니다. **Red Hat Virtualization**에서는 **Red Hat Virtualization Manager**에서 **Satellite**에서 관리하는 호스트를 가상화 호스트로 추가하고 사용할 수 있습니다. **Manager**에 **Satellite** 인스턴스를 추가한 후 새 호스트를 추가할 때 해당 **Satellite** 인스턴스에서 사용 가능한 호스트를 검색하여 **Satellite** 인스턴스에서 관리하는 호스트를 추가할 수 있습니다. **Red Hat Satellite** 설치 및 **Red Hat Satellite**를 사용하여 호스트 관리에 대한 자세한 내용은 [Red Hat Satellite 설치 가이드](#) 및 [Red Hat Satellite 호스트 구성 가이드](#)를 참조하십시오.

#### 이미지 관리를 위한 OpenStack Image Service(Glance)

**OpenStack Image Service**는 가상 머신 이미지 카탈로그를 제공합니다. **Red Hat Virtualization**에서는 이러한 이미지를 **Red Hat Virtualization Manager**로 가져오고 부동 디스크로 사용하거나 가상 머신에 연결하여 템플릿으로 변환할 수 있습니다. **Manager**에 **OpenStack Image Service**를 추가한 후 데이터 센터에 연결되지 않은 스토리지 도메인으로 표시됩니다. **Red Hat Virtualization** 환경의 가상 디스크는 가상 디스크로 **OpenStack Image Service**에 내보낼 수도 있습니다.

#### 네트워크 프로비저닝을 위한 OpenStack Networking(Neutron)

**OpenStack Networking**은 소프트웨어 정의 네트워크를 제공합니다. **Red Hat Virtualization**에서는 **OpenStack Networking**에서 제공하는 네트워크를 **Red Hat Virtualization Manager**로 가져올 수 있으며 모든 유형의 트래픽을 전송하고 복잡한 네트워크 토폴로지를 생성하는 데 사용됩니다. **Manager**에 **OpenStack Networking**을 추가한 후에는 **OpenStack Networking**에서 제공하는 네트워크에 수동으로 가져와서 액세스할 수 있습니다.

#### 스토리지 관리를 위한 OpenStack 볼륨(Cinder)

**OpenStack Volume**은 가상 하드 드라이브의 영구 블록 스토리지 관리를 제공합니다. **OpenStack Cinder** 볼륨은 **Ceph Storage**에서 프로비저닝합니다. **Red Hat Virtualization**에서는 부동 디스크로 사용하거나 가상 머신에 연결할 수 있는 **OpenStack** 볼륨 스토리지에 디스크를 만들 수 있습니다. **Manager**에 **OpenStack Volume**을 추가한 후 **OpenStack Volume**에서 제공하는 스토리지에 디스크를 생성할 수 있습니다.

#### VMware for Virtual Machine Provisioning

**VMware**에서 생성된 가상 머신은 **V2V(virt-v2v)**를 사용하여 변환하고 **Red Hat Virtualization** 환경으로 가져올 수 있습니다. **Manager**에 **VMware** 공급자를 추가한 후 제공하는 가상 머신을 가져올 수

있습니다. **V2V** 변환은 지정된 프록시 호스트에서 가져오기 작업의 일부로 수행됩니다.

### 가상 머신 프로비저닝을 위한 **RHEL 5ECDHE**

**RHEL 5ECDHE**에서 생성된 가상 머신은 **V2V(virt-v2v)**를 사용하여 변환하고 **Red Hat Virtualization** 환경으로 가져올 수 있습니다. **RHEL 5ECDHE** 호스트를 **Manager**에 추가한 후 제공하는 가상 머신을 가져올 수 있습니다. **V2V** 변환은 지정된 프록시 호스트에서 가져오기 작업의 일부로 수행됩니다.

### 가상 머신 프로비저닝 **KVM**

**KVM**에서 생성된 가상 머신을 **Red Hat Virtualization** 환경으로 가져올 수 있습니다. **Manager**에 **KVM** 호스트를 추가한 후 제공하는 가상 머신을 가져올 수 있습니다.

### 네트워크 프로비저닝을 위한 **OVN(Open Virtual Network)**

**OVN(Open Virtual Network)**은 소프트웨어 정의 네트워크를 제공하는 **OVS(Open vSwitch)** 확장입니다. **Manager**에 **OVN**을 추가한 후 기존 **OVN** 네트워크를 가져와서 **Manager**에서 새 **OVN** 네트워크를 생성할 수 있습니다. **engine-setup**을 사용하여 **Manager**에 **OVN**을 자동으로 설치할 수도 있습니다.

### 네트워크 프로비저닝을 위한 외부 네트워크 공급자

지원되는 외부 **software-defined** 네트워크 공급자에는 **OpenStack Neutron REST API**를 구현하는 모든 공급자가 포함됩니다. **OpenStack Networking(Neutron)**과 달리 **Neutron** 에이전트는 호스트에서 가상 인터페이스 드라이버 구현으로 사용되지 않습니다. 대신 외부 네트워크 공급자의 구현자가 가상 인터페이스 드라이버를 제공해야 합니다.

모든 외부 리소스 공급자는 입력에 적응할 단일 창을 사용하여 추가됩니다. **Red Hat Virtualization** 환경에서 제공하는 리소스를 사용하기 전에 리소스 공급자를 추가해야 합니다.

## 14.2. 외부 공급자 추가

### 14.2.1. 호스트 프로비저닝을 위한 **Red Hat Satellite** 인스턴스 추가

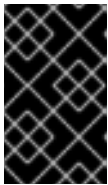
**Red Hat Virtualization Manager**에 호스트 프로비저닝을 위한 **Satellite** 인스턴스를 추가합니다. **Red Hat Virtualization 4.2**는 **Red Hat Satellite 6.1**에서 지원됩니다.

#### 호스트 프로비저닝을 위한 **Satellite** 인스턴스 추가

1.

**Administration** → **Providers** (관리 공급자)를 클릭합니다.

2. 추가를 클릭합니다.
3. 이름 및 설명을 입력합니다.
4. 유형 드롭다운 목록에서 **FlexVolume /Satellite** 를 선택합니다.
5. 공급자 **URL** 텍스트 필드에 **Satellite** 인스턴스가 설치된 시스템의 **URL** 또는 정규화된 도메인 이름을 입력합니다. 포트 번호를 지정할 필요가 없습니다.



중요

**IP** 주소는 **Satellite** 인스턴스를 추가하는 데 사용할 수 없습니다.

6. **Requires Authentication (인증 필요)** 확인란을 선택합니다.
7. **Username (사용자 이름)** 및 **Password (암호)**를 **Satellite** 인스턴스에 입력합니다. **Satellite** 프로비저닝 포털에 로그인하는 데 사용하는 것과 동일한 사용자 이름 및 암호를 사용해야 합니다.
8. 자격 증명을 테스트합니다.
  - a. 테스트를 클릭하여 제공된 자격 증명을 사용하여 **Satellite** 인스턴스에서 성공적으로 인증할 수 있는지 테스트합니다.
  - b. **Satellite** 인스턴스에서 **SSL**을 사용하는 경우 공급자 인증서 가져오기 창이 열립니다. 확인을 클릭하여 관리자가 인스턴스와 통신할 수 있도록 **Satellite** 인스턴스에서 제공하는 인증서를 가져옵니다.
9. **OK**를 클릭합니다.

#### 14.2.2. 이미지 관리를 위해 **OpenStack Image(Glance)** 인스턴스 추가

이미지 관리를 위해 **OpenStack Image(Glance)** 인스턴스를 **Red Hat Virtualization Manager**에 추가합니다.



이미지 관리를 위해 **OpenStack Image(Glance)** 인스턴스 추가

1. **Administration** → **Providers** (관리 공급자)를 클릭합니다.
2. 추가를 클릭하고 일반 설정 탭에 세부 정보를 입력합니다. 이러한 필드에 대한 자세한 내용은 **14.2.10절. “설명되는 공급자 일반 설정 추가”** 을 참조하십시오.
3. 이름 및 설명을 입력합니다.
4. 유형 드롭다운 목록에서 **OpenStack Image** 를 선택합니다.
5. **OpenStack Image** 인스턴스가 공급자 **URL** 텍스트 필드에 설치된 시스템의 **URL** 또는 정규화된 도메인 이름을 입력합니다.
6. 선택적으로 **Requires Authentication** (인증 필요) 확인란을 선택하고 **Keystone**에 등록된 **OpenStack Image** 인스턴스 사용자에게 대해 **Username** 및 **Password** 를 입력합니다. 또한 프로토콜 (**HTTP**), 호스트 이름, **API** 포트를 정의하여 **Keystone** 서버의 인증 **URL** 을 정의해야 합니다.  
  
**OpenStack Image** 인스턴스의 테넌트 를 입력합니다.
7. 자격 증명을 테스트합니다.
  - a. 테스트를 클릭하여 제공된 자격 증명을 사용하여 **OpenStack Image** 인스턴스로 성공적으로 인증할 수 있는지 테스트합니다.
  - b. **OpenStack Image** 인스턴스에서 **SSL** 을 사용하는 경우 공급자 인증서 가져오기 창이 열립니다. 확인을 클릭하여 **Manager**가 인스턴스와 통신할 수 있도록 **OpenStack Image** 인스턴스에서 제공하는 인증서를 가져옵니다.
8. **OK**를 클릭합니다.

### 14.2.3. 네트워크 프로비저닝을 위해 OpenStack Networking(Neutron) 인스턴스 추가

Red Hat Virtualization Manager에 네트워크 프로비저닝을 위한 OpenStack Networking(neutron) 인스턴스를 추가합니다. OpenStack Neutron REST API를 구현하는 다른 타사 네트워크 공급자를 추가하려면 14.2.9절. “외부 네트워크 공급자 추가” 을 참조하십시오.

중요

Red Hat Virtualization은 Red Hat OpenStack Platform 버전 10, 13 및 14를 외부 네트워크 공급자로 지원합니다.

- OpenStack 10은 OVS 드라이버를 사용하여 배포해야 합니다.
- OpenStack 13은 OVS, OVN 또는 ODL 드라이버를 사용하여 배포해야 합니다.
- OpenStack 14는 OVN 또는 ODL 드라이버를 사용하여 배포해야 합니다.

neutron 네트워크를 사용하려면 호스트에 neutron 에이전트가 구성되어 있어야 합니다. 에이전트를 수동으로 구성하거나 Red Hat OpenStack Platform director를 사용하여 네트워크 노드를 호스트로 추가하기 전에 Networker 역할을 배포할 수 있습니다. director를 사용하는 것이 좋습니다. New Host (새 호스트) 창의 Network Provider 탭을 통해 neutron 에이전트를 자동으로 배포할 수 없습니다.

동일한 클러스터에서 네트워크 노드와 일반 호스트를 사용할 수 있지만 neutron 네트워크를 사용하는 가상 머신은 네트워크 노드에서만 실행할 수 있습니다.

네트워크 노드를 호스트로 추가

1. Red Hat OpenStack Platform director를 사용하여 네트워크 노드에 Networker 역할을 배포합니다. Red Hat OpenStack Platform Advanced Overcloud Customization 가이드에서 새 역할 및 네트워크 생성 을 참조하십시오.
2. 필요한 리포지토리를 활성화합니다.
  - a. 메시지가 표시되면 Content Delivery Network에 시스템을 등록하고 고객 포털 사용자

이름과 암호를 입력합니다.

```
# subscription-manager register
```

- b. **Red Hat Enterprise Linux Server** 및 **Red Hat Virtualization** 서브스크립션 풀을 찾아 풀 ID를 기록합니다.

```
# subscription-manager list --available
```

- c. 풀 ID를 사용하여 서브스크립션을 시스템에 연결합니다.

```
# subscription-manager attach --pool=poolid
```

- d. 리포지토리를 구성합니다.

```
# subscription-manager repos \
  --disable='*' \
  --enable=rhel-7-server-rpms \
  --enable=rhel-7-server-rhv-4-mgmt-agent-rpms \
  --enable=rhel-7-server-ansible-2.9-rpms
```

- e. 현재 설치된 모든 패키지가 최신 상태인지 확인합니다.

```
# yum update
```

- f. 커널 패키지가 업데이트된 경우 시스템을 재부팅합니다.

3. **Openstack Networking** 후크를 설치합니다.

```
# yum install vdsm-hook-openstacknet
```

4. 네트워크 노드를 **Manager**에 호스트로 추가합니다. **10.5.1 절. “Red Hat Virtualization Manager에 표준 호스트 추가”** 을 참조하십시오.

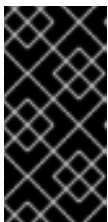


중요

네트워크 공급자 탭에서 **OpenStack Networking** 공급자를 선택하지 마십시오. 이는 현재 지원되지 않습니다.

네트워크 프로비저닝을 위해 **OpenStack Networking(Neutron)** 인스턴스 추가

1. **Administration** → **Providers** (관리 공급자)를 클릭합니다.
2. 추가를 클릭하고 일반 설정 탭에 세부 정보를 입력합니다. 이러한 필드에 대한 자세한 내용은 **14.2.10절. “설명되는 공급자 일반 설정 추가”** 을 참조하십시오.
3. 이름 및 설명을 입력합니다.
4. 유형 드롭다운 목록에서 **OpenStack Networking** 을 선택합니다.
5. 네트워킹 플러그인 필드에서 **Open vSwitch** 가 선택되어 있는지 확인합니다.
6. 필요한 경우 자동 동기화 확인란을 선택합니다. 이를 통해 외부 네트워크 공급자를 기존 네트워크와 자동 동기화할 수 있습니다.
7. **OpenStack Networking** 인스턴스가 설치된 시스템의 **URL** 또는 정규화된 도메인 이름을 공급자 **URL** 텍스트 필드에 입력한 다음 포트 번호를 입력합니다. 읽기 전용 확인란이 기본적으로 선택됩니다. 이로 인해 사용자가 **OpenStack Networking** 인스턴스를 수정하지 않습니다.



중요

**Red Hat**에서 지원하려면 설정을 위해 **Read-Only** 확인란을 선택해야 합니다.

8. 선택적으로 **Requires Authentication** (인증 필요) 확인란을 선택하고 **Keystone**에 등록된 **OpenStack Networking** 사용자의 **Username** 및 **Password** 를 입력합니다. 또한 프로토콜, 호스트 이름, **API** 포트 및 **API** 버전을 정의하여 **Keystone** 서버의 인증 **URL** 을 정의해야 합니다.

**API 버전 2.0**의 경우 **OpenStack Networking** 인스턴스의 테넌트 를 입력합니다. **API 버전 3**의 경우 사용자 도메인 이름, 프로젝트 이름 및 프로젝트 도메인 이름을 입력합니다.

9.

자격 증명을 테스트합니다.

a.

테스트를 클릭하여 제공된 자격 증명을 사용하여 **OpenStack Networking** 인스턴스로 성공적으로 인증할 수 있는지 테스트합니다.

b.

**OpenStack Networking** 인스턴스에서 **SSL**을 사용하는 경우 공급자 인증서 가져오기 창이 열립니다. **OK** 를 클릭하여 **OpenStack Networking** 인스턴스에서 제공하는 인증서를 가져와서 **Manager**가 인스턴스와 통신할 수 있는지 확인합니다.

10.

에이전트 구성 탭을 클릭합니다.



주의

다음 단계는 기술 프리뷰로만 제공됩니다. **Red Hat Virtualization**은 사전 구성된 **neutron** 호스트만 지원합니다.

11.

**Interface Mappings** 필드에 **Open vSwitch** 에이전트의 인터페이스 매핑 목록을 쉼표로 구분하여 입력합니다.

12.

브로커 유형 목록에서 **OpenStack Networking** 인스턴스가 사용하는 메시지 브로커 유형을 선택합니다.

13.

메시지 브로커가 호스트 필드에서 호스팅되는 호스트의 **URL** 또는 정규화된 도메인 이름을 입력합니다.

14.

메시지 브로커에 연결할 포트 를 입력합니다. 이 포트 번호는 기본적으로 메시지 브로커가 **SSL**을 사용하도록 구성되지 않고 **SSL**을 사용하도록 구성된 경우 **5761**입니다.

15. 메시지 브로커 인스턴스에 등록된 **OpenStack Networking** 사용자의 **Username** 및 **Password** 를 입력합니다.
16. **OK**를 클릭합니다.

**Red Hat Virtualization Manager**에 **OpenStack Networking** 인스턴스가 추가되었습니다. 제공하는 네트워크를 사용하려면 먼저 **Manager**로 네트워크를 가져옵니다. **9.3.1절. “외부 공급자에서 네트워크 가져오기”** 을 참조하십시오.

#### 14.2.4. 스토리지 관리를 위해 **OpenStack Block Storage(Cinder)** 인스턴스 추가



##### 중요

스토리지 관리에 **OpenStack Block Storage(Cinder)** 인스턴스를 사용하는 것은 기술 프리뷰 기능입니다. 기술 프리뷰 기능은 **Red Hat** 프로덕션 서비스 수준 계약(**SLA**)에서 지원되지 않으며, 기능상 완전하지 않을 수 있어 프로덕션에 사용하지 않는 것이 좋습니다. 이러한 기능을 사용하면 향후 제품 기능을 조기에 이용할 수 있어 개발 과정에서 고객이 기능을 테스트하고 피드백을 제공할 수 있습니다.

**Red Hat** 기술 프리뷰 기능 지원 범위에 대한 자세한 내용은 <https://access.redhat.com/support/offerings/techpreview/>를 참조하십시오.

**Red Hat Virtualization Manager**에 스토리지 관리를 위한 **OpenStack Block Storage(Cinder)** 인스턴스를 추가합니다. **OpenStack Cinder** 볼륨은 **Ceph Storage**에서 프로비저닝합니다.

#### 스토리지 관리를 위해 **OpenStack Block Storage(Cinder)** 인스턴스 추가

1. **Administration** → **Providers** (관리 공급자)를 클릭합니다.
2. 추가를 클릭하고 일반 설정 탭에 세부 정보를 입력합니다. 이러한 필드에 대한 자세한 내용은 **14.2.10절. “설명되는 공급자 일반 설정 추가”** 을 참조하십시오.
3. 이름 및 설명을 입력합니다.

4. 유형 드롭다운 목록에서 **OpenStack Block Storage** 를 선택합니다.
5. **OpenStack Block Storage** 볼륨이 연결될 데이터 센터를 선택합니다.
6. **OpenStack Block Storage** 인스턴스가 설치된 시스템의 **URL** 또는 정규화된 도메인 이름을 입력하고 공급자 **URL** 텍스트 필드에 포트 번호를 입력합니다.
7. 필요한 경우 **Authentication(인증 필요)** 확인란을 선택하고 **Keystone**에 등록된 **OpenStack Block Storage** 인스턴스 사용자에게 대해 **Username** 및 **Password** 를 입력합니다. 프로토콜 (**HTTP**), 호스트 이름, **API** 포트 를 정의하여 **Keystone** 서버의 인증 **URL** 을 정의합니다.  
  
**OpenStack Block Storage** 인스턴스의 테넌트 를 입력합니다.
8. 테스트를 클릭하여 제공된 자격 증명을 사용하여 **OpenStack Block Storage** 인스턴스로 성공적으로 인증할 수 있는지 테스트합니다.
9. **OK**를 클릭합니다.
10. 클라이언트 **Ceph** 인증(**cephx**)이 활성화된 경우 다음 단계도 완료해야 합니다. **cephx** 프로토콜은 기본적으로 활성화되어 있습니다.
  - a. **Ceph** 서버에서 **ceph auth get-or-create** 명령을 사용하여 **client.cinder** 사용자에게 대한 새 시크릿 키를 생성합니다. **cephx** 에 대한 자세한 내용은 **Cephx 구성 참조** 및 새 사용자의 키 생성에 대한 자세한 내용은 사용자 관리를 참조하십시오. **client.cinder** 사용자에게 대한 키가 이미 있는 경우 동일한 명령을 사용하여 검색합니다.
  - b. 관리 포털의 공급자 목록에서 새로 생성된 **Cinder** 외부 공급자를 선택합니다.
  - c. 인증 키 탭을 클릭합니다.
  - d. 새로 생성을 클릭합니다.

- e. **Value** 필드에 시크릿 키를 입력합니다.
- f. 자동으로 생성된 **UUID** 를 복사하거나 텍스트 필드에 기존 **UUID**를 입력합니다.
- g. **Cinder** 서버에서 이전 단계의 **UUID**와 **cinder** 사용자를 **/etc/cinder/cinder.conf** 에 추가합니다.

```

rbd_secret_uuid = UUID
rbd_user = cinder
    
```

**OpenStack Block Storage(Cinder)** 디스크 생성에 대한 자세한 내용은 **13.6.1절. “가상 디스크 생성”** 을 참조하십시오.

#### 14.2.5. VMware 인스턴스를 가상 머신 공급자로 추가

**VMware**에서 **Red Hat Virtualization Manager**로 가상 머신을 가져올 **VMware vCenter** 인스턴스를 추가합니다.

**Red Hat Virtualization**은 **V2V**를 사용하여 **VMware** 가상 머신을 가져오기 전에 올바른 형식으로 변환합니다. **virt-v2v** 패키지가 하나 이상의 호스트에 설치되어 있어야 합니다. **virt-v2v** 패키지는 기본적으로 **RHVH(Red Hat Virtualization Host)**에서 사용할 수 있으며 **Red Hat Virtualization** 환경에 추가할 때 **VDSM**의 종속성으로 **Red Hat Enterprise Linux** 호스트에 설치됩니다. **Red Hat Enterprise Linux** 호스트는 **Red Hat Enterprise Linux 7.2** 이상이어야 합니다.



참고

**virt-v2v** 패키지는 **ppc64le** 아키텍처에서 사용할 수 없습니다. 이러한 호스트는 프록시 호스트로 사용할 수 없습니다.

#### **VMware vCenter** 인스턴스를 가상 머신 공급자로 추가

1. **Administration** → **Providers (관리 공급자)**를 클릭합니다.
2. 추가를 클릭합니다.



3. 이름 및 설명을 입력합니다.
4. 유형 드롭다운 목록에서 **VMware** 를 선택합니다.
5. **VMware** 가상 머신을 가져올 데이터 센터를 선택하거나 대신 개별 가져오기 작업 중에 대상 데이터 센터를 지정할 데이터 센터를 선택합니다.
6. **vCenter** 필드에 **VMware vCenter** 인스턴스의 **IP** 주소 또는 정규화된 도메인 이름을 입력합니다.
7. **ESXi** 필드에서 가상 머신을 가져올 호스트의 **IP** 주소 또는 정규화된 도메인 이름을 입력합니다.
8. 지정된 **ESXi** 호스트가 데이터 센터 필드에 상주하는 데이터 센터의 이름을 입력합니다.
9. **ESXi** 호스트와 **Manager** 간에 **SSL** 인증서를 교환한 경우 **Verify server**의 **SSL** 인증서 확인란을 선택하여 **ESXi** 호스트의 인증서를 확인합니다. 그렇지 않은 경우 확인란을 지웁니다.
10. 가상 머신 가져오기 작업 중에 **virt-v2v** 가 설치된 선택한 데이터 센터에서 호스트를 선택하여 프록시 호스트 역할을 합니다. 이 호스트는 **VMware vCenter** 외부 공급자의 네트워크에 연결할 수도 있어야 합니다. 위의 모든 데이터 센터를 선택한 경우 여기에서 호스트를 선택할 수 없지만 개별 가져오기 작업 중에 호스트를 지정할 수 있습니다.
11. **VMware vCenter** 인스턴스에 **Username** 및 **Password** 를 입력합니다. 사용자는 가상 시스템이 상주하는 **VMware** 데이터 센터 및 **ESXi** 호스트에 액세스할 수 있어야 합니다.
12. 자격 증명을 테스트합니다.
  - a. 테스트를 클릭하여 제공된 자격 증명을 사용하여 **VMware vCenter** 인스턴스로 성공적으로 인증할 수 있는지 테스트합니다.
  - b. **VMware vCenter** 인스턴스에서 **SSL**을 사용하는 경우 공급자 인증서 가져오기 창이 열립니다. 확인 을 클릭하여 **VMware vCenter** 인스턴스에서 제공하는 인증서를 가져와서

**Manager**가 인스턴스와 통신할 수 있는지 확인합니다.

13.

**OK**를 클릭합니다.

**VMware** 외부 공급자에서 가상 머신을 가져오려면 가상 머신 관리 가이드의 **VMware** 공급자에서 가상 머신 가져오기를 참조하십시오.

#### 14.2.6. RHEL 5ECDHE Host를 가상 머신 공급자로 추가

**RHEL 5ECDHE** 호스트를 추가하여**ECDHE**에서 **Red Hat Virtualization**으로 가상 머신을 가져옵니다.

**Red Hat Virtualization**에서는 **V2V**를 사용하여 **RHEL 5ECDHE** 가상 머신을 가져오기 전에 올바른 형식으로 변환합니다. **virt-v2v** 패키지가 하나 이상의 호스트에 설치되어 있어야 합니다. **virt-v2v** 패키지는 기본적으로 **RHVH(Red Hat Virtualization Host)**에서 사용할 수 있으며 **Red Hat Virtualization** 환경에 추가할 때 **VDSM**의 종속성으로 **Red Hat Enterprise Linux** 호스트에 설치됩니다. **Red Hat Enterprise Linux** 호스트는 **Red Hat Enterprise Linux 7.2** 이상이어야 합니다.



참고

**virt-v2v** 패키지는 **ppc64le** 아키텍처에서 사용할 수 없습니다. 이러한 호스트는 프록시 호스트로 사용할 수 없습니다.

#### RHEL 5ECDHE 인스턴스를 가상 머신 공급자로 추가

1.

프록시 호스트와 **RHEL 5ECDHE** 호스트 간의 공개 키 인증을 활성화합니다.

a.

프록시 호스트에 로그인하고 **vds**m 사용자에게 대한 **SSH** 키를 생성합니다.

```
# sudo -u vds m ssh-keygen
```

b.

**vds**m 사용자의 공개 키를 **RHEL 5ECDHE** 호스트에 복사합니다. 프록시 호스트의 **known\_hosts** 파일도 **RHEL 5ECDHE** 호스트의 호스트 키를 포함하도록 업데이트됩니다.

```
# sudo -u vds m ssh-copy-id root@xenhost.example.com
```

c.

**RHEL 5ECDHE** 호스트에 로그인하여 로그인이 올바르게 작동하는지 확인합니다.

```
# sudo -u vdsm ssh root@xenhost.example.com
```

2.

**Administration** → **Providers** (관리 공급자)를 클릭합니다.

3.

추가를 클릭합니다.

4.

이름 및 설명을 입력합니다.

5.

유형 드롭다운 목록에서 **XEN** 을 선택합니다.

6.

**DestinationRule** 가상 머신을 가져올 데이터 센터를 선택하거나 **Any Data Center** 를 선택하여 개별 가져오기 작업 중에 대상 데이터 센터를 지정합니다.

7.

**RHEL 5ECDHE** 호스트의 **URI**를 **URI** 필드에 입력합니다.

8.

가상 머신 가져오기 작업 중에 **virt-v2v** 가 설치된 선택한 데이터 센터에서 호스트를 선택하여 프록시 호스트 역할을 합니다. 이 호스트는 **RHEL 5ECDHE** 외부 공급자의 네트워크에도 연결할 수 있어야 합니다. 위의 모든 데이터 센터를 선택한 경우 여기에서 호스트를 선택할 수 없지만 개별 가져오기 작업 중에 호스트를 지정할 수 있습니다.

9.

테스트를 클릭하여 **RHEL 5ECDHE** 호스트로 성공적으로 인증할 수 있는지 테스트합니다.

10.

**OK**를 클릭합니다.

**RHEL 5ECDHE** 외부 공급자에서 가상 머신을 가져오려면 [가상 머신 관리 가이드의 RHEL 5ECDHE 호스트에서 가상 머신 가져오기](#) 를 참조하십시오.

#### 14.2.7. KVM 호스트를 가상 머신 공급자로 추가

**KVM**에서 **Red Hat Virtualization Manager**로 가상 머신을 가져올 **KVM** 호스트를 추가합니다.

**KVM** 호스트를 가상 머신 공급자로 추가

1.

프록시 호스트와 **KVM** 호스트 간에 공개 키 인증을 활성화합니다.

a.

프록시 호스트에 로그인하고 **vds**m 사용자에게 대한 **SSH** 키를 생성합니다.

```
# sudo -u vds m ssh-keygen
```

b.

**vds**m 사용자의 공개 키를 **KVM** 호스트에 복사합니다. 프록시 호스트의 **known\_hosts** 파일도 **KVM** 호스트의 호스트 키를 포함하도록 업데이트됩니다.

```
# sudo -u vds m ssh-copy-id root@kvmhost.example.com
```

c.

**KVM** 호스트에 로그인하여 로그인이 올바르게 작동하는지 확인합니다.

```
# sudo -u vds m ssh root@kvmhost.example.com
```

2.

**Administration** → **Providers** (관리 공급자)를 클릭합니다.

3.

추가를 클릭합니다.

4.

이름 및 설명을 입력합니다.

5.

유형 드롭다운 목록에서 **KVM** 을 선택합니다.

6.

**KVM** 가상 머신을 가져올 데이터 센터를 선택하거나 **Any Data Center** 를 선택하여 개별 가져오기 작업 중에 대상 데이터 센터를 지정합니다.

7.

**URI** 필드에 **KVM** 호스트의 **URI** 를 입력합니다.

```
qemu+ssh://root@host.example.com/system
```

8.

가상 시스템 가져오기 작업 중에 선택한 데이터 센터에서 호스트를 선택하여 프록시 호스트 역할을 합니다. 이 호스트는 **KVM** 외부 공급자의 네트워크에도 연결할 수 있어야 합니다. 위의 데이터 센터 필드에서 모든 데이터 센터를 선택한 경우 여기에서 호스트를 선택할 수 없습니다. 이 필드는 회색으로 표시되고 데이터 센터에 임의 호스트 나 표시됩니다. 대신 개별 가져오기 작업 중에 호스트를 지정할 수 있습니다.

9.

필요한 경우 인증 필요 확인란을 선택하고 **KVM** 호스트에 **Username and Password** 를 입력합니다. 사용자는 가상 머신이 상주하는 **KVM** 호스트에 액세스할 수 있어야 합니다.

10.

테스트를 클릭하여 제공된 자격 증명을 사용하여 **KVM** 호스트로 성공적으로 인증할 수 있는지 테스트합니다.

11.

**OK**를 클릭합니다.

**KVM** 외부 공급자에서 가상 머신을 가져오려면 가상 머신 관리 가이드의 **KVM 호스트에서 가상 머신 가져오기** 를 참조하십시오.

#### 14.2.8. OVN(Open Virtual Network)을 외부 네트워크 공급자로 추가

**OVN(Open Virtual Network)**을 사용하면 **VLAN**을 추가하거나 인프라를 변경하지 않고도 네트워크를 만들 수 있습니다. **OVN**은 가상 **L2** 및 **L3** 오버레이에 대한 기본 **OVS** 지원을 추가하여 가상 네트워크를 지원하는 **OVS(Open vSwitch)** 확장입니다.

새 **OVN** 네트워크 공급자를 설치하거나 기존 **OVN** 네트워크 공급자를 추가할 수 있습니다.

**OVN** 네트워크를 기본 **Red Hat Virtualization** 네트워크에 연결할 수도 있습니다. 자세한 내용은 **14.2.8.5절. "OVN 네트워크를 물리적 네트워크에 연결"**를 참조하십시오. 이 기능은 기술 프리뷰로만 사용할 수 있습니다.

**Neutron**과 유사한 **REST API**는 **ovirt-provider-ovn** 에 의해 노출되므로 네트워크, 서버넷, 포트 및 라우터를 생성할 수 있습니다(자세한 내용은 **OpenStack Networking API v2.0** 참조). 이러한 오버레이 네트워크를 사용하면 가상 머신 간 통신을 사용할 수 있습니다.



## 참고

**OVN**은 **OpenStack(Neutron) API**를 사용하여 **CloudForms**에서 외부 공급자로 지원됩니다. 자세한 내용은 **Red Hat CloudForms의 네트워크 관리자: 공급업체 관리**를 참조하십시오.

**OVS** 및 **OVN**에 대한 자세한 내용은 **OVS** 설명서 <http://docs.openvswitch.org/en/latest/> 및 <http://openvswitch.org/support/dist-docs/> 를 참조하십시오.

## 14.2.8.1. 새 OVN 네트워크 공급자 설치



## 주의

**openvswitch** 패키지가 이미 설치되어 있고 버전이 **1:2.6.1(버전 2.6.1, epoch 1)**인 경우 최신 **openvswitch** 패키지를 설치하려고 하면 **OVN** 설치가 실패합니다. 자세한 내용과 해결 방법은 **BZ#1505398** 의 **Doc** 텍스트를 참조하십시오.

**engine-setup** 을 사용하여 **OVN**을 설치하면 다음 단계가 자동화됩니다.

- **Manager** 시스템에서 **OVN** 중앙 서버를 설정합니다.
- **Red Hat Virtualization**에 외부 네트워크 공급자로 **OVN** 추가
- 기본 클러스터의 기본 네트워크 공급자를 **ovirt-provider-ovn** 로 설정합니다.
- 클러스터에 추가할 때 **OVN**과 통신하도록 호스트를 구성합니다.

**engine-setup** 과 함께 사전 구성된 응답 파일을 사용하는 경우 다음 항목을 추가하여 **OVN**을 설치할 수 있습니다.

```
OVESETUP_OVN/ovirtProviderOvn=bool:True
```

## 새 OVN 네트워크 공급자 설치

1.

**engine-setup**을 사용하여 **Manager**에 **OVN**을 설치합니다. 설치 중에 **engine-setup**은 다음과 같은 질문을 합니다.

```
# Install ovirt-provider-ovn(Yes, No) [Yes]?:
```

•

예 인 경우 **engine-setup**은 **ovirt-provider-ovn**을 설치합니다. **engine-setup**이 시스템을 업데이트하는 경우 **ovirt-provider-ovn**이 이전에 설치되지 않은 경우에만 이 프롬프트가 표시됩니다.

•

**No.**가 없으면 다음 **engine-setup** 실행 시 다시 묻지 않습니다. 이 옵션을 보려면 **engine-setup --reconfigure-optional-components**를 실행합니다.

```
# Use default credentials (admin@internal) for ovirt-provider-ovn(Yes, No) [Yes]?:
```

예 인 경우 **engine-setup**은 설정 프로세스의 이전에 지정된 기본 엔진 사용자와 암호를 사용합니다. 이 옵션은 새 설치 중에만 사용할 수 있습니다.

```
# oVirt OVN provider user[admin]:
```

```
# oVirt OVN provider password[empty]:
```

기본값을 사용하거나 **oVirt OVN** 공급자 사용자 및 암호를 지정할 수 있습니다.



## 참고

나중에 인증 방법을 변경하려면 **/etc/ovirt-provider-ovn/conf.d/10\_engine\_setup.conf** 파일을 편집하거나 새 **/etc/ovirt-provider-ovn/conf.d/20\_engine\_setup.conf** 파일을 생성할 수 있습니다. 변경 사항을 적용하려면 **ovirt-provider-ovn** 서비스를 다시 시작합니다. **OVN** 인증에 대한 자세한 내용은 <https://github.com/oVirt/ovirt-provider-ovn/blob/master/README.adoc>를 참조하십시오.

2.

기본 클러스터에 호스트를 추가합니다. 이 클러스터에 추가된 호스트는 **OVN**과 통신하도록 자동으로 구성됩니다. 새 호스트를 추가하려면 **10.5.1절. “Red Hat Virtualization Manager에 표준 호스트 추가”**을 참조하십시오.

기본이 아닌 기존 네트워크를 사용하도록 호스트를 구성하려면 **14.2.8.4절. “OVNECDHE**

네트워크의 호스트 구성”을 참조하십시오.

3.

**Default** 클러스터에 네트워크를 추가합니다. **9.1.2절. “데이터 센터 또는 클러스터에서 새 논리 네트워크 생성”**을 보고 **Create on external provider** (외부 공급자에서 생성) 확인란을 선택합니다. **ovirt-provider-ovn** 은 기본적으로 선택됩니다.

4.

**OVN** 네트워크를 기본 **Red Hat Virtualization** 네트워크에 연결 하려면 물리적 네트워크에 연결 확인란을 선택하고 사용할 **Red Hat Virtualization** 네트워크를 지정합니다. 자세한 내용 및 사전 요구 사항은 **14.2.8.5절. “OVN 네트워크를 물리적 네트워크에 연결”**을 참조하십시오.

5.

네트워크가 보안 그룹 드롭다운의 보안 그룹을 사용해야 하는지 여부를 정의합니다. 사용 가능한 옵션에 대한 자세한 내용은 **9.1.7절. “설명되는 논리적 네트워크 일반 설정”**에서 참조하십시오. 이제 **OVN** 네트워크를 사용하는 가상 머신을 생성할 수 있습니다.

#### 14.2.8.2. 기존 OVN 네트워크 공급자 추가

**Red Hat Virtualization**에서 외부 네트워크 공급자로 기존 **OVN** 중앙 서버를 추가하려면 다음 주요 단계를 수행해야 합니다.

- 

**Manager**에서 **OVN**과 상호 작용하기 위해 사용하는 프록시인 **OVN** 공급자를 설치합니다. **OVN** 공급자는 모든 시스템에 설치할 수 있지만 **OVN** 중앙 서버 및 관리자와 통신할 수 있어야 합니다.

- 

**OVN** 공급자를 **Red Hat Virtualization**에 외부 네트워크 공급자로 추가합니다.

- 

**OVN**을 기본 네트워크 공급자로 사용하는 새 클러스터를 만듭니다. 이 클러스터에 추가된 호스트는 **OVN**과 통신하도록 자동으로 구성됩니다.

사전 요구 사항

다음 패키지는 **OVN** 공급자에 필요하며 공급자 시스템에서 사용할 수 있어야 합니다.

- 

**openvswitch-ovn-central**



- **openvswitch**
- **openvswitch-ovn-common**
- **python-openvswitch**

공급자 시스템에서 이미 활성화된 리포지토리에서 이러한 패키지를 사용할 수 없는 경우 **OVS** 웹 사이트 <http://openvswitch.org/download/> 에서 다운로드할 수 있습니다.

#### 기존 **OVN** 네트워크 공급자 추가

1. **OVN** 공급자를 설치하고 구성합니다.

- a. 공급자 시스템에 공급자를 설치합니다.

```
# yum install ovirt-provider-ovn
```

- b. **Manager**와 동일한 머신에 공급자를 설치하지 않는 경우 `/etc/ovirt-provider-ovn/conf.d/10_engine_setup.conf` 파일에 다음 항목을 추가합니다(아직 존재하지 않는 경우 이 파일 생성).

```
[OVIRT]
ovirt-host=https://Manager_host_name
```

인증이 활성화된 경우 인증에 사용됩니다.

- c. **OVN** 중앙 서버와 동일한 머신에 공급자를 설치하지 않는 경우 `/etc/ovirt-provider-ovn/conf.d/10_engine_setup.conf` 파일에 다음 항목을 추가합니다(아직 존재하지 않는 경우 이 파일 생성).

```
[OVN REMOTE]
ovn-remote=tcp:OVN_central_server_IP:6641
```

- d. **OVN** 공급자, **OVN** 중앙 서버 및 **Manager** 간 통신을 허용하도록 방화벽에서 포트 **9696**, **6641** 및 **6642**를 엽니다. 이 작업은 수동으로 수행하거나 `ovirt-provider-ovn` 및 `ovirt-`

**provider-ovn-central** 서비스를 적절한 영역에 추가하여 수행할 수 있습니다.

```
# firewall-cmd --zone=ZoneName --add-service=ovirt-provider-ovn --permanent
# firewall-cmd --zone=ZoneName --add-service=ovirt-provider-ovn-central --permanent
# firewall-cmd --reload
```

e.

서비스를 시작하고 활성화합니다.

```
# systemctl start ovirt-provider-ovn
# systemctl enable ovirt-provider-ovn
```

f.

포트 **6642** 및 **6641**의 요청을 수신 대기하도록 **OVN** 중앙 서버를 구성합니다.

```
# ovn-sbctl set-connection tcp:6642
# ovn-nbctl set-connection tcp:6641
```

2.

관리 포털에서 관리 공급자를 클릭합니다.

3.

추가를 클릭하고 일반 설정 탭에 세부 정보를 입력합니다. 이러한 필드에 대한 자세한 내용은 [14.2.10절. “설명되는 공급자 일반 설정 추가”](#) 을 참조하십시오.

4.

이름 및 설명을 입력합니다.

5.

유형 목록에서 외부 네트워크 공급자를 선택합니다.

6.

네트워킹 플러그인 텍스트 상자를 클릭하고 드롭다운 메뉴에서 **oVirt Network Provider for OVN** 을 선택합니다.

7.

필요한 경우 자동 동기화 확인란을 선택합니다. 이를 통해 외부 네트워크 공급자를 기존 네트워크와 자동 동기화할 수 있습니다.



참고

자동 동기화는 **engine-setup** 툴로 생성된 **ovirt-provider-ovn** 네트워크 공급자에서 기본적으로 활성화됩니다.

8. **OVN** 공급자의 **URL** 또는 정규화된 도메인 이름을 공급자 **URL** 텍스트 필드에 입력하고 포트 번호를 입력합니다. **OVN** 공급자 및 **OVN** 중앙 서버가 별도의 시스템에 있는 경우 중앙 서버가 아닌 공급자 시스템의 **URL**입니다. **OVN** 공급자가 **Manager**와 동일한 시스템에 있는 경우 **URL**이 기본 <http://localhost:9696> 으로 남아 있을 수 있습니다.
9. **Red Hat Virtualization Manager**에서 새 **OVN** 네트워크를 생성할 수 있도록 읽기 전용 확인란의 선택을 해제합니다.
10. 선택적으로 **Requires Authentication** (인증 필요) 확인란을 선택하고 **Keystone**에 등록된 외부 네트워크 공급자 사용자에게 대해 **Username** (사용자 이름) 및 **Password** 를 입력합니다. 또한 프로토콜, 호스트 이름 및 **API** 포트를 정의하여 **Keystone** 서버의 인증 **URL**을 정의해야 합니다.  
  
필요한 경우 외부 네트워크 공급자에 대한 테넌트를 입력합니다.  
  
인증 방법은 `/etc/ovirt-provider-ovn/conf.d/10_engine_setup.conf` 파일에 구성해야 합니다(아직 존재하지 않는 경우 이 파일 생성). 변경 사항을 적용하려면 **ovirt-provider-ovn** 서비스를 다시 시작합니다. **OVN** 인증에 대한 자세한 내용은 <https://github.com/oVirt/ovirt-provider-ovn/blob/master/README.adoc> 를 참조하십시오.
11. 자격 증명을 테스트합니다.
  - a. 테스트를 클릭하여 제공된 자격 증명을 사용하여 **OVN**으로 성공적으로 인증할 수 있는지 테스트합니다.
  - b. **OVN** 인스턴스에서 **SSL**을 사용하는 경우 공급자 인증서 가져오기 창이 열립니다. **OK**를 클릭하여 **Manager**가 인스턴스와 통신할 수 있도록 **OVN** 인스턴스에서 제공하는 인증서를 가져옵니다.
12. **OK**를 클릭합니다.
13. **OVN**을 기본 네트워크 공급자로 사용하는 새 클러스터를 만듭니다. **8.2.1**절. “새 클러스터 생성”을 확인하고 기본 네트워크 공급자 드롭다운 목록에서 **OVN** 네트워크 공급자를 선택합니다.
14. 클러스터에 호스트를 추가합니다. 이 클러스터에 추가된 호스트는 **OVN**과 통신하도록 자동으로 구성됩니다. 새 호스트를 추가하려면 **10.5.1**절. “Red Hat Virtualization Manager에 표준

호스트 추가”을 참조하십시오.

15.

**OVN** 네트워크를 새 클러스터에 가져오거나 추가합니다. 네트워크를 가져오려면 **네트워크 가져오기를 참조하십시오**. **OVN**을 사용하여 새 네트워크를 생성하려면 **데이터 센터 또는 클러스터에서 새 논리적 네트워크 생성**을 확인하고 외부 공급자 생성 확인란을 선택합니다. **ovirt-provider-ovn**은 기본적으로 선택됩니다.

기본이 아닌 기존 네트워크를 사용하도록 호스트를 구성하려면 **14.2.8.4절. “OVNCDHE 네트워크의 호스트 구성”**을 참조하십시오.

**OVN** 네트워크를 기본 **Red Hat Virtualization** 네트워크에 연결하려면 물리적 네트워크에 연결 확인란을 선택하고 사용할 **Red Hat Virtualization** 네트워크를 지정합니다. 자세한 내용 및 사전 요구 사항은 **14.2.8.5절. “OVN 네트워크를 물리적 네트워크에 연결”**을 참조하십시오.

이제 **OVN** 네트워크를 사용하는 가상 머신을 생성할 수 있습니다.

### 14.2.8.3. Ansible 플레이북을 사용하여 OVN 터널 네트워크 수정

**ovirt-provider-ovn-driver Ansible** 플레이북을 사용하여 **OVN** 컨트롤러의 터널 네트워크를 수정하려면 긴 이름을 사용할 수 있습니다.

#### OVN 터널 네트워크를 수정하는 Ansible Playbook

```
# ansible-playbook --key-file <path_to_key_file> -i <path_to_inventory> --extra-vars " cluster_name=
<cluster_name> ovn_central=<ovn_central_ip_address> ovirt_network=<ovirt network name>
ovn_tunneling_interface=<vdsm_network_name>" ovirt-provider-ovn-driver.yml
```

매개 변수

#### key-file

호스트에 로그인할 키 파일입니다. 기본 키 파일은 일반적으로 **/etc/pki/ovirt-engine/keys** 디렉터리에 있습니다.

인벤토리

**oVirt VM** 인벤토리입니다. 인벤토리 값을 찾으려면 이 스크립트를 사용합니다.  
**/usr/share/ovirt-engine-metrics/bin/ovirt-engine-hosts-ansible-inventory.**

**cluster\_name**

이름을 업데이트할 클러스터의 이름입니다.

### **ovn\_central**

**OVN** 중앙 서버의 **IP** 주소입니다. 이 **IP** 주소는 모든 호스트에서 액세스할 수 있어야 합니다.

### **ovirt\_network**

**oVirt** 네트워크 이름입니다.

### **ovn\_tunneling\_interface**

**VDSM** 네트워크 이름입니다.



참고

**ovirt-provider-ovn-driver Ansible Playbook**은 **ovirt\_network** 매개변수 또는 **ovn\_tunneling\_interface** 매개변수 사용을 지원합니다. 두 매개변수가 동일한 플레이북에 있는 경우 이 플레이북이 실패합니다.

**ovirt\_network** 매개변수가 있는 플레이북

```
# ansible-playbook --key-file /etc/pki/ovirt-engine/keys/engine_id_rsa -i /usr/share/ovirt-engine-
metrics/bin/ovirt-engine-hosts-ansible-inventory --extra-vars " cluster_name=test-cluster
ovn_central=192.168.200.2 ovirt_network=\"Long Network Name with \ character | |\" ovirt-
provider-ovn-driver.yml
```

**ovn\_tunneling\_interface** 매개변수가 있는 플레이북

```
# ansible-playbook --key-file /etc/pki/ovirt-engine/keys/engine_id_rsa -i /usr/share/ovirt-engine-
metrics/bin/ovirt-engine-hosts-ansible-inventory --extra-vars " cluster_name=test-cluster
ovn_central=192.168.200.2 ovn_tunneling_interface=on703ea21ddbc34" ovirt-provider-ovn-driver.yml
```

**Manager** 시스템에서 **/usr/share/ovirt-engine/playbooks** 디렉터리로 이동하여 **Ansible** 플레이북을 실행합니다.

#### 14.2.8.4. OVNECDHE 네트워크의 호스트 구성

**ovirt-provider-ovn-driver Ansible** 플레이북과 함께 기본 **ovirtmgmt** 네트워크 이외의 기존 네트워크를 사용하도록 호스트를 구성할 수 있습니다. 클러스터의 모든 호스트에서 네트워크에 액세스할 수 있

어야 합니다.



참고

**ovirt-provider-ovn-driver Ansible Playbook**은 기존 호스트를 업데이트합니다. 클러스터에 새 호스트를 추가하는 경우 플레이북을 다시 실행해야 합니다.

## OVNECDHE 네트워크의 호스트 구성

1. **Manager** 시스템에서 **Playbook** 디렉터리로 이동합니다.

```
# cd /usr/share/ovirt-engine/playbooks
```

2. 다음 매개변수를 사용하여 **ansible-playbook** 명령을 실행합니다.

```
# ansible-playbook --private-key=/etc/pki/ovirt-engine/keys/engine_id_rsa -i /usr/share/ovirt-engine-metrics/bin/ovirt-engine-hosts-ansible-inventory --extra-vars "cluster_name=Cluster_Name ovn_central=OVN_Central_IP ovn_tunneling_interface=VDSM_Network_Name" ovirt-provider-ovn-driver.yml
```

예를 들면 다음과 같습니다.

```
# ansible-playbook --private-key=/etc/pki/ovirt-engine/keys/engine_id_rsa -i /usr/share/ovirt-engine-metrics/bin/ovirt-engine-hosts-ansible-inventory --extra-vars "cluster_name=MyCluster ovn_central=192.168.0.1 ovn_tunneling_interface=MyNetwork" ovirt-provider-ovn-driver.yml
```



참고

**OVN\_Central\_IP** 는 새 네트워크에 있을 수 있지만 필수 사항은 아닙니다. **OVN\_Central\_IP** 는 모든 호스트에서 액세스할 수 있어야 합니다.

**VDSM\_Network\_Name** 은 15자로 제한됩니다. 15자를 초과하거나 **ASCII**가 아닌 문자가 포함된 논리 네트워크 이름을 정의한 경우 15자 이름이 자동으로 생성됩니다. 이러한 이름의 매핑을 표시하는 방법은 **VDSM** 이름을 논리 네트워크 이름에 매핑 매핑을 참조하십시오.

단일 호스트에서 **OVN networks** 업데이트

**vdsms-tool** 을 사용하여 단일 호스트에서 **OVN** 터널 네트워크를 업데이트할 수 있습니다.

```
# vdsms-tool ovn-config OVN_Central_IP Tunneling_IP_or_Network_Name
```

예 **14.1. vdsms-tool**으로 호스트 업데이트

```
# vdsms-tool ovn-config 192.168.0.1 MyNetwork
```

#### 14.2.8.5. OVN 네트워크를 물리적 네트워크에 연결

중요

이 기능은 **Red Hat Virtualization**에서 기술 프리뷰로만 사용할 수 있는 **Open vSwitch** 지원을 사용합니다. 기술 프리뷰 기능은 **Red Hat** 프로덕션 서비스 수준 계약 (**SLA**)에서 지원되지 않으며 기능적으로 완전하지 않을 수 있으며 프로덕션에는 사용하지 않는 것이 좋습니다. 이러한 기능을 사용하면 향후 제품 기능을 조기에 이용할 수 있어 개발 과정에서 고객이 기능을 테스트하고 피드백을 제공할 수 있습니다.

**Red Hat** 기술 프리뷰 기능 지원 범위에 대한 자세한 내용은 <https://access.redhat.com/support/offerings/techpreview/>를 참조하십시오.

기본 **Red Hat Virtualization** 네트워크를 오버레이하는 외부 공급자 네트워크를 생성하여 각 가상 머신이 동일한 서브넷을 공유할 수 있도록 할 수 있습니다.

중요

**OVN** 네트워크의 서브넷을 생성한 경우 해당 네트워크를 사용하는 가상 머신은 해당 **IP** 주소를 수신합니다. 물리적 네트워크에서 **IP** 주소를 할당하려면 **OVN** 네트워크의 서브넷을 생성하지 마십시오.

사전 요구 사항

- 

클러스터에 스위치 유형으로 **OVS**가 선택되어 있어야 합니다. 이 클러스터에 추가된 호스트에는 **ovirtmgmt** 브리지와 같이 기존 **Red Hat Virtualization** 네트워크가 구성되어 있지 않아

야 합니다.

- 물리적 네트워크는 호스트에서 사용할 수 있어야 합니다. 물리적 네트워크를 클러스터에 필요한 대로 설정(네트워크 관리 창 또는 **New Logical Network** 창의 **Cluster** 탭에서) 설정하여 적용할 수 있습니다.

물리적 네트워크에 연결된 새 외부 네트워크 생성

1. **Compute** → **Clusters** 를 클릭합니다.
  2. 클러스터 이름을 클릭하여 세부 정보 보기를 엽니다.
  3. **Logical Networks** 탭을 클릭하고 네트워크 추가를 클릭합니다.
  4. 네트워크의 이름을 입력합니다.
  5. 외부 공급자 생성 확인란을 선택합니다. **ovirt-provider-ovn** 은 기본적으로 선택됩니다.
  6. 기본적으로 선택되어 있지 않은 경우 물리적 네트워크에 연결 확인란을 선택합니다.
  7. 새 네트워크를 연결할 물리적 네트워크를 선택합니다.
- **Data Center Network** (데이터 센터 네트워크) 라디오 버튼을 클릭하고 드롭다운 목록에서 물리적 네트워크를 선택합니다. 이는 권장되는 옵션입니다.
  - **Custom** (사용자 지정) 라디오 버튼을 클릭하고 실제 네트워크의 이름을 입력합니다. 물리적 네트워크에 **VLAN** 태그 지정이 활성화된 경우 **VLAN** 태그 활성화 확인란을 선택하고 물리적 네트워크의 **VLAN** 태그를 입력해야 합니다.





## 중요

물리적 네트워크 이름은 **15**자를 초과하거나 특수 문자를 포함해서는 안 됩니다.

8. **OK**를 클릭합니다.

### 14.2.9. 외부 네트워크 공급자 추가

**OpenStack Neutron REST API**를 구현하는 모든 네트워크 공급자를 **Red Hat Virtualization**에 추가할 수 있습니다. 가상 인터페이스 드라이버는 외부 네트워크 공급자의 구현자가 제공해야 합니다. 네트워크 공급자 및 가상 인터페이스 드라이버의 참조 구현은 <https://github.com/mmirecki/ovirt-provider-mock> 및 [https://github.com/mmirecki/ovirt-provider-mock/blob/master/docs/driver\\_instalation](https://github.com/mmirecki/ovirt-provider-mock/blob/master/docs/driver_instalation)에서 확인할 수 있습니다.

네트워크 프로비저닝을 위한 외부 네트워크 공급자 추가

1. **Administration** → **Providers** (관리 공급자)를 클릭합니다.
2. 추가를 클릭하고 일반 설정 탭에 세부 정보를 입력합니다. 이러한 필드에 대한 자세한 내용은 **14.2.10절. “설명되는 공급자 일반 설정 추가”**을 참조하십시오.
3. 이름 및 설명을 입력합니다.
4. 유형 드롭다운 목록에서 외부 네트워크 공급자를 선택합니다.
5. 선택적으로 **Networking Plugin** 텍스트 상자를 클릭하고 드롭다운 메뉴에서 적절한 드라이버를 선택합니다.
6. 필요한 경우 자동 동기화 확인란을 선택합니다. 이를 통해 외부 네트워크 공급자를 기존 네트워크와 자동 동기화할 수 있습니다. 이 기능은 외부 네트워크 공급자를 추가할 때 기본적으로 비활성화되어 있습니다.



참고

자동 동기화는 **engine-setup** 툴로 생성된 **ovirt-provider-ovn** 네트워크 공급자에서 기본적으로 활성화됩니다.

7.

외부 네트워크 공급자가 설치된 시스템의 **URL** 또는 정규화된 도메인 이름을 입력한 다음 포트 번호를 입력합니다. 읽기 전용 확인란이 기본적으로 선택됩니다. 이로 인해 사용자가 외부 네트워크 공급자를 수정하지 않습니다.



중요

**Red Hat**에서 지원하려면 설정을 위해 **Read-Only** 확인란을 선택해야 합니다.

8.

선택적으로 **Requires Authentication (인증 필요)** 확인란을 선택하고 **Keystone**에 등록된 외부 네트워크 공급자 사용자에게 대해 **Username (사용자 이름)** 및 **Password** 를 입력합니다. 또한 **프로토콜, 호스트 이름 및 API 포트** 를 정의하여 **Keystone** 서버의 인증 **URL** 을 정의해야 합니다.

필요한 경우 외부 네트워크 공급자에 대한 테넌트 를 입력합니다.

9.

자격 증명을 테스트합니다.

a.

테스트를 클릭하여 제공된 자격 증명을 사용하여 외부 네트워크 공급자로 성공적으로 인증할 수 있는지 테스트합니다.

b.

외부 네트워크 공급자가 **SSL** 을 사용하는 경우 공급자 인증서 가져오기 창이 열립니다. 확인을 클릭하여 **Manager** 가 인스턴스와 통신할 수 있도록 외부 네트워크 공급자가 제공하는 인증서를 가져옵니다.

10.

**OK** 를 클릭합니다.

이 공급자의 네트워크를 사용하려면 먼저 호스트에 가상 인터페이스 드라이버를 설치하고 네트워크를 가져와야 합니다. 네트워크를 가져오려면 **9.3.1 절. “외부 공급자에서 네트워크 가져오기”** 에서 참조하십시오.

## 14.2.10. 설명되는 공급자 일반 설정 추가

공급자 추가 창의 일반 탭에서 외부 공급자의 핵심 세부 정보를 등록할 수 있습니다.

표 14.1. 공급자 추가: 일반 설정

설정	설명
이름	Manager의 공급자를 나타내는 이름입니다.
설명	사용자가 읽을 수 있는 일반 텍스트로 공급자의 설명입니다.
유형	<p>외부 공급자의 유형입니다. 이 설정을 변경하면 공급자 구성에 사용 가능한 필드가 변경됩니다.</p> <p><b>FlexVolume/Satellite</b></p> <ul style="list-style-type: none"> <li>● <b>공급자 URL:</b> Satellite 인스턴스를 호스팅하는 시스템의 URL 또는 정규화된 도메인 이름입니다. 포트 번호를 URL 또는 정규화된 도메인 이름의 끝에 추가할 필요가 없습니다.</li> <li>● <b>Requires Authentication:</b> 공급자에 인증이 필요한지 여부를 지정할 수 있습니다. <b>ECDHE/Satellite</b> 가 선택된 경우 인증이 필요합니다.</li> <li>● <b>username:</b> Satellite 인스턴스에 연결하기 위한 사용자 이름입니다. 이 사용자 이름은 Satellite 인스턴스의 프로비저닝 포털에 로그인하는 데 사용되는 사용자 이름이어야 합니다.</li> <li>● <b>password:</b> 위 사용자 이름을 인증할 암호입니다. 이 암호는 Satellite 인스턴스의 프로비저닝 포털에 로그인하는 데 사용되는 암호여야 합니다.</li> </ul> <p><b>OpenStack Image</b></p> <ul style="list-style-type: none"> <li>● <b>공급자 URL:</b> OpenStack 이미지 서비스가 호스팅되는 시스템의 URL 또는 정규화된 도메인 이름입니다. OpenStack Image 서비스의 포트 번호를 URL 또는 정규화된 도메인 이름의 끝에 추가해야 합니다. 기본적으로 이 포트 번호는 9292입니다.</li> <li>● <b>Authentication 필요:</b> 인증이 OpenStack 이미지 서비스에 액세스하는 데 필요한지 여부를 지정할 수 있습니다.</li> <li>● <b>사용자 이름:</b> Keystone 서버에 연결하기 위한 사용자 이름입니다. 이 사용자 이름은 OpenStack Image 서비스가 멤버인 Keystone 인스턴스에 등록된 OpenStack 이미지 서비스의 사용자 이름이어야 합니다.</li> </ul>

설정	설명
	<ul style="list-style-type: none"> <li>● <b>password:</b> 위 사용자 이름을 인증할 암호입니다. 이 암호는 OpenStack Image 서비스가 멤버인 Keystone 인스턴스에 등록된 OpenStack 이미지 서비스의 암호여야 합니다.</li> <li>● <b>protocol:</b> Keystone 서버와 통신하는 데 사용되는 프로토콜입니다. <b>HTTP</b> 로 설정해야 합니다.</li> <li>● <b>hostname:</b> Keystone 서버의 IP 주소 또는 호스트 이름</li> <li>● <b>API 포트:</b> Keystone 서버의 API 포트 번호입니다.</li> <li>● <b>API 버전:</b> Keystone 서비스의 버전입니다. 값은 <b>v2.0</b> 이고 필드가 비활성화되어 있습니다.</li> <li>● <b>테넌트 이름:</b> OpenStack Image 서비스가 멤버인 OpenStack 테넌트의 이름입니다.</li> </ul> <p><b>OpenStack Networking</b></p> <ul style="list-style-type: none"> <li>● <b>네트워킹 플러그인:</b> OpenStack Networking 서버에 연결할 네트워킹 플러그인입니다. OpenStack Networking의 경우 <b>Open vSwitch</b> 는 유일한 옵션이며 기본적으로 선택됩니다.</li> <li>● <b>자동 동기화:</b> 공급자가 기존 네트워크와 자동으로 동기화되는지 여부를 지정할 수 있습니다.</li> <li>● <b>공급자 URL:</b> OpenStack Networking 인스턴스가 호스팅되는 시스템의 URL 또는 정규화된 도메인 이름입니다. OpenStack Networking 인스턴스의 포트 번호를 URL 또는 정규화된 도메인 이름 끝에 추가해야 합니다. 기본적으로 이 포트는 9696입니다.</li> <li>● <b>읽기 전용:</b> 관리 포털에서 OpenStack Networking 인스턴스를 수정할 수 있는지 여부를 지정할 수 있습니다.</li> <li>● <b>Authentication 필요:</b> 인증이 OpenStack Networking 서비스에 액세스하는 데 필요한지 여부를 지정할 수 있습니다.</li> <li>● <b>username:</b> OpenStack Networking 인스턴스에 연결하기 위한 사용자 이름입니다. 이 사용자 이름은 OpenStack Networking 인스턴스가 멤버인 Keystone 인스턴스에 등록된 OpenStack Networking의 사용자 이름이어야 합니다.</li> <li>● <b>password:</b> 위 사용자 이름을 인증할 암호입니다. 이 암호는 OpenStack Networking 인스턴스가 멤버인 Keystone 인스턴스에 등록된 OpenStack Networking의 암호여야 합니다.</li> <li>● <b>protocol:</b> Keystone 서버와 통신하는 데 사용되는 프로토콜입니다. 기본값은 <b>HTTPS</b> 입니다.</li> </ul>

설정	설명
	<ul style="list-style-type: none"> <li>● <b>hostname:</b> Keystone 서버의 IP 주소 또는 호스트 이름</li> <li>● <b>API 포트:</b> Keystone 서버의 API 포트 번호입니다.</li> <li>● <b>API 버전:</b> Keystone 서버의 버전입니다. 이는 URL에 나타납니다. v2.0이 표시되면 <b>v2.0</b>을 선택합니다. v3이 표시되면 <b>v3</b>를 선택합니다.</li> </ul> <p><b>API 버전</b> 필드에서 <b>v3</b>을 선택하면 다음 필드가 표시됩니다.</p> <ul style="list-style-type: none"> <li>● <b>User Domain Name:</b> 도메인에 정의된 사용자의 이름 Keystone API v3에서는 도메인이 OpenStack에서 서비스 엔티티의 관리 경계를 결정하는 데 사용됩니다. 도메인을 사용하면 도메인별 구성 또는 보안 옵션 설정과 같은 다양한 목적으로 사용자를 그룹화할 수 있습니다. 자세한 내용은 Red Hat OpenStack Platform <i>아키텍처 가이드</i>의 <a href="#">OpenStack Identity(keystone)</a>를 참조하십시오.</li> <li>● <b>프로젝트 이름:</b> OpenStack Identity API v3의 프로젝트 이름을 정의합니다.</li> <li>● <b>프로젝트 도메인 이름:</b> OpenStack Identity API v3에 대한 프로젝트의 도메인 이름을 정의합니다.</li> </ul> <p><b>API 버전</b> 필드에서 <b>v2.0</b>을 선택하면 다음 필드가 표시됩니다.</p> <ul style="list-style-type: none"> <li>● <b>테넌트 이름:</b> <b>API 버전</b> 필드에서 v2를 선택한 경우에만 나타납니다. OpenStack Networking 인스턴스가 멤버인 OpenStack 테넌트의 이름입니다.</li> </ul> <p><b>OpenStack Volume</b></p> <ul style="list-style-type: none"> <li>● <b>Data Center:</b> OpenStack 볼륨 스토리지 볼륨이 연결될 데이터 센터입니다.</li> <li>● <b>공급자 URL:</b> OpenStack 볼륨 인스턴스가 호스팅되는 시스템의 URL 또는 정규화된 도메인 이름입니다. OpenStack Volume 인스턴스의 포트 번호를 URL 또는 정규화된 도메인 이름의 끝에 추가해야 합니다. 기본적으로 이 포트 번호는 8776입니다.</li> <li>● <b>Authentication 필요:</b> 인증이 OpenStack 볼륨 서비스에 액세스하는 데 필요한지 여부를 지정할 수 있습니다.</li> <li>● <b>사용자 이름:</b> Keystone 서버에 연결하기 위한 사용자 이름입니다. 이 사용자 이름은 OpenStack Volume 인스턴스가 멤버인 Keystone 인스턴스에 등록된 OpenStack Volume의 사용자 이름이어야 합니다.</li> <li>● <b>password:</b> 위 사용자 이름을 인증할 암호입니다. 이 암호는 OpenStack Volume 인스턴스가 멤버인 Keystone 인스턴스에 등록된 OpenStack Volume의 암호여야 합니다.</li> </ul>

설정	설명
	<ul style="list-style-type: none"> <li>● <b>protocol:</b> Keystone 서버와 통신하는 데 사용되는 프로토콜입니다. <b>HTTP</b> 로 설정해야 합니다.</li> <li>● <b>hostname:</b> Keystone 서버의 IP 주소 또는 호스트 이름</li> <li>● <b>API 포트:</b> Keystone 서버의 API 포트 번호입니다.</li> <li>● <b>API 버전:</b> Keystone 서버의 버전입니다. 값은 <b>v2.0</b> 이고 필드가 비활성화되어 있습니다.</li> <li>● <b>테넌트 이름:</b> OpenStack Volume 인스턴스가 멤버인 OpenStack 테넌트의 이름입니다.</li> </ul> <p><b>VMware</b></p> <ul style="list-style-type: none"> <li>● <b>데이터 센터:</b> VMware 가상 머신을 가져올 데이터 센터를 지정하거나 <b>Any Data Center</b> 를 선택하여 개별 가져오기 작업 중에 대상 데이터 센터를 지정합니다(가상 머신 탭에서 가져오기 기능 사용).</li> <li>● <b>vCenter:</b> VMware vCenter 인스턴스의 IP 주소 또는 정규화된 도메인 이름입니다.</li> <li>● <b>ESXi:</b> 가상 머신을 가져올 호스트의 IP 주소 또는 정규화된 도메인 이름입니다.</li> <li>● <b>Data Center:</b> 지정된 ESXi 호스트가 상주하는 데이터 센터의 이름입니다.</li> <li>● <b>cluster:</b> 지정된 ESXi 호스트가 상주하는 클러스터의 이름입니다.</li> <li>● <b>서버의 SSL 인증서 확인:</b> 연결 시 ESXi 호스트의 인증서가 확인되는지 여부를 지정합니다.</li> <li>● <b>프록시 호스트:</b> 가상 머신 가져오기 작업 중에 호스트 역할을 하도록 <b>virt-v2v</b> 가 설치된 선택한 데이터 센터에서 호스트를 선택합니다. 이 호스트는 VMware vCenter 외부 공급자의 네트워크에 연결할 수도 있어야 합니다. <b>Any Data Center</b> 를 선택한 경우 여기에서 호스트를 선택할 수 없지만 개별 가져오기 작업 중에 호스트를 지정할 수 있습니다(가상 머신 탭에서 가져오기 기능 사용).</li> <li>● <b>사용자 이름:</b> VMware vCenter 인스턴스에 연결하기 위한 사용자 이름입니다. 사용자는 가상 시스템이 상주하는 VMware 데이터 센터 및 ESXi 호스트에 액세스할 수 있어야 합니다.</li> <li>● <b>password:</b> 위 사용자 이름을 인증할 암호입니다.</li> </ul> <p><b>RHEL 5ECDHE</b></p> <ul style="list-style-type: none"> <li>● <b>Data Center:</b> FlexVolume 가상 머신을 가져올 데이터 센터를 지정하거나, 가상 머신 탭에서 가져오기 기능 사용(가상 머신 탭에서 가져오기 기능 사용)에서 개별 가져오기 작업 중에 대상 데이터 센터를 대신 지정할 데이터 센터를 선택합니다.</li> </ul>

설정	설명
	<ul style="list-style-type: none"> <li>● URI: RHEL 5ECDHE 호스트의 URI입니다.</li> <li>● 프록시 호스트: 가상 머신 가져오기 작업 중에 호스트 역할을 하도록 <b>virt-v2v</b> 가 설치된 선택한 데이터 센터에서 호스트를 선택합니다. 이 호스트는 RHEL 5ECDHE 외부 공급자의 네트워크에도 연결할 수 있어야 합니다. Any Data Center 를 선택한 경우 여기에서 호스트를 선택할 수 없지만, 대신 가상 머신 탭에서 가져오기 함수를 사용하여 개별 가져오기 작업 중에 호스트를 지정할 수 있습니다.</li> </ul> <p><b>KVM</b></p> <ul style="list-style-type: none"> <li>● 데이터 센터: KVM 가상 머신을 가져올 데이터 센터를 지정하거나, 가상 머신 탭에서 가져오기 기능을 사용하여 개별 가져오기 작업 중에 대상 데이터 센터를 대신 지정할 데이터 센터를 선택합니다.</li> <li>● URI: KVM 호스트의 URI입니다.</li> <li>● 프록시 호스트: 가상 머신 가져오기 작업 중에 선택한 데이터 센터에서 호스트로 사용할 호스트를 선택합니다. 이 호스트는 KVM 외부 공급자의 네트워크에도 연결할 수 있어야 합니다. Any Data Center 를 선택한 경우 여기에서 호스트를 선택할 수 없지만, 대신 가상 머신 탭에서 가져오기 함수를 사용하여 개별 가져오기 작업 중에 호스트를 지정할 수 있습니다.</li> <li>● Authentication 필요: 인증이 KVM 호스트에 액세스하는 데 필요한지 여부를 지정할 수 있습니다.</li> <li>● 사용자 이름: KVM 호스트에 연결하기 위한 사용자 이름입니다.</li> <li>● password: 위 사용자 이름을 인증할 암호입니다.</li> </ul> <p><b>외부 네트워크 공급자</b></p> <ul style="list-style-type: none"> <li>● 네트워킹 플러그인: NIC 작업을 처리하기 위해 호스트에서 사용할 드라이버 구현을 결정합니다. <b>oVirt Network Provider for OVN</b> 플러그인이 있는 외부 네트워크 공급자가 클러스터의 기본 네트워크 공급자로 추가되는 경우 클러스터에 추가된 호스트에 설치할 드라이버도 결정합니다.</li> <li>● 자동 동기화: 공급자가 기존 네트워크와 자동으로 동기화되는지 여부를 지정할 수 있습니다.</li> <li>● 공급자 URL: 외부 네트워크 공급자가 호스팅되는 시스템의 URL 또는 정규화된 도메인 이름입니다. 외부 네트워크 공급자의 포트 번호를 URL 또는 정규화된 도메인 이름의 끝에 추가해야 합니다. 기본적으로 이 포트는 9696입니다.</li> <li>● 읽기 전용: 관리 포털에서 외부 네트워크 공급자를 수정할 수 있는지 여부를 지정할 수 있습니다.</li> </ul>

설정	설명
	<ul style="list-style-type: none"> <li>● <b>Authentication 필요:</b> 인증이 외부 네트워크 공급자에 액세스하는 데 필요한지 여부를 지정할 수 있습니다.</li> <li>● <b>사용자 이름:</b> 외부 네트워크 공급자에 연결하기 위한 사용자 이름입니다. Active Directory를 사용하여 인증하는 경우 사용자 이름은 기본 사용자 이름 @ 도메인 대신 사용자이름 @domain@_profile 형식이어야 합니다.</li> <li>● <b>password:</b> 위 사용자 이름을 인증할 암호입니다.</li> <li>● <b>protocol:</b> Keystone 서버와 통신하는 데 사용되는 프로토콜입니다. 기본값은 HTTPS 입니다.</li> <li>● <b>hostname:</b> Keystone 서버의 IP 주소 또는 호스트 이름</li> <li>● <b>API 포트:</b> Keystone 서버의 API 포트 번호입니다.</li> <li>● <b>API 버전:</b> Keystone 서버의 버전입니다. 값은 v2.0 이고 필드가 비활성화되어 있습니다.</li> <li>● <b>테넌트 이름:</b> 선택 사항. 외부 네트워크 공급자가 멤버인 테넌트의 이름입니다.</li> </ul>
테스트	사용자가 지정된 자격 증명을 테스트할 수 있습니다. 이 버튼은 모든 공급자 유형에서 사용할 수 있습니다.

#### 14.2.11. 설명 된 공급자 에이전트 구성 설정 추가

**Add Provider** (프로바이더 추가) 창의 에이전트 구성 탭에서는 사용자가 네트워킹 플러그인에 대한 세부 정보를 등록할 수 있습니다. 이 탭은 **OpenStack Networking** 공급자 유형에서만 사용할 수 있습니다.

표 14.2. 공급자 추가: 에이전트 설정

설정	설명
인터페이스 매핑	레이블: 인터페이스 형식으로 된 심볼로 구분된 매핑 목록입니다.
브로커 유형	OpenStack Networking 인스턴스에서 사용하는 메시지 브로커 유형입니다. RabbitMQ 또는 ECDHE를 선택합니다.
호스트	메시지 브로커가 설치된 시스템의 URL 또는 정규화된 도메인 이름입니다.



설정	설명
포트	위 호스트와의 연결을 수행할 원격 포트입니다. 기본적으로 이 포트는 호스트에서 SSL을 활성화하지 않고 SSL이 활성화된 경우 5761입니다.
사용자 이름	위의 메시지 브로커를 사용하여 OpenStack Networking 인스턴스를 인증하기 위한 사용자 이름입니다. 기본적으로 이 사용자 이름은 <b>neutron</b> 입니다.
암호	위의 사용자 이름을 인증할 암호입니다.

### 14.3. 외부 공급자 편집

#### 외부 공급자 편집

1. **Administration** → **Providers (프로바이더)**를 클릭하고 편집할 외부 공급자를 선택합니다.
2. 편집을 클릭합니다.
3. 공급자의 현재 값을 기본 값으로 변경합니다.
4. **OK**를 클릭합니다.

### 14.4. 외부 공급자 제거

#### 외부 공급자 제거

1. **Administration** → **Providers (프로바이더)**를 클릭하고 제거할 외부 공급자를 선택합니다.
2. 제거를 클릭합니다.

3.

**OK**를 클릭합니다.

III 부. 환경

## 15장. 자체 호스트 엔진

## 15.1. 셀프 호스트 엔진 유지 관리

## 셀프 호스트 엔진 유지 관리 모드

유지 관리 모드를 사용하면 고가용성 에이전트와 간섭 없이 **Manager** 가상 시스템을 시작, 중지 및 수정하고 **Manager**를 방해하지 않고 환경에서 자체 호스팅 엔진 노드를 재시작 및 수정할 수 있습니다.

적용할 수 있는 유지 관리 모드는 다음 세 가지가 있습니다.

- **global - Manager** 가상 시스템의 상태를 모니터링하지 못하도록 클러스터의 모든 고가용성 에이전트가 비활성화됩니다. 이후 **Red Hat Virtualization** 버전으로의 업그레이드와 같이 **ovirt-engine** 서비스를 중지해야 하는 설정 또는 업그레이드 작업에 글로벌 유지 관리 모드를 적용해야 합니다.
- **local** - 명령을 실행하는 노드의 고가용성 에이전트는 **Manager** 가상 머신의 상태를 모니터링하지 않습니다. 노드는 **Manager** 가상 머신을 호스팅하는 동안 로컬 유지 관리 모드에서 제외됩니다. 이 모드에 배치될 때 **Manager** 가상 머신을 호스팅하는 경우 **Manager**는 사용 가능한 노드가 있는 경우 다른 노드로 마이그레이션됩니다. 시스템 변경 또는 업데이트를 셀프 호스트 엔진 노드에 적용하는 경우 로컬 유지 관리 모드를 사용하는 것이 좋습니다.
- **none** - 유지 관리 모드를 비활성화하여 고가용성 에이전트가 작동하는지 확인합니다.

## 로컬 유지 관리 설정

단일 셀프 호스트 엔진 노드에서 고가용성 에이전트를 중지합니다.

## 관리 포털에서 로컬 유지 관리 모드 설정

1. 자체 호스팅 엔진 노드를 로컬 유지 관리 모드로 설정합니다.
  - a. 관리 포털에서 **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 자체 호스팅 엔진 노드를 선택합니다.
  - b. **Management** → **Maintenance** 를 클릭합니다. 해당 노드에 대해 로컬 유지보수 모드가 자동으로 트리거됩니다.

2. 유지 관리 작업을 완료한 후 유지 관리 모드를 비활성화합니다.
  - a. 관리 포털에서 **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 자체 호스팅 엔진 노드를 선택합니다.
  - b. **Management** → **Activate** (활성화)를 클릭합니다.

#### 명령줄에서 로컬 유지 관리 모드 설정

1. 셀프 호스팅 엔진 노드에 로그인하여 로컬 유지 관리 모드로 설정합니다.

```
# hosted-engine --set-maintenance --mode=local
```

2. 유지 관리 작업을 완료한 후 유지 관리 모드를 비활성화합니다.

```
# hosted-engine --set-maintenance --mode=none
```

#### 글로벌 유지 관리 설정

클러스터의 모든 자체 호스팅 엔진 노드에서 고가용성 에이전트를 중지합니다.

#### 관리 포털에서 글로벌 유지보수 모드 설정

1. 모든 셀프 호스팅 엔진 노드를 글로벌 유지 관리 모드로 설정합니다.
  - a. 관리 포털에서 **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 자체 호스팅 엔진 노드를 선택합니다.
  - b.
    - 더 많은 작업 (
      - ⋮
 )을 클릭한 다음 글로벌 **HA** 유지 관리 사용을 클릭합니다.
2. 유지 관리 작업을 완료한 후 유지 관리 모드를 비활성화합니다.
  - a. 관리 포털에서 **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 자체 호스팅 엔진 노드를 선택합니다.

b.

더 많은 작업 (

⋮

)을 클릭한 다음 글로벌 **HA** 유지 관리 비활성화 를 클릭합니다.

명령줄에서 글로벌 유지보수 모드 설정

1.

자체 호스팅 엔진 노드에 로그인하고 이를 글로벌 유지 관리 모드로 설정합니다.

```
# hosted-engine --set-maintenance --mode=global
```

2.

유지 관리 작업을 완료한 후 유지 관리 모드를 비활성화합니다.

```
# hosted-engine --set-maintenance --mode=none
```

## 15.2. MANAGER 가상 머신

**hosted-engine** 유틸리티는 **Manager** 가상 머신을 관리하는 데 도움이 되는 여러 명령을 제공합니다. 자체 호스팅 엔진 노드에서 **hosted-engine** 을 실행할 수 있습니다. 사용 가능한 모든 명령을 보려면 **hosted-engine --help** 를 실행합니다. 특정 명령에 대한 자세한 내용은 **hosted-engine --command --help** 를 실행하십시오.

### 15.2.1. 셀프 호스트 엔진 구성 업데이트

자체 호스팅 엔진 구성을 업데이트하려면 **hosted-engine --set-shared-config** 명령을 사용합니다. 이 명령은 초기 배포 후 공유 스토리지 도메인에서 셀프 호스트 엔진 구성을 업데이트합니다.

현재 구성 값을 보려면 **hosted-engine --get-shared-config** 명령을 사용합니다.

사용 가능한 모든 구성 키 및 해당 유형의 목록을 보려면 다음 명령을 입력합니다.

```
# hosted-engine --set-shared-config key --type=type --help
```

여기서 **type** 은 다음 중 하나입니다.

<b>he_local</b>	로컬 호스트에서 <b>etc/ovirt-hosted-engine/hosted-engine.conf</b> 의 로컬 인스턴스에 있는 값을 설정하므로 해당 호스트에서만 새 값을 사용합니다. 새 값을 활성화하려면 <b>ovirt-ha-agent</b> 및 <b>ovirt-ha-broker</b> 서비스를 다시 시작합니다.
<b>he_shared</b>	공유 스토리지에 <b>etc/ovirt-hosted-engine/hosted-engine.conf</b> 의 값을 설정하여 구성 변경 후 배포된 모든 호스트에서 이러한 값을 사용합니다. 호스트에서 새 값을 활성화하려면 해당 호스트를 재배포합니다.
<b>ha</b>	로컬 스토리지의 <b>/var/lib/ovirt-hosted-engine-ha/ha.conf</b> 에 값을 설정합니다. 새 설정은 즉시 적용됩니다.
<b>broker</b>	로컬 스토리지의 <b>/var/lib/ovirt-hosted-engine-ha/broker.conf</b> 에 값을 설정합니다. <b>ovirt-ha-broker</b> 서비스를 다시 시작하여 새 설정을 활성화합니다.

### 15.2.2. 이메일 알림 구성

자체 호스팅 엔진 노드에서 **HA** 상태 전환에 대해 **SMTP**를 사용하여 이메일 알림을 구성할 수 있습니다. 업데이트할 수 있는 키는 **smtp-server, smtp-port, source-email, destination-emails, state\_transition** 등입니다.

이메일 알림을 구성하려면 다음을 수행합니다.

1.

자체 호스팅 엔진 노드에서 **smtp-server** 키를 원하는 **SMTP** 서버 주소로 설정합니다.

```
# hosted-engine --set-shared-config smtp-server smtp.example.com --type=broker
```



참고

자체 호스팅 엔진 구성 파일이 업데이트되었는지 확인하려면 다음을 실행합니다.

```
# hosted-engine --get-shared-config smtp-server --type=broker
broker : smtp.example.com, type : broker
```

2.

기본 **SMTP** 포트 (포트 **25**)가 구성되어 있는지 확인합니다.

```
# hosted-engine --get-shared-config smtp-port --type=broker
broker : 25, type : broker
```

3.

**SMTP** 서버가 이메일 알림을 보내는 데 사용할 이메일 주소를 지정합니다. 하나의 주소만 지정할 수 있습니다.

```
# hosted-engine --set-shared-config source-email source@example.com --type=broker
```

4.

이메일 알림을 수신할 대상 이메일 주소를 지정합니다. 여러 이메일 주소를 지정하려면 각 주소를 쉼표로 구분합니다.

```
# hosted-engine --set-shared-config destination-emails
destination1@example.com,destination2@example.com --type=broker
```

자체 호스팅 엔진 환경에 대해 **SMTP**가 올바르게 구성되었는지 확인하려면 셀프 호스팅 엔진 노드에서 **HA** 상태를 변경하고 이메일 알림이 전송되었는지 확인합니다. 예를 들어 **HA** 에이전트를 유지 관리 모드로 배치하여 **HA** 상태를 변경할 수 있습니다. 자세한 내용은 [15.1절. “셀프 호스팅 엔진 유지 관리”](#)를 참조하십시오.

### 15.3. 추가 호스트의 셀프 호스팅 엔진에 예약된 메모리 슬롯 구성

**Manager** 가상 머신이 종료되거나 마이그레이션해야 하는 경우 **Manager** 가상 머신을 재시작하거나 마이그레이션할 수 있도록 셀프 호스팅 엔진 노드에 충분한 메모리가 있어야 합니다. 이 메모리는 스케줄링 정책을 사용하여 여러 자체 호스팅 엔진 노드에 예약할 수 있습니다. 스케줄링 정책은 가상 머신을 시작하거나 마이그레이션하기 전에 **Manager** 가상 머신을 시작하기에 충분한 메모리가 지정된 수의 추가 호스팅 엔진 노드에 남아 있는지 확인합니다. 스케줄링 정책에 대한 자세한 내용은 관리 가이드에서 스케줄링 정책 생성을 참조하십시오.

**Red Hat Virtualization Manager**에 셀프 호스팅 엔진 노드를 추가하려면 [15.4절. “Red Hat Virtualization Manager에 자체 호스팅 엔진 노드 추가”](#)을 참조하십시오.

#### 추가 호스트의 셀프 호스팅 엔진에 예약된 메모리 슬롯 구성

1. **Compute** → **Clusters (클러스터)**를 클릭하고 자체 호스팅 엔진 노드가 포함된 클러스터를 선택합니다.
2. 편집을 클릭합니다.
3. 스케줄링 정책 탭을 클릭합니다.



4. **+** 를 클릭하고 **HeSparesCount** 를 선택합니다.
5. **Manager** 가상 시스템을 시작하기에 충분한 여유 메모리를 예약할 추가 셀프 호스트 엔진 노드 수를 입력합니다.
6. **OK**를 클릭합니다.

#### 15.4. RED HAT VIRTUALIZATION MANAGER에 자체 호스팅 엔진 노드 추가

셀프 호스트 엔진 노드는 표준 호스트와 동일한 방식으로 추가되며 추가 단계로 호스트를 호스트 엔진 노드로 배포합니다. 공유 스토리지 도메인이 자동으로 감지되고 노드가 필요한 경우 **Manager** 가상 머신을 호스팅하는 장애 조치 호스트로 사용할 수 있습니다. 표준 호스트를 자체 호스트 엔진 환경에 연결할 수도 있지만 **Manager** 가상 시스템을 호스팅할 수 없습니다. **Red Hat**은 **Manager** 가상 시스템을 고가용성으로 유지하기 위해 최소 두 개의 자체 호스팅 엔진 노드를 사용하는 것이 좋습니다. **REST API**를 사용하여 추가 호스트를 추가할 수도 있습니다. **REST API** 가이드의 호스트를 참조하십시오.

[https://access.redhat.com/documentation/en-us/red\\_hat\\_virtualization/4.3/html/rest\\_api\\_guide/services#services-hosts](https://access.redhat.com/documentation/en-us/red_hat_virtualization/4.3/html/rest_api_guide/services#services-hosts)

#### 사전 요구 사항

- 셀프 호스트 엔진 노드를 재사용하는 경우 기존 셀프 호스트 엔진 구성을 삭제합니다. 셀프 호스트 엔진 환경에서 호스트 제거를 참조하십시오.



#### 중요

정적 IPv6 주소를 사용하는 관리 브릿지를 생성할 때 호스트를 추가하기 전에 인터페이스 구성(**ifcfg**) 파일에서 네트워크 관리자 제어를 비활성화합니다. 자세한 내용은 <https://access.redhat.com/solutions/3981311> 을 참조하십시오.

#### 절차

1. 관리 포털에서 **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭합니다.
2. 새로 생성을 클릭합니다.

추가 호스트 설정에 대한 자세한 내용은 관리 가이드의 새 호스트의 설정 및 제어 설명 및 호스트 편집 을 참조하십시오.

3. 드롭다운 목록을 사용하여 새 호스트의 **Data Center** (데이터 센터) 및 **Host Cluster** 를 선택합니다.
4. 새 호스트의 이름 및 주소를 입력합니다. 표준 **SSH** 포트인 포트 **22**는 **SSH** 포트 필드에 자동으로 채워집니다.
5. **Manager**에서 호스트에 액세스하는 데 사용할 인증 방법을 선택합니다.
  - **root** 사용자 암호를 입력하여 암호 인증을 사용합니다.
  - 또는 **SSH PublicKey** 필드에 표시되는 키를 호스트의 `/root/.ssh/authorized_keys` 에 복사하여 공개 키 인증을 사용합니다.
6. 선택적으로 호스트에 지원되는 전원 관리 카드가 있는 전원 관리를 구성합니다. 전원 관리 구성에 대한 자세한 내용은 관리 가이드에 설명된 [호스트 전원 관리 설정](#)을 참조하십시오.
7. 호스팅 엔진 탭을 클릭합니다.
8. 배포를 선택합니다.
9. **OK**를 클릭합니다.

### 15.5. 기존 호스트를 셀프 호스트 엔진 노드로 다시 설치

자체 호스팅 엔진 환경의 기존 표준 호스트를 **Manager** 가상 머신을 호스팅할 수 있는 자체 호스팅 엔진 노드로 변환할 수 있습니다.

#### 절차

1. **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.
2. **Management** → **Maintenance** 를 클릭하고 **OK** 를 클릭합니다.

3. **Installation** → **Reinstall** 을 클릭합니다.
4. 호스팅 엔진 탭을 클릭하고 드롭다운 목록에서 **DEPLOY** 를 선택합니다.
5. **OK**를 클릭합니다.

호스트는 자체 호스트 엔진 구성으로 다시 설치되며 관리 포털의 왕관 아이콘이 표시됩니다.

### 15.6. 복구 모드에서 관리자 가상 머신 부팅

이 주제에서는 **Manager** 가상 머신을 시작되지 않을 때 복구 모드로 부팅하는 방법을 설명합니다. 자세한 내용은 **Red Hat Enterprise Linux** 시스템 관리자 가이드의 **Rescue Mode**로 부팅 을 참조하십시오.

1. **hosted-engine** 노드 중 하나에 연결합니다.

```
$ ssh root@host_address
```

2. 자체 호스팅 엔진을 글로벌 유지 관리 모드로 설정합니다.

```
# hosted-engine --set-maintenance --mode=global
```

3. **Manager** 가상 머신의 실행 중인 인스턴스가 이미 있는지 확인합니다.

```
# hosted-engine --vm-status
```

**Manager** 가상 머신 인스턴스가 실행 중인 경우 호스트에 연결합니다.

```
# ssh root@host_address
```

4. 가상 머신을 종료합니다.

```
# hosted-engine --vm-shutdown
```



## 참고

가상 머신이 종료되지 않으면 다음 명령을 실행합니다.

```
# hosted-engine --vm-poweroff
```

5. **Manager** 가상 머신을 일시 정지 모드로 시작합니다.

```
hosted-engine --vm-start-paused
```

6. 임시 **VNC** 암호를 설정합니다.

```
hosted-engine --add-console-password
```

이 명령은 **VNC**를 사용하여 **Manager** 가상 머신에 로그인하는 데 필요한 정보를 출력합니다.

7. **VNC**를 사용하여 **Manager** 가상 머신에 로그인합니다. **Manager** 가상 머신이 여전히 일시 중지되어 있으므로 동결된 것처럼 보입니다.

8. 호스트에서 다음 명령을 사용하여 **Manager** 가상 머신을 재시작합니다.



## 주의

다음 명령을 실행하면 부트 로더 메뉴가 표시됩니다. 부트 로더가 일반 부팅 프로세스를 진행하기 전에 복구 모드로 전환해야 합니다. 이 명령을 진행하기 전에 복구 모드로 전환하는 방법에 대한 다음 단계를 읽으십시오.

```
# /usr/bin/virsh -c qemu:///system?authfile=/etc/ovirt-hosted-engine/virsh_auth.conf resume HostedEngine
```

9. **Manager** 가상 머신을 복구 모드로 부팅합니다.

10.

글로벌 유지보수 모드 비활성화

```
# hosted-engine --set-maintenance --mode=none
```

**Manager** 가상 머신에서 복구 작업을 실행할 수 있습니다.

### 15.7. 자체 호스트 엔진 환경에서 호스트 제거

환경에서 셀프 호스트 엔진 노드를 제거하려면 노드를 유지보수 모드에 배치하고 노드 배포를 취소한 다음 선택적으로 삭제합니다. **HA** 서비스가 중지된 후 노드는 일반 호스트로 관리할 수 있으며 자체 호스팅 엔진 구성 파일이 제거되었습니다.

자체 호스트 엔진 환경에서 호스트 제거

1.

관리 포털에서 **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 자체 호스팅 엔진 노드를 선택합니다.

2.

**Management** → **Maintenance** 를 클릭하고 **OK** 를 클릭합니다.

3.

**Installation** → **Reinstall** 을 클릭합니다.

4.

호스팅 엔진 탭을 클릭하고 드롭다운 목록에서 **UNDEPLOY** 를 선택합니다. 이 작업은 **ovirt-ha-agent** 및 **ovirt-ha-broker** 서비스를 중지하고 자체 호스팅 엔진 구성 파일을 제거합니다.

5.

**OK**를 클릭합니다.

6.

선택적으로 제거를 클릭하여 호스트 제거 확인 창을 열고 확인을 클릭합니다.

### 15.8. 자체 호스트 엔진 업데이트

현재 버전 **4.3**에서 최신 버전 **4.3**으로 셀프 호스트 엔진을 업데이트하려면 환경을 글로벌 유지 관리 모드로 배치한 다음 마이너 버전 간에 업데이트하는 표준 절차를 따라야 합니다.

## 글로벌 유지 관리 모드 활성화

**Manager** 가상 머신에서 설정 또는 업그레이드 작업을 수행하기 전에 자체 호스팅 엔진 환경을 전역 유지 관리 모드에 배치해야 합니다.

### 절차

1. 셀프 호스팅 엔진 노드 중 하나에 로그인하고 글로벌 유지보수 모드를 활성화합니다.

```
# hosted-engine --set-maintenance --mode=global
```

2. 계속하기 전에 환경이 유지보수 모드에 있는지 확인합니다.

```
# hosted-engine --vm-status
```

클러스터가 유지보수 모드에 있음을 나타내는 메시지가 표시되어야 합니다.

## Red Hat Virtualization Manager 업데이트

**Red Hat Virtualization Manager** 업데이트는 콘텐츠 전송 네트워크를 통해 릴리스됩니다.

### 절차

1. **Manager** 가상 머신에 로그인합니다.

2. 업데이트된 패키지를 사용할 수 있는지 확인합니다.

```
# engine-upgrade-check
```

3. 설정 패키지를 업데이트합니다.

```
# yum update ovirt\*setup\* rh\*vm-setup-plugins
```

4. **Red Hat Virtualization Manager**를 **engine-setup** 스크립트로 업데이트합니다. **engine-setup** 스크립트에서는 일부 구성 질문을 요청한 다음 **ovirt-engine** 서비스를 중지하고, 업데이트된 패키지를 다운로드하여 설치하고, 데이터베이스를 백업 및 업데이트하고, 설치 후 구성을 수행하며 **ovirt-engine** 서비스를 시작합니다.

## # engine-setup

스크립트가 성공적으로 완료되면 다음 메시지가 표시됩니다.

### Execution of setup completed successfully



#### 참고

**engine-setup** 스크립트는 **Red Hat Virtualization Manager** 설치 프로세스 중에도 사용되며 제공된 구성 값을 저장합니다. 업데이트 중에 구성을 미리 볼 때 저장된 값이 표시되고, 설치 후 **engine-config** 가 구성을 업데이트하는 데 사용된 경우 최신 버전이 아닐 수 있습니다. 예를 들어 설치 후 **engine-config** 를 **true** 로 업데이트하는 데 **engine-config** 를 사용하면 **engine-setup** 은 구성 프리뷰에서 "삭제: **False** 후 기본 **SAN delete: False**"가 출력됩니다. 그러나 업데이트된 값은 **engine-setup** 으로 덮어 쓰지 않습니다.



#### 중요

업데이트 프로세스에 시간이 다소 걸릴 수 있습니다. 완료되기 전에 프로세스를 중지하지 마십시오.

5.

**Manager**에 설치된 기본 운영 체제 및 선택적 패키지를 업데이트합니다.

## # yum update



#### 중요

커널 패키지가 업데이트된 경우:

1.           글로벌 유지보수 모드 비활성화
2.           시스템을 재부팅하여 업데이트를 완료합니다.

관련 정보

[글로벌 유지 관리 모드 비활성화](#)

### 글로벌 유지 관리 모드 비활성화

#### 절차

1. **Manager** 가상 머신에 로그인하여 종료합니다.
2. 셀프 호스트 엔진 노드 중 하나에 로그인하고 글로벌 유지 관리 모드를 비활성화합니다.

```
# hosted-engine --set-maintenance --mode=none
```

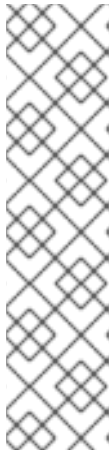
글로벌 유지 관리 모드를 종료하면 **ovirt-ha-agent**가 **Manager** 가상 시스템을 시작한 다음 **Manager**가 자동으로 시작됩니다. **Manager**를 시작하는 데 최대 10분이 걸릴 수 있습니다.

3. 환경이 실행 중인지 확인합니다.

```
# hosted-engine --vm-status
```

나열된 정보에는 엔진 상태가 포함됩니다. 엔진 상태 값은 다음과 같아야 합니다.

```
{"health": "good", "vm": "up", "detail": "Up"}
```



#### 참고

가상 머신이 계속 부팅되고 **Manager**가 아직 시작되지 않은 경우 **Engine** 상태는 다음과 같습니다.

```
{"reason": "bad vm status", "health": "bad", "vm": "up", "detail": "Powering up"}
```

이 경우 몇 분 기다렸다가 다시 시도합니다.

### 15.9. 셀프 호스트 엔진에서 관리자의 FQDN 변경

**ovirt-engine-rename** 명령을 사용하여 **Manager**의 정규화된 도메인 이름(FQDN) 레코드를 업데이트할 수 있습니다.



자세한 내용은 [22.1.3절. “oVirt Engine Rename Tool으로 관리자 이름 변경”](#)에서 참조하십시오.

## 16장. 백업 및 마이그레이션

### 16.1. RED HAT VIRTUALIZATION MANAGER 백업 및 복원

#### 16.1.1. Red Hat Virtualization Manager 백업 - 개요

**engine-backup** 도구를 사용하여 **Red Hat Virtualization Manager**의 정기적인 백업을 수행합니다. 틀은 엔진 데이터베이스 및 구성 파일을 단일 파일로 백업하고 **ovirt-engine** 서비스를 중단하지 않고 실행할 수 있습니다.

#### 16.1.2. engine-backup 명령의 구문

**engine-backup** 명령은 다음 두 가지 기본 모드 중 하나로 작동합니다.

```
# engine-backup --mode=backup
```

```
# engine-backup --mode=restore
```

이러한 두 가지 모드는 백업의 범위와 엔진 데이터베이스의 다른 자격 증명을 지정할 수 있는 매개변수 세트로 추가로 확장됩니다. 전체 매개변수 및 해당 함수에 대해 **engine-backup --help** 를 실행합니다.

#### 기본 옵션

##### --mode

명령이 백업 작업 또는 복원 작업을 수행할지 여부를 지정합니다. 백업 및 복원 의 두 가지 옵션을 사용할 수 있습니다. 필수 매개변수입니다.

##### --file

백업 모드에서 백업을 수행할 파일의 경로와 이름과 복원 모드에서 백업 데이터를 읽을 파일의 경로와 이름을 지정합니다. 이는 백업 모드 및 복원 모드에서 필수 매개 변수입니다.

##### --log

백업 또는 복원 작업의 로그를 작성할 파일의 경로와 이름을 지정합니다. 이 매개변수는 백업 모드와 복원 모드에서 모두 필요합니다.

##### --scope

백업 또는 복원 작업의 범위를 지정합니다. 모든 데이터베이스 및 구성 데이터를 백업하거나 복원하는 모든 옵션은 모두 있습니다. 즉, 시스템의 파일만 백업하거나 복원하는 파일, **Manager** 데이터

베이스만 백업하거나 복원하는 **db.** 및 **dwhdb.** 이 데이터베이스 백업 또는 복원 데이터베이스만 백업하거나 복원하는 **dwhdb**입니다. 기본 범위는 **all** 입니다.

**--scope** 매개변수는 동일한 **engine-backup** 명령에서 여러 번 지정할 수 있습니다.

### 관리자 데이터베이스 옵션

다음 옵션은 복원 모드에서 **engine-backup** 명령을 사용하는 경우에만 사용할 수 있습니다. 아래 옵션 구문은 **Manager** 데이터베이스 복원에 적용됩니다. 데이터 **ECDHE** 데이터베이스를 복원하는 데 동일한 옵션이 있습니다. **DataECDHE** 옵션 구문은 **engine-backup --help** 를 참조하십시오.

#### **--provision-db**

복원할 **Manager** 데이터베이스 백업에 사용할 **PostgreSQL** 데이터베이스를 만듭니다. 이 매개변수는 원격 호스트에서 백업을 복원하거나 **PostgreSQL** 데이터베이스가 아직 구성되지 않은 새로 설치할 때 필요합니다.

#### **--change-db-credentials**

백업 자체에 저장된 자격 증명 이외의 다른 자격 증명을 사용하여 **Manager** 데이터베이스를 복원하는 대체 자격 증명을 지정할 수 있습니다. 이 매개변수에 필요한 추가 매개변수는 **engine-backup --help** 를 참조하십시오.

#### **--restore-permissions** 또는 **--no-restore-permissions**

데이터베이스 사용자의 권한을 복원(또는 복원하지 않음)합니다. 백업을 복원할 때 이러한 매개변수 중 하나가 필요합니다.



#### 참고

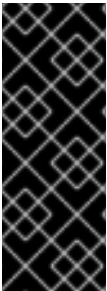
백업에 추가 데이터베이스 사용자에게 대한 부여가 포함된 경우 **--restore-permissions** 및 **--provision-db** (또는 **--provision-dwh-db**) 옵션으로 백업을 복원하면 임의의 암호가 있는 추가 사용자가 생성됩니다. 추가 사용자가 복원된 시스템에 액세스해야 하는 경우 이러한 암호를 수동으로 변경해야 합니다.

<https://access.redhat.com/articles/2686731> 을 참조하십시오.

### 16.1.3. **engine-backup** 명령을 사용하여 백업 생성

**Red Hat Virtualization Manager**는 **Manager**가 활성 상태인 동안 **engine-backup** 명령을 사용하여 백업할 수 있습니다. 수행할 백업을 지정하려면 **--scope** 에 다음 옵션 중 하나를 추가합니다.

- 모두: **Manager**에서 모든 데이터베이스 및 구성 파일의 전체 백업
- **files:** 시스템에 있는 파일만 백업
- **DB: Manager** 데이터베이스만 백업
- **dwhdb:** 데이터**ECDHE** 데이터베이스만 백업



중요

**Red Hat Virtualization Manager**의 새로 설치로 데이터베이스를 복원하려면 데이터베이스 백업만으로는 충분하지 않습니다. 관리자도 구성 파일에 액세스해야 합니다. 기본 값이 아닌 범위를 지정하는 모든 백업은 파일 범위 또는 파일 시스템 백업을 동반해야 합니다.

**engine-backup** 명령의 사용 예

1. **Red Hat Virtualization Manager**를 실행하는 머신에 로그인합니다.
2. 백업을 생성합니다.

예 **16.1.** 전체 백업 생성

```
# engine-backup --scope=all --mode=backup --file=file_name --log=log_file_name
```

예 **16.2.** 관리자 데이터베이스 백업 생성

```
# engine-backup --scope=files --scope=db --mode=backup --file=file_name --log=log_file_name
```

**db** 옵션을 **dwhdb** 로 교체하여 **DataECDHE** 데이터베이스를 백업합니다.

백업이 포함된 **tar** 파일은 지정된 경로와 파일 이름을 사용하여 생성됩니다.

백업이 포함된 **tar** 파일을 사용하여 환경을 복원할 수 있습니다.

#### 16.1.4. engine-backup 명령을 사용하여 백업 복원

**engine-backup** 명령을 사용하여 백업을 복원하려면 복원 대상에 따라 백업을 생성하는 것보다 더 많은 단계가 필요합니다. 예를 들어 **engine-backup** 명령을 사용하여 **Red Hat Virtualization**을 새로 설치한 후 로컬 또는 원격 데이터베이스를 사용하여 백업을 **Red Hat Virtualization**을 새로 설치할 수 있습니다.



#### 중요

백업은 백업과 동일한 주요 릴리스 환경으로만 복원할 수 있습니다. 예를 들어 **Red Hat Virtualization** 버전 4.2 환경 백업은 다른 **Red Hat Virtualization** 버전 4.2 환경으로만 복원할 수 있습니다. 백업 파일에 포함된 **Red Hat Virtualization** 버전을 보려면 백업 파일의 압축을 풀고 압축 해제된 파일의 루트 디렉터리에 있는 버전 파일의 값을 읽습니다.

#### 16.1.5. 새로 고침으로 백업 복원

**engine-backup** 명령을 사용하여 **Red Hat Virtualization Manager** 신규 설치에 백업을 복원할 수 있습니다. 기본 운영 체제가 설치되어 있고 **Red Hat Virtualization Manager**에 필요한 패키지가 설치되어 있지만 **engine-setup** 명령이 아직 실행되지 않은 시스템에서 다음 절차를 수행해야 합니다. 이 절차에서는 백업을 복원할 시스템에서 백업 파일 또는 파일에 액세스할 수 있다고 가정합니다.

##### 새로 고침으로 백업 복원

1.

**Manager** 시스템에 로그인합니다. 엔진 데이터베이스를 원격 호스트에 복원하는 경우 해당 호스트에 로그인하여 관련 작업을 수행해야 합니다. 마찬가지로 데이터를 원격 호스트에 복원할 수도 있는 경우 해당 호스트에 로그인하여 해당 호스트에서 관련 작업을 수행해야 합니다.

2.

전체 백업 또는 데이터베이스 전용 백업을 복원합니다.

•

전체 백업을 복원하십시오.

```
# engine-backup --mode=restore --file=file_name --log=log_file_name --provision-db --restore-permissions
```

전체 백업의 일부로 데이터도 복원되는 경우 추가 데이터베이스를 프로비저닝합니다.

```
engine-backup --mode=restore --file=file_name --log=log_file_name --provision-db --
provision-dwh-db --restore-permissions
```

구성 파일 및 데이터베이스 백업을 복원하여 데이터베이스 전용 백업을 복원합니다.

```
# engine-backup --mode=restore --scope=files --scope=db --file=file_name --
log=log_file_name --provision-db --restore-permissions
```

위의 예제에서는 **Manager** 데이터베이스의 백업을 복원합니다.

```
# engine-backup --mode=restore --scope=files --scope=dwhdb --file=file_name --
log=log_file_name --provision-dwh-db --restore-permissions
```

위의 예에서는 데이터 **ECDHE** 데이터베이스의 백업을 복원합니다.

성공하면 다음 출력이 표시됩니다.

```
You should now run engine-setup.
Done.
```

3.

다음 명령을 실행하고 프롬프트에 따라 복원된 관리자를 구성합니다.

```
# engine-setup
```

**Red Hat Virtualization Manager**가 백업에 보존된 버전으로 복원되었습니다. 새 **Red Hat Virtualization** 시스템의 정규화된 도메인 이름을 변경하려면 [22.1.1 절. “oVirt Engine Rename Tool”](#) 을 참조하십시오.

### 16.1.6. 기존 설치를 덮어쓰도록 백업 복원

**engine-backup** 명령은 **Red Hat Virtualization Manager**가 이미 설치 및 설정된 머신으로 백업을 복원할 수 있습니다. 이 기능은 환경을 백업하고, 해당 환경에서 변경한 작업을 수행한 다음 백업에서 환경을 복원하여 변경 사항을 실행 취소하려는 경우에 유용합니다.

복원된 환경에 호스트 추가 또는 제거와 같은 백업이 수행되었기 때문에 환경에 대한 변경 사항이 표시되지 않습니다. 이러한 변경 사항을 다시 실행해야 합니다.

### 절차

1. **Manager** 시스템에 로그인합니다.
2. 구성 파일을 제거하고 **Manager**와 관련된 데이터베이스를 정리합니다.

```
# engine-cleanup
```

**engine-cleanup** 명령은 **Manager** 데이터베이스만 정리합니다. 데이터베이스를 삭제하거나 해당 데이터베이스를 소유한 사용자를 삭제하지 않습니다.

3. 전체 백업 또는 데이터베이스 전용 백업을 복원합니다. 사용자와 데이터베이스가 이미 있으므로 새 데이터베이스를 만들거나 데이터베이스 자격 증명을 지정할 필요가 없습니다.

- 전체 백업을 복원하십시오.

```
# engine-backup --mode=restore --file=file_name --log=log_file_name --restore-permissions
```

- 구성 파일과 데이터베이스 백업을 복원하여 데이터베이스 전용 백업을 복원합니다.

```
# engine-backup --mode=restore --scope=files --scope=db --scope=dwhdb --file=file_name --log=log_file_name --restore-permissions
```



### 참고

**Manager** 데이터베이스만 복원하려면 (예: **DataECDHE** 데이터베이스가 다른 시스템에 있는 경우) **--scope=dwhdb** 매개변수를 생략할 수 있습니다.

성공하면 다음 출력이 표시됩니다.

```
You should now run engine-setup.  
Done.
```

4.

관리자를 재구성하십시오.

# engine-setup

**16.1.7. 다른 자격 증명을 사용하여 백업 복원**

**engine-backup** 명령은 **Red Hat Virtualization Manager**가 이미 설치 및 설정된 머신에 백업을 복원할 수 있지만 백업에 있는 데이터베이스의 자격 증명은 백업을 복원할 머신의 데이터베이스와 다릅니다. 이 기능은 설치 백업을 수행하고 백업에서 다른 시스템으로 설치를 복원하려는 경우에 유용합니다.

중요

백업을 복원하여 기존 설치를 덮어쓸 때 **engine-cleanup** 명령을 실행하여 **engine-backup** 명령을 사용하기 전에 기존 설치를 정리해야 합니다. **engine-cleanup** 명령은 엔진 데이터베이스만 정리하고 데이터베이스를 삭제하거나 해당 데이터베이스를 소유한 사용자를 삭제하지 않습니다. 따라서 새 데이터베이스를 만들거나 데이터베이스 자격 증명을 지정할 필요가 없습니다. 그러나 엔진 데이터베이스의 소유자에 대한 자격 증명을 알 수 없는 경우 백업을 복원하기 전에 변경해야 합니다.

**다른 자격 증명을 사용하여 백업 복원**

1.

**Red Hat Virtualization Manager** 시스템에 로그인합니다.

2.

다음 명령을 실행하고 프롬프트에 따라 **Manager**의 구성 파일을 제거하고 **Manager**의 데이터베이스를 정리합니다.

# engine-cleanup

3.

해당 사용자의 자격 증명을 알 수 없는 경우 **engine** 데이터베이스 소유자의 암호를 변경합니다.

a.

**postgresql** 명령줄을 입력합니다.

# su - postgres -c 'scl enable rh-postgresql10 -- psql'

b.

**engine** 데이터베이스를 소유한 사용자의 암호를 변경합니다.



```
postgres=# alter role user_name encrypted password 'new_password';
```

필요한 경우 **ovirt\_engine\_history** 데이터베이스를 소유한 사용자에게 대해 이 작업을 반복합니다.

4.

**--change-db-credentials** 매개변수를 사용하여 전체 백업 또는 데이터베이스 전용 백업을 복원하여 새 데이터베이스의 인증 정보를 전달합니다. **Manager**에 로컬인 데이터베이스의 **database\_location** 은 **localhost** 입니다.



참고

다음 예제에서는 각 데이터베이스에 **--password** 옵션을 사용하여 암호를 지정하지 않고 각 데이터베이스의 암호를 입력하라는 메시지를 표시합니다. 또는 대화형 프롬프트 없이 각 데이터베이스에 **--passfile=password\_file** 옵션을 사용하여 암호를 **engine-backup** 툴에 안전하게 전달할 수 있습니다.

•

전체 백업을 복원하십시오.

```
# engine-backup --mode=restore --file=file_name --log=log_file_name --change-db-credentials --db-host=database_location --db-name=database_name --db-user=engine -db-password --no-restore-permissions
```

전체 백업의 일부로 데이터도 복원되는 경우 추가 데이터베이스에 대한 수정된 자격 증명을 포함합니다.

```
engine-backup --mode=restore --file=file_name --log=log_file_name --change-db-credentials --db-host=database_location --db-name=database_name --db-user=engine -db-password --change-dwh-db-credentials --dwh-db-host=database_location --dwh-db-name=database_name --dwh-db-user=ovirt_engine_history --dwh-db-password --no-restore-permissions
```

•

구성 파일과 데이터베이스 백업을 복원하여 데이터베이스 전용 백업을 복원합니다.

```
# engine-backup --mode=restore --scope=files --scope=db --file=file_name --log=log_file_name --change-db-credentials --db-host=database_location --db-name=database_name --db-user=engine --db-password --no-restore-permissions
```

위의 예제에서는 **Manager** 데이터베이스의 백업을 복원합니다.

```
# engine-backup --mode=restore --scope=files --scope=dwhdb --file=file_name --
log=log_file_name --change-dwh-db-credentials --dwh-db-host=database_location --
dwh-db-name=database_name --dwh-db-user=ovirt_engine_history --dwh-db-password
--no-restore-permissions
```

위의 예에서는 데이터 **ECDHE** 데이터베이스의 백업을 복원합니다.

성공하면 다음 출력이 표시됩니다.

```
You should now run engine-setup.
Done.
```

5.

다음 명령을 실행하고 프롬프트에 따라 방화벽을 재구성하고 **ovirt-engine** 서비스가 올바르게 구성되었는지 확인합니다.

```
# engine-setup
```

### 16.1.8. 셀프 호스트 엔진 백업 및 복원

셀프 호스트 엔진을 백업하고 새 자체 호스팅 환경에서 복원할 수 있습니다. 다른 스토리지 유형의 새 셀프 호스트 엔진 스토리지 도메인으로 환경을 마이그레이션하는 등의 작업에는 다음 절차를 사용하십시오.

배포 중에 백업 파일을 지정하면 새로운 자체 호스팅 엔진 스토리지 도메인이 있는 새 **Manager** 가상 머신에서 백업 파일이 복원됩니다. 이전 **Manager**가 제거되고 이전 자체 호스팅 엔진 스토리지 도메인의 이름이 변경되어 새 환경이 올바르게 작동하는지 확인한 후 수동으로 제거할 수 있습니다. 새 호스트에 배포하는 것이 매우 좋습니다. 백업 환경에 사용된 호스트가 백업된 환경에 존재하는 경우 새 환경의 충돌을 방지하기 위해 복원된 데이터베이스에서 제거됩니다.

백업 및 복원 작업에는 다음 주요 작업이 포함됩니다.

1.

**engine-backup** 도구를 사용하여 원래 관리자를 백업합니다.

2.

새 셀프 호스트 엔진을 배포하고 백업을 복원합니다.

3.

새 **Manager** 가상 머신에서 **Manager** 리포지토리를 활성화합니다.

4. **셀프 호스트 엔진 노드를 다시 설치하여 구성을 업데이트합니다.**
5. **이전 자체 호스팅 엔진 스토리지 도메인을 제거합니다.**

이 절차에서는 사용자가 액세스할 수 있고 원래 관리자를 변경할 수 있다고 가정합니다.

#### 사전 요구 사항

- 관리자 및 호스트를 위해 준비된 정규화된 도메인 이름입니다. 정방향 및 역방향 조회 레코드를 모두 **DNS**에서 설정해야 합니다. 새 **Manager**는 원래 관리자와 동일한 정규화된 도메인 이름을 가져야 합니다.
- 원래 **Manager**는 최신 마이너 버전으로 업데이트해야 합니다. 백업 파일의 **Manager** 버전이 새 **Manager**의 버전과 일치해야 합니다. 업그레이드 가이드에서 **Red Hat Virtualization Manager** 업데이트를 참조하십시오.
- 환경에는 하나 이상의 일반 호스트가 있어야 합니다. 이 호스트(및 기타 일반 호스트)는 **ECDHE** 역할 및 실행 중인 가상 머신을 호스팅하는 데 활성 상태로 유지됩니다. 일반 호스트가 아직 설치되어 있지 않은 경우 일반 호스트를 선택하고 **Management** → **Select asECDHE**를 클릭하여 백업을 생성하기 전에 **ECDHE** 역할을 이동합니다.

일반 호스트를 사용할 수 없는 경우 다음 두 가지 방법으로 추가할 수 있습니다.

- 노드에서 셀프 호스트 엔진 구성을 제거합니다(하지만 환경에서 노드를 제거하지 마십시오). **15.7절. “자체 호스트 엔진 환경에서 호스트 제거”**을 참조하십시오.
- 새 일반 호스트를 추가합니다. **10.5.1절. “Red Hat Virtualization Manager에 표준 호스트 추가”**을 참조하십시오.

#### 16.1.8.1. 원래 관리자 백업

**engine-backup** 명령을 사용하여 원래 **Manager**를 백업하고 백업 파일을 프로세스 중 어느 시점에서든 액세스할 수 있도록 별도의 위치에 복사합니다.

**engine-backup --mode=backup** 옵션에 대한 자세한 내용은 관리 가이드의 **Red Hat Virtualization Manager 백업 및 복원**을 참조하십시오.

## 절차

1. 셸프 호스트 엔진 노드 중 하나에 로그인하고 환경을 전역 유지 관리 모드로 이동합니다.

```
# hosted-engine --set-maintenance --mode=global
```

2. 원래 **Manager**에 로그인하고 **ovirt-engine** 서비스를 중지합니다.

```
# systemctl stop ovirt-engine
# systemctl disable ovirt-engine
```



## 참고

원래 **Manager**를 실행에서 중지하는 것은 필수 사항은 아니지만 백업이 생성된 후 환경에 대한 변경 사항이 발생하지 않도록 하는 것이 좋습니다. 또한 원래 **Manager**와 새 **Manager**가 기존 리소스를 동시에 관리하는 것을 방지합니다.

3. 생성할 백업 파일의 이름과 백업 로그를 저장할 로그 파일의 이름을 지정하여 **engine-backup** 명령을 실행합니다.

```
# engine-backup --mode=backup --file=file_name --log=log_file_name
```

4. 파일을 외부 서버에 복사합니다. 다음 예에서 **storage.example.com**은 필요할 때까지 백업을 저장할 네트워크 스토리지 서버의 정규화된 도메인 이름이며 **/backup/**는 지정된 폴더 또는 경로입니다.

```
# scp -p file_name log_file_name storage.example.com:/backup/
```

5. 다른 용도로 **Manager** 시스템이 필요하지 않은 경우 **Red Hat Subscription Manager**에서 등록 해제하십시오.

```
# subscription-manager unregister
```

6. 셸프 호스트 엔진 노드 중 하나에 로그인하고 원래 **Manager** 가상 시스템을 종료합니다.

```
# hosted-engine --vm-shutdown
```

**Manager**를 백업한 후 새 자체 호스팅 엔진을 배포하고 새 가상 시스템에 백업을 복원합니다.

### 16.1.8.2. 새 자체 호스팅 엔진에서 백업 복원

새 호스트에서 **hosted-engine** 스크립트를 실행하고 **--restore-from-file=** 경로/to/file\_name 옵션을 사용하여 배포 중에 **Manager** 백업을 복원합니다.

#### 중요

**iSCSI** 스토리지를 사용하고 있고 이니시에이터의 **ACL**에 따라 **iSCSI** 대상 연결을 필터링하는 경우 **STORAGE\_DOMAIN\_UNREACHABLE** 오류로 인해 배포가 실패할 수 있습니다. 이를 방지하려면 자체 호스팅 엔진 배포를 시작하기 전에 **iSCSI** 구성을 업데이트해야 합니다.

- 기존 호스트에 재배포하는 경우 `/etc/iscsi/initiatorname.iscsi` 에서 호스트의 **iSCSI** 이니시에이터 설정을 업데이트해야 합니다. 이니시에이터 **IQN**은 **iSCSI** 대상에 이전에 매핑된 것과 동일하거나 해당하는 경우 새 **IQN**으로 업데이트해야 합니다.
- 새 호스트에 배포하는 경우 해당 호스트의 연결을 수락하도록 **iSCSI** 대상 구성을 업데이트해야 합니다.

**IQN**은 호스트 측(**iSCSI** 이니시에이터) 또는 스토리지 측(**iSCSI** 대상)에서 업데이트할 수 있습니다.

#### 절차

1. 백업 파일을 새 호스트에 복사합니다. 다음 예에서 **host.example.com** 은 호스트의 **FQDN** 이고 **/backup/** 는 임의의 지정된 폴더 또는 경로입니다.

```
# scp -p file_name host.example.com:/backup/
```

2. 새 호스트에 로그인합니다. **Red Hat Virtualization Host**에 배포하는 경우 기본적으로 자체 호스팅 엔진 배포 툴을 사용할 수 있습니다. **Red Hat Enterprise Linux**에 배포하는 경우 패키지를 설치해야 합니다.

```
# yum install ovirt-hosted-engine-setup
```

3.

네트워크 또는 터미널 중단 시 세션이 손실되지 않도록 화면 창 관리자를 사용하여 스크립트를 실행하는 것이 좋습니다. 설치 및 실행 화면:

```
# yum install screen
# screen
```

세션 시간 초과 또는 연결 중단이 발생하는 경우 **screen -d -r** 을 실행하여 배포 세션을 복구합니다.

4.

**hosted-engine** 스크립트를 실행하여 백업 파일의 경로를 지정합니다.

```
# hosted-engine --deploy --restore-from-file=backup/file_name
```

언제든지 스크립트를 이스케이프하려면 **CTRL+D** 사용하여 배포를 중단합니다.

5.

**Yes** 를 선택하여 배포를 시작합니다.

6.

네트워크를 구성합니다. 스크립트는 환경에 대한 관리 브릿지로 사용할 수 있는 **NIC**를 감지합니다.

7.

가상 시스템 설치에 사용자 지정 어플라이언스를 사용하려면 **OVA** 아카이브 경로를 입력합니다. 그렇지 않으면 이 필드를 비워 두고 **RHV-MECDHE**를 사용합니다.

8.

**Manager** 가상 머신의 **FQDN**을 지정합니다.

9.

**Manager**의 루트 암호를 입력합니다.

10.

**Manager**에 **root** 사용자로 로그인할 수 있는 **SSH** 공개 키를 입력하고 **root** 사용자에게 대한 **SSH** 액세스 활성화 여부를 지정합니다.

11.

가상 머신의 **CPU** 및 메모리 구성을 입력합니다.

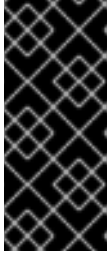
12.

**Manager** 가상 머신의 **MAC** 주소를 입력하거나 무작위로 생성된 가상 머신을 수락합니다.

**Manager** 가상 머신에 **DHCP**를 통해 **IP** 주소를 제공하려면 이 **MAC** 주소에 대한 유효한 **DHCP** 예약이 있어야 합니다. 배포 스크립트는 **DHCP** 서버를 구성하지 않습니다.

13.

가상 머신의 네트워킹 세부 정보를 입력합니다. **Static** 을 지정하는 경우 **Manager**의 **IP** 주소를 입력합니다.



중요

고정 **IP** 주소는 호스트와 동일한 서브넷에 속해야 합니다. 예를 들어 호스트가 **10.1.1.0/24**인 경우 **Manager** 가상 시스템의 **IP**는 동일한 서브넷 범위(**10.1.1.1-254/24**)에 있어야 합니다.

14.

**Manager** 가상 머신의 항목을 가상 시스템의 **/etc/hosts** 파일에 추가할지 여부를 지정합니다. 호스트 이름을 확인할 수 있는지 확인해야 합니다.

15.

**SMTP** 서버의 이름 및 **TCP** 포트 번호, 이메일 알림을 보내는 데 사용되는 이메일 주소, 이러한 알림을 수신할 쉽표로 구분된 이메일 주소 목록을 제공합니다.

16.

**admin@internal** 사용자의 암호를 입력하여 관리 포털에 액세스합니다.

이 스크립트는 가상 머신을 생성합니다. **RHV-MECDHE**를 설치해야 하는 경우 시간이 걸릴 수 있습니다.

17.

사용할 스토리지 유형을 선택합니다.

•

**NFS**의 경우 **version**, 전체 주소 및 스토리지 경로, 모든 마운트 옵션을 입력합니다.



주의

가상 시스템 데이터가 손실될 위험이 있으므로 새 스토리지 도메인에 기존 셀프 호스트 엔진 스토리지 도메인의 마운트 지점을 사용하지 마십시오.

- **iSCSI**의 경우 포털 세부 정보를 입력하고 자동 감지된 목록에서 대상 및 **LUN**을 선택합니다. 배포 중에 **iSCSI** 대상 하나를 선택할 수 있지만 동일한 포털 그룹의 모든 포털을 연결하려면 다중 경로가 지원됩니다.



참고

두 개 이상의 **iSCSI** 대상을 지정하려면 자체 호스팅 엔진을 배포하기 전에 다중 경로를 활성화해야 합니다. 자세한 내용은 **Red Hat Enterprise Linux DM Multipath** 를 참조하십시오. 다양한 옵션으로 다중 경로를 설치하고 구성하는 스크립트를 생성하는 **Multipath Helper** 도구도 있습니다.

- **Gluster** 스토리지의 경우 전체 주소 및 스토리지 경로를 입력하고 모든 마운트 옵션을 입력합니다.



주의

가상 시스템 데이터가 손실될 위험이 있으므로 새 스토리지 도메인에 기존 셀프 호스팅 엔진 스토리지 도메인의 마운트 지점을 사용하지 마십시오.





## 중요

복제본 **3 Gluster** 스토리지만 지원됩니다. 다음 구성이 있는지 확인합니다.

○

세 개의 **Gluster** 서버의 `/etc/glusterfs/glusterd.vol` 파일에서 `rpc-auth-allow-insecure` 를 **on** 으로 설정합니다.

```
option rpc-auth-allow-insecure on
```

○

다음과 같이 볼륨을 구성합니다.

```
gluster volume set _volume_ cluster.quorum-type auto
gluster volume set _volume_ network.ping-timeout 10
gluster volume set _volume_ auth.allow \*
gluster volume set _volume_ group virt
gluster volume set _volume_ storage.owner-uid 36
gluster volume set _volume_ storage.owner-gid 36
gluster volume set _volume_ server.allow-insecure on
```

●

파이버 채널의 경우 자동 감지 목록에서 **LUN**을 선택합니다. 호스트 버스 어댑터를 구성하고 연결해야 하며 **LUN**에는 기존 데이터가 포함되어 있지 않아야 합니다. 기존 **LUN**을 재사용하려면 관리 가이드 의 **LUN** 재사용을 참조하십시오.

18.

**Manager** 디스크 크기를 입력합니다.

이 스크립트는 배포가 완료될 때까지 계속됩니다.

19.

배포 프로세스에서 관리자의 **SSH** 키를 변경합니다. 클라이언트 머신이 **SSH** 오류 없이 새 **Manager**에 액세스할 수 있도록 하려면 원래 **Manager** 항목을 원래 **Manager**에 액세스하는 모든 클라이언트 머신의 `.ssh/known_hosts` 파일에서 제거합니다.

배포가 완료되면 새 **Manager** 가상 시스템에 로그인하고 필요한 리포지토리를 활성화합니다.

### 16.1.8.3. Red Hat Virtualization Manager 리포지토리 활성화

**Red Hat Subscription Manager**에 시스템을 등록하고 **Red Hat Virtualization Manager** 서브스크립션을 연결한 다음 **Manager** 리포지토리를 활성화합니다.

## 절차

1. 메시지가 표시되면 **Content Delivery Network**에 시스템을 등록하고 고객 포털 사용자 이름과 암호를 입력합니다.

```
# subscription-manager register
```



참고

**IPv6** 네트워크를 사용하는 경우 **IPv6** 전환 메커니즘을 사용하여 **Content Delivery Network** 및 서브스크립션 관리자에 액세스합니다.

2. **Red Hat Virtualization Manager** 서브스크립션 풀을 찾아서 풀 **ID**를 기록합니다.

```
# subscription-manager list --available
```

3. 풀 **ID**를 사용하여 서브스크립션을 시스템에 연결합니다.

```
# subscription-manager attach --pool=pool_id
```



참고

현재 연결된 서브스크립션을 보려면 다음을 수행합니다.

```
# subscription-manager list --consumed
```

활성화된 모든 리포지토리를 나열하려면 다음을 수행합니다.

```
# yum repolist
```

4. 리포지토리를 구성합니다.

```
# subscription-manager repos \
  --disable='*' \
  --enable=rhel-7-server-rpms \
  --enable=rhel-7-server-supplementary-rpms \
  --enable=rhel-7-server-rhv-4.3-manager-rpms \
```

```
--enable=rhel-7-server-rhv-4-manager-tools-rpms \
--enable=rhel-7-server-ansible-2.9-rpms \
--enable=jb-eap-7.2-for-rhel-7-server-rpms
```

**Manager** 및 해당 리소스가 새 자체 호스팅 환경에서 실행됩니다. 자체 호스팅 엔진 구성을 업데이트 하려면 **Manager**에서 셀프 호스팅 엔진 노드를 다시 설치해야 합니다. 표준 호스트는 영향을 받지 않습니다. 각 자체 호스팅 엔진 노드에 대해 다음 절차를 수행합니다.

#### 16.1.8.4. 호스트 재설치

관리 포털에서 **RHVH(Red Hat Virtualization Host)** 및 **Red Hat Enterprise Linux** 호스트를 다시 설치합니다. 이 절차에는 호스트를 중지하고 다시 시작하는 작업이 포함됩니다.

##### 사전 요구 사항

- 클러스터 수준에서 마이그레이션을 활성화하면 가상 머신이 클러스터의 다른 호스트로 자동 마이그레이션됩니다. 따라서 호스트 사용량이 상대적으로 낮은 시점에 호스트 재설치를 수행하는 것이 좋습니다.
- 호스트가 유지보수를 수행하기 위해 클러스터에 충분한 메모리 예약이 있는지 확인합니다. 클러스터에 충분한 메모리가 없으면 가상 머신 마이그레이션 작업이 중단되고 실패합니다. 호스트를 유지 관리로 이동하기 전에 일부 또는 모든 가상 머신을 종료하여 이 작업의 메모리 사용량을 줄일 수 있습니다.
- 다시 설치를 수행하기 전에 클러스터에 두 개 이상의 호스트가 포함되어 있는지 확인합니다. 하나의 호스트를 스토리지 풀 관리자(**SPM**) 작업을 수행하는 데 사용할 수 있어야 하므로 모든 호스트를 동시에 다시 설치하지 마십시오.

##### 절차

1. **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.
2. **Management** → **Maintenance** 를 클릭합니다.
3. **Installation** → **Reinstall** 을 클릭하여 호스트 설치 창을 엽니다.
4. 호스팅 엔진 탭을 클릭하고 드롭다운 목록에서 **DEPLOY** 를 선택합니다.

5.

**OK** 를 클릭하여 호스트를 다시 설치합니다.

성공적으로 다시 설치되면 호스트에 **Up** 상태가 표시됩니다. 호스트를 마이그레이션한 모든 가상 머신을 이제 다시 마이그레이션할 수 있습니다.



중요

**Red Hat Virtualization Host**를 **Red Hat Virtualization Manager**에 성공적으로 등록한 후 다시 설치한 후 **Install Failed** (설치 실패) 상태의 관리 포털에 잘못 표시될 수 있습니다. **Management** → **Activate** (활성화) 를 클릭하면 호스트는 **Up** 상태로 변경되고 사용할 준비가 됩니다.

자체 호스팅 엔진 노드를 다시 설치한 후 노드 중 하나에서 다음 명령을 실행하여 새 환경의 상태를 확인할 수 있습니다.

```
# hosted-engine --vm-status
```

복원 중에 이전 셀프 호스트 엔진 스토리지 도메인의 이름이 변경되었지만 복원에 결함이 있는 경우 새 환경에서 제거되지 않았습니다. 환경이 정상적으로 실행되고 있는지 확인한 후 이전 자체 호스팅 엔진 스토리지 도메인을 제거할 수 있습니다.

### 16.1.8.5. 스토리지 도메인 제거

가상화 환경에서 삭제할 데이터 센터에 스토리지 도메인이 있습니다.

절차

1. 스토리지 도메인을 클릭합니다.
2. 스토리지 도메인을 유지 관리 모드로 이동하고 분리합니다.
  - a. 스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.
  - b. 데이터 센터 탭을 클릭합니다.

- c. 유지 관리를 클릭한 다음 확인을 클릭합니다.
  - d. 분리 를 클릭한 다음 확인을 클릭합니다.
3. 제거를 클릭합니다.
  4. 선택적으로 형식 도메인 (예: **Storage Content will be lost!**) 확인란을 선택하여 도메인의 콘텐츠를 해제합니다.
  5. **OK**를 클릭합니다.

스토리지 도메인은 환경에서 영구적으로 제거됩니다.

#### 16.1.9. 기존 백업에서 자체 호스팅 엔진 복구

복구할 수 없는 문제로 인해 자체 호스팅 엔진을 사용할 수 없는 경우 문제가 발생하기 전에 수행된 백업을 사용하여 새 자체 호스팅 환경으로 복원할 수 있습니다.

배포 중에 백업 파일을 지정하면 새로운 자체 호스팅 엔진 스토리지 도메인이 있는 새 **Manager** 가상 머신에서 백업 파일이 복원됩니다. 이전 **Manager**가 제거되고 이전 자체 호스팅 엔진 스토리지 도메인의 이름이 변경되어 새 환경이 올바르게 작동하는지 확인한 후 수동으로 제거할 수 있습니다. 새 호스트에 배포하는 것이 매우 좋습니다. 백업 환경에 사용된 호스트가 백업된 환경에 존재하는 경우 새 환경의 충돌을 방지하기 위해 복원된 데이터베이스에서 제거됩니다.

셀프 호스트 엔진을 복원하려면 다음 주요 작업이 필요합니다.

1. 새 셀프 호스트 엔진을 배포하고 백업을 복원합니다.
2. 새 **Manager** 가상 머신에서 **Manager** 리포지토리를 활성화합니다.
3. 셀프 호스트 엔진 노드를 다시 설치하여 구성을 업데이트합니다.

4.

이전 자체 호스팅 엔진 스토리지 도메인을 제거합니다.

이 절차에서는 원래 **Manager**에 액세스할 수 없으며 새 호스트가 백업 파일에 액세스할 수 있다고 가정합니다.

사전 요구 사항

- 

관리자 및 호스트를 위해 준비된 정규화된 도메인 이름입니다. 정방향 및 역방향 조회 레코드를 모두 **DNS**에서 설정해야 합니다. 새 **Manager**는 원래 관리자와 동일한 정규화된 도메인 이름을 가져야 합니다.

16.1.9.1. 새 자체 호스팅 엔진에서 백업 복원

새 호스트에서 **hosted-engine** 스크립트를 실행하고 **--restore-from-file=** 경로/**to/file\_name** 옵션을 사용하여 배포 중에 **Manager** 백업을 복원합니다.



중요

**iSCSI** 스토리지를 사용하고 있고 이니시에이터의 **ACL**에 따라 **iSCSI** 대상 연결을 필터링하는 경우 **STORAGE\_DOMAIN\_UNREACHABLE** 오류로 인해 배포가 실패할 수 있습니다. 이를 방지하려면 자체 호스팅 엔진 배포를 시작하기 전에 **iSCSI** 구성을 업데이트해야 합니다.

- 

기존 호스트에 재배포하는 경우 **/etc/iscsi/initiatorname.iscsi**에서 호스트의 **iSCSI** 이니시에이터 설정을 업데이트해야 합니다. 이니시에이터 **IQN**은 **iSCSI** 대상에 이전에 매핑된 것과 동일하거나 해당하는 경우 새 **IQN**으로 업데이트해야 합니다.

- 

새 호스트에 배포하는 경우 해당 호스트의 연결을 수락하도록 **iSCSI** 대상 구성을 업데이트해야 합니다.

**IQN**은 호스트 측(**iSCSI** 이니시에이터) 또는 스토리지 측(**iSCSI** 대상)에서 업데이트할 수 있습니다.

절차

1.

백업 파일을 새 호스트에 복사합니다. 다음 예에서 **host.example.com** 은 호스트의 **FQDN** 이고 **/backup/** 는 임의의 지정된 폴더 또는 경로입니다.

```
# scp -p file_name host.example.com:/backup/
```

2.

새 호스트에 로그인합니다. **Red Hat Virtualization Host**에 배포하는 경우 기본적으로 자체 호스팅 엔진 배포 툴을 사용할 수 있습니다. **Red Hat Enterprise Linux**에 배포하는 경우 패키지를 설치해야 합니다.

```
# yum install ovirt-hosted-engine-setup
```

3.

네트워크 또는 터미널 중단 시 세션이 손실되지 않도록 화면 창 관리자를 사용하여 스크립트를 실행하는 것이 좋습니다. 설치 및 실행 화면:

```
# yum install screen
# screen
```

세션 시간 초과 또는 연결 중단이 발생하는 경우 **screen -d -r** 을 실행하여 배포 세션을 복구합니다.

4.

**hosted-engine** 스크립트를 실행하여 백업 파일의 경로를 지정합니다.

```
# hosted-engine --deploy --restore-from-file=backup/file_name
```

언제든지 스크립트를 이스케이프하려면 **CTRL+D** 사용하여 배포를 중단합니다.

5.

**Yes** 를 선택하여 배포를 시작합니다.

6.

네트워크를 구성합니다. 스크립트는 환경에 대한 관리 브릿지로 사용할 수 있는 **NIC**를 감지합니다.

7.

가상 시스템 설치에 사용자 지정 어플라이언스를 사용하려면 **OVA** 아카이브 경로를 입력합니다. 그렇지 않으면 이 필드를 비워 두고 **RHV-MECDHE**를 사용합니다.

8.

**Manager** 가상 머신의 **FQDN**을 지정합니다.

9.

**Manager**의 루트 암호를 입력합니다.

10.

**Manager**에 **root** 사용자로 로그인할 수 있는 **SSH** 공개 키를 입력하고 **root** 사용자에게 대한 **SSH** 액세스 활성화 여부를 지정합니다.

11.

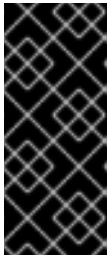
가상 머신의 **CPU** 및 메모리 구성을 입력합니다.

12.

**Manager** 가상 머신의 **MAC** 주소를 입력하거나 무작위로 생성된 가상 머신을 수락합니다. **Manager** 가상 머신에 **DHCP**를 통해 **IP** 주소를 제공하려면 이 **MAC** 주소에 대한 유효한 **DHCP** 예약이 있어야 합니다. 배포 스크립트는 **DHCP** 서버를 구성하지 않습니다.

13.

가상 머신의 네트워킹 세부 정보를 입력합니다. **Static** 을 지정하는 경우 **Manager**의 **IP** 주소를 입력합니다.



중요

고정 **IP** 주소는 호스트와 동일한 서브넷에 속해야 합니다. 예를 들어 호스트가 **10.1.1.0/24**인 경우 **Manager** 가상 시스템의 **IP**는 동일한 서브넷 범위(**10.1.1.1-254/24**)에 있어야 합니다.

14.

**Manager** 가상 머신의 항목을 가상 시스템의 **/etc/hosts** 파일에 추가할지 여부를 지정합니다. 호스트 이름을 확인할 수 있는지 확인해야 합니다.

15.

**SMTP** 서버의 이름 및 **TCP** 포트 번호, 이메일 알림을 보내는 데 사용되는 이메일 주소, 이러한 알림을 수신할 쉽표로 구분된 이메일 주소 목록을 제공합니다.

16.

**admin@internal** 사용자의 암호를 입력하여 관리 포털에 액세스합니다.

이 스크립트는 가상 머신을 생성합니다. **RHV-MECDHE**를 설치해야 하는 경우 시간이 걸릴 수 있습니다.

17.

사용할 스토리지 유형을 선택합니다.



• **NFS**의 경우 **version**, 전체 주소 및 스토리지 경로, 모든 마운트 옵션을 입력합니다.



주의

가상 시스템 데이터가 손실될 위험이 있으므로 새 스토리지 도메인에 기존 셀프 호스트 엔진 스토리지 도메인의 마운트 지점을 사용하지 마십시오.

• **iSCSI**의 경우 포털 세부 정보를 입력하고 자동 감지된 목록에서 대상 및 **LUN**을 선택합니다. 배포 중에 **iSCSI** 대상 하나를 선택할 수 있지만 동일한 포털 그룹의 모든 포털을 연결하려면 다중 경로가 지원됩니다.



참고

두 개 이상의 **iSCSI** 대상을 지정하려면 자체 호스팅 엔진을 배포하기 전에 다중 경로를 활성화해야 합니다. 자세한 내용은 **Red Hat Enterprise Linux DM Multipath**를 참조하십시오. 다양한 옵션으로 다중 경로를 설치하고 구성하는 스크립트를 생성하는 **Multipath Helper** 도구도 있습니다.

• **Gluster** 스토리지의 경우 전체 주소 및 스토리지 경로를 입력하고 모든 마운트 옵션을 입력합니다.



주의

가상 시스템 데이터가 손실될 위험이 있으므로 새 스토리지 도메인에 기존 셀프 호스트 엔진 스토리지 도메인의 마운트 지점을 사용하지 마십시오.



중요

복제본 3 Gluster 스토리지만 지원됩니다. 다음 구성이 있는지 확인합니다.

- 

세 개의 Gluster 서버의 /etc/glusterfs/glusterd.vol 파일에서 rpc-auth-allow-insecure 를 on 으로 설정합니다.

```
option rpc-auth-allow-insecure on
```

- 

다음과 같이 볼륨을 구성합니다.

```
gluster volume set _volume_ cluster.quorum-type auto
gluster volume set _volume_ network.ping-timeout 10
gluster volume set _volume_ auth.allow \*
gluster volume set _volume_ group virt
gluster volume set _volume_ storage.owner-uid 36
gluster volume set _volume_ storage.owner-gid 36
gluster volume set _volume_ server.allow-insecure on
```

- 

파이버 채널의 경우 자동 감지 목록에서 LUN을 선택합니다. 호스트 버스 어댑터를 구성하고 연결해야 하며 LUN에는 기존 데이터가 포함되어 있지 않아야 합니다. 기존 LUN을 재사용하려면 관리 가이드 의 LUN 재사용을 참조하십시오.

18.

**Manager** 디스크 크기를 입력합니다.

이 스크립트는 배포가 완료될 때까지 계속됩니다.

19.

배포 프로세스에서 관리자의 SSH 키를 변경합니다. 클라이언트 머신이 SSH 오류 없이 새 Manager에 액세스할 수 있도록 하려면 원래 Manager 항목을 원래 Manager에 액세스하는 모든 클라이언트 머신의 .ssh/known\_hosts 파일에서 제거합니다.

배포가 완료되면 새 Manager 가상 시스템에 로그인하고 필요한 리포지토리를 활성화합니다.

### 16.1.9.2. Red Hat Virtualization Manager 리포지토리 활성화

Red Hat Subscription Manager에 시스템을 등록하고 Red Hat Virtualization Manager 서브스크립션을 연결한 다음 Manager 리포지토리를 활성화합니다.

## 절차

1. 메시지가 표시되면 **Content Delivery Network**에 시스템을 등록하고 고객 포털 사용자 이름과 암호를 입력합니다.

```
# subscription-manager register
```



참고

**IPv6** 네트워크를 사용하는 경우 **IPv6** 전환 메커니즘을 사용하여 **Content Delivery Network** 및 서브스크립션 관리자에 액세스합니다.

2. **Red Hat Virtualization Manager** 서브스크립션 풀을 찾아서 풀 **ID**를 기록합니다.

```
# subscription-manager list --available
```

3. 풀 **ID**를 사용하여 서브스크립션을 시스템에 연결합니다.

```
# subscription-manager attach --pool=pool_id
```



참고

현재 연결된 서브스크립션을 보려면 다음을 수행합니다.

```
# subscription-manager list --consumed
```

활성화된 모든 리포지토리를 나열하려면 다음을 수행합니다.

```
# yum repolist
```

4. 리포지토리를 구성합니다.

```
# subscription-manager repos \
  --disable='*' \
  --enable=rhel-7-server-rpms \
  --enable=rhel-7-server-supplementary-rpms \
  --enable=rhel-7-server-rhv-4.3-manager-rpms \
```

```
--enable=rhel-7-server-rhv-4-manager-tools-rpms \
--enable=rhel-7-server-ansible-2.9-rpms \
--enable=jb-eap-7.2-for-rhel-7-server-rpms
```

**Manager** 및 해당 리소스가 새 자체 호스팅 환경에서 실행됩니다. 자체 호스팅 엔진 구성을 업데이트하려면 **Manager**에서 셀프 호스팅 엔진 노드를 다시 설치해야 합니다. 표준 호스트는 영향을 받지 않습니다. 각 자체 호스팅 엔진 노드에 대해 다음 절차를 수행합니다.

### 16.1.9.3. 호스트 재설치

관리 포털에서 **RHVH(Red Hat Virtualization Host)** 및 **Red Hat Enterprise Linux** 호스트를 다시 설치합니다. 이 절차에는 호스트를 중지하고 다시 시작하는 작업이 포함됩니다.

#### 사전 요구 사항

- 클러스터 수준에서 마이그레이션을 활성화하면 가상 머신이 클러스터의 다른 호스트로 자동 마이그레이션됩니다. 따라서 호스트 사용량이 상대적으로 낮은 시점에 호스트 재설치를 수행하는 것이 좋습니다.
- 호스트가 유지보수를 수행하기 위해 클러스터에 충분한 메모리 예약이 있는지 확인합니다. 클러스터에 충분한 메모리가 없으면 가상 머신 마이그레이션 작업이 중단되고 실패합니다. 호스트를 유지 관리로 이동하기 전에 일부 또는 모든 가상 머신을 종료하여 이 작업의 메모리 사용량을 줄일 수 있습니다.
- 다시 설치를 수행하기 전에 클러스터에 두 개 이상의 호스트가 포함되어 있는지 확인합니다. 하나의 호스트를 스토리지 풀 관리자(**SPM**) 작업을 수행하는 데 사용할 수 있어야 하므로 모든 호스트를 동시에 다시 설치하지 마십시오.

#### 절차

1. **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.
2. **Management** → **Maintenance** 를 클릭합니다.
3. **Installation** → **Reinstall** 을 클릭하여 호스트 설치 창을 엽니다.
4. 호스팅 엔진 탭을 클릭하고 드롭다운 목록에서 **DEPLOY** 를 선택합니다.

5.

**OK** 를 클릭하여 호스트를 다시 설치합니다.

성공적으로 다시 설치되면 호스트에 **Up** 상태가 표시됩니다. 호스트를 마이그레이션한 모든 가상 머신을 이제 다시 마이그레이션할 수 있습니다.



중요

**Red Hat Virtualization Host**를 **Red Hat Virtualization Manager**에 성공적으로 등록한 후 다시 설치한 후 **Install Failed** (설치 실패) 상태의 관리 포털에 잘못 표시될 수 있습니다. **Management** → **Activate** (활성화) 를 클릭하면 호스트는 **Up** 상태로 변경되고 사용할 준비가 됩니다.

자체 호스팅 엔진 노드를 다시 설치한 후 노드 중 하나에서 다음 명령을 실행하여 새 환경의 상태를 확인할 수 있습니다.

```
# hosted-engine --vm-status
```

복원 중에 이전 셀프 호스팅 엔진 스토리지 도메인의 이름이 변경되었지만 복원에 결함이 있는 경우 새 환경에서 제거되지 않았습니다. 환경이 정상적으로 실행되고 있는지 확인한 후 이전 자체 호스팅 엔진 스토리지 도메인을 제거할 수 있습니다.

#### 16.1.9.4. 스토리지 도메인 제거

가상화 환경에서 삭제할 데이터 센터에 스토리지 도메인이 있습니다.

절차

1. 스토리지 도메인을 클릭합니다.
2. 스토리지 도메인을 유지 관리 모드로 이동하고 분리합니다.
  - a. 스토리지 도메인의 이름을 클릭하여 세부 정보 보기를 엽니다.
  - b. 데이터 센터 탭을 클릭합니다.

- c. 유지 관리를 클릭한 다음 확인을 클릭합니다.
  - d. 분리 를 클릭한 다음 확인을 클릭합니다.
3. 제거를 클릭합니다.
4. 선택적으로 형식 도메인 (예: **Storage Content will be lost!**) 확인란을 선택하여 도메인의 콘텐츠를 해제합니다.
5. **OK**를 클릭합니다.

스토리지 도메인은 환경에서 영구적으로 제거됩니다.

### 16.1.10. 기존 백업에서 자체 호스트 엔진 덮어쓰기

셀프 호스트 엔진에 액세스할 수 있지만 데이터베이스 손상 또는 롤백하기 어려운 구성 오류와 같은 문제가 발생하는 경우 문제가 발생하기 전에 수행된 백업을 사용하여 환경을 이전 상태로 복원할 수 있습니다.

셀프 호스트 엔진의 이전 상태를 복원하려면 다음 단계가 포함됩니다.

- 1. 환경을 글로벌 유지 관리 모드로 배치합니다.
- 2. **Manager** 가상 머신에서 백업을 복원합니다.
- 3. 글로벌 유지보수 모드를 비활성화합니다.

**engine-backup --mode=restore** 옵션에 대한 자세한 내용은 16.1 절. “Red Hat Virtualization Manager 백업 및 복원” 을 참조하십시오.

#### 16.1.10.1. 글로벌 유지 관리 모드 활성화

**Manager** 가상 머신에서 설정 또는 업그레이드 작업을 수행하기 전에 자체 호스팅 엔진 환경을 전역 유지 관리 모드에 배치해야 합니다.

#### 절차

1. 셀프 호스팅 엔진 노드 중 하나에 로그인하고 글로벌 유지보수 모드를 활성화합니다.

```
# hosted-engine --set-maintenance --mode=global
```

2. 계속하기 전에 환경이 유지보수 모드에 있는지 확인합니다.

```
# hosted-engine --vm-status
```

클러스터가 유지보수 모드에 있음을 나타내는 메시지가 표시되어야 합니다.

#### 16.1.10.2. 기존 설치를 덮어쓰도록 백업 복원

**engine-backup** 명령은 **Red Hat Virtualization Manager**가 이미 설치 및 설정된 머신으로 백업을 복원할 수 있습니다. 이 기능은 환경을 백업하고, 해당 환경에서 변경한 작업을 수행한 다음 백업에서 환경을 복원하여 변경 사항을 실행 취소하려는 경우에 유용합니다.

복원된 환경에 호스트 추가 또는 제거와 같은 백업이 수행되었기 때문에 환경에 대한 변경 사항이 표시되지 않습니다. 이러한 변경 사항을 다시 실행해야 합니다.

#### 절차

1. **Manager** 시스템에 로그인합니다.
2. 구성 파일을 제거하고 **Manager**와 관련된 데이터베이스를 정리합니다.

```
# engine-cleanup
```

**engine-cleanup** 명령은 **Manager** 데이터베이스만 정리합니다. 데이터베이스를 삭제하거나 해당 데이터베이스를 소유한 사용자를 삭제하지 않습니다.

3. 전체 백업 또는 데이터베이스 전용 백업을 복원합니다. 사용자와 데이터베이스가 이미 있으

므로 새 데이터베이스를 만들거나 데이터베이스 자격 증명을 지정할 필요가 없습니다.

- 전체 백업을 복원하십시오.

```
# engine-backup --mode=restore --file=file_name --log=log_file_name --restore-permissions
```

- 구성 파일과 데이터베이스 백업을 복원하여 데이터베이스 전용 백업을 복원합니다.

```
# engine-backup --mode=restore --scope=files --scope=db --scope=dwhdb --file=file_name --log=log_file_name --restore-permissions
```



참고

**Manager** 데이터베이스만 복원하려면 (예: **DataECDHE** 데이터베이스가 다른 시스템에 있는 경우) **--scope=dwhdb** 매개변수를 생략할 수 있습니다.

성공하면 다음 출력이 표시됩니다.

```
You should now run engine-setup.  
Done.
```

4. 관리자를 재구성하십시오.

```
# engine-setup
```

### 16.1.10.3. 글로벌 유지 관리 모드 비활성화

절차

1. **Manager** 가상 머신에 로그인하여 종료합니다.
2. 셀프 호스트 엔진 노드 중 하나에 로그인하고 글로벌 유지 관리 모드를 비활성화합니다.

```
# hosted-engine --set-maintenance --mode=none
```



글로벌 유지 관리 모드를 종료하면 **ovirt-ha-agent**가 **Manager** 가상 시스템을 시작한 다음 **Manager**가 자동으로 시작됩니다. **Manager**를 시작하는 데 최대 10분이 걸릴 수 있습니다.

3.

환경이 실행 중인지 확인합니다.

```
# hosted-engine --vm-status
```

나열된 정보에는 엔진 상태가 포함됩니다. 엔진 상태 값은 다음과 같아야 합니다.

```
{"health": "good", "vm": "up", "detail": "Up"}
```



참고

가상 머신이 계속 부팅되고 **Manager**가 아직 시작되지 않은 경우 **Engine** 상태는 다음과 같습니다.

```
{"reason": "bad vm status", "health": "bad", "vm": "up", "detail": "Powering up"}
```

이 경우 몇 분 기다렸다가 다시 시도합니다.

환경이 다시 실행 중이면 중지된 가상 머신을 시작하고 환경의 리소스가 예상대로 작동하는지 확인할 수 있습니다.

## 16.2. RED HAT VIRTUALIZATION 데이터베이스를 원격 서버로 마이그레이션

### 16.2.1. Manager 데이터베이스를 원격 서버로 마이그레이션

**Red Hat Virtualization Manager**를 처음 구성한 후 **Manager (engine)** 데이터베이스를 원격 데이터베이스 서버로 마이그레이션할 수 있습니다. **engine-backup**을 사용하여 데이터베이스 백업을 생성하고 새 데이터베이스 서버에서 복원합니다.

새 데이터베이스 서버에는 **Red Hat Enterprise Linux 7**이 설치되어 있고 필요한 리포지토리가 활성화되어 있어야 합니다.

#### 16.2.1.1. Red Hat Virtualization Manager 리포지토리 활성화

**Red Hat Subscription Manager**에 시스템을 등록하고 **Red Hat Virtualization Manager** 서브스크립션을 연결한 다음 **Manager** 리포지토리를 활성화합니다.

절차

1. 메시지가 표시되면 **Content Delivery Network**에 시스템을 등록하고 고객 포털 사용자 이름과 암호를 입력합니다.

```
# subscription-manager register
```



참고

**IPv6** 네트워크를 사용하는 경우 **IPv6** 전환 메커니즘을 사용하여 **Content Delivery Network** 및 서브스크립션 관리자에 액세스합니다.

2. **Red Hat Virtualization Manager** 서브스크립션 풀을 찾아서 풀 **ID**를 기록합니다.

```
# subscription-manager list --available
```

3. 풀 **ID**를 사용하여 서브스크립션을 시스템에 연결합니다.

```
# subscription-manager attach --pool=pool_id
```



참고

현재 연결된 서브스크립션을 보려면 다음을 수행합니다.

```
# subscription-manager list --consumed
```

활성화된 모든 리포지토리를 나열하려면 다음을 수행합니다.

```
# yum repolist
```

4. 리포지토리를 구성합니다.

```
# subscription-manager repos \
```

```
--disable='*' \
--enable=rhel-7-server-rpms \
--enable=rhel-7-server-supplementary-rpms \
--enable=rhel-7-server-rhv-4.3-manager-rpms \
--enable=rhel-7-server-rhv-4-manager-tools-rpms \
--enable=rhel-7-server-ansible-2.9-rpms \
--enable=jb-eap-7.2-for-rhel-7-server-rpms
```

### Manager 데이터베이스를 원격 서버로 마이그레이션

1.

**Red Hat Virtualization Manager** 시스템에 로그인하고 **ovirt-engine** 서비스를 중지하여 엔진 백업을 방해하지 않도록 합니다.

```
# systemctl stop ovirt-engine.service
```

2.

엔진 데이터베이스 백업을 생성합니다.

```
# engine-backup --scope=files --scope=db --mode=backup --file=file_name --
log=log_file_name
```

3.

백업 파일을 새 데이터베이스 서버에 복사합니다.

```
# scp /tmp/engine.dump root@new.database.server.com:/tmp
```

4.

새 데이터베이스 서버에 로그인하고 **engine-backup** 을 설치합니다.

```
# yum install ovirt-engine-tools-backup
```

5.

새 데이터베이스 서버에서 데이터베이스를 복원합니다. **file\_name** 은 **Manager**에서 복사한 백업 파일입니다.

```
# engine-backup --mode=restore --scope=files --scope=db --file=file_name --
log=log_file_name --provision-db --no-restore-permissions
```

6.

데이터베이스가 마이그레이션되었으므로 **ovirt-engine** 서비스를 시작합니다.

```
# systemctl start ovirt-engine.service
```

### 16.2.2. 셀프 호스트 엔진 데이터베이스를 원격 서버로 마이그레이션

**Red Hat Virtualization Manager**를 처음 구성한 후 자체 호스팅 엔진의 엔진 데이터베이스를 원격 데이터베이스 서버로 마이그레이션할 수 있습니다. **engine-backup** 을 사용하여 데이터베이스 백업을 생성하고 새 데이터베이스 서버에서 복원합니다.

새 데이터베이스 서버에는 **Red Hat Enterprise Linux 7**이 설치되어 있고 필요한 리포지토리가 활성화되어 있어야 합니다.

### Red Hat Virtualization Manager 리포지토리 활성화

**Red Hat Subscription Manager**에 시스템을 등록하고 **Red Hat Virtualization Manager** 서브스크립션을 연결한 다음 **Manager** 리포지토리를 활성화합니다.

#### 절차

1. 메시지가 표시되면 **Content Delivery Network**에 시스템을 등록하고 고객 포털 사용자 이름과 암호를 입력합니다.

```
# subscription-manager register
```



참고

**IPv6** 네트워크를 사용하는 경우 **IPv6** 전환 메커니즘을 사용하여 **Content Delivery Network** 및 서브스크립션 관리자에 액세스합니다.

2. **Red Hat Virtualization Manager** 서브스크립션 풀을 찾아서 풀 **ID**를 기록합니다.

```
# subscription-manager list --available
```

3. 풀 **ID**를 사용하여 서브스크립션을 시스템에 연결합니다.

```
# subscription-manager attach --pool=pool_id
```



## 참고

현재 연결된 서브스크립션을 보려면 다음을 수행합니다.

```
# subscription-manager list --consumed
```

활성화된 모든 리포지토리를 나열하려면 다음을 수행합니다.

```
# yum repolist
```

4.

리포지토리를 구성합니다.

```
# subscription-manager repos \
  --disable='*' \
  --enable=rhel-7-server-rpms \
  --enable=rhel-7-server-supplementary-rpms \
  --enable=rhel-7-server-rhv-4.3-manager-rpms \
  --enable=rhel-7-server-rhv-4-manager-tools-rpms \
  --enable=rhel-7-server-ansible-2.9-rpms \
  --enable=jib-eap-7.2-for-rhel-7-server-rpms
```

셀프 호스트 엔진 데이터베이스를 원격 서버로 마이그레이션

1.

셀프 호스트 엔진 노드에 로그인하고 환경을 글로벌 유지 관리 모드로 배치합니다. 이렇게 하면 **High Availability** 에이전트를 비활성화하고 **Manager** 가상 머신이 절차 중 마이그레이션되지 않습니다.

```
# hosted-engine --set-maintenance --mode=global
```

2.

**Red Hat Virtualization Manager** 시스템에 로그인하고 **ovirt-engine** 서비스를 중지하여 엔진 백업을 방해하지 않도록 합니다.

```
# systemctl stop ovirt-engine.service
```

3.

엔진 데이터베이스 백업을 생성합니다.

```
# engine-backup --scope=files --scope=db --mode=backup --file=file_name --
log=backup_log_name
```

4.

백업 파일을 새 데이터베이스 서버에 복사합니다.

```
# scp /tmp/engine.dump root@new.database.server.com:/tmp
```

5.

새 데이터베이스 서버에 로그인하고 **engine-backup** 을 설치합니다.

```
# yum install ovirt-engine-tools-backup
```

6.

새 데이터베이스 서버에서 데이터베이스를 복원합니다. **file\_name** 은 **Manager**에서 복사한 백업 파일입니다.

```
# engine-backup --mode=restore --scope=files --scope=db --file=file_name --log=restore_log_name --provision-db --no-restore-permissions
```

7.

데이터베이스가 마이그레이션되었으므로 **ovirt-engine** 서비스를 시작합니다.

```
# systemctl start ovirt-engine.service
```

8.

셀프 호스트 엔진 노드에 로그인하고 유지 관리 모드를 끄고 고가용성 에이전트를 활성화합니다.

```
# hosted-engine --set-maintenance --mode=none
```

### 16.2.3. 분리 머신으로 데이터 마이그레이션

이 섹션에서는 **Red Hat Virtualization Manager**에서 별도의 머신으로 데이터 **EC2DHE** 데이터베이스 및 서비스를 마이그레이션하는 방법에 대해 설명합니다. 별도의 머신에서 **DataEC2DHE** 서비스를 호스팅하는 경우 개별 머신의 부하를 줄이고 각 서비스에서 **CPU** 및 메모리 리소스를 다른 프로세스와 공유하여 발생할 수 있는 충돌을 방지할 수 있습니다.

**DataEC2DHE** 서비스를 마이그레이션하고 기존 데이터 **EC2DHE** 데이터베이스(**ovirt\_engine\_history**)와 연결하거나 **DataEC2DHE** 서비스를 마이그레이션하기 전에 **DataEC2DHE** 데이터베이스를 별도의 머신으로 마이그레이션할 수 있습니다. **DataEC2DHE** 데이터베이스가 **Manager**에서 호스팅되는 경우 **DataEC2DHE** 서비스 외에도 데이터베이스를 마이그레이션하면 **Manager** 머신의 리소스에 대한 경쟁이 더 줄어들습니다. 데이터베이스를 **Manager** 머신과 새 데이터 **group** 서비스 시스템과 별도의 시스템으로 마이그레이션할 동일한 머신으로 마이그레이션할 수 있습니다.

#### 16.2.3.1. 분리 머신으로 데이터 데이터베이스 마이그레이션

데이터**EC**DHE 서비스를 마이그레이션하기 전에 **DataEC**DHE 데이터베이스(**ovirt\_engine\_history**)를 마이그레이션합니다. **engine-backup** 을 사용하여 데이터베이스 백업을 생성하고 새 데이터베이스 시스템에 복원합니다. **engine-backup** 에 대한 자세한 내용을 보려면 **engine-backup --help** 를 실행하십시오.

**DataEC**DHE 서비스만 마이그레이션하려면 16.2.3.2절. “**DataEC**DHE 서비스를 분리 머신으로 마이그레이션” 을 참조하십시오.

새 데이터베이스 서버에 **Red Hat Enterprise Linux 7**이 설치되어 있어야 합니다. 새 데이터베이스 서버에서 필요한 리포지토리를 활성화합니다.

### **Red Hat Virtualization Manager** 리포지토리 활성화

**Red Hat Subscription Manager**에 시스템을 등록하고 **Red Hat Virtualization Manager** 서브스크립션을 연결한 다음 **Manager** 리포지토리를 활성화합니다.

#### 절차

1. 메시지가 표시되면 **Content Delivery Network**에 시스템을 등록하고 고객 포털 사용자 이름과 암호를 입력합니다.

```
# subscription-manager register
```



참고

**IPv6** 네트워크를 사용하는 경우 **IPv6** 전환 메커니즘을 사용하여 **Content Delivery Network** 및 서브스크립션 관리자에 액세스합니다.

2. **Red Hat Virtualization Manager** 서브스크립션 풀을 찾아서 풀 **ID**를 기록합니다.

```
# subscription-manager list --available
```

3. 풀 **ID**를 사용하여 서브스크립션을 시스템에 연결합니다.

```
# subscription-manager attach --pool=pool_id
```



## 참고

현재 연결된 서브스크립션을 보려면 다음을 수행합니다.

```
# subscription-manager list --consumed
```

활성화된 모든 리포지토리를 나열하려면 다음을 수행합니다.

```
# yum repolist
```

4.

리포지토리를 구성합니다.

```
# subscription-manager repos \
  --disable='*' \
  --enable=rhel-7-server-rpms \
  --enable=rhel-7-server-supplementary-rpms \
  --enable=rhel-7-server-rhv-4.3-manager-rpms \
  --enable=rhel-7-server-rhv-4-manager-tools-rpms \
  --enable=rhel-7-server-ansible-2.9-rpms \
  --enable=jb-eap-7.2-for-rhel-7-server-rpms
```

분리 머신으로 데이터 데이터베이스 마이그레이션

1.

**Manager**에서 데이터 **ECDHE** 데이터베이스 및 구성 파일의 백업을 생성합니다.

```
# engine-backup --mode=backup --scope=dwhdb --scope=files --file=file_name --
log=log_file_name
```

2.

**Manager**에서 새 머신으로 백업 파일을 복사합니다.

```
# scp /tmp/file_name root@new.dwh.server.com:/tmp
```

3.

새 머신에 **engine-backup** 을 설치합니다.

```
# yum install ovirt-engine-tools-backup
```

4.

**PostgreSQL** 서버 패키지를 설치합니다.

```
# yum install rh-postgresql10 rh-postgresql10-postgresql-contrib
```



5.

**PostgreSQL** 데이터베이스를 초기화하고 **postgresql** 서비스를 시작한 후 이 서비스가 부팅 시 시작되는지 확인합니다.

```
# scl enable rh-postgresql10 -- postgresql-setup --initdb
# systemctl enable rh-postgresql10-postgresql
# systemctl start rh-postgresql10-postgresql
```

6.

새 시스템에서 데이터 **ECDE** 데이터베이스를 복원합니다. **file\_name** 은 **Manager**에서 복사한 백업 파일입니다.

```
# engine-backup --mode=restore --scope=files --scope=dwhdb --file=file_name --
log=log_file_name --provision-dwh-db --no-restore-permissions
```

이제 데이터 베이스가 **Manager**가 호스팅되는 별도의 시스템에서 호스팅됩니다. **DataECDE** 데이터베이스를 복원한 후 프롬프트에서 **engine-setup** 명령을 실행하도록 지시합니다. 이 명령을 실행하기 전에 데이터 **ECDE** 서비스를 마이그레이션합니다.

### 16.2.3.2. DataECDE 서비스를 분리 머신으로 마이그레이션

**Red Hat Virtualization Manager**에 설치 및 구성된 **DataECDE** 서비스를 별도의 시스템으로 마이그레이션할 수 있습니다. 별도의 머신에서 데이터 **ECDE** 서비스를 호스팅하는 것은 **Manager** 머신의 부하를 줄이는 데 도움이 됩니다.

이 절차에서는 **DataECDE** 서비스만 마이그레이션합니다.

**DataECDE** 서비스를 마이그레이션하기 전에 **DataECDE** 데이터베이스(**ovirt\_engine\_history**)를 마이그레이션하려면 [16.2.3.1절](#). “분리 머신으로 데이터 데이터베이스 마이그레이션” 을 참조하십시오.

#### 사전 요구 사항

- 동일한 시스템에 **Manager** 및 **DataECDE**를 설치하고 구성해야 합니다.
- 새 데이터 **ECDE** 머신을 설정하려면 다음이 있어야 합니다.
  - 관리자의 **/etc/ovirt-engine/engine.conf.d/10-setup-database.conf** 파일의 암호입니다.

- **Manager** 데이터베이스 시스템의 **TCP** 포트 **5432**로 데이터**ECDHE** 시스템에서 허용된 액세스 허용
- **Manager**의 `/etc/ovirt-engine-dwh/ovirt-engine-dwhd.conf.d/10-setup-database.conf` 파일에서 데이터**ECDHE** 데이터베이스의 사용자 이름 및 암호입니다. **16.2.3.1절**. “분리 머신으로 데이터 데이터베이스 마이그레이션” 을 사용하여 **ovirt\_engine\_history** 데이터베이스를 마이그레이션하면 백업에 해당 시스템의 데이터베이스 설정 중에 정의한 이러한 인증 정보가 포함됩니다.

이 시나리오를 설치하려면 다음 네 가지 단계가 필요합니다.

1. 새 데이터 제공 시스템 설정
2. **Manager** 시스템에서 **DataECDHE** 서비스 중지
3. 새 데이터**ECDHE** 머신 구성
4. 관리자 시스템에서 데이터 보호 패키지 비활성화

### 16.2.3.2.1. 새 데이터 제공 시스템 설정

**Red Hat Virtualization** 리포지토리를 활성화하고 **Red Hat Enterprise Linux 7** 시스템에 **DataECDHE** 설정 패키지를 설치합니다.

1. 필요한 리포지토리를 활성화합니다.
  - a. 메시지가 표시되면 **Content Delivery Network**에 시스템을 등록하고 고객 포털 사용자 이름과 암호를 입력합니다.

```
# subscription-manager register
```

- b. **Red Hat Virtualization Manager** 서브스크립션 풀을 찾아서 풀 **ID**를 기록합니다.

```
# subscription-manager list --available
```

c.

풀 ID를 사용하여 서브스크립션을 시스템에 연결합니다.

```
# subscription-manager attach --pool=pool_id
```

d.

리포지토리를 구성합니다.

```
# subscription-manager repos \
--disable='*' \
--enable=rhel-7-server-rpms \
--enable=rhel-7-server-supplementary-rpms \
--enable=rhel-7-server-rhv-4.3-manager-rpms \
--enable=rhel-7-server-rhv-4-manager-tools-rpms \
--enable=rhel-7-server-ansible-2.9-rpms \
--enable=jb-eap-7.2-for-rhel-7-server-rpms
```

2.

현재 설치된 모든 패키지가 최신 상태인지 확인합니다.

```
# yum update
```

3.

**ovirt-engine-dwh-setup** 패키지를 설치합니다.

```
# yum install ovirt-engine-dwh-setup
```

### 16.2.3.2.2. 관리자 머신에서 데이터 **sharing** 서비스 중지

1.

**DataECDHE** 서비스를 중지합니다.

```
# systemctl stop ovirt-engine-dwhd.service
```

2.

데이터베이스가 원격 시스템에서 호스팅되는 경우 **postgres.conf** 파일을 편집하여 수동으로 액세스 권한을 부여해야 합니다. **/var/opt/rh/rh-postgresql10/lib/pgsql/data/postgresql.conf** 파일을 편집하고 **listen\_addresses** 행을 다음과 일치하도록 수정합니다.

```
listen_addresses = '*'
```

행이 없거나 주석 처리된 경우 수동으로 추가합니다.

데이터베이스가 **Manager** 시스템에서 호스팅되고 **Red Hat Virtualization Manager** 정리

설정 중에 구성된 경우 기본적으로 액세스가 부여됩니다.

데이터 **EC2DHE** 데이터베이스를 구성하고 마이그레이션하는 방법에 대한 자세한 내용은 [16.2.3.1절. “분리 머신으로 데이터 데이터베이스 마이그레이션”](#) 를 참조하십시오.

3.

**postgresql** 서비스를 다시 시작하십시오.

```
# systemctl restart rh-postgresql10-postgresql
```

### 16.2.3.2.3. 새 데이터 제공 시스템 구성

이 섹션에 표시된 옵션 또는 설정 순서는 환경에 따라 다를 수 있습니다.

1.

**ovirt\_engine\_history** 데이터베이스와 **DataEC2DHE** 서비스를 동일한 머신으로 마이그레이션하는 경우 다음을 실행합니다. 그렇지 않으면 다음 단계를 진행합니다.

```
# sed -i '/^ENGINE_DB_/d' \
    /etc/ovirt-engine-dwh/ovirt-engine-dwhd.conf.d/10-setup-database.conf

# sed -i \
    -e 's;^(OVESETUP_ENGINE_CORE/enable=bool):True;1:False;' \
    -e '/^OVESETUP_CONFIG/fqdn/d' \
    /etc/ovirt-engine-setup.conf.d/20-setup-ovirt-post.conf
```

2.

**engine-setup** 명령을 실행하여 시스템에서 **DataEC2DHE** 설정을 시작합니다.

```
# engine-setup
```

3.

**Enter** 를 눌러 데이터를 구성합니다.

```
Configure Data Warehouse on this host (Yes, No) [Yes]:
```

4.

**Enter** 를 눌러 자동으로 감지된 호스트 이름을 수락하거나 대체 호스트 이름을 입력하고 **Enter** 키를 누릅니다.

```
Host fully qualified DNS name of this server [autodetected host name]:
```

5.

**Enter** 를 눌러 방화벽을 자동으로 구성하거나 **No** 를 입력하고 **Enter** 를 눌러 기존 설정을

유지합니다.

*Setup can automatically configure the firewall on this system.*

*Note: automatic configuration of the firewall may overwrite current settings.*

*Do you want Setup to configure the firewall? (Yes, No) [Yes]:*

방화벽을 자동으로 구성하도록 선택하고 방화벽 관리자가 활성화되어 있지 않은 경우 지원되는 옵션 목록에서 선택한 방화벽 관리자를 선택하라는 메시지가 표시됩니다. 방화벽 관리자의 이름을 입력하고 **Enter** 를 누릅니다. 이는 하나의 옵션만 나열된 경우에도 적용됩니다.

6.

**Manager**에 대해 정규화된 도메인 이름 및 암호를 입력합니다. **Enter** 를 눌러 서로 다른 필드에서 기본값을 적용합니다.

*Host fully qualified DNS name of the engine server []: engine-fqdn*

*Setup needs to do some actions on the remote engine server. Either automatically, using ssh as root to access it, or you will be prompted to manually perform each such action.*

*Please choose one of the following:*

*1 - Access remote engine server using ssh as root*

*2 - Perform each action manually, use files to copy content around*

*(1, 2) [1]:*

*ssh port on remote engine server [22]:*

*root password on remote engine server engine-fqdn: password*

7.

**Manager** 데이터베이스 시스템의 **FQDN** 및 암호를 입력합니다. **Enter** 를 눌러 서로 다른 필드에서 기본값을 적용합니다.

*Engine database host []: manager-db-fqdn*

*Engine database port [5432]:*

*Engine database secured connection (Yes, No) [No]:*

*Engine database name [engine]:*

*Engine database user [engine]:*

*Engine database password: password*

8.

설치 설정을 확인합니다.

*Please confirm installation settings (OK, Cancel) [OK]:*

이제 데이터 서비스가 원격 머신에 구성되어 있습니다. **Manager** 시스템에서 **DataECDHE** 서비스를 비활성화합니다.

#### 16.2.3.2.4. 관리자 머신에서 데이터 **hieradata** 서비스 비활성화

1.

**Manager** 시스템에서 관리자를 다시 시작합니다.

```
# service ovirt-engine restart
```

2.

다음 명령을 실행하여 `/etc/ovirt-engine-setup.conf.d/20-setup-ovirt-post.conf` 파일을 수정하고 옵션을 **False** 로 설정합니다.

```
# sed -i \
-e 's;^\(OVESETUP_DWH_CORE/enable=bool\):True;\1:False;' \
-e 's;^\(OVESETUP_DWH_CONFIG/remoteEngineConfigured=bool\):True;\1:False;' \
/etc/ovirt-engine-setup.conf.d/20-setup-ovirt-post.conf
```

3.

**DataECDHE** 서비스를 비활성화합니다.

```
# systemctl disable ovirt-engine-dwhd.service
```

4.

데이터**ECDHE** 파일을 삭제합니다.

```
# rm -f /etc/ovirt-engine-dwh/ovirt-engine-dwhd.conf.d/* .conf /var/lib/ovirt-engine-dwh/backups/*
```

이제 데이터 브릿지 서비스가 **Manager**와 별도의 시스템에서 호스팅됩니다.

## 16.3. 백업 스토리지 도메인을 사용하여 가상 머신 백업 및 복원

### 16.3.1. 백업 스토리지 도메인 설명

백업 스토리지 도메인은 재해 복구, 마이그레이션 또는 기타 백업/복원 사용 모델을 위해 가상 머신 및 가상 머신 템플릿을 저장하고 마이그레이션하는 데 특히 사용할 수 있습니다. 백업 도메인은 백업 도메인의 모든 가상 시스템이 전원이 꺼진 상태라는 점에서 백업되지 않은 도메인과 다릅니다. 백업 도메인에서는 가상 머신을 실행할 수 없습니다.

모든 데이터 스토리지 도메인을 백업 도메인으로 설정할 수 있습니다. 도메인 관리 대화 상자에서 확인란을 선택하거나 선택 해제하여 이 설정을 활성화 또는 비활성화할 수 있습니다. 이 설정은 해당 스토리지 도메인의 모든 가상 머신이 중지된 후에만 활성화할 수 있습니다.

백업 도메인에 저장된 가상 머신을 시작할 수 없습니다. **Manager**는 이 작업과 백업을 무효화할 수 있는 다른 작업을 차단합니다. 그러나 가상 머신의 디스크가 백업 도메인의 일부가 아닌 경우 백업 도메인에

저장된 템플릿을 기반으로 가상 머신을 실행할 수 있습니다.

다른 스토리지 도메인 유형과 마찬가지로 백업 도메인을 데이터 센터로 연결하거나 분리할 수 있습니다. 따라서 백업 저장 외에도 백업 도메인을 사용하여 데이터 센터 간에 가상 머신을 마이그레이션할 수 있습니다.

## 이점

내보내기 도메인이 아닌 백업 도메인을 사용해야 하는 몇 가지 이유는 다음과 같습니다.

- 하나의 내보내기 도메인과 달리 데이터 센터에 백업 스토리지 도메인을 여러 개 보유할 수 있습니다.
- 백업 및 재해 복구에 사용할 백업 스토리지 도메인을 전용으로 지정할 수 있습니다.
- 가상 머신, 템플릿 또는 스냅샷의 백업을 백업 스토리지 도메인으로 전송할 수 있습니다.
- 내보내기 도메인보다 백업 도메인에서 다수의 가상 머신, 템플릿 또는 **OVF** 파일을 마이그레이션하는 것이 훨씬 빠릅니다.
- 백업 도메인은 내보내기 도메인보다 디스크 공간을 더 효율적으로 사용합니다.
- 백업 도메인은 파일 스토리지(**NFS, Gluster**) 및 블록 스토리지(**Fiber Channel** 및 **iSCSI**)를 모두 지원합니다. 이는 파일 스토리지만 지원하는 내보내기 도메인과 다릅니다.
- 제한 사항을 고려하여 스토리지 도메인에 대한 백업 설정을 동적으로 활성화 및 비활성화할 수 있습니다.

## 제한 사항

- **\_backup** 도메인의 모든 가상 머신 또는 템플릿에는 동일한 도메인에 모든 디스크가 있어야 합니다.
- 백업 도메인으로 설정하려면 스토리지 도메인의 모든 가상 머신의 전원을 꺼야 합니다.

- 이렇게 하면 디스크의 데이터를 조작할 수 있으므로 백업 도메인에 저장된 가상 머신을 실행할 수 없습니다.
- 메모리 볼륨은 활성 가상 머신에서만 지원되므로 백업 도메인은 메모리 볼륨의 대상이 될 수 없습니다.
- 백업 도메인에서 가상 머신을 미리 볼 수 없습니다.
- 가상 머신을 백업 도메인으로 실시간 마이그레이션할 수 없습니다.
- 백업 도메인을 마스터 도메인으로 설정할 수 없습니다.
- 셀프 호스트 엔진의 도메인을 백업 도메인으로 설정할 수 없습니다.
- 기본 스토리지 도메인을 백업 도메인으로 사용하지 마십시오.

### 16.3.2. 데이터 스토리지 도메인을 백업 도메인으로 설정

#### 사전 요구 사항

- 스토리지 도메인의 가상 머신 또는 템플릿에 속하는 모든 디스크는 동일한 도메인에 있어야 합니다.
- 도메인의 모든 가상 머신의 전원이 꺼야 합니다.

#### 절차

1. 관리 포털에서 **Storage** → **Domains** 를 선택합니다.
2. 새 스토리지 도메인을 생성하거나 기존 스토리지 도메인을 선택하고 **Manage Domain** 을 클릭합니다. 도메인 관리 대화 상자가 열립니다.



3.

**Advanced Parameters** 에서 **Backup** 확인란을 선택합니다.

이제 도메인이 백업 도메인입니다.

### 16.3.3. 백업 도메인을 사용하여 가상 머신 또는 스냅샷 백업 또는 복원

전원이 꺼진 가상 머신 또는 스냅샷을 백업할 수 있습니다. 그런 다음 동일한 데이터 센터에 백업을 저장하고 필요에 따라 복원하거나 다른 데이터 센터로 마이그레이션할 수 있습니다.

절차: 가상 머신 백업

1.

백업 도메인을 생성합니다. **16.3.2절**. “데이터 스토리지 도메인을 백업 도메인으로 설정” 을 참조하십시오.

2.

백업하려는 가상 머신을 기반으로 새 가상 머신을 생성합니다.

•

스냅샷을 백업하려면 먼저 스냅샷에서 가상 머신을 생성합니다. [가상 머신 관리 가이드의 스냅샷에서 가상 머신 생성](#) 을 참조하십시오.

•

가상 머신을 백업하려면 먼저 가상 머신을 복제합니다. [가상 머신 관리 가이드의 가상 머신 확장 가이드](#) 를 참조하십시오. 계속하기 전에 복제본의 전원이 꺼졌는지 확인합니다.

3.

새 가상 머신을 백업 도메인으로 내보냅니다. [가상 머신 관리 가이드의 데이터 도메인으로 가상 머신 내보내기](#) 를 참조하십시오.

절차: 가상 머신 복원

1.

가상 머신 백업을 저장하는 백업 스토리지 도메인이 데이터 센터에 연결되어 있는지 확인합니다.

2.

백업 도메인에서 가상 머신을 가져옵니다. **11.7.5절**. “가져온 데이터 스토리지 도메인에서 가상 머신 가져오기” 을 참조하십시오.

관련 정보

- [11.7.2절. “스토리지 도메인 가져오기”](#)
- [11.7.3절. “동일한 환경에서 데이터 센터 간 스토리지 도메인 마이그레이션”](#)
- [11.7.4절. “다른 환경에서 데이터 센터 간 스토리지 도메인 마이그레이션”](#)
- [11.7.5절. “가져온 데이터 스토리지 도메인에서 가상 머신 가져오기”](#)

**16.4. 백업 및 복원 API를 사용하여 가상 머신 백업 및 복원**

**16.4.1. 백업 및 복원 API**

백업 및 복원 **API**는 가상 머신의 전체 또는 파일 수준 백업 및 복원을 수행할 수 있는 함수 컬렉션입니다. **API**는 라이브 스냅샷 및 **REST API**와 같은 **Red Hat Virtualization**의 여러 구성 요소를 결합하여 독립 소프트웨어 공급자가 제공하는 백업 소프트웨어가 포함된 가상 머신에 연결할 수 있는 임시 볼륨을 생성하고 사용합니다.

지원되는 타사 백업 벤더는 [Red Hat Virtualization Ecosystem](#) 에서 참조하십시오.

**16.4.2. 가상 머신 백업**

백업 및 복원 **API**를 사용하여 가상 머신을 백업합니다. 이 절차에서는 백업할 가상 머신과 백업 관리용 소프트웨어가 설치된 가상 머신이라는 두 개의 가상 머신이 있다고 가정합니다.

가상 머신 백업

1. **REST API**를 사용하여 백업할 가상 머신의 스냅샷을 생성합니다.

```

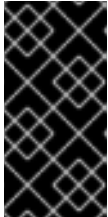
POST /api/vms/{vm:id}/snapshots/ HTTP/1.1
Accept: application/xml
Content-type: application/xml

<snapshot>
  <description>BACKUP</description>
</snapshot>
    
```



## 참고

- 여기에서 **{vm:id}** 를 스냅샷을 만든 가상 시스템의 **VM ID**로 바꿉니다. 이 **ID**는 새 가상 머신의 일반 탭에서 사용할 수 있으며 관리 포털 및 **VM** 포털의 가상 머신 편집 창에서 사용할 수 있습니다.
- 가상 머신의 스냅샷을 사용하면 스냅샷 아래의 초기화 에 있는 구성 속성의 현재 구성 데이터가 구성 속성의 데이터 속성에 저장됩니다.



## 중요

공유 가능 또는 직접 **LUN** 디스크를 기반으로 표시된 디스크의 스냅샷을 가져올 수 없습니다.

2.

스냅샷 아래의 **data** 속성에서 가상 머신의 구성 데이터를 검색합니다.

```
GET /api/vms/{vm:id}/snapshots/{snapshot:id} HTTP/1.1
All-Content: true
Accept: application/xml
Content-type: application/xml
```



## 참고

- 여기에서 **{vm:id}** 를 이전에 만든 가상 시스템의 **ID**로 바꿉니다. **{snapshot:id}** 를 스냅샷 **ID**로 교체합니다.
- 응답에서 추가 **OVF** 데이터를 검색하려면 **All-Content: true** 헤더를 추가합니다. **XML** 응답의 **OVF** 데이터는 **VM** 구성 요소인 **<initialization>** **<configuration>** 내에 있습니다. 나중에 이 데이터를 사용하여 가상 머신을 복원합니다.

3.

스냅샷 **ID**를 가져옵니다.

```
GET /api/vms/{vm:id}/snapshots/ HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

4.

스냅샷의 디스크 **ID**를 확인합니다.

```
GET /api/vms/{vm:id}/snapshots/{snapshot:id}/disks HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

5.

올바른 인터페이스 유형(예: **virtio\_scsi**)을 사용하여 백업 가상 머신에 활성 디스크 연결로 스냅샷을 연결합니다.

```
POST /api/vms/{vm:id}/diskattachments/ HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

```
<disk_attachment>
<active>true</active>
<interface>_virtio_scsi_</interface>
<disk id="{disk:id}">
<snapshot id="{snapshot:id}"/>
</disk>
</disk_attachment>
```



참고

여기에서 **{vm:id}** 를 이전에 만든 스냅샷이 아닌 백업 가상 시스템의 **ID**로 바꿉니다. **{disk:id}** 를 디스크 **ID**로 바꿉니다. **{snapshot:id}** 를 스냅샷 **ID**로 교체합니다.

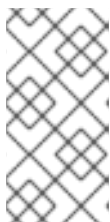
6.

백업 가상 머신의 백업 소프트웨어를 사용하여 스냅샷 디스크의 데이터를 백업합니다.

7.

백업 가상 머신에서 스냅샷 디스크 연결을 제거합니다.

```
DELETE /api/vms/{vm:id}/diskattachments/{snapshot:id} HTTP/1.1
Accept: application/xml
Content-type: application/xml
```



참고

여기에서 **{vm:id}** 를 이전에 만든 스냅샷이 아닌 백업 가상 시스템의 **ID**로 바꿉니다. **{snapshot:id}** 를 스냅샷 **ID**로 교체합니다.

8.

필요한 경우 스냅샷을 삭제합니다.

```
DELETE /api/vms/{vm:id}/snapshots/{snapshot:id} HTTP/1.1
Accept: application/xml
Content-type: application/xml
```



참고

여기에서 **{vm:id}** 를 이전에 만든 가상 시스템의 **ID**로 바꿉니다.  
**{snapshot:id}** 를 스냅샷 **ID**로 교체합니다.

별도의 가상 머신에 설치된 백업 소프트웨어를 사용하여 고정된 시점에 가상 머신의 상태를 백업했습니다.

### 16.4.3. 가상 머신 복원

백업 및 복원 **API**를 사용하여 백업한 가상 머신을 복원합니다. 이 절차에서는 이전 백업을 관리하는 데 사용된 백업 가상 머신이 설치되어 있다고 가정합니다.

#### 가상 머신 복원

1. 관리 포털에서 백업을 복원할 부동 디스크를 생성합니다. 유동 디스크를 만드는 방법에 대한 자세한 내용은 [13.6.1절. “가상 디스크 생성”](#)을 참조하십시오.
2. 백업 가상 머신에 디스크를 연결합니다.

```
POST /api/vms/{vm:id}/disks/ HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

```
<disk id="{disk:id}">
</disk>
```



참고

여기에서 **{vm:id}** 를 이전에 만든 스냅샷이 아닌 이 백업 가상 시스템의 **ID**로 바꿉니다. 가상 머신을 백업하는 동안 **{disk:id}** 를 가져온 디스크 **ID**로 교체합니다.

3.

백업 소프트웨어를 사용하여 백업을 디스크에 복원합니다.

4.

백업 가상 머신에서 디스크를 분리합니다.

```
DELETE /api/vms/{vm:id}/disks/{disk:id} HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

```
<action>
  <detach>true</detach>
</action>
```

**+ NOTE: here, replace {vm:id} with the ID of this backup virtual machine, not the virtual machine whose snapshot you made earlier. {disk:id} 를 디스크 ID로 바꿉니다.**

5.

복원 중인 가상 머신의 구성 데이터를 사용하여 새 가상 머신을 생성합니다.

```
POST /api/vms/ HTTP/1.1
Accept: application/xml
Content-type: application/xml
```

```
<vm>
  <cluster>
    <name>cluster_name</name>
  </cluster>
  <name>_NAME_</name>
  <initialization>
    <configuration>
<data>
<!-- omitting long ovf data -->
</data>
    <type>ovf</type>
  </configuration>
  </initialization>
  ...
</vm>
```



참고

가상 머신을 생성하는 동안 **ovf**의 값을 재정의하려면 초기화 요소 전후의 요소를 다시 정의합니다. 초기화 요소 내에 없습니다.

6.

디스크를 새 가상 머신에 연결합니다.

```
POST /api/vms/{vm:id}/disks/ HTTP/1.1
```

```
Accept: application/xml
```

```
Content-type: application/xml
```

```
<disk id="{disk:id}">
```

```
</disk>
```



#### 참고

여기에서 **{vm:id}** 를 이전에 만든 스냅샷이 아닌 새 가상 시스템의 **ID**로 바꿉니다. **{disk:id}** 를 디스크 **ID**로 바꿉니다.

백업 및 복원 **API**를 사용하여 생성된 백업을 사용하여 가상 머신을 복원했습니다.

## 17장. RED HAT SATELLITE를 통한 에라타 관리

**Red Hat Virtualization Manager**에서 **Red Hat Satellite**에서 에라타를 볼 수 있도록 **Red Hat Virtualization**을 구성할 수 있습니다. 이를 통해 관리자는 **Red Hat Satellite** 공급자와 연결된 호스트, 가상 시스템 및 **Manager**에 대해 사용 가능한 에라타에 대한 업데이트를 받을 수 있습니다. 그러면 관리자는 필요한 호스트, 가상 시스템 또는 **Manager**에서 업데이트를 실행하여 업데이트를 적용하도록 선택할 수 있습니다. **Red Hat Satellite**에 대한 자세한 내용은 [Red Hat Satellite 문서](#)를 참조하십시오.

**Red Hat Virtualization 4.3**에서는 **Red Hat Satellite 6.5**를 통한 에라타 관리를 지원합니다.

### 중요

**Manager**, 호스트 및 가상 시스템은 해당 **FQDN**으로 **Satellite** 서버에서 식별됩니다. 이렇게 하면 **Red Hat Virtualization**에서 외부 콘텐츠 호스트 ID를 유지 관리할 필요가 없습니다.

**Manager**, 호스트 및 가상 시스템을 관리하는 데 사용되는 **Satellite** 계정에는 관리자 권한 및 기본 조직 세트가 있어야 합니다.

### Red Hat Virtualization 에라타 구성

**Manager**, 호스트 및 가상 머신을 **Red Hat Satellite** 공급자와 연결하려면 먼저 **Manager**를 공급자와 연결해야 합니다. 그러면 호스트가 동일한 공급자와 연결되고 구성됩니다. 마지막으로 가상 머신은 동일한 공급자와 연결되고 구성됩니다.

1.

필요한 **Satellite** 서버를 외부 공급자로 추가하여 **Manager**를 연결합니다. 자세한 내용은 [14.2.1절. “호스트 프로비저닝을 위한 Red Hat Satellite 인스턴스 추가”](#)를 참조하십시오.

### 참고

**Manager**는 콘텐츠 호스트로 **Satellite** 서버에 등록하고 **katello-agent** 패키지가 설치되어 있어야 합니다.

호스트 등록을 구성하는 방법에 대한 자세한 내용은 [Red Hat Satellite 문서 호스트 관리에서 호스트 등록](#)을 참조하십시오.



2. 필요한 경우 사용 가능한 에라타를 표시하도록 필수 호스트를 구성합니다. 자세한 내용은 **10.5.3절. “호스트에 대한 Satellite 에라타 관리 구성”**를 참조하십시오.
  
3. 필요한 경우 사용 가능한 에라타를 표시하도록 필요한 가상 머신을 구성합니다. 필요한 가상 머신을 구성하기 전에 관련 호스트를 구성해야 합니다. 자세한 내용은 가상 머신 **관리 가이드**에서 가상 머신용 **Red Hat Satellite 에라타 관리 구성**을 참조하십시오.

### **Red Hat Virtualization Manager 에라타 보기**

1. **Administration → Errata** )를 클릭합니다.
  
2. 보안, 버그 또는 기능 개선 확인란을 선택하여 해당 에라타 유형만 확인합니다.

호스트에 사용 가능한 에라타 보기에 대한 자세한 내용은 **10.5.21절. “호스트 에라타 보기”** 및 가상 머신의 경우 가상 머신 관리 가이드에서 **가상 머신용 Red Hat Satellite 에라타 보기**를 참조하십시오.

## 18장. ANSIBLE을 사용하여 구성 작업 자동화

**Ansible**은 시스템 구성, 소프트웨어 배포, 롤링 업데이트를 수행하는 데 사용되는 자동화 틀입니다. **Ansible**에는 **Red Hat Virtualization**에 대한 지원이 포함되어 있으며 데이터 센터 설정 및 구성, 사용자 또는 가상 머신 작업과 같은 설치 후 작업을 자동화할 수 있는 **Ansible** 모듈을 사용할 수 있습니다.

**Ansible**은 **REST API** 및 **SDK**에 비해 **Red Hat Virtualization** 구성을 더 쉽게 자동화하고 다른 **Ansible** 모듈과 통합할 수 있는 방법을 제공합니다. **Red Hat Virtualization**에서 사용할 수 있는 **Ansible** 모듈에 대한 자세한 내용은 **Ansible** 설명서의 **Ovirt 모듈**을 참조하십시오.



참고

**Ansible Tower**는 웹 인터페이스 및 **Ansible**용 **REST API**를 통해 액세스할 수 있는 그래픽으로 활성화된 프레임워크입니다. **Ansible Tower**에 대한 지원을 원한다면 **Red Hat Virtualization** 서브스크립션에 포함되지 않은 **Ansible Tower** 라이선스가 있어야 합니다.

**Ansible**은 **Red Hat Virtualization**과 함께 제공됩니다. **Ansible**을 설치하려면 **Manager** 시스템에서 다음 명령을 실행합니다.

```
# yum install ansible
```

대체 설치 지침은 **Ansible** 설명서 및 **Ansible** 사용에 대한 정보를 참조하십시오.



참고

**Ansible** 플레이북을 실행할 때 관리자의 세부 정보 수준을 영구적으로 늘리려면 다음 행을 사용하여 **/etc/ovirt-engine/engine.conf.d/**에 구성 파일을 만듭니다.

```
ANSIBLE_PLAYBOOK_VERBOSE_LEVEL=4
```

**systemctl restart ovirt-engine**을 실행하여 파일을 생성한 후 **Manager**를 다시 시작해야 합니다.

### 18.1. ANSIBLE 역할

**Red Hat Virtualization** 인프라의 다양한 부분을 구성하고 관리하는 데 도움이 되는 여러 **Ansible** 역할

할을 사용할 수 있습니다. **Ansible** 역할은 대규모 플레이북을 다른 사용자와 공유할 수 있는 작고 재사용 가능한 파일로 분할하여 **Ansible** 코드를 모듈화하는 방법을 제공합니다.

**Red Hat Virtualization**에서 사용할 수 있는 **Ansible** 역할은 다양한 내부 구성 요소로 분류됩니다. **Ansible** 역할에 대한 자세한 내용은 **oVirt Ansible Roles** 설명서를 참조하십시오. **Ansible** 역할로 설치된 문서는 **18.1.1절. “Ansible 역할 설치”**을 참조하십시오.

### 18.1.1. Ansible 역할 설치

**Red Hat Virtualization Manager** 리포지토리에서 **Red Hat Virtualization**에 대한 **Ansible** 역할을 설치할 수 있습니다. 다음 명령을 사용하여 **Manager** 시스템에 **Ansible** 역할을 설치합니다.

```
# yum install ovirt-ansible-roles
```

기본적으로 역할은 **/usr/share/ansible/roles**에 설치됩니다. **ovirt-ansible-roles** 패키지의 구조는 다음과 같습니다.

- **/usr/share/ansible/roles** - 역할을 저장합니다.
- **/usr/share/doc/ovirt-ansible-roles/** - 예제, 기본 개요 및 권한 부여를 저장합니다.
- **/usr/share/doc/ansible/roles/role\_name** - 역할과 관련된 문서를 저장합니다.

### 18.1.2. Ansible 역할을 사용하여 Red Hat Virtualization 구성

다음 절차에서는 **Ansible** 역할을 사용하여 **Red Hat Virtualization**을 구성하는 플레이북을 생성 및 실행하는 방법을 안내합니다. 이 예에서는 **Ansible**을 사용하여 로컬 시스템의 **Manager**에 연결하고 새 데이터 센터를 생성합니다.

사전 요구 사항

- **/etc/ansible/ansible.cfg**의 **roles\_path** 옵션이 **Ansible** 역할의 위치 (**/usr/share/ansible/roles**)를 가리키는지 확인합니다.

- **Playbook**을 실행하는 머신에 **Python SDK**가 설치되어 있는지 확인합니다.

### **Ansible** 역할을 사용하여 **Red Hat Virtualization** 구성

1. **Red Hat Virtualization Manager** 사용자 암호를 저장할 파일을 작업 디렉터리에 만듭니다.

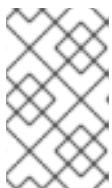
```
# cat passwords.yml
---
engine_password: youruserpassword
```

2. 사용자 암호를 암호화합니다. **Vault** 암호를 요청합니다.

```
# ansible-vault encrypt passwords.yml
New Vault password:
Confirm New Vault password:
```

3. **URL**, 인증서 위치 및 사용자와 같은 관리자 세부 정보를 저장하는 파일을 생성합니다.

```
# cat engine_vars.yml
---
engine_url: https://example.engine.redhat.com/ovirt-engine/api
engine_user: admin@internal
engine_cafile: /etc/pki/ovirt-engine/ca.pem
```



참고

원하는 경우 이러한 변수를 플레이북에 직접 추가할 수 있습니다.

4. 플레이북을 생성합니다. 이를 단순화하기 위해 **/usr/share/doc/ovirt-ansible-roles/examples**의 예제를 복사하고 수정할 수 있습니다.

```
# cat rhv_infra.yml
---
- name: RHV infrastructure
  hosts: localhost
  connection: local
  gather_facts: false

  vars_files:
    # Contains variables to connect to the Manager
```

```

- engine_vars.yml
# Contains encrypted engine_password variable using ansible-vault
- passwords.yml

```

*pre\_tasks:*

```

- name: Login to RHV
  ovirt_auth:
    url: "{{ engine_url }}"
    username: "{{ engine_user }}"
    password: "{{ engine_password }}"
    ca_file: "{{ engine_cafile | default(omit) }}"
    insecure: "{{ engine_insecure | default(true) }}"
  tags:
    - always

```

*vars:*

```

data_center_name: mydatacenter
data_center_description: mydatacenter
data_center_local: false
compatibility_version: 4.1

```

*roles:*

```

- ovirt-datacenters

```

*post\_tasks:*

```

- name: Logout from RHV
  ovirt_auth:
    state: absent
    ovirt_auth: "{{ ovirt_auth }}"
  tags:
    - always

```

5.

플레이북을 실행합니다.

```

# ansible-playbook --ask-vault-pass rhv_infra.yml

```

**ovirt-datacenters Ansible** 역할을 사용하여 **mydatacenter** 라는 데이터 센터를 생성합니다.

## 19장. 사용자 및 역할

### 19.1. 사용자 소개

**Red Hat Virtualization**에는 로컬 도메인과 외부 도메인의 두 가지 유형의 사용자 도메인이 있습니다. 내부 도메인이라는 기본 로컬 도메인과 기본 사용자 **admin**은 **Manager** 설치 프로세스 중에 생성됩니다.

**ovirt-aaa-jdbc-tool**을 사용하여 내부 도메인에 추가 사용자를 생성할 수 있습니다. 로컬 도메인에서 생성된 사용자 계정을 로컬 사용자라고 합니다. **Red Hat Directory Server, Active Directory, OpenLDAP** 및 기타 지원되는 많은 옵션과 같은 외부 디렉터리 서버를 **Red Hat Virtualization** 환경에 연결하여 외부 도메인으로 사용할 수도 있습니다. 외부 도메인에서 생성된 사용자 계정을 디렉터리 사용자라고 합니다.

로컬 사용자와 디렉터리 사용자는 모두 환경에서 작동하기 전에 관리 포털을 통해 적절한 역할 및 권한을 사용하여 할당해야 합니다. 사용자 역할에는 최종 사용자와 관리자의 두 가지 주요 유형이 있습니다. 최종 사용자 역할은 **VM** 포털에서 가상 리소스를 사용하고 관리합니다. 관리자 역할은 관리 포털을 사용하여 시스템 인프라를 유지 관리합니다. 가상 머신 및 호스트와 같은 개별 리소스 또는 클러스터 및 데이터 센터와 같은 오브젝트 계층에 대해 역할을 사용자에게 할당할 수 있습니다.

### 19.2. 디렉터리 서버 소개

설치하는 동안 **Red Hat Virtualization Manager**는 내부 도메인에 관리자를 생성합니다. 사용자를 **admin@internal**라고도 합니다. 이 계정은 처음 환경을 구성하고 문제 해결을 위해 사용할 때 사용하기 위한 것입니다. 외부 디렉터리 서버를 연결하고 디렉터리 사용자를 추가하고 적절한 역할 및 권한으로 할당할 후 필수가 아닌 경우 **admin@internal** 사용자를 비활성화할 수 있습니다. 지원되는 디렉터리 서버는 다음과 같습니다.

- **389ds**
- **389ds RFC-2307 Schema**
- **Active Directory**
- **IBM Security Directory Server**
- **IBM Security Directory Server RFC-2307 스키마**

- **FreeIPA**
- **iDM**
- **Novell eDirectory RFC-2307 스키마**
- **OpenLDAP RFC-2307 Schema**
- **OpenLDAP 표준 스키마**
- **Oracle Unified Directory RFC-2307 스키마**
- **RFC-2307 스키마(Generic)**
- **Red Hat Directory Server(RHDS)**
- **RHDS(Red Hat Directory Server) RFC-2307 스키마**
- **iPlanet**



중요

동일한 시스템에 **Red Hat Virtualization Manager(ECDHE)** 및 **IdM(ipa-server)**을 설치할 수 없습니다. **IdM**은 **Red Hat Virtualization Manager**에 필요한 **mod\_ssl** 패키지와 호환되지 않습니다.

**중요**

**Active Directory**를 디렉터리 서버로 사용하고 템플릿 및 가상 머신 생성에 **sysprep**을 사용하려면 **Red Hat Virtualization** 관리 사용자가 다음을 위해 도메인에 대한 제어를 위임해야 합니다.

- 도메인에 컴퓨터를 조인합니다.
- 그룹의 멤버십 수정

**Active Directory**에서 사용자 계정 생성에 대한 자세한 내용은 <http://technet.microsoft.com/en-us/library/cc732336.aspx>의 내용을 참조하십시오.

**Active Directory**에서 제어 위임에 대한 자세한 내용은 <http://technet.microsoft.com/en-us/library/cc732524.aspx>를 참조하십시오.

**19.3. 외부 LDAP 공급자 구성****19.3.1. 외부 LDAP 공급자 구성(Interactive Setup)**

**ovirt-engine-extension-aaa-ldap** 확장을 통해 사용자는 외부 디렉터리 설정을 쉽게 사용자 지정할 수 있습니다. **ovirt-engine-extension-aaa-ldap** 확장은 다양한 **LDAP** 서버 유형을 지원하며, 대부분의 **LDAP** 유형에 대한 설정을 지원하기 위해 대화형 설정 스크립트가 제공됩니다.

**LDAP** 서버 유형이 대화형 설정 스크립트에 나열되지 않았거나 사용자 지정을 추가하려는 경우 구성 파일을 수동으로 편집할 수 있습니다. 자세한 내용은 **19.3.3절. “외부 LDAP 공급자 구성(Manual Method)”**를 참조하십시오.

**Active Directory** 예제는 **19.3.2절. “Active Directory 연결”**를 참조하십시오.

## 사전 요구 사항

- **DNS** 또는 **LDAP** 서버의 도메인 이름을 알아야 합니다.



**LDAP** 서버와 **Manager** 간 보안 연결을 설정하려면 **PEM** 인코딩 **CA** 인증서가 준비되었는지 확인합니다.

- **LDAP** 서버에 대한 검색 및 로그인 쿼리를 수행할 준비가 된 계정 이름과 암호 세트가 하나 이상 있어야 합니다.

#### 외부 **LDAP** 공급자 구성

1. **Red Hat Virtualization Manager**에서 **LDAP** 확장 패키지를 설치합니다.

```
# yum install ovirt-engine-extension-aaa-ldap-setup
```

2. **ovirt-engine-extension-aaa-ldap-setup** 을 실행하여 대화형 설정을 시작합니다.

```
# ovirt-engine-extension-aaa-ldap-setup
```

3. 해당 번호를 입력하여 **LDAP** 유형을 선택합니다. **LDAP** 서버가 어떤 스키마인지 확실하지 않은 경우 **LDAP** 서버 유형의 표준 스키마를 선택합니다. **Active Directory**의 경우 **19.3.2절**. “**Active Directory** 연결”의 절차를 따르십시오.

```
Available LDAP implementations:
```

```
1 - 389ds
2 - 389ds RFC-2307 Schema
3 - Active Directory
4 - IBM Security Directory Server
5 - IBM Security Directory Server RFC-2307 Schema
6 - IPA
7 - Novell eDirectory RFC-2307 Schema
8 - OpenLDAP RFC-2307 Schema
9 - OpenLDAP Standard Schema
10 - Oracle Unified Directory RFC-2307 Schema
11 - RFC-2307 Schema (Generic)
12 - RHDS
13 - RHDS RFC-2307 Schema
14 - iPlanet
Please select:
```

4. **Enter** 를 눌러 기본값을 수락하고 **LDAP** 서버 이름에 대한 도메인 이름 확인을 구성합니다.

```
It is highly recommended to use DNS resolution for LDAP server.
If for some reason you intend to use hosts or plain address disable DNS usage.
Use DNS (Yes, No) [Yes]:
```

5.

**DNS 정책 방법을 선택합니다.**

- 옵션 1의 경우 **/etc/resolv.conf** 에 나열된 **DNS** 서버를 사용하여 **IP** 주소를 확인합니다. **/etc/resolv.conf** 파일이 올바른 **DNS** 서버로 업데이트되었는지 확인합니다.
- 옵션 2의 경우 **FQDN**(정규화된 도메인 이름) 또는 **LDAP** 서버의 **IP** 주소를 입력합니다. **dig** 명령을 **SRV** 레코드와 함께 사용하여 도메인 이름을 확인할 수 있습니다. **SRV** 레코드는 다음 형식을 취합니다.

```
_service._protocol.domain_name
```

예: **dig \_ldap.\_tcp.redhat.com SRV.**

- 옵션 3의 경우 공백으로 구분된 **LDAP** 서버 목록을 입력합니다. 서버의 **FQDN** 또는 **IP** 주소를 사용합니다. 이 정책은 **LDAP** 서버 간 로드 밸런싱을 제공합니다. 쿼리는 라운드 로빈 알고리즘에 따라 모든 **LDAP** 서버에 배포됩니다.
- 옵션 4의 경우 공백으로 구분된 **LDAP** 서버 목록을 입력합니다. 서버의 **FQDN** 또는 **IP** 주소를 사용합니다. 이 정책은 쿼리에 응답할 첫 번째 **LDAP** 서버를 기본 **LDAP** 서버로 정의합니다. 첫 번째 서버를 사용할 수 없는 경우 쿼리는 목록의 다음 **LDAP** 서버로 이동합니다.

```
1 - Single server
2 - DNS domain LDAP SRV record
3 - Round-robin between multiple hosts
4 - Failover between multiple hosts
Please select:
```

6.

**LDAP** 서버에서 지원하는 보안 연결 방법을 선택하고 **PEM** 인코딩 **CA** 인증서를 가져올 방법을 지정합니다.

- 파일을 사용하면 인증서의 전체 경로를 제공할 수 있습니다.
- **URL** 을 사용하면 인증서의 **URL** 을 지정할 수 있습니다.
- 인라인 을 사용하면 인증서 콘텐츠를 터미널에 붙여넣을 수 있습니다.

- 시스템을 사용하면 모든 **CA** 파일의 기본 위치를 지정할 수 있습니다.
- 비보안 스루핑은 인증서 검증을 수행하지만, **TLS**를 사용하여 연결은 계속 암호화됩니다.

**NOTE:**

*It is highly recommended to use secure protocol to access the LDAP server.*

*Protocol startTLS is the standard recommended method to do so.*

*Only in cases in which the startTLS is not supported, fallback to non standard ldaps protocol.*

*Use plain for test environments only.*

*Please select protocol to use (startTLS, ldaps, plain) [startTLS]: startTLS*

*Please select method to obtain PEM encoded CA certificate (File, URL, Inline, System, Insecure):*

*Please enter the password:*



참고

**LDAPS**는 보안 소켓 링크를 통한 **Lightweight Directory Access Protocol**의 약자입니다. **SSL** 연결의 경우 **ldaps** 옵션을 선택합니다.

7.

검색 사용자 고유 이름(**DN**)을 입력합니다. 사용자에게 디렉터리 서버의 모든 사용자 및 그룹을 탐색할 수 있는 권한이 있어야 합니다. 검색 사용자는 **LDAP** 주석에 지정해야 합니다. 익명 검색이 허용되는 경우 입력 없이 **Enter** 를 누릅니다.

*Enter search user DN (for example uid=username,dc=example,dc=com or leave empty for anonymous): uid=user1,ou=Users,ou=department-1,dc=example,dc=com*

*Enter search user password:*

8.

기본 **DN**을 입력합니다.

*Please enter base DN (dc=redhat,dc=com) [dc=redhat,dc=com]: ou=department-1,dc=redhat,dc=com*

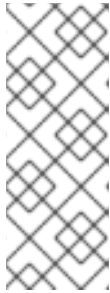
9.

가상 머신에 대해 **Single Sign-On**을 구성하려면 **Yes** 를 선택합니다. 이 기능은 관리 포털에 대한 **SSO(Single Sign-On)** 기능과 함께 사용할 수 없습니다. 이 스크립트는 프로파일 이름이 도메인 이름과 일치해야 함을 알립니다. 가상 머신 관리 가이드의 **가상 머신용 SSO(Single Sign-On)** 구성의 지침을 계속 따라야 합니다.

*Are you going to use Single Sign-On for Virtual Machines (Yes, No) [Yes]:*

10.

프로필 이름을 지정합니다. 프로필 이름은 로그인 페이지의 사용자에게 표시됩니다. 이 예에서는 **redhat.com** 을 사용합니다.

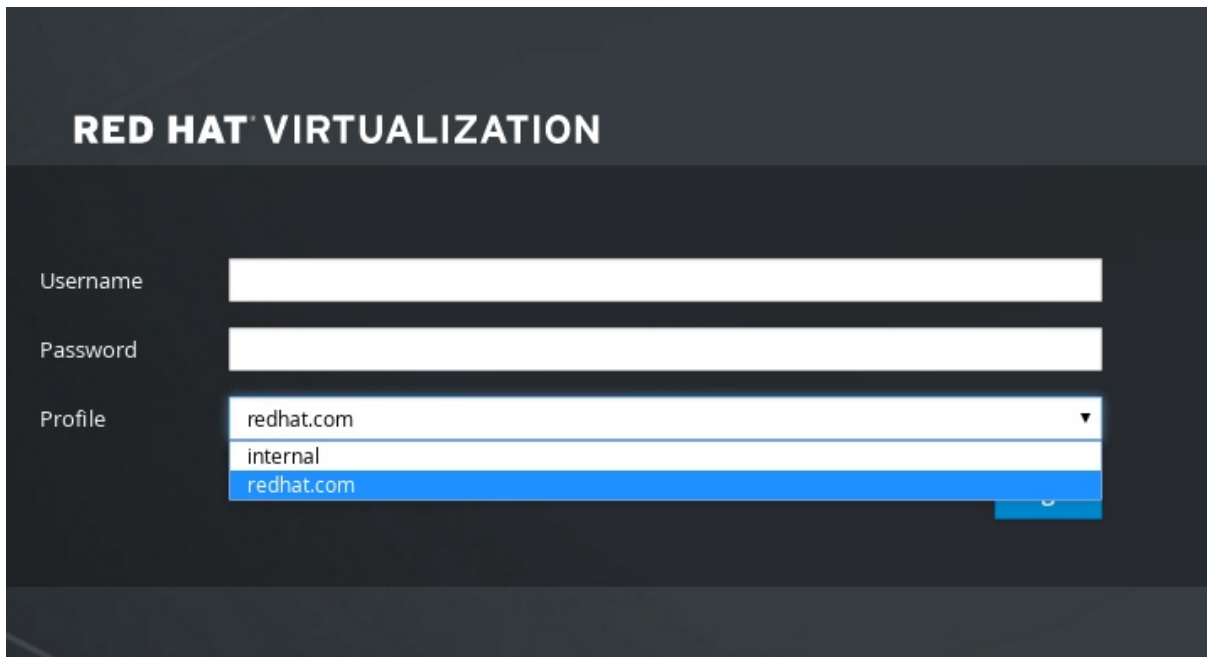


참고

도메인이 구성된 후 프로필 이름을 바꾸려면 **/etc/ovirt-engine/extensions.d/redhat.com-authn.properties** 파일에서 **ovirt.engine.aaa.authn.profile.name** 속성을 편집합니다. 변경 사항을 적용하려면 **ovirt-engine** 서비스를 다시 시작합니다.

Please specify profile name that will be visible to users: redhat.com

그림 19.1. 관리 포털 로그인 페이지



참고

사용자는 처음 로그인할 때 드롭다운 목록에서 프로필을 선택해야 합니다. 이 정보는 브라우저 쿠키에 저장되고 다음에 사용자가 로그인할 때 미리 선택됩니다.

11. 로그인 기능을 테스트하여 **LDAP** 서버가 **Red Hat Virtualization** 환경에 올바르게 연결되어 있는지 확인합니다. 로그인 쿼리의 경우 사용자 이름과 암호를 입력합니다.

**NOTE:**  
It is highly recommended to test drive the configuration before applying it into engine. Login sequence is executed automatically, but it is recommended to also execute Search sequence manually after successful Login sequence.

```

Please provide credentials to test login flow:
Enter user name:
Enter user password:
[ INFO ] Executing login sequence...
...
[ INFO ] Login sequence executed successfully

```

12.

사용자 세부 정보가 올바른지 확인합니다. 사용자 정보가 올바르지 않으면 **Abort** 를 선택합니다.

```

Please make sure that user details are correct and group membership meets expectations
(search for PrincipalRecord and GroupRecord titles).
Abort if output is incorrect.
Select test sequence to execute (Done, Abort, Login, Search) [Abort]:

```

13.

검색 기능을 수동으로 테스트하는 것이 좋습니다. 검색 쿼리의 경우 사용자 계정 또는 그룹 계정의 보안 주체를 선택합니다. 사용자 계정의 그룹 계정 정보를 반환하려면 예 를 선택하여 그룹을 해결하십시오. 세 개의 구성 파일이 생성되고 화면 출력에 표시됩니다.

```

Select test sequence to execute (Done, Abort, Login, Search) [Search]: Search
Select entity to search (Principal, Group) [Principal]:
Term to search, trailing '*' is allowed: testuser1
Resolve Groups (Yes, No) [No]:

```

14.

완료 를 선택하여 설정을 완료합니다.

```

Select test sequence to execute (Done, Abort, Login, Search) [Abort]: Done
[ INFO ] Stage: Transaction setup
[ INFO ] Stage: Misc configuration
[ INFO ] Stage: Package installation
[ INFO ] Stage: Misc configuration
[ INFO ] Stage: Transaction commit
[ INFO ] Stage: Closing up
CONFIGURATION SUMMARY
Profile name is: redhat.com
The following files were created:
  /etc/ovirt-engine/aaa/redhat.com.properties
  /etc/ovirt-engine/extensions.d/redhat.com.properties
  /etc/ovirt-engine/extensions.d/redhat.com-authn.properties
[ INFO ] Stage: Clean up
Log file is available at /tmp/ovirt-engine-extension-aaa-ldap-setup-20171004101225-
mmneib.log:
[ INFO ] Stage: Pre-termination
[ INFO ] Stage: Termination

```

15.

**ovirt-engine** 서비스를 다시 시작합니다. 생성된 프로파일은 이제 관리 포털 및 **VM** 포털 로그인 페이지에서 사용할 수 있습니다. 예를 들어 **VM** 포털에 로그인하기 위한 **LDAP** 서버 적절한 역

할 및 권한에 사용자 계정을 할당하려면 **19.7절. “관리 포털에서 사용자 작업”**을 참조하십시오.

```
# systemctl restart ovirt-engine.service
```



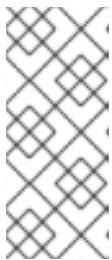
참고

자세한 내용은 `/usr/share/doc/ovirt-engine-extension-aaa-ldap-버전의 LDAP 인증 및 권한 부여 README` 파일을 참조하십시오.

### 19.3.2. Active Directory 연결

사전 요구 사항

- **Active Directory** 마운트 이름을 알아야 합니다. 가상 도메인 이름도 루트 도메인 이름이라고도 합니다.



참고

`ovirt-engine-extension-aaa-ldap-setup` 툴을 사용하여 구성할 수 없는 가장 일반적인 **Active Directory** 구성의 예는 `/usr/share/ovirt-engine-extension-aaa-ldap/README.md`에 제공되어 있습니다.

- 관리자의 `/etc/resolv.conf` 파일에 **Active Directory** 오케스트라 이름을 확인할 수 있는 **DNS** 서버를 추가하거나 **Active Directory DNS** 서버를 적어 둔 후 대화형 설정 스크립트로 메시지가 표시되면 이를 입력해야 합니다.
- **LDAP** 서버와 **Manager** 간 보안 연결을 설정하려면 **PEM** 인코딩 **CA** 인증서가 준비되었는지 확인합니다. 자세한 내용은 **D.2절. “관리자와 LDAP 서버 간 암호화 통신 설정”**를 참조하십시오.
- 익명 검색이 지원되지 않는 경우, 모든 사용자 및 그룹을 검색할 수 있는 권한이 있는 사용자는 검색 사용자로 사용하려면 **Active Directory**에서 사용할 수 있어야 합니다. 검색 사용자의 고유 이름(DN)을 기록해 둡니다. **Active Directory**에 관리자 사용자를 사용하지 마십시오.
- **Active Directory**에 대한 검색 및 로그인 쿼리를 수행할 준비가 된 계정 이름과 암호가 하나 이상 있어야 합니다.
-

**Active Directory** 배포가 여러 도메인에 걸쳐 있는 경우 `/usr/share/ovirt-engine-extension-aaa-ldap/profiles/ad.properties` 파일에 설명된 제한을 알고 있어야 합니다.

#### 외부 **LDAP** 공급자 구성

1. **Red Hat Virtualization Manager**에서 **LDAP** 확장 패키지를 설치합니다.

```
# yum install ovirt-engine-extension-aaa-ldap-setup
```

2. **ovirt-engine-extension-aaa-ldap-setup** 을 실행하여 대화형 설정을 시작합니다.

```
# ovirt-engine-extension-aaa-ldap-setup
```

3. 해당 번호를 입력하여 **LDAP** 유형을 선택합니다. 이 단계 이후 **LDAP** 관련 질문은 다른 **LDAP** 유형에 따라 다릅니다.

```
Available LDAP implementations:
```

```
1 - 389ds
2 - 389ds RFC-2307 Schema
3 - Active Directory
4 - IBM Security Directory Server
5 - IBM Security Directory Server RFC-2307 Schema
6 - IPA
7 - Novell eDirectory RFC-2307 Schema
8 - OpenLDAP RFC-2307 Schema
9 - OpenLDAP Standard Schema
10 - Oracle Unified Directory RFC-2307 Schema
11 - RFC-2307 Schema (Generic)
12 - RHDS
13 - RHDS RFC-2307 Schema
14 - iPlanet
Please select: 3
```

4. **Active Directory** 도메인 이름을 입력합니다. **Manager**의 **DNS**에서 마운트 이름을 확인할 수 없는 경우 스크립트에서 공백으로 구분된 **Active Directory DNS** 서버 이름 목록을 입력하라는 메시지가 표시됩니다.

```
Please enter Active Directory Forest name: ad-example.redhat.com
[ INFO ] Resolving Global Catalog SRV record for ad-example.redhat.com
[ INFO ] Resolving LDAP SRV record for ad-example.redhat.com
```

5. **LDAP** 서버에서 지원하는 보안 연결 방법을 선택하고 **PEM** 인코딩 **CA** 인증서를 가져올 방법을 지정합니다. **file** 옵션을 사용하면 인증서의 전체 경로를 제공할 수 있습니다. **URL** 옵션을 사

용하면 인증서에 대한 **URL**을 지정할 수 있습니다. 인라인 옵션을 사용하여 인증서의 콘텐츠를 터미널에 붙여넣습니다. 시스템 옵션을 사용하면 모든 **CA** 파일의 위치를 지정할 수 있습니다. 비보안 옵션을 사용하면 비보안 모드에서 **startTLS**를 사용할 수 있습니다.

**NOTE:**  
 It is highly recommended to use secure protocol to access the LDAP server.  
 Protocol startTLS is the standard recommended method to do so.  
 Only in cases in which the startTLS is not supported, fallback to non standard ldaps protocol.  
 Use plain for test environments only.  
 Please select protocol to use (startTLS, ldaps, plain) [startTLS]: startTLS  
 Please select method to obtain PEM encoded CA certificate (File, URL, Inline, System, Insecure): File  
 Please enter the password:



참고

**LDAPS**는 보안 소켓 링크를 통한 **Lightweight Directory Access Protocol**의 약자입니다. **SSL** 연결의 경우 **ldaps** 옵션을 선택합니다.

**PEM** 인코딩 **CA** 인증서 생성에 대한 자세한 내용은 **D.2절. “관리자와 LDAP 서버 간 암호화 통신 설정”**을 참조하십시오.

6. 검색 사용자 고유 이름(**DN**)을 입력합니다. 사용자에게 디렉터리 서버의 모든 사용자 및 그룹을 탐색할 수 있는 권한이 있어야 합니다. 검색 사용자는 **LDAP** 주석이어야 합니다. 익명 검색이 허용되는 경우 입력 없이 **Enter**를 누릅니다.

Enter search user DN (empty for anonymous):  
 cn=user1,ou=Users,dc=test,dc=redhat,dc=com  
 Enter search user password:

7. 가상 머신에 **Single Sign-On** 사용 여부를 지정합니다. 이 기능은 기본적으로 활성화되어 있지만 관리 포털에 대한 **SSO(Single Sign-On)**가 활성화된 경우에는 사용할 수 없습니다. 이 스크립트는 프로필 이름이 도메인 이름과 일치해야 함을 알립니다. 가상 머신 관리 가이드의 **가상 머신용 SSO(Single Sign-On)** 구성의 지침을 계속 따라야 합니다.

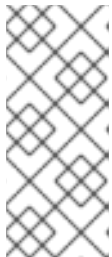
Are you going to use Single Sign-On for Virtual Machines (Yes, No) [Yes]:

8. 프로필 이름을 지정합니다. 프로필 이름은 로그인 페이지의 사용자에게 표시됩니다. 이 예에서는 **redhat.com**을 사용합니다.

Please specify profile name that will be visible to users:redhat.com



그림 19.2. 관리 포털 로그인 페이지



## 참고

사용자는 처음으로 로그인할 때 드롭다운 목록에서 원하는 프로필을 선택해야 합니다. 그러면 이 정보는 브라우저 쿠키에 저장되고 다음에 사용자가 로그인할 때 미리 선택됩니다.

9.

검색 및 로그인 기능을 테스트하여 **LDAP** 서버가 **Red Hat Virtualization** 환경에 올바르게 연결되어 있는지 확인합니다. 로그인 쿼리의 경우 계정 이름과 암호를 입력합니다. 검색 쿼리의 경우 사용자 계정의 주체를 선택하고 그룹 계정 그룹을 선택합니다. 사용자 계정에 대한 그룹 계정 정보를 반환하려면 **Yes** 를 **Resolve Groups** 에 입력합니다. 완료 를 선택하여 설정을 완료합니다. 세 개의 구성 파일이 생성되고 화면 출력에 표시됩니다.

**NOTE:**

*It is highly recommended to test drive the configuration before applying it into engine. Login sequence is executed automatically, but it is recommended to also execute Search sequence manually after successful Login sequence.*

*Select test sequence to execute (Done, Abort, Login, Search) [Abort]: Login*

*Enter search user name: testuser1*

*Enter search user password:*

*[ INFO ] Executing login sequence...*

*...*

*Select test sequence to execute (Done, Abort, Login, Search) [Abort]: Search*

*Select entity to search (Principal, Group) [Principal]:*

*Term to search, trailing '\*' is allowed: testuser1*

*Resolve Groups (Yes, No) [No]:*

*[ INFO ] Executing login sequence...*

*...*

*Select test sequence to execute (Done, Abort, Login, Search) [Abort]: Done*

*[ INFO ] Stage: Transaction setup*

*[ INFO ] Stage: Misc configuration*

```
[ INFO ] Stage: Package installation
[ INFO ] Stage: Misc configuration
[ INFO ] Stage: Transaction commit
[ INFO ] Stage: Closing up
CONFIGURATION SUMMARY
Profile name is: redhat.com
The following files were created:
    /etc/ovirt-engine/aaa/redhat.com.properties
    /etc/ovirt-engine/extensions.d/redhat.com-authz.properties
    /etc/ovirt-engine/extensions.d/redhat.com-authn.properties
[ INFO ] Stage: Clean up
    Log file is available at /tmp/ovirt-engine-extension-aaa-ldap-setup-20160114064955-
    1yar9i.log:
[ INFO ] Stage: Pre-termination
[ INFO ] Stage: Termination
```

10.

생성된 프로파일은 이제 관리 포털 및 VM 포털 로그인 페이지에서 사용할 수 있습니다. 예를 들어 VM 포털에 로그인하기 위한 LDAP 서버 적절한 역할 및 권한에 사용자 계정을 할당하려면 **19.7절. “관리 포털에서 사용자 작업”** 을 참조하십시오.



참고

자세한 내용은 `/usr/share/doc/ovirt-engine-extension-aaa-ldap-` 버전의 **LDAP** 인증 및 권한 부여 **README** 파일을 참조하십시오.

**19.3.3. 외부 LDAP 공급자 구성(Manual Method)**

**ovirt-engine-extension-aaa-ldap** 확장에서는 **LDAP** 프로토콜을 사용하여 디렉터리 서버에 액세스 하며 완전히 사용자 지정할 수 있습니다. VM 포털 또는 관리 포털에 대한 **SSO(Single Sign-On)**를 활성화 하려면 **Kerberos** 인증이 필요하지 않습니다.

이전 섹션의 대화형 설정 방법에 사용 사례가 적용되지 않는 경우 구성 파일을 수동으로 수정하여 **LDAP** 서버를 연결할 수 있습니다. 다음 절차에서는 일반 세부 정보를 사용합니다. 특정 값은 설정에 따라 다릅니다.

수동으로 외부 **LDAP** 공급자 구성

1.

**Red Hat Virtualization Manager**에서 **LDAP** 확장 패키지를 설치합니다.

```
# yum install ovirt-engine-extension-aaa-ldap
```

2.

**LDAP** 구성 템플릿 파일을 `/etc/ovirt-engine` 디렉터리에 복사합니다. 템플릿 파일은 활성 디

렉터리(**ad**) 및 기타 디렉터리 유형(단순)에 사용할 수 있습니다. 이 예에서는 간단한 구성 템플릿을 사용합니다.

```
# cp -r /usr/share/ovirt-engine-extension-aaa-ldap/examples/simple/. /etc/ovirt-engine
```

3.

관리 포털 및 **VM** 포털 로그인 페이지의 사용자에게 표시되는 프로필 이름과 일치하도록 구성 파일의 이름을 변경합니다.

```
# mv /etc/ovirt-engine/aaa/profile1.properties /etc/ovirt-engine/aaa/example.properties
# mv /etc/ovirt-engine/extensions.d/profile1-authn.properties /etc/ovirt-engine/extensions.d/example-authn.properties
# mv /etc/ovirt-engine/extensions.d/profile1-authz.properties /etc/ovirt-engine/extensions.d/example-authz.properties
```

4.

**LDAP** 서버 유형의 주석을 제거하고 도메인 및 암호 필드를 업데이트하여 **LDAP** 속성 구성 파일을 편집합니다.

```
# vi /etc/ovirt-engine/aaa/example.properties
```

예 19.1. 프로필: **LDAP** 서버 섹션 예

```
# Select one
#
include = <openldap.properties>
#include = <389ds.properties>
#include = <rhds.properties>
#include = <ipa.properties>
#include = <iplanet.properties>
#include = <rfc2307-389ds.properties>
#include = <rfc2307-rhds.properties>
#include = <rfc2307-openldap.properties>
#include = <rfc2307-edir.properties>
#include = <rfc2307-generic.properties>

# Server
#
vars.server = ldap1.company.com

# Search user and its password.
#
vars.user = uid=search,cn=users,cn=accounts,dc=company,dc=com
vars.password = 123456

pool.default.serverset.single.server = ${global:vars.server}
pool.default.auth.simple.bindDN = ${global:vars.user}
pool.default.auth.simple.password = ${global:vars.password}
```

**TLS** 또는 **SSL** 프로토콜을 사용하여 **LDAP** 서버와 상호 작용하려면 **LDAP** 서버의 루트 **CA** 인증서를 가져오고 이를 사용하여 공개 키 저장소 파일을 생성합니다. 다음 줄의 주석을 제거하고 공개 키 저장소 파일의 전체 경로와 파일에 액세스할 암호를 지정합니다.



참고

공개 키 저장소 파일 생성에 대한 자세한 내용은 **D.2절. “관리자와 LDAP 서버 간 암호화 통신 설정”** 을 참조하십시오.

예 19.2. profile: keystore 섹션의 예

```
# Create keystore, import certificate chain and uncomment
# if using tls.
pool.default.ssl.startTLS = true
pool.default.ssl.truststore.file = /full/path/to/myrootca.jks
pool.default.ssl.truststore.password = password
```

5.

인증 구성 파일을 검토합니다. 관리 포털의 사용자에게 표시되는 프로필 이름과 **VM** 포털 로그인 페이지는 **ovirt.engine.aaa.authn.profile.name** 에 의해 정의됩니다. 구성 프로필 위치는 **LDAP** 구성 파일 위치와 일치해야 합니다. 모든 필드는 기본값으로 남겨 둘 수 있습니다.

```
# vi /etc/ovirt-engine/extensions.d/example-authn.properties
```

예 19.3. 인증 구성 파일 예

```
ovirt.engine.extension.name = example-authn
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-
extensions.aaa.ldap
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.ldap.AuthnExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Authn
ovirt.engine.aaa.authn.profile.name = example
ovirt.engine.aaa.authn.authz.plugin = example-authz
config.profile.file.1 = ../aaa/example.properties
```

6.

권한 부여 구성 파일을 검토합니다. 구성 프로필 위치는 **LDAP** 구성 파일 위치와 일치해야 합니다. 모든 필드는 기본값으로 남겨 둘 수 있습니다.

```
# vi /etc/ovirt-engine/extensions.d/example-authz.properties
```

예 19.4. 권한 부여 구성 파일의 예

```
ovirt.engine.extension.name = example-authz
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-
extensions.aaa.ldap
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.ldap.AuthzExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Authz
config.profile.file.1 = ../aaa/example.properties
```

7.

구성 프로파일의 소유권 및 권한이 적절한지 확인합니다.

```
# chown ovirt:ovirt /etc/ovirt-engine/aaa/example.properties
# chmod 600 /etc/ovirt-engine/aaa/example.properties
```

8.

**engine** 서비스를 다시 시작하십시오.

```
# systemctl restart ovirt-engine.service
```

9.

생성된 예제 프로파일은 이제 관리 포털 및 **VM** 포털 로그인 페이지에서 사용할 수 있습니다. 예를 들어 **VM** 포털에 로그인하기 위한 **LDAP** 서버의 적절한 권한을 사용자 계정에 지정하려면 [19.7절. “관리 포털에서 사용자 작업”](#)을 참조하십시오.



참고

자세한 내용은 `/usr/share/doc/ovirt-engine-extension-aaa-ldap-버전의 LDAP 인증 및 권한 부여 README` 파일을 참조하십시오.

#### 19.3.4. 외부 LDAP 공급자 제거

다음 절차에서는 외부 구성된 **LDAP** 공급자 및 해당 사용자를 제거하는 방법을 설명합니다.

외부 **LDAP** 공급자 제거

1.

**LDAP** 공급자 구성 파일을 제거하고 기본 이름 **profile1** 을 대체합니다.

```
# rm /etc/ovirt-engine/extensions.d/profile1-authn.properties
# rm /etc/ovirt-engine/extensions.d/profile1-authz.properties
# rm /etc/ovirt-engine/aaa/profile1.properties
```

2. **ovirt-engine** 서비스를 다시 시작합니다.

```
# systemctl restart ovirt-engine
```

3. 관리 포털의 **Users** 리소스 탭에서 이 공급자의 사용자(**Authority whose Authorization provider is profile1-authz**)를 선택하고 **Remove** 를 클릭합니다.

#### 19.4. SSO(SINGLE SIGN-ON)의 LDAP 및 KERBEROS 구성

**Single Sign-On**을 사용하면 암호를 다시 사용하지 않고도 **VM** 포털 또는 관리 포털에 로그인할 수 있습니다. 인증 자격 증명은 **Kerberos** 서버에서 가져옵니다. 관리 포털과 **VM** 포털에 단일 사인온을 구성하려면 **ovirt-engine-extension-aaa-misc** 및 **ovirt-engine-extension-aaa-ldap** 라는 두 가지 확장 기능을 구성하고 **Apache** 모듈 **mod\_auth\_gssapi** 및 **mod\_session** 을 구성해야 합니다. **Kerberos**를 포함하지 않는 **SSO(Single Sign-On)**를 구성할 수 있지만 이 문서의 범위를 벗어납니다.



참고

**VM** 포털에 대한 **SSO(Single Sign-On)**가 활성화된 경우 가상 머신에 대한 **SSO(Single Sign-On)**를 사용할 수 없습니다. **VM** 포털에 **SSO(Single Sign-On)**가 활성화된 상태에서 **VM** 포털에서 암호를 수락할 필요가 없으므로 가상 머신에 로그인할 암호를 위임할 수 없습니다.

이 예에서는 다음을 가정합니다.

- 기존 **KDC(Key Distribution Center)** 서버는 **MIT** 버전의 **Kerberos 5**를 사용합니다.
- **KDC** 서버에 대한 관리 권한이 있습니다.
- **Kerberos** 클라이언트가 **Red Hat Virtualization Manager** 및 사용자 시스템에 설치되어 있습니다.
- **kadmin** 유틸리티는 **Kerberos** 서비스 주체 및 키 **tab** 파일을 만드는 데 사용됩니다.

다음 구성 요소가 포함됩니다.

**KDC 서버에서**

- **Red Hat Virtualization Manager**에서 **Apache** 서비스에 대한 서비스 주체와 키탭 파일을 생성합니다.

**Red Hat Virtualization Manager**

- 인증 및 권한 부여 확장 패키지 및 **Apache Kerberos** 인증 모듈을 설치합니다.
- 확장 파일을 구성합니다.

**Apache** 서비스에 대한 **Kerberos** 구성

1. **KDC** 서버에서 **kadmin** 유틸리티를 사용하여 **Red Hat Virtualization Manager**에서 **Apache** 서비스에 대한 서비스 주체를 생성합니다. 서비스 주체는 **Apache** 서비스의 **KDC**에 대한 참조 ID입니다.

```
# kadmin
kadmin> addprinc -randkey HTTP/fqdn-of-rhev@REALM.COM
```

2. **Apache** 서비스의 **keytab** 파일을 생성합니다. **keytab** 파일은 공유 보안 키를 저장합니다.



참고

**engine-backup** 명령에는 백업 및 복원 시 **/etc/httpd/http.keytab** 파일이 포함됩니다. **keytab** 파일에 다른 이름을 사용하는 경우 백업하고 복원해야 합니다.

```
kadmin> ktadd -k /tmp/http.keytab HTTP/fqdn-of-rhev@REALM.COM
kadmin> quit
```

3. **KDC** 서버에서 **Red Hat Virtualization Manager**로 **keytab** 파일을 복사합니다.

```
# scp /tmp/http.keytab root@rhev.example.com:/etc/httpd
```

## VM 포털 또는 관리 포털에 **Single Sign-On** 구성

1. **Red Hat Virtualization Manager**에서 키 탭의 소유권 및 권한이 적절한지 확인합니다.

```
# chown apache /etc/httpd/http.keytab
# chmod 400 /etc/httpd/http.keytab
```

2. 인증 확장 패키지, **LDAP** 확장 패키지 및 **mod\_auth\_gssapi** 및 **mod\_session Apache** 모듈을 설치합니다.

```
# yum install ovirt-engine-extension-aaa-misc ovirt-engine-extension-aaa-ldap
mod_auth_gssapi mod_session
```

3. **SSO** 구성 템플릿 파일을 **/etc/ovirt-engine** 디렉터리에 복사합니다. 템플릿 파일은 **Active Directory(ad-ss)** 및 기타 디렉터리 유형(**simple-ss**)에 사용할 수 있습니다. 이 예에서는 간단한 **SSO** 구성 템플릿을 사용합니다.

```
# cp -r /usr/share/ovirt-engine-extension-aaa-ldap/examples/simple-ss/. /etc/ovirt-engine
```

4. **ovirt-ss.conf** 를 **Apache** 구성 디렉터리로 이동합니다.



참고

**engine-backup** 명령에는 백업 및 복원할 때 **/etc/httpd/conf.d/ovirt-ss.conf** 파일이 포함되어 있습니다. 이 파일에 다른 이름을 사용하는 경우 백업하고 복원하십시오.

```
# mv /etc/ovirt-engine/aaa/ovirt-ss.conf /etc/httpd/conf.d
```

5. 인증 방법 파일을 검토합니다. 키가 **keytab** 파일에서 자동으로 가져오기 때문에 이 파일을 편집할 필요가 없습니다.

```
# vi /etc/httpd/conf.d/ovirt-ss.conf
```

예 **19.5.** 인증 방법 파일 예

```
<LocationMatch ^/ovirt-engine/ss/(interactive-login-negotiate|oauth/token-http-
auth)|^/ovirt-engine/api>
<If "req('Authorization') !~ /^(Bearer|Basic)/">
```



```

RewriteEngine on
RewriteCond %{LA-U:REMOTE_USER} ^(.*)$
RewriteRule ^(.*)$ - [L,NS,P,E=REMOTE_USER:%1]
RequestHeader set X-Remote-User %{REMOTE_USER}s

AuthType GSSAPI
AuthName "Kerberos Login"

# Modify to match installation
GssapiCredStore keytab:/etc/httpd/http.keytab
GssapiUseSessions On
Session On
SessionCookieName ovirt_gssapi_session path=/private;httponly;secure;

Require valid-user
ErrorDocument 401 "<html><meta http-equiv=\"refresh\" content=\"0; url=/ovirt-engine/sso/login-unauthorized\"/><body><a href=\"/ovirt-engine/sso/login-unauthorized\">Here</a></body></html>"
</If>
</LocationMatch>

```

6.

관리 포털 및 **VM** 포털 로그인 페이지의 사용자에게 표시되는 프로필 이름과 일치하도록 구성 파일의 이름을 변경합니다.

```
# mv /etc/ovirt-engine/aaa/profile1.properties /etc/ovirt-engine/aaa/example.properties
```

```
# mv /etc/ovirt-engine/extensions.d/profile1-http-authn.properties /etc/ovirt-engine/extensions.d/example-http-authn.properties
```

```
# mv /etc/ovirt-engine/extensions.d/profile1-http-mapping.properties /etc/ovirt-engine/extensions.d/example-http-mapping.properties
```

```
# mv /etc/ovirt-engine/extensions.d/profile1-authz.properties /etc/ovirt-engine/extensions.d/example-authz.properties
```

7.

**LDAP** 서버 유형의 주석을 제거하고 도메인 및 암호 필드를 업데이트하여 **LDAP** 속성 구성 파일을 편집합니다.

```
# vi /etc/ovirt-engine/aaa/example.properties
```

예 19.6. 프로필: **LDAP** 서버 섹션 예

```

# Select one
include = <openldap.properties>
#include = <389ds.properties>
#include = <rhds.properties>
#include = <ipa.properties>
#include = <iplanet.properties>

```

```

#include = <rfc2307-389ds.properties>
#include = <rfc2307-rhds.properties>
#include = <rfc2307-openldap.properties>
#include = <rfc2307-edir.properties>
#include = <rfc2307-generic.properties>

# Server
#
vars.server = ldap1.company.com

# Search user and its password.
#
vars.user = uid=search,cn=users,cn=accounts,dc=company,dc=com
vars.password = 123456

pool.default.serverset.single.server = ${global:vars.server}
pool.default.auth.simple.bindDN = ${global:vars.user}
pool.default.auth.simple.password = ${global:vars.password}

```

**TLS** 또는 **SSL** 프로토콜을 사용하여 **LDAP** 서버와 상호 작용하려면 **LDAP** 서버의 루트 **CA** 인증서를 가져오고 이를 사용하여 공개 키 저장소 파일을 생성합니다. 다음 줄의 주석을 제거하고 공개 키 저장소 파일의 전체 경로와 파일에 액세스할 암호를 지정합니다.



#### 참고

공개 키 저장소 파일 생성에 대한 자세한 내용은 [D.2절. “관리자와 LDAP 서버 간 암호화 통신 설정”](#) 을 참조하십시오.

#### 예 19.7. profile: keystore 섹션의 예

```

# Create keystore, import certificate chain and uncomment
# if using ssl/tls.
pool.default.ssl.startTLS = true
pool.default.ssl.truststore.file = /full/path/to/myrootca.jks
pool.default.ssl.truststore.password = password

```

8.

인증 구성 파일을 검토합니다. 관리 포털의 사용자에게 표시되는 프로필 이름과 **VM** 포털 로그인 페이지는 **ovirt.engine.aaa.authn.profile.name** 에 의해 정의됩니다. 구성 프로필 위치는 **LDAP** 구성 파일 위치와 일치해야 합니다. 모든 필드는 기본값으로 남겨 둘 수 있습니다.

```
# vi /etc/ovirt-engine/extensions.d/example-http-authn.properties
```

#### 예 19.8. 인증 구성 파일 예

```
ovirt.engine.extension.name = example-http-authn
```

```
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-
extensions.aaa.misc
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.misc.http.AuthnExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Authn
ovirt.engine.aaa.authn.profile.name = example-http
ovirt.engine.aaa.authn.authz.plugin = example-authz
ovirt.engine.aaa.authn.mapping.plugin = example-http-mapping
config.artifact.name = HEADER
config.artifact.arg = X-Remote-User
```

9.

권한 부여 구성 파일을 검토합니다. 구성 프로파일 위치는 **LDAP** 구성 파일 위치와 일치해야 합니다. 모든 필드는 기본값으로 남겨 둘 수 있습니다.

```
# vi /etc/ovirt-engine/extensions.d/example-authz.properties
```

예 19.9. 권한 부여 구성 파일의 예

```
ovirt.engine.extension.name = example-authz
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-
extensions.aaa ldap
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.ldap.AuthzExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Authz
config.profile.file.1 = ../aaa/example.properties
```

10.

인증 매핑 구성 파일을 검토합니다. 구성 프로파일 위치는 **LDAP** 구성 파일 위치와 일치해야 합니다. 구성 프로파일 확장 이름은 인증 구성 파일의 **ovirt.engine.aaa.authn.mapping.plugin** 값과 일치해야 합니다. 모든 필드는 기본값으로 남겨 둘 수 있습니다.

```
# vi /etc/ovirt-engine/extensions.d/example-http-mapping.properties
```

예 19.10. 인증 매핑 구성 파일의 예

```
ovirt.engine.extension.name = example-http-mapping
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-
extensions.aaa.misc
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.misc.mapping.MappingExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Mapping
config.mapAuthRecord.type = regex
config.mapAuthRecord.regex.mustMatch = true
```

```
config.mapAuthRecord.regex.pattern = ^(?<user>.*?)(\\|\\|(?<at>@)(?<suffix>.*?)@.*)((?<realm>@.*))$
config.mapAuthRecord.regex.replacement = ${user}${at}${suffix}
```

11.

구성 파일의 소유권 및 권한이 적절한지 확인합니다.

```
# chown ovirt:ovirt /etc/ovirt-engine/aaa/example.properties
```

```
# chown ovirt:ovirt /etc/ovirt-engine/extensions.d/example-http-authn.properties
```

```
# chown ovirt:ovirt /etc/ovirt-engine/extensions.d/example-http-mapping.properties
```

```
# chown ovirt:ovirt /etc/ovirt-engine/extensions.d/example-authz.properties
```

```
# chmod 600 /etc/ovirt-engine/aaa/example.properties
```

```
# chmod 640 /etc/ovirt-engine/extensions.d/example-http-authn.properties
```

```
# chmod 640 /etc/ovirt-engine/extensions.d/example-http-mapping.properties
```

```
# chmod 640 /etc/ovirt-engine/extensions.d/example-authz.properties
```

12.

**Apache** 서비스 및 **ovirt-engine** 서비스를 다시 시작합니다.

```
# systemctl restart httpd.service
# systemctl restart ovirt-engine.service
```

## 19.5. RED HAT SINGLE SIGN-ON 설치 및 구성

**Red Hat Single Sign-On**을 인증 방법으로 사용하려면 다음이 필요합니다.

- **Install Red Hat SSO.**
- **LDAP** 그룹 매핑을 구성합니다.
- **Manager**에서 **Apache**를 구성합니다.

- **OVN** 공급자 인증 정보를 구성합니다.



참고

**Red Hat SSO**가 구성된 경우 한 번에 하나의 권한 부여 프로토콜만 사용할 수 있으므로 **Red Hat SSO**의 이전 **LDAP** 기호가 작동하지 않습니다.

### 19.5.1. Red Hat Single Sign-On 설치

**ZIP** 파일을 다운로드하고 압축을 풀거나 **RPM** 파일을 사용하여 **Red Hat Single Sign-On**을 설치할 수 있습니다.

**Red Hat SSO**설치의 설치 지침을 따르십시오.

다음 정보를 준비합니다.

- **Open ID Connect** 서버의 경로/ 위치.
- 올바른 리포지토리의 서브스크립션 채널입니다.
- 유효한 **Red Hat** 서브스크립션 로그인 자격 증명.

### 19.5.2. LDAP 그룹 매핑 구성

1. 다음 정보를 사용하여 **LDAP** 그룹 매핑을 추가합니다.

- 이름: **ldapgroups**
- 매핑 유형: **group-ldap-mapper**
- **LDAP** 그룹 DN: **ou=groups,dc=example,dc=com**

- **Groupof uniquenesses (LDAP 서버 설정에 따라 이 클래스를 조정)**
  - **LDAP 속성 멤버십: 고유member (LDAP 서버 설정에 따라 이 클래스를 조정)**
2. **저장을 클릭합니다.**
  3. **Sync LDAP Groups to KeyCloak 을 클릭합니다.**
  4. **사용자 페더링 공급자 페이지 하단에서 모든 사용자 동기화 를 클릭합니다.**
  5. **클라이언트 탭 의 클라이언트 탭에서 클라이언트 ID 로 **ovirt-engine** 을 추가하고 루트 URL 로 엔진 URL을 입력합니다.**
  6. **클라이언트 프로토콜 을 **openid-connect** 로 수정하고 액세스 유형을 기밀 로 수정합니다.**
  7. **클라이언트 탭의 **Ovirt-engine >** 고급 설정에서 액세스 토큰 수명 주기를 늘립니다.**
  8. **<https://rhvm.example.com:443/>\* 을 유효한 리디렉션 URI로 추가합니다.**
  9. **클라이언트 시크릿이 생성되며 자격 증명 탭에서 볼 수 있습니다.**
  10. **다음과 같은 설정으로 매핑 프로토콜 생성 아래의 클라이언트 탭에서 매핑을 생성합니다.**
- **이름: **groups****
  - **매핑 유형: 그룹 멤버십**
  - **토큰 클레임 이름: **groups****

- 전체 그룹 경로:설정
  - **ID 토큰에 추가:ON**
  - 액세스 토큰에 추가:ON
  - **userinfo :ON**에 추가
11. 사용자 이름에 내장 프로토콜 매핑을 추가합니다.
  12. **ovirt-engine,ovirt-app-api** 및 **ovirt-app-admin** 에 필요한 범위를 생성합니다.
  13. 이전 단계에서 만든 범위를 사용하여 **ovirt-engine** 클라이언트에 대한 선택적 클라이언트 범위를 설정합니다.

### 19.5.3. 관리자에서 Apache 구성

1. **Manager**에서 **Apache**를 구성합니다.

```
# yum install mod_auth_openidc
```

2. 다음 콘텐츠를 사용하여 **/etc/httpd/conf.d** 에 새 **httpd** 구성 파일 **ovirt-openidc.conf** 를 생성합니다.

```
LoadModule auth_openidc_module modules/mod_auth_openidc.so

OIDCProviderMetadataURL https://SSO.example.com/auth/realms/master/.well-known/openid-configuration
OIDCSSLValidateServer Off

OIDCClientID ovirt-engine
OIDCClientSecret <client_SSO_generated_key>
OIDCRedirectURI https://rhvm.example.com/ovirt-engine/callback
OIDCDefaultURL https://rhvm.example.com/ovirt-engine/login?scope=ovirt-app-admin+ovirt-app-portal+ovirt-ext%3Dauth%3Asequence-priority%3D%7E

# maps the preferred_username claim to the REMOTE_USER environment variable:
```

```
OIDCRemoteUserClaim <preferred_username>
OIDCCryptoPassphrase <random1234>
```

```
<LocationMatch ^/ovirt-engine/sso/(interactive-login-negotiate|oauth/token-http-auth)|^/ovirt-
engine/callback>
  <If "req('Authorization') !~ /^(Bearer|Basic)/i">
```

```
    Require valid-user
    AuthType openid-connect
```

```
    ErrorDocument 401 "<html><meta http-equiv=\"refresh\" content=\"0; url=/ovirt-
engine/sso/login-unauthorized\"/><body><a href=\"/ovirt-engine/sso/login-
unauthorized\">Here</a></body></html>"
  </If>
</LocationMatch>
```

```
OIDCOAuthIntrospectionEndpoint
https://SSO.example.com/auth/realms/master/protocol/openid-connect/token/introspect
OIDCOAuthSSLValidateServer Off
OIDCOAuthIntrospectionEndpointParams token_type_hint=access_token
OIDCOAuthClientID ovirt-engine
OIDCOAuthClientSecret <client_SSO_generated_key>
OIDCOAuthRemoteUserClaim sub
```

```
<LocationMatch ^/ovirt-engine/(api$/api/)>
  AuthType oauth20
  Require valid-user
</LocationMatch>
```

3.

구성 변경 사항을 저장하려면 **httpd** 및 **ovirt-engine** 을 다시 시작하십시오.

```
# systemctl restart httpd
# systemctl restart ovirt-engine
```

4.

다음 콘텐츠를 사용하여 **/etc/ovirt-engine/extensions.d/** 에 **openidc-authn.properties** 파일을 만듭니다.

```
ovirt.engine.extension.name = openidc-authn
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-extensions.aaa.misc
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.misc.http.AuthnExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Authn
ovirt.engine.aaa.authn.profile.name = openidchttp
ovirt.engine.aaa.authn.authz.plugin = openidc-authz
ovirt.engine.aaa.authn.mapping.plugin = openidc-http-mapping
config.artifact.name = HEADER
config.artifact.arg = OIDC_CLAIM_preferred_username
```

5.

다음 콘텐츠를 사용하여 **/etc/ovirt-engine/extensions.d/** 에 **openidc-http-**



**mapping.properties** 파일을 만듭니다.

```
ovirt.engine.extension.name = openid-http-mapping
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-extensions.aaa.misc
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.misc.mapping.MappingExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Mapping
config.mapAuthRecord.type = regex
config.mapAuthRecord.regex.mustMatch = false
config.mapAuthRecord.regex.pattern = ^(?<user>. *?)(\\|\\(|?<at>@)(?<suffix>. *?)@. *?)(?<realm>@. *?)$
config.mapAuthRecord.regex.replacement = ${user}${at}${suffix}
```

6.

다음 콘텐츠를 사용하여 **/etc/ovirt-engine/extensions.d/** 에 **openid-authz.properties** 파일을 만듭니다.

```
ovirt.engine.extension.name = openid-authz
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-extensions.aaa.misc
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.aaa.misc.http.AuthzExtension
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.aaa.Authz
config.artifact.name.arg = OIDC_CLAIM_preferred_username
config.artifact.groups.arg = OIDC_CLAIM_groups
```

7.

다음 콘텐츠를 사용하여 **/etc/ovirt-engine/engine.conf.d/** 에 **99-enable-external-auth.conf** 파일을 만듭니다.

```
ENGINE_SSO_ENABLE_EXTERNAL_SSO=true
ENGINE_SSO_EXTERNAL_SSO_LOGOUT_URI="${ENGINE_URI}/callback"
EXTERNAL_OIDC_USER_INFO_END_POINT=https://SSO.example.com/auth/realms/master/protocol/openid-connect/userinfo
EXTERNAL_OIDC_TOKEN_END_POINT=https://SSO.example.com/auth/realms/master/protocol/openid-connect/token
EXTERNAL_OIDC_LOGOUT_END_POINT=https://SSO.example.com/auth/realms/master/protocol/openid-connect/logout
EXTERNAL_OIDC_CLIENT_ID=ovirt-engine
EXTERNAL_OIDC_CLIENT_SECRET="<client_sso_generated_key>"
EXTERNAL_OIDC_HTTPS_PKI_TRUST_STORE="/etc/pki/java/cacerts"
EXTERNAL_OIDC_HTTPS_PKI_TRUST_STORE_PASSWORD=""
EXTERNAL_OIDC_SSL_VERIFY_CHAIN=false
EXTERNAL_OIDC_SSL_VERIFY_HOST=false
```

#### 19.5.4. OVN 구성

**Manager**에서 **ovirt-ovn-provider** 를 구성한 경우 **OVN** 공급자 인증 정보를 구성해야 합니다.

1. **user1**이 LDAP 그룹 **ovirt-administrator**에 속하는 다음 콘텐츠를 사용하여 **20-setup-ovirt-provider-ovn.conf**에 파일 **20-setup-ovirt-provider-ovn.conf**를 만듭니다. 여기서 **user1**은 **aaa-ldap-misc** 용으로 구성된 프로필입니다.

```
[OVIRT]
# ovirt-admin-user-name=user1@openidhttp
```

2. **ovirt-provider-ovn**을 다시 시작합니다.

```
# systemctl restart ovirt-provider-ovn
```

3. 관리 포털에 로그인하고 관리 공급자로 이동하여 **ovirt-provider-ovn**을 선택하고 편집을 클릭하여 **ovn** 공급자의 암호를 업데이트합니다.

## 19.6. 사용자 인증

### 19.6.1. 사용자 권한 부여 모델

**Red Hat Virtualization**은 다음 세 가지 구성 요소의 조합에 따라 권한 부여 제어를 적용합니다.

- 작업을 수행하는 사용자
- 수행 중인 작업 유형입니다.
- 작업을 수행할 오브젝트입니다.

### 19.6.2. 사용자 작업

작업을 성공적으로 수행하려면 사용자에게 적절한 권한이 있어야 합니다. 각 유형의 작업에는 해당 권한이 있습니다.

일부 작업은 둘 이상의 오브젝트에서 수행됩니다. 예를 들어 템플릿을 다른 스토리지 도메인에 복사하면 템플릿과 대상 스토리지 도메인 모두에 영향을 미칩니다. 작업을 수행하는 사용자는 작업에 영향을 미치는 모든 오브젝트에 대한 적절한 권한이 있어야 합니다.

## 19.7. 관리 포털에서 사용자 작업

### 19.7.1. 사용자 추가 및 VM 포털 권한 할당

사용자는 이미 생성되어 역할 및 권한을 할당할 수 있습니다. 이 절차에서 할당된 역할 및 권한은 사용자에게 VM 포털에 로그인하고 가상 머신 생성을 시작할 수 있는 권한을 부여합니다. 절차는 그룹 계정에 적용됩니다.

#### 사용자 추가 및 VM 포털 권한 할당

1. 헤더 표시줄에서 관리 → 구성을 클릭하여 구성 창을 엽니다.
2. **System Permissions** 를 클릭합니다.
3. 추가를 클릭하여 **Add System Permission to User** 창을 엽니다.
4. 검색에서 프로필을 선택합니다. 프로필은 검색할 도메인입니다. 검색 텍스트 필드에 이름 또는 이름 부분을 입력하고 **GO** 를 클릭합니다. 또는 **GO** 를 클릭하여 모든 사용자 및 그룹 목록을 봅니다.
5. 적절한 사용자 또는 그룹의 확인란을 선택합니다.
6. **Role to Assign** 에 할당할 적절한 역할을 선택합니다. **UserRole** 역할은 사용자 계정에 VM 포털에 로그인할 수 있는 권한을 부여합니다.
7. **OK**를 클릭합니다.

VM 포털에 로그인하여 사용자 계정에 로그인할 수 있는 권한이 있는지 확인합니다.

### 19.7.2. 사용자 정보 보기

#### 사용자 정보 보기

1. **Administration** → **Users (사용자)**를 클릭하여 권한 있는 사용자 목록을 표시합니다.
2. 사용자 이름을 클릭하여 세부 정보 보기를 엽니다. 일반적으로 일반 탭에서 사용자의 도메인 이름, 이메일 및 상태와 같은 일반 정보를 표시합니다.
3. 기타 탭을 사용하면 사용자에 대한 그룹, 권한, 할당량 및 이벤트를 볼 수 있습니다.

예를 들어 사용자가 속한 그룹을 보려면 디렉터리 그룹 탭을 클릭합니다.

### 19.7.3. 리소스에서 사용자 권한 보기

사용자에게 특정 리소스 또는 리소스 계층에 대한 권한을 할당할 수 있습니다. 할당된 사용자 및 각 리소스에 대한 권한을 볼 수 있습니다.

리소스에서 사용자 권한 보기

1. 리소스 이름을 찾아서 클릭하여 세부 정보 보기를 엽니다.
2. 권한 탭을 클릭하여 할당된 사용자, 사용자의 역할 및 선택한 리소스에 대한 상속된 권한을 나열합니다.

### 19.7.4. 사용자 제거

사용자 계정이 더 이상 필요하지 않은 경우 **Red Hat Virtualization**에서 제거합니다.

사용자 제거

1. **Administration** → **Users (사용자)**를 클릭하여 권한 있는 사용자 목록을 표시합니다.
2. 제거할 사용자를 선택합니다. 사용자가 가상 머신을 실행되고 있지 않은지 확인합니다.

3. 제거를 클릭한 다음 확인을 클릭합니다.

사용자는 **Red Hat Virtualization**에서 제거되지만 외부 디렉터리에서는 제거되지 않습니다.

### 19.7.5. 로그인한 사용자 보기

세션 시간 및 기타 세부 정보와 함께 현재 로그인한 사용자를 볼 수 있습니다. **Administration** → **Active User Sessions** 를 클릭하여 세션 **DB ID( Session DB ID, 사용자 이름, 권한 부여 공급자, 사용자 ID, 소스 IP, 세션 시작 시간 )** 및 로그인한 각 사용자에 대해 세션 마지막 활성 시간을 확인합니다.

### 19.7.6. 사용자 세션 종료

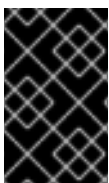
현재 로그인한 사용자의 세션을 종료할 수 있습니다.

사용자 세션 종료

1. **Administration** → **Active User Sessions (활성 사용자 세션)**를 클릭합니다.
2. 종료할 사용자 세션을 선택합니다.
3. 세션 종료 를 클릭합니다.
4. **OK**를 클릭합니다.

## 19.8. 명령줄에서 사용자 작업

**ovirt-aaa-jdbc-tool** 툴을 사용하여 내부 도메인의 사용자 계정을 관리할 수 있습니다. 툴을 사용하여 변경한 내용이 즉시 적용되며 **ovirt-engine** 서비스를 다시 시작할 필요가 없습니다. 전체 사용자 옵션 목록은 **ovirt-aaa-jdbc-tool user --help** 를 실행합니다. 이 섹션에는 일반적인 예제가 제공됩니다.



중요

**Manager** 머신에 로그인해야 합니다.

### 19.8.1. 새 사용자 생성

새 사용자 계정을 만들 수 있습니다. 선택적 **--attribute** 명령은 계정 세부 정보를 지정합니다. 전체 옵션 목록은 **ovirt-aaa-jdbc-tool 사용자 add --help** 를 실행합니다.

```
# ovirt-aaa-jdbc-tool user add test1 --attribute=firstName=John --attribute=lastName=Doe
adding user test1...
user added successfully
```

관리 포털에서 새로 생성된 사용자를 추가하고 사용자에게 적절한 역할 및 권한을 할당할 수 있습니다. 자세한 내용은 [19.7.1 절. “사용자 추가 및 VM 포털 권한 할당”](#)를 참조하십시오.

### 19.8.2. 사용자 암호 설정

암호를 생성할 수 있습니다. **--password-valid-to** 의 값을 설정해야 합니다. 그러지 않으면 암호 만료 시간이 현재 시간으로 설정됩니다. 날짜 형식은 **yyyy-MM-dd HH:mm:ssX** 입니다. 이 예에서 **-0800** 은 **GMT**에서 **8** 시간을 나타냅니다. 자세한 내용은 **ovirt-aaa-jdbc-tool 사용자 password-reset --help** 를 실행합니다.

```
# ovirt-aaa-jdbc-tool user password-reset test1 --password-valid-to="2025-08-01 12:00:00-0800"
Password:
updating user test1...
user updated successfully
```

#### 참고

기본적으로 내부 도메인의 사용자 계정에 대한 암호 정책에는 다음과 같은 제한이 있습니다.

- 최소 **6**자 이상
- 이전 암호 세 개는 암호를 변경하는 동안 다시 설정할 수 없습니다.

암호 정책 및 기타 기본 설정에 대한 자세한 내용을 보려면 **ovirt-aaa-jdbc-tool** 설정이 표시됨 을 실행합니다.

### 19.8.3. 사용자 시간 제한 설정

사용자 제한 기간을 설정할 수 있습니다.

```
# engine-config --set UserSessionTimeoutInterval=integer
```

#### 19.8.4. 사용자 암호 사전 암호화

**ovirt-engine-crypto-tool** 스크립트를 사용하여 사전 암호화된 사용자 암호를 생성할 수 있습니다. 이 옵션은 스크립트가 있는 데이터베이스에 사용자 및 암호를 추가하는 경우 유용합니다.

##### 참고

암호는 **Manager** 데이터베이스에 암호화된 형식으로 저장됩니다. 모든 암호를 동일한 알고리즘으로 암호화해야 하므로 **ovirt-engine-crypto-tool** 스크립트가 사용됩니다.

암호가 미리 암호화된 경우 암호 유효성 테스트를 수행할 수 없습니다. 암호 검증 정책을 준수하지 않는 경우에도 암호는 승인됩니다.

1.

다음 명령을 실행합니다.

```
# /usr/share/ovirt-engine/bin/ovirt-engine-crypto-tool.sh pbe-encode
```

이 스크립트는 암호를 입력하라는 메시지를 표시합니다.

또는 **--password=file:file** 옵션을 사용하여 파일의 첫 번째 줄로 표시되는 단일 암호를 암호화할 수 있습니다. 이 옵션은 자동화에 유용합니다. 다음 예에서 **file** 은 암호화를 위한 단일 암호를 포함하는 텍스트 파일입니다.

```
# /usr/share/ovirt-engine/bin/ovirt-engine-crypto-tool.sh pbe-encode --password=file:file
```

2.

**--encrypted** 옵션을 사용하여 **ovirt-aaa-jdbc-tool** 스크립트를 사용하여 새 암호를 설정합니다.

```
# ovirt-aaa-jdbc-tool user password-reset test1 --password-valid-to="2025-08-01 12:00:00-0800" --encrypted
```

3.

를 입력하고 암호화된 암호를 확인합니다.

```

Password:
Reenter password:
updating user test1...
user updated successfully

```

### 19.8.5. 사용자 정보 보기

자세한 사용자 계정 정보를 볼 수 있습니다.

```
# ovirt-aaa-jdbc-tool user show test1
```

이 명령은 관리 포털 → 사용자 화면에 있는 것보다 자세한 정보를 표시합니다.

### 19.8.6. 사용자 정보 편집

이메일 주소와 같은 사용자 정보를 업데이트할 수 있습니다.

```
# ovirt-aaa-jdbc-tool user edit test1 --attribute=email=jdoe@example.com
```

### 19.8.7. 사용자 제거

사용자 계정을 삭제할 수 있습니다.

```
# ovirt-aaa-jdbc-tool user delete test1
```

관리 포털에서 사용자를 제거합니다. 자세한 내용은 [19.7.4절](#). “사용자 제거”를 참조하십시오.

### 19.8.8. 내부 사용자 비활성화

**engine-setup** 중에 생성된 **admin@internal** 사용자를 포함하여 로컬 도메인에서 사용자를 비활성화할 수 있습니다. 기본 **admin** 사용자를 비활성화하기 전에 전체 관리 권한이 있는 **environment**에 하나 이상의 사용자가 있어야 합니다.

내부 사용자 비활성화



1. **Red Hat Virtualization Manager**가 설치된 시스템에 로그인합니다.
2. **SuperUser** 역할이 있는 다른 사용자가 환경에 추가되었는지 확인합니다. 자세한 내용은 **19.7.1절. “사용자 추가 및 VM 포털 권한 할당”**를 참조하십시오.
3. 기본 **admin** 사용자를 비활성화합니다.

```
# ovirt-aaa-jdbc-tool user edit admin --flag=+disabled
```



참고

비활성화된 사용자를 활성화하려면 **ovirt-aaa-jdbc-tool** 사용자가 사용자 이름 **--flag=-disabled**를 실행합니다.

### 19.8.9. 그룹 관리

**ovirt-aaa-jdbc-tool** 툴을 사용하여 내부 도메인의 그룹 계정을 관리할 수 있습니다. 그룹 계정 관리는 사용자 계정 관리와 유사합니다. 그룹 옵션의 전체 목록은 **ovirt-aaa-jdbc-tool group --help**를 실행합니다. 이 섹션에는 일반적인 예제가 제공됩니다.

#### 그룹 생성

이 절차에서는 그룹 계정을 생성하고, 사용자를 그룹에 추가하고, 그룹 세부 정보를 확인하는 방법을 설명합니다.

1. **Red Hat Virtualization Manager**가 설치된 시스템에 로그인합니다.
  2. 새 그룹을 생성합니다.
- ```
# ovirt-aaa-jdbc-tool group add group1
```
3. 그룹에 사용자를 추가합니다. 사용자는 이미 생성되어 있어야 합니다.

```
# ovirt-aaa-jdbc-tool group-manage useradd group1 --user=test1
```



## 참고

**group-manage** 옵션의 전체 목록은 **ovirt-aaa-jdbc-tool group-manage --help** 를 실행합니다.

4.

그룹 계정 세부 정보 보기:

```
# ovirt-aaa-jdbc-tool group show group1
```

5.

관리 포털에 새로 생성된 그룹을 추가하고 그룹에 적절한 역할 및 권한을 할당합니다. 그룹의 사용자는 그룹의 역할 및 권한을 상속합니다. 자세한 내용은 [19.7.1 절. “사용자 추가 및 VM 포털 권한 할당”](#)를 참조하십시오.

## 중첩 그룹 생성

다음 절차에서는 그룹 내에서 그룹을 생성하는 방법을 설명합니다.

1.

**Red Hat Virtualization Manager**가 설치된 시스템에 로그인합니다.

2.

첫 번째 그룹을 생성합니다.

```
# ovirt-aaa-jdbc-tool group add group1
```

3.

두 번째 그룹을 만듭니다.

```
# ovirt-aaa-jdbc-tool group add group1-1
```

4.

두 번째 그룹을 첫 번째 그룹에 추가합니다.

```
# ovirt-aaa-jdbc-tool group-manage groupadd group1 --group=group1-1
```

5.

관리 포털에 첫 번째 그룹을 추가하고 적절한 역할 및 권한을 그룹에 할당합니다. 자세한 내용은 [19.7.1 절. “사용자 추가 및 VM 포털 권한 할당”](#)를 참조하십시오.

### 19.8.10. 사용자 및 그룹 쿼리

**query** 모듈을 사용하면 사용자 및 그룹 정보를 쿼리할 수 있습니다. 전체 옵션 목록은 **ovirt-aaa-jdbc-tool query --help** 를 실행합니다.

모든 사용자 또는 그룹 계정 정보 나열

다음 절차에서는 모든 계정 정보를 나열하는 방법을 설명합니다.

1. **Red Hat Virtualization Manager**가 설치된 시스템에 로그인합니다.

2. 계정 세부 정보를 나열합니다.

- 모든 사용자 계정 세부 정보:

```
# ovirt-aaa-jdbc-tool query --what=user
```

- 모든 그룹 계정 세부 정보:

```
# ovirt-aaa-jdbc-tool query --what=group
```

필터링된 계정 세부 정보 나열

다음 절차에서는 계정 정보를 나열할 때 필터를 적용하는 방법을 설명합니다.

1. **Red Hat Virtualization Manager**가 설치된 시스템에 로그인합니다.

2. **--pattern** 매개변수를 사용하여 계정 세부 정보를 필터링합니다.

- 문자 **j**로 시작하는 이름으로 사용자 계정 세부 정보를 나열합니다.

```
# ovirt-aaa-jdbc-tool query --what=user --pattern="name=j*"
```

- **department** 특성이 **marketing** 으로 설정된 그룹 나열 :

```
# ovirt-aaa-jdbc-tool query --what=group --pattern="department=marketing"
```

### 19.8.11. 계정 설정 관리

기본 계정 설정을 변경하려면 **ovirt-aaa-jdbc-tool** 설정 모듈을 사용합니다.

계정 설정 업데이트

다음 절차에서는 기본 계정 설정을 업데이트하는 방법을 설명합니다.

1. **Red Hat Virtualization Manager**가 설치된 시스템에 로그인합니다.
2. 다음 명령을 실행하여 사용 가능한 모든 설정을 표시합니다.

```
# ovirt-aaa-jdbc-tool settings show
```

3. 원하는 설정을 변경합니다.

- 이 예제에서는 모든 사용자 계정의 세션 시간을 **60**분으로 기본 로그로 업데이트합니다. 기본값은 **10080**분입니다.

```
# ovirt-aaa-jdbc-tool settings set --name=MAX_LOGIN_MINUTES --value=60
```

- 이 예제에서는 사용자 계정이 잠기 전에 사용자가 수행할 수 있는 실패한 로그인 시도 횟수를 업데이트합니다. 기본값은 **5**입니다.

```
# ovirt-aaa-jdbc-tool settings set --name=MAX_FAILURES_SINCE_SUCCESS --value=3
```



참고

잠금 해제된 사용자 계정의 잠금을 해제하려면 **ovirt-aaa-jdbc-tool** 사용자가 **test1**의 잠금을 해제합니다.

## 19.9. 추가 로컬 도메인 구성

기본 내부 도메인 이외의 추가 로컬 도메인도 지원됩니다. 이 작업은 **ovirt-engine-extension-aaa-jdbc** 확장을 사용하여 수행할 수 있으며 엔터프라이즈 환경에서 사용 사례는 일반적이지 않을 수 있지만 외부 디렉터리 서버를 연결하지 않고 여러 도메인을 생성할 수 있습니다.

또한 생성된 로컬 도메인은 표준 **Red Hat Virtualization** 업그레이드 중에 자동 업그레이드되지 않으므로 향후 릴리스마다 수동으로 업그레이드해야 합니다. 추가 로컬 도메인을 생성하고 도메인을 업그레이드하는 방법에 대한 자세한 내용은 `/usr/share/doc/ovirt-engine-extension-aaa-jdbc-버전/README.admin`의 **README** 파일을 참조하십시오.

## 20장. 할당량 및 서비스 수준 계약 정책

### 20.1. 할당량 소개

할당량은 **Red Hat Virtualization**에서 제공하는 리소스 제한 틀입니다. 할당량은 사용자 권한에 의해 설정된 제한 계층에 대한 제한 계층으로 간주될 수 있습니다.

할당량은 데이터 센터 오브젝트입니다.

**Red Hat Virtualization** 환경의 관리자는 할당량을 통해 메모리, **CPU** 및 스토리지에 대한 사용자 액세스를 제한할 수 있습니다. 할당량은 관리자가 사용자를 할당할 수 있는 메모리 리소스 및 스토리지 리소스를 정의합니다. 따라서 사용자는 할당된 리소스만 사용할 수 있습니다. 할당량 리소스가 소모되면 **Red Hat Virtualization**에서 추가 사용자 작업을 허용하지 않습니다.

할당량의 두 가지 종류가 있습니다.

표 20.1. 할당량의 두 가지 다른 Kinds

| 할당량 유형   | 정의                                       |
|----------|------------------------------------------|
| 런타임 할당량  | 이 할당량은 CPU 및 메모리와 같은 런타임 리소스의 사용을 제한합니다. |
| 스토리지 할당량 | 이 할당량은 사용 가능한 스토리지의 양을 제한합니다.            |

**SELinux**와 같은 할당량에는 다음 세 가지 모드가 있습니다.

표 20.2. 할당량 모드

| Quota Mode | 함수                                                                                                                             |
|------------|--------------------------------------------------------------------------------------------------------------------------------|
| enforced   | 이 모드에서는 감사 모드로 설정한 할당량에 영향을 미치므로 할당량의 영향을 받는 그룹 또는 사용자로 리소스를 제한합니다.                                                            |
| audit      | 이 모드는 사용자를 차단하지 않고 할당량 위반을 기록하며 할당량을 테스트하는 데 사용할 수 있습니다. 감사 모드에서는 런타임 할당량의 양과 영향을 받는 사용자가 사용할 수 있는 스토리지 할당량 양을 늘리거나 줄일 수 있습니다. |

| Quota Mode | 함수                                         |
|------------|--------------------------------------------|
| disabled   | 이 모드는 할당량으로 정의된 런타임 및 스토리지 제한 사항을 비활성화합니다. |

사용자가 가상 머신 실행을 시도하면 가상 머신의 사양이 스토리지 허용과 비교되고 해당 할당량에 런타임 허용이 설정됩니다.

가상 머신을 시작하면 할당량에서 처리하는 모든 가상 머신의 집계 리소스가 할당량에 정의된 허용 범위를 초과하지 않으면 **Manager**에서 가상 머신의 실행을 거부합니다.

사용자가 새 디스크를 생성하면 요청된 디스크 크기가 해당 할당량에 적용되는 기타 모든 디스크의 집계된 디스크 사용량에 추가됩니다. 새 디스크가 할당량에서 허용하는 것보다 총 집계된 디스크 사용량을 가져오는 경우 디스크 생성이 실패합니다.

할당량을 사용하면 동일한 하드웨어의 리소스를 공유할 수 있습니다. 하드 및 소프트웨어 임계값을 지원합니다. 관리자는 할당량을 사용하여 리소스에 임계값을 설정할 수 있습니다. 이러한 임계값은 해당 리소스의 **100%** 사용으로 사용자 관점에서 표시됩니다. 고객이 이 임계값을 예기치 않게 초과할 때 오류를 방지하기 위해 인터페이스에서 임계값을 간단히 초과할 수 있는 **"Grace"** 양을 지원합니다. 임계값을 초과하면 고객에게 전송된 경고가 표시됩니다.

**중요**

할당량은 가상 머신 실행 시 제한 사항을 적용합니다. 이러한 제한 사항을 무시하면 가상 머신 및 가상 디스크를 사용할 수 없는 상황이 발생할 수 있습니다.

할당량이 적용된 모드에서 실행 중인 경우 할당량이 할당되지 않은 가상 시스템 및 디스크를 사용할 수 없습니다.

가상 시스템의 전원을 켜려면 해당 가상 시스템에 할당량을 할당해야 합니다.

가상 머신의 스냅샷을 생성하려면 가상 시스템과 연결된 디스크에 할당량이 할당되어 있어야 합니다.

가상 머신에서 템플릿을 생성할 때 템플릿에서 사용할 할당량을 선택하라는 메시지가 표시됩니다. 그러면 템플릿이 생성되는 가상 머신 및 디스크와 다른 할당량을 사용하도록 템플릿(및 템플릿에서 생성된 모든 시스템)을 설정할 수 있습니다.

**20.2. 공유 할당량 및 개별적으로 정의 할당량**

**SuperUser** 권한이 있는 사용자는 개별 사용자 또는 그룹의 할당량을 생성할 수 있습니다.

그룹 할당량은 **Active Directory** 사용자에게 대해 설정할 수 있습니다. **10** 명의 사용자 그룹에 **1TB**의 스토리지 할당량이 부여되고 **10**명의 사용자가 전체 테라바이트를 채우면 전체 그룹이 할당량을 초과하고 **10**명의 사용자가 해당 그룹과 연결된 스토리지를 사용할 수 없습니다.

개별 사용자의 할당량은 개인용으로만 설정됩니다. 개별 사용자가 자신의 스토리지 또는 런타임 할당량을 모두 사용하면 사용자가 할당량을 초과하고 사용자가 더 이상 할당량과 연결된 스토리지를 사용할 수 없습니다.

**20.3. 할당량 계정**

할당량이 소비자 또는 리소스에 할당되면 해당 소비자 또는 스토리지, **vCPU** 또는 메모리를 포함한 리소스에서 각 작업으로 인해 할당량 소비 또는 할당량 릴리스가 생성됩니다.

할당량은 사용자의 리소스에 대한 액세스를 제한하는 상한의 역할을 하므로 할당량 계산은 현재 사용자의 실제 사용과 다를 수 있습니다. 할당량은 현재 사용량이 아닌 최대 증가 가능성에 대해 계산됩니다.



**예 20.1. 회계 예**

사용자는 **1개의 vCPU** 및 **1024MB** 메모리가 있는 가상 머신을 실행합니다. 이 작업은 **1개의 vCPU** 및 **1024MB**의 할당량을 해당 사용자에게 할당합니다. 가상 시스템이 **1 vCPU**가 중지되고 **1024MB**의 **RAM**이 해당 사용자에게 할당된 할당량으로 다시 릴리스됩니다. 런타임 할당량 소비는 소비자의 실제 실행 시간 동안만 계산됩니다.

사용자가 **10GB**의 가상 썬 프로비저닝 디스크를 생성합니다. 실제 디스크 사용량은 **3GB**의 디스크만 실제로 사용 중임을 나타낼 수 있습니다. 그러나 할당량 소비는 해당 디스크의 최대 증가 가능성이 **10GB**가 됩니다.

**20.4. 데이터 센터에서 할당량 모드 활성화 및 변경**

이 절차에서는 데이터 센터에서 할당량 모드를 활성화하거나 변경할 수 있습니다. 할당량을 정의하려면 할당량 모드를 선택해야 합니다. 이 절차의 단계에 따라 관리 포털에 로그인해야 합니다.

감사 모드를 사용하여 할당량을 테스트하여 예상대로 작동하는지 확인합니다. 할당량을 생성하거나 변경하려면 감사 모드에 할당량이 없어야 합니다.

**데이터 센터에서 할당량 활성화 및 변경**

1. **Compute** → **Data Centers** (데이터 센터)를 클릭하고 데이터 센터를 선택합니다.
2. 편집 을 클릭합니다.
3. **Quota Mode** 드롭다운 목록에서 할당량 모드를 **Enforced** 로 변경합니다.
4. **OK**를 클릭합니다.

테스트 중에 할당량 모드를 감사로 설정하는 경우 할당량 설정을 적용하려면 **Enforced** 로 변경해야 합니다.

## 20.5. 새 할당량 정책 생성

감사 또는 강제 모드에서 할당량 모드가 활성화되었습니다. 데이터 센터의 리소스 사용량을 관리하기 위한 할당량 정책을 정의하려고 합니다.

### 새 할당량 정책 생성

1. **Administration** → **Quota (할당량)** 를 클릭합니다.
2. **추가** 를 클릭합니다.
3. **Name** 및 **Description** 필드를 작성합니다.
4. 데이터 센터를 선택합니다.
5. **Memory & CPU** 섹션에서 녹색 수행을 사용하여 **Cluster Threshold** 를 설정합니다.
6. **Memory & CPU** 섹션에서 **BlueAdapter** 를 사용하여 **Cluster Grace** 를 설정합니다.
7. **All Clusters** 또는 **Specific Clusters (특정 클러스터)** 라디오 버튼을 클릭합니다. **Specific Clusters** 를 선택하는 경우 할당량 정책을 추가할 클러스터의 확인란을 선택합니다.
8. **Edit (편집)** 를 클릭하여 **Edit Quota** 창을 엽니다.
  - a. **Memory** 필드에서 **Unlimited (무제한)** 라디오 버튼을 선택하여 (클러스터에서 메모리 리소스를 제한 없이 사용할 수 있음)를 선택하거나, 이 할당량에 의해 설정된 메모리 양을 설정하려면 **radio** 버튼을 선택합니다. 라디오 버튼에 제한을 선택하는 경우 **MB** 필드에 메모리 할당량을 메가바이트(**MB**)로 입력합니다.
  - b. **CPU** 필드에서 **Unlimited radio button** 또는 **radio** 버튼을 선택하여 이 할당량에 설정된 **CPU** 양을 설정합니다. 라디오 버튼에 제한을 선택하는 경우 **vCpus** 필드에 다수의 **vCPU**를 입력합니다.

- c. 할당량 편집 창에서 확인을 클릭합니다.
9. 스토리지 섹션에서 녹색 수행을 사용하여 스토리지 임계값을 설정합니다.
10. 스토리지 섹션에서 **BlueAdapter**를 사용하여 스토리지 그레이스를 설정합니다.
11. **All Storage Domains** 또는 **Specific Storage Domains** (특정 스토리지 도메인) 라디오 버튼을 클릭합니다. **Specific Storage Domains** 를 선택하는 경우 할당량 정책을 추가할 스토리지 도메인의 확인란을 선택합니다.
12. **Edit** (편집)를 클릭하여 **Edit Quota** 창을 엽니다.
- a. **Storage Quota** (스토리지 할당량) 필드에서 **Unlimited radio** 버튼 (스토리지를 제한 없이 사용할 수 있음) 또는 **radio** 버튼을 선택하여 할당량이 사용자를 제한하는 스토리지의 양을 설정합니다. 무선 버튼 제한을 선택하는 경우 **GB** 필드에 스토리지 할당량 크기를 기가바이트(**GB**)로 입력합니다.
- b. 할당량 편집 창에서 확인을 클릭합니다.
13. 새 할당량 창에서 확인을 클릭합니다.

## 20.6. 할당량 임계값 설정에 대한 설명

표 20.3. 할당량 임계값 및 **grace**

| 설정         | 정의                                                          |
|------------|-------------------------------------------------------------|
| 클러스터 임계값   | 데이터 센터별로 사용 가능한 클러스터 리소스의 양입니다.                             |
| 클러스터 Grace | 데이터 센터의 Cluster Threshold가 소진된 후 데이터 센터에 사용 가능한 클러스터의 양입니다. |
| 스토리지 임계값   | 데이터 센터당 사용 가능한 스토리지 리소스의 양입니다.                              |
| 스토리지 Grace | 데이터 센터의 스토리지 임계값이 소진된 후 데이터 센터에 사용 가능한 스토리지의 양입니다.          |

할당량이 **20% Grace**인 **100GB**로 설정된 경우 사용자는 **120GB**의 스토리지를 사용한 후 스토리지 사용을 차단합니다. 동일한 할당량이 **70 %**로 설정된 임계값이 설정된 경우 사용자는 **70GB**의 스토리지 사용을 초과할 때 경고를 받지만 **120GB**의 스토리지 소비에 도달할 때까지 스토리지를 계속 사용할 수 있습니다.) 할당량에 상대적으로 **"Threshold"** 및 **"Grace"**가 모두 설정됩니다. **"소유"**는 **"소프트 제한"**으로 간주될 수 있으며 초과하여 경고를 생성합니다. **"grace"**는 **"하드 제한"**으로 간주될 수 있으며 초과하여 더 이상 스토리지 리소스를 사용할 수 없습니다.

## 20.7. 오브젝트에 할당량 할당

### 가상 머신에 할당량 할당

1. **Compute** → **Virtual Machines** (가상 머신)를 클릭하고 가상 머신을 선택합니다.
2. 편집 을 클릭합니다.
3. **Quota (할당량)** 드롭다운 목록에서 가상 시스템을 사용할 할당량을 선택합니다.
4. **OK**를 클릭합니다.

### 가상 디스크에 할당량 할당

1. **Compute** → **Virtual Machines** 를 클릭합니다.
2. 가상 머신의 이름을 클릭하여 세부 정보 보기를 엽니다.
3. 디스크 탭을 클릭하고 할당량과 연결할 디스크를 선택합니다.
4. 편집 을 클릭합니다.
5. **Quota (할당량)** 드롭다운 목록에서 가상 디스크를 사용할 할당량을 선택합니다.

6.

**OK**를 클릭합니다.

중요

해당 가상 머신이 작동하려면 가상 시스템과 연결된 모든 오브젝트에 대해 할당량을 선택해야 합니다. 가상 머신과 연결된 오브젝트에 대한 할당량을 선택하지 못하면 가상 머신이 작동하지 않습니다. **Manager**가 이 상황에서 **throw**하는 오류는 일반적이므로 할당량을 가상 시스템과 연결된 모든 오브젝트와 연결하지 않았기 때문에 오류가 발생했는지 확인하기 어렵습니다. 할당량이 할당되지 않은 가상 머신의 스냅샷을 가져올 수 없습니다. 가상 디스크에 할당량이 할당되지 않은 가상 머신의 템플릿을 생성할 수 없습니다.

### 20.8. 할당량을 사용하여 사용자별 리소스 제한 사용

이 절차에서는 할당량을 사용하여 사용자가 액세스할 수 있는 리소스를 제한하는 방법을 설명합니다.

할당량에 사용자 할당

1.

**Administration** → **Quota** (할당량) 를 클릭합니다.

2.

대상 할당량의 이름을 클릭하여 세부 정보 보기를 엽니다.

3.

소비자 탭을 클릭합니다.

4.

추가를 클릭합니다.

5.

**Search** 필드에 할당량과 연결할 사용자의 이름을 입력합니다.

6.

**GO** 를 클릭합니다.

7.

사용자 이름 옆에 있는 확인란을 선택합니다.

8.

**OK**를 클릭합니다.

잠시 후 사용자는 세부 정보 보기의 소비자 탭에 표시됩니다.

## 20.9. 할당량 편집

다음 절차에서는 기존 할당량을 변경하는 방법을 설명합니다.

### 할당량 편집

1. **Administration** → **Quota (할당량)**를 클릭하고 할당량을 선택합니다.
2. **편집**을 클릭합니다.
3. 필요에 따라 필드를 편집합니다.
4. **OK**를 클릭합니다.

## 20.10. 할당량 제거

다음 절차에서는 할당량을 제거하는 방법을 설명합니다.

### 할당량 제거

1. **Administration** → **Quota (할당량)**를 클릭하고 할당량을 선택합니다.
2. **제거**를 클릭합니다.
3. **OK**를 클릭합니다.

## 20.11. 서비스 수준 계약 정책 시행

다음 절차에서는 서비스 수준 계약 **CPU** 기능을 설정하는 방법을 설명합니다.

#### 서비스 수준 계약 **CPU** 정책 설정

1. **Compute** → **Virtual Machines** 를 클릭합니다.
2. 새로 생성을 클릭하거나 가상 머신을 선택하고 편집 을 클릭합니다.
3. 리소스 할당 탭을 클릭합니다.
4. **CPU DestinationRule**을 지정합니다. 가능한 옵션은 낮음, 중간, 높음, 사용자 정의 및 **Disabled** 입니다. **High** 로 설정된 가상 머신은 중간 단위로 두 배 많은 공유를 수신하며, 가상 머신은 가상 머신이 **Low** 로 설정된 두 배의 공유를 수신합니다. **disabled** 는 **VDSM**에서 공유 분배를 결정하기 위해 이전 알고리즘을 사용하도록 지시합니다. 일반적으로 이러한 조건에서 분배되는 공유 수는 **1020**입니다.

이제 사용자의 **CPU** 사용량이 설정한 정책에 의해 관리됩니다.

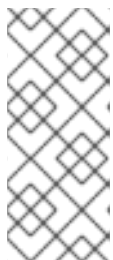
## 21장. 이벤트 알림

### 21.1. 관리 포털에서 이벤트 알림 구성

**Red Hat Virtualization Manager**는 **Red Hat Virtualization Manager**가 관리하는 환경에서 특정 이벤트가 발생할 때 지정된 사용자에게 이메일을 통해 알릴 수 있습니다. 이 기능을 사용하려면 메시지를 전달할 이메일 전송 에이전트를 설정해야 합니다. 관리 포털을 통해 이메일 알림만 구성할 수 있습니다. **Manager** 시스템에서 **SNMP** 트랩을 구성해야 합니다.

#### 이벤트 알림 구성

1. **Manager**에서 자동 메시지를 수락하고 배포 목록에 전달할 수 있는 이메일 서버에 액세스할 수 있는지 확인합니다.
2. **Administration** → **Users** (사용자)를 클릭하고 사용자를 선택합니다.
3. 사용자의 사용자 이름을 클릭하여 세부 정보 페이지로 이동합니다.
4. 이벤트 알림 탭에서 이벤트 관리를 클릭합니다.
5. 모두 확장 버튼 또는 주제별 확장 버튼을 사용하여 이벤트를 볼 수 있습니다.
6. 적절한 확인란을 선택합니다.
7. **mail Recipient** 필드에 이메일 주소를 입력합니다.



#### 참고

이메일 주소는 문자 메시지 이메일 주소(예: **1234567890@carrierdomainname.com**) 또는 이메일 주소 및 텍스트 메시지 이메일 주소가 포함된 이메일 그룹 주소일 수 있습니다.



8.

**OK**를 클릭합니다.

9.

**Manager** 시스템에서 **ovirt-engine-notifier.conf** 를 **90-email-notify.conf** 라는 새 파일에 복사합니다.

```
# cp /usr/share/ovirt-engine/services/ovirt-engine-notifier/ovirt-engine-notifier.conf /etc/ovirt-engine/notifier/notifier.conf.d/90-email-notify.conf
```

10.

**90-email-notify.conf** 를 편집하여 **EMAIL** 알림 섹션을 제외한 모든 항목을 삭제합니다.

11.

아래 예제와 같이 올바른 이메일 변수를 입력합니다. 이 파일은 원래 **ovirt-engine-notifier.conf** 파일의 값을 덮어씁니다.

```
#-----#
# EMAIL Notifications #
#-----#

# The SMTP mail server address. Required.
MAIL_SERVER=myemailserver.example.com

# The SMTP port (usually 25 for plain SMTP, 465 for SMTP with SSL, 587 for SMTP with
# TLS)
MAIL_PORT=25

# Required if SSL or TLS enabled to authenticate the user. Used also to specify 'from' user
# address if mail server
# supports, when MAIL_FROM is not set. Address is in RFC822 format
MAIL_USER=

# Required to authenticate the user if mail server requires authentication or if SSL or TLS is
# enabled
SENSITIVE_KEYS="${SENSITIVE_KEYS},MAIL_PASSWORD"
MAIL_PASSWORD=

# Indicates type of encryption (none, ssl or tls) should be used to communicate with mail
# server.
MAIL_SMTP_ENCRYPTION=none

# If set to true, sends a message in HTML format.
HTML_MESSAGE_FORMAT=false

# Specifies 'from' address on sent mail in RFC822 format, if supported by mail server.
MAIL_FROM=rhev2017@example.com

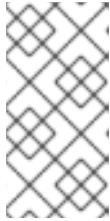
# Specifies 'reply-to' address on sent mail in RFC822 format.
MAIL_REPLY_TO=

# Interval to send smtp messages per # of IDLE_INTERVAL
```

```
MAIL_SEND_INTERVAL=1
```

```
# Amount of times to attempt sending an email before failing.
```

```
MAIL_RETRIES=4
```



참고

자세한 내용은 `/etc/ovirt-engine/notifier/notifier.conf.d/README` 를 참조하십시오.

12.

**ovirt-engine-notifier** 서비스를 활성화하고 다시 시작하여 변경 사항을 활성화합니다.

```
# systemctl daemon-reload
# systemctl enable ovirt-engine-notifier.service
# systemctl restart ovirt-engine-notifier.service
```

지정된 사용자는 이제 **Red Hat Virtualization** 환경의 이벤트를 기반으로 이메일을 수신합니다. 선택한 이벤트는 해당 사용자의 이벤트 **Notifier** 탭에 표시됩니다.

## 21.2. 관리 포털에서 이벤트 알림 취소

사용자는 불필요한 이메일 알림을 구성하고 이를 취소하고자 합니다.

이벤트 알림 취소

1.

**Administration** → **Users** 를 클릭합니다.

2.

사용자의 사용자 이름을 클릭하여 세부 정보 보기를 엽니다.

3.

이벤트 알림 탭을 클릭하여 사용자가 이메일 알림을 수신하는 이벤트를 나열합니다.

4.

이벤트 관리를 클릭합니다.

5.

모두 확장 버튼 또는 주제별 확장 버튼을 사용하여 이벤트를 볼 수 있습니다.

6. 해당 이벤트에 대한 알림을 제거하려면 적절한 확인란을 지웁니다.
7. **OK**를 클릭합니다.

### 21.3. OVIRT-ENGINE-NOTIFIER.CONF의 이벤트 알림 매개변수

이벤트 알림기 구성 파일은 `/usr/share/ovirt-engine/services/ovirt-engine-notifier/ovirt-engine-notifier.conf` 에서 확인할 수 있습니다.

표 21.1. `ovirt-engine-notifier.conf` variables

| 변수 이름                  | 기본값                                 | marks                                                    |
|------------------------|-------------------------------------|----------------------------------------------------------|
| SENSITIVE_KEYS         | none                                | 기록되지 않는 쉼표로 구분된 키 목록입니다.                                 |
| JBOSS_HOME             | /opt/rh/eap7/root/usr/share/wildfly | Manager에서 사용하는 JBoss 애플리케이션 서버의 위치입니다.                   |
| ENGINE_ETC             | /etc/ovirt-engine                   | Manager에서 사용하는 <b>etc</b> 디렉토리의 위치입니다.                   |
| ENGINE_LOG             | /var/log/ovirt-engine               | Manager에서 사용하는 로그 디렉토리의 위치입니다.                           |
| ENGINE_USR             | /usr/share/ovirt-engine             | Manager에서 사용하는 <b>ECDHE</b> 디렉토리의 위치입니다.                 |
| ENGINE_JAVA_MODULEPATH | \${ENGINE_USR}/modules              | JBoss 모듈이 추가되는 파일 경로입니다.                                 |
| NOTIFIER_DEBUG_ADDRESS | none                                | 알림기가 사용하는 Java 가상 머신의 원격 디버깅을 수행하는 데 사용할 수 있는 머신의 주소입니다. |
| NOTIFIER_STOP_TIME     | 30                                  | 서비스 시간이 초과되는 시간(초)입니다.                                   |
| NOTIFIER_STOP_INTERVAL | 1                                   | 시간 초과 카운터가 증가되는 시간(초)입니다.                                |
| INTERVAL_IN_SECONDS    | 120                                 | 메시지를 구독자에게 디스패치하는 인스턴스 간 간격(초)입니다.                       |

| 변수 이름                                  | 기본값       | marks                                                                                                                                                           |
|----------------------------------------|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| IDLE_INTERVAL                          | 30        | 우선순위가 낮은 작업을 수행할 간격(초)입니다.                                                                                                                                      |
| DAYS_TO_KEEP_HISTORY                   | 0         | 이 변수는 디스패치된 이벤트 수를 기록 테이블에 보존합니다. 이 변수를 설정하지 않으면 이벤트가 기록 테이블에 무기한 유지됩니다.                                                                                        |
| FAILED_QUERIES_NOTIFICATION_THRESHOLD  | 30        | 알림 이메일을 보낸 후 실패한 쿼리 수입니다. 알림 이메일은 알림을 가져오지 못한 후 전송되며, 이 변수에 의해 지정된 실패 수에 도달할 때마다 한 번 전송됩니다. 값을 <b>0</b> 또는 <b>1</b> 로 지정하면 각 실패와 함께 이메일이 전송됩니다.                 |
| FAILED_QUERIES_NOTIFICATION_RECIPIENTS | none      | 통지 이메일이 전송될 수신자의 이메일 주소입니다. 이메일 주소는 쉼표로 구분해야 합니다. 이 항목은 <b>FILTER</b> 변수에서 더 이상 사용되지 않습니다.                                                                      |
| DAYS_TO_SEND_ON_STARTUP                | 0         | 알림기가 시작될 때 처리 및 전송되는 이전 이벤트의 일 수입니다.                                                                                                                            |
| FILTER                                 | exclude:* | 이메일 알림의 트리거 및 LinuxONE을 결정하는 데 사용되는 알고리즘입니다. 이 변수의 값은 포함 또는 제외되는 이벤트, 이벤트 및 수신자의 조합으로 구성됩니다. 예를 들면 <b>:VDC_START(smtp:mail@example.com) \${FILTER}</b> 를 포함합니다. |
| MAIL_SERVER                            | none      | SMTP 메일 서버 주소입니다. 필수 항목입니다.                                                                                                                                     |
| MAIL_PORT                              | 25        | 통신에 사용되는 포트입니다. 가능한 값에는 일반 SMTP의 경우 <b>25</b> , SSL이 있는 SMTP의 경우 <b>465</b> , TLS를 사용하는 SMTP의 경우 <b>587</b> 이 포함됩니다.                                            |

| 변수 이름                | 기본값                               | marks                                                                                                                                                      |
|----------------------|-----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MAIL_USER            | none                              | 사용자를 인증하기 위해 SSL을 활성화하면 이 변수를 설정해야 합니다. 이 변수는 MAIL_FROM 변수가 설정되지 않은 경우 "from" 사용자 주소를 지정하는 데도 사용됩니다. 일부 메일 서버는 이 기능을 지원하지 않습니다. 이 주소는 RFC822 형식으로 되어 있습니다. |
| SENSITIVE_KEYS       | `\${SENSITIVE_KEYS},MAIL_PASSWORD | 메일 서버에 인증이 필요하거나 SSL 또는 TLS가 활성화된 경우 사용자를 인증하는 데 필요합니다.                                                                                                    |
| MAIL_PASSWORD        | none                              | 메일 서버에 인증이 필요하거나 SSL 또는 TLS가 활성화된 경우 사용자를 인증하는 데 필요합니다.                                                                                                    |
| MAIL_SMTP_ENCRYPTION | none                              | 통신에 사용할 암호화 유형입니다. 가능한 값은 <b>none ssl,tls</b> 입니다.                                                                                                         |
| HTML_MESSAGE_FORMAT  | false                             | 이 변수가 <b>true</b> 로 설정된 경우 메일 서버는 HTML 형식으로 메시지를 보냅니다.                                                                                                     |
| MAIL_FROM            | none                              | 이 변수는 메일 서버에서 지원하는 경우 RFC822 형식으로 발신자 주소를 지정합니다.                                                                                                           |
| MAIL_REPLY_TO        | none                              | 이 변수는 메일 서버에서 지원하는 경우 전송된 메일에서 RFC822 형식으로 회신 주소를 지정합니다.                                                                                                   |
| MAIL_SEND_INTERVAL   | 1                                 | 각 IDLE_INTERVAL에 대해 전송할 SMTP 메시지 수                                                                                                                         |
| MAIL_RETRIES         | 4                                 | 실패하기 전에 이메일 전송을 시도하는 횟수입니다.                                                                                                                                |
| SNMP_MANAGER         | none                              | SNMP 관리자 역할을 할 시스템의 IP 주소 또는 정규화된 도메인 이름입니다. 항목은 공백으로 구분되어야 하며 포트 번호를 포함할 수 있습니다. 예:<br><b>manager1.example.com</b><br><b>manager2.example.com:164</b>     |

| 변수 이름                              | 기본값                                    | marks                                                                                                                             |
|------------------------------------|----------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| SNMP_COMMUNITY                     | public                                 | 기본 SNMP 커뮤니티                                                                                                                      |
| SNMP_OID                           | 1.3.6.1.4.1.2312.13.11                 | 경고에 대한 기본 트랩 오브젝트 식별자입니다. 이 OID가 정의된 경우 모든 트랩 유형은 이벤트 정보가 추가되어 SNMP 관리자에게 전송됩니다. 기본 트랩을 변경하면 생성된 트랩이 Manager의 관리 정보 기반을 따르지 않습니다. |
| ENGINE_INTERVAL_IN_SECONDS         | 300                                    | Manager가 설치된 시스템을 모니터링하는 간격(초)입니다. 모니터링이 완료된 시점부터 간격이 측정됩니다.                                                                      |
| ENGINE_MONITOR_RETRIES             | 3                                      | 알림기가 실패 후 지정된 간격으로 Manager가 설치된 시스템의 상태를 모니터링하려고 하는 횟수입니다.                                                                        |
| ENGINE_TIMEOUT_IN_SECONDS          | 30                                     | 알림기가 실패 후 지정된 간격으로 Manager가 설치된 머신의 상태를 모니터링하기 전에 대기하는 시간(초)입니다.                                                                  |
| IS_HTTPS_PROTOCOL                  | false                                  | JBoss가 보안 모드에서 실행되는 경우 이 항목을 <b>true</b> 로 설정해야 합니다.                                                                              |
| SSL_PROTOCOL                       | TLS                                    | SSL이 활성화된 경우 JBoss 구성 커넥터에서 사용하는 프로토콜입니다.                                                                                         |
| SSL_IGNORE_CERTIFICATE_ERRORS      | false                                  | JBoss가 보안 모드에서 실행되고 있고 SSL 오류를 무시하려면 이 값을 <b>true</b> 로 설정해야 합니다.                                                                 |
| SSL_IGNORE_HOST_VERIFICATION       | false                                  | JBoss가 보안 모드에서 실행 중이고 호스트 이름 확인을 무시하려면 이 값을 <b>true</b> 로 설정해야 합니다.                                                               |
| REPEAT_NON_RESPONSIVE_NOTIFICATION | false                                  | 이 변수는 Manager가 설치된 머신이 무응답인 경우 구독자에게 반복적으로 오류 메시지를 보낼지 여부를 지정합니다.                                                                 |
| ENGINE_PID                         | /var/lib/ovirt-engine/ovirt-engine.pid | Manager의 PID의 경로 및 파일 이름입니다.                                                                                                      |

## 21.4. SNMP TRAPS를 전송하도록 RED HAT VIRTUALIZATION MANAGER 구성

**Red Hat Virtualization Manager**를 구성하여 간단한 네트워크 관리 프로토콜 트랩을 하나 이상의 외부 **SNMP** 관리자에게 보냅니다. **SNMP** 트랩에는 시스템 이벤트 정보가 포함되어 있으며 **Red Hat Virtualization** 환경을 모니터링하는 데 사용됩니다. **SNMP** 관리자에게 전송된 트랩 수 및 유형은 **Red Hat Virtualization Manager** 내에서 정의할 수 있습니다.

이 절차에서는 트랩을 수신하도록 하나 이상의 외부 **SNMP** 관리자를 구성하고 다음 세부 정보가 있다고 가정합니다.

- **SNMP** 관리자 역할을 할 시스템의 **IP** 주소 또는 정규화된 도메인 이름입니다. 필요한 경우 **Manager**에서 트랩 알림을 수신하는 포트를 결정합니다. 기본적으로 **UDP** 포트 **162**입니다.
- **SNMP** 커뮤니티 여러 개의 **SNMP** 관리자가 단일 커뮤니티에 속할 수 있습니다. 관리 시스템 및 에이전트는 동일한 커뮤니티 내에 있는 경우에만 통신할 수 있습니다. 기본 커뮤니티는 **public**입니다.
- 경고에 대한 트랩 오브젝트 식별자입니다. **Red Hat Virtualization Manager**는 기본 **OID 1.3.6.1.4.1.2312.13.1.1**을 제공합니다. 이 **OID**가 정의된 경우 모든 트랩 유형은 이벤트 정보가 추가되어 **SNMP** 관리자에게 전송됩니다. 기본 트랩을 변경하면 생성된 트랩이 **Manager**의 관리 정보 기반을 따르지 않습니다.

참고

**Red Hat Virtualization Manager**는 **/usr/share/doc/ovirt-engine/mibs/OVIRT-MIB.txt** 및 **/usr/share/doc/ovirt-engine/mibs/REDHAT-MIB.txt**에서 관리 정보 기반을 제공합니다. 계속하기 전에 **SNMP** 관리자에 **LLB**를 로드합니다.

이벤트 알림 데몬 구성 파일 **/usr/share/ovirt-engine/services/ovirt-engine-notifier/ovirt-engine-notifier.conf**의 **Manager**에 기본 **SNMP** 구성 값이 있습니다. 다음 절차에 설명된 값은 해당 파일에 제공된 기본값 또는 예제 값을 기반으로 합니다. 시스템 업그레이드와 같은 변경 후 구성 옵션을 유지하려면 **ovirt-engine-notifier.conf** 파일을 편집하지 않고 재정의 파일을 정의하는 것이 좋습니다.

관리자에서 **SNMP** 트랩 구성

1. **Manager**에서 **SNMP** 구성 파일을 만듭니다.

```
# vi /etc/ovirt-engine/notifier/notifier.conf.d/20-snmp.conf
```

2.

**SNMP** 관리자, **SNMP** 커뮤니티, **OID**를 다음 형식으로 지정합니다.

```
SNMP_MANAGERS="manager1.example.com manager2.example.com:162"
SNMP_COMMUNITY=public
SNMP_OID=1.3.6.1.4.1.2312.13.1.1
```

3.

**SNMP** 관리자에게 보낼 이벤트를 정의합니다.

예 **21.1.** 이벤트 예

모든 이벤트를 기본 **SNMP** 프로파일로 보냅니다.

```
FILTER="include:*(snmp:) ${FILTER}"
```

심각도가 높은 모든 이벤트를 기본 **SNMP** 프로파일로 **ERROR** 또는 **ALERT** 로 보냅니다.

```
FILTER="include:*:ERROR(snmp:) ${FILTER}"
```

```
FILTER="include:*:ALERT(snmp:) ${FILTER}"
```

**VDC\_START** 에 대한 이벤트를 지정된 이메일 주소로 보냅니다.

```
FILTER="include:VDC_START(snmp:mail@example.com) ${FILTER}"
```

**VDC\_START** 를 기본 **SNMP** 프로파일에 제외한 모든 항목에 대한 이벤트를 보냅니다.

```
FILTER="exclude:VDC_START include:*(snmp:) ${FILTER}"
```

**ovirt-engine-notifier.conf** 에 정의된 기본 필터입니다. 이 필터를 비활성화하거나 덮어쓰기 필터를 적용하지 않으면 알림이 전송되지 않습니다.

```
FILTER="exclude:*"
```

**Vdc\_START** 는 사용 가능한 감사 로그 메시지의 예입니다. 전체 감사 로그 메시지는 **/usr/share/doc/ovirt-engine/AuditLogMessages.properties** 에서 확인할 수 있습니다. 또는



**SNMP** 관리자 내에서 결과를 필터링합니다.

4. 파일을 저장합니다.
5. **ovirt-engine-notifier** 서비스를 시작하고 이 서비스가 부팅 시 시작되는지 확인합니다.

```
# systemctl start ovirt-engine-notifier.service
# systemctl enable ovirt-engine-notifier.service
```

**SNMP** 관리자를 확인하여 트랩이 수신되고 있는지 확인합니다.



참고

**SNMP\_MANAGERS,MAIL\_SERVER**, 또는 둘 다 **/usr/share/ovirt-engine/services/ovirt-engine-notifier/ovirt-engine-notifier.conf** 에 올바르게 정의되어야 합니다.

## 22장. 유틸리티

### 22.1. OVIRT ENGINE RENAME TOOL

#### 22.1.1. oVirt Engine Rename Tool

**engine-setup** 명령을 정리 환경에서 실행하면 명령에서 설정 프로세스 중에 제공된 **Manager**의 정규화된 도메인 이름을 사용하는 여러 인증서와 키를 생성합니다. 관리자의 정규화된 도메인 이름을 나중에 변경해야 하는 경우(예: **Manager**를 다른 도메인으로 호스팅하는 머신 마이그레이션 때문에) 새 이름을 반영하도록 정규화된 도메인 이름의 레코드를 업데이트해야 합니다. **ovirt-engine-rename** 명령은 이 작업을 자동화합니다.

**ovirt-engine-rename** 명령은 다음 위치에서 **Manager**의 정규화된 도메인 이름 레코드를 업데이트합니다.

- **/etc/ovirt-engine/engine.conf.d/10-setup-protocols.conf**
- **/etc/ovirt-engine/isouploader.conf.d/10-engine-setup.conf**
- **/etc/ovirt-engine/logcollector.conf.d/10-engine-setup.conf**
- **/etc/pki/ovirt-engine/cert.conf**
- **/etc/pki/ovirt-engine/cert.template**
- **/etc/pki/ovirt-engine/certs/apache.cer**
- **/etc/pki/ovirt-engine/keys/apache.key.nopass**
- **/etc/pki/ovirt-engine/keys/apache.p12**



주의

**ovirt-engine-rename** 명령은 **Manager**를 실행하는 웹 서버에 대한 새 인증서를 생성하는 반면 **Manager** 또는 인증 기관의 인증서에는 영향을 미치지 않습니다. 이로 인해 **ovirt-engine-rename** 명령, 특히 **Red Hat Enterprise Virtualization 3.2** 및 이전 버전에서 업그레이드된 환경에서 **ovirt-engine-rename** 명령을 사용할 위험이 있습니다. 따라서 **engine-cleanup** 및 **engine-setup** 을 실행하여 **Manager**의 정규화된 도메인 이름을 변경하는 것이 좋습니다.



주의

업그레이드 프로세스 중에 이전 호스트 이름을 확인할 수 있어야 합니다. **oVirt Engine Rename Tool**이 **[ ERROR ] Host name is not valid: <OLD FQDN>** 이 IP 주소로 확인되지 않은 경우, **/etc/hosts** 파일에 이전 호스트 이름을 추가하고 **oVirt Engine Rename Tool**을 사용한 다음 **/etc/hosts** 파일에서 이전 호스트 이름을 제거합니다.

### 22.1.2. oVirt Engine Rename 명령의 구문

**ovirt-engine-rename** 명령의 기본 구문은 다음과 같습니다.

```
# /usr/share/ovirt-engine/setup/bin/ovirt-engine-rename
```

명령에서는 다음 옵션도 허용합니다.

**--newname=[new name]**

사용자 상호 작용 없이 **Manager**에 대해 정규화된 새 도메인 이름을 지정할 수 있습니다.

**--log=[file]**

이름 변경 작업 로그를 작성할 파일의 경로와 이름을 지정할 수 있습니다.

**--config=[file]**

이름 변경 작업에 로드할 구성 파일의 경로와 파일 이름을 지정할 수 있습니다.

### **--config-append=[file]**

이름 변경 작업에 추가할 구성 파일의 경로와 파일 이름을 지정할 수 있습니다. 이 옵션을 사용하여 이름 변경 작업을 자동화하기 위해 기존 응답 파일의 경로와 파일 이름을 지정할 수 있습니다.

### **--generate-answer=[file]**

응답과 **ovirt-engine-rename** 명령으로 변경된 값과 파일의 경로와 파일 이름을 지정할 수 있습니다.

## 22.1.3. oVirt Engine Rename Tool으로 관리자 이름 변경

**ovirt-engine-rename** 명령을 사용하여 **Manager**의 정규화된 도메인 이름(**FQDN**) 레코드를 업데이트할 수 있습니다.



### 중요

**ovirt-engine-rename** 명령은 **imageio-proxy** 또는 **websocket-proxy** 와 같은 **SSL** 인증서를 업데이트하지 않습니다. **ovirt-engine-rename** 을 실행한 후 수동으로 업데이트해야 합니다. 아래에서 **SSL 인증서 업데이트**를 참조하십시오.

틀은 관리자가 로컬 **ISO** 또는 데이터 스토리지 도메인을 제공하는지 여부를 확인합니다. 이 경우 도구를 사용하면 작업을 계속하기 전에 스토리지에 연결된 가상 머신 또는 스토리지 도메인을 제거, 종료 또는 유지 관리 모드로 전환하라는 메시지가 표시됩니다. 이렇게 하면 가상 머신의 가상 디스크 연결이 손실되지 않으며 이름 변경 프로세스 중에 **ISO** 스토리지 도메인이 연결이 손실되지 않습니다.

### **oVirt Engine Rename Tool** 사용

1. 새 **FQDN**에 대해 모든 **DNS** 및 기타 관련 레코드를 준비합니다.
2. **DHCP**가 사용되는 경우 **DHCP** 서버 구성을 업데이트합니다.
3. **Manager**에서 호스트 이름을 업데이트합니다.

4.

다음 명령을 실행합니다.

```
# /usr/share/ovirt-engine/setup/bin/ovirt-engine-rename
```

5.

프롬프트가 표시되면 **Enter** 를 눌러 엔진 서비스를 중지합니다.

```
During execution engine service will be stopped (OK, Cancel) [OK]:
```

6.

프롬프트가 표시되면 **Manager**에 새 **FQDN**을 입력합니다.

```
New fully qualified server name:new_engine_fqdn
```

**ovirt-engine-rename** 명령은 **Manager**의 **FQDN** 레코드를 업데이트합니다.

셀프 호스트 엔진의 경우 다음 추가 단계를 완료합니다.

1.

모든 기존 셀프 호스트 엔진 노드에서 다음 명령을 실행합니다.

```
# hosted-engine --set-shared-config fqdn new_engine_fqdn --type=he_local
```

이 명령은 각 자체 호스팅 엔진 노드의 **/etc/ovirt-hosted-engine-ha/hosted-engine.conf**의 **FQDN**을 수정합니다.

2.

셀프 호스트 엔진 노드 중 하나에서 다음 명령을 실행합니다.

```
# hosted-engine --set-shared-config fqdn new_engine_fqdn --type=he_shared
```

이 명령은 공유 스토리지 도메인의 **/etc/ovirt-hosted-engine-ha/hosted-engine.conf**의 마스터 사본의 **FQDN**을 수정합니다.

이제 모든 신규 및 기존 셀프 호스트 엔진 노드에서 새 **FQDN**을 사용합니다.

**SSL certificates** 업데이트

**ovirt-engine-rename** 명령 다음에 다음 명령을 실행하여 **SSL** 인증서를 업데이트합니다.

```
1. # names="websocket-proxy imageio-proxy"

2. # subject="$(\
  openssl x509 \
  -in /etc/pki/ovirt-engine/certs/apache.cer \
  -noout \
  -subject | \
  sed \
    's;subject= \(.*\);|1;'
)"

3. # ./usr/share/ovirt-engine/bin/engine-prolog.sh

4. # for name in $names; do
  /usr/share/ovirt-engine/bin/pki-enroll-pkcs12.sh \
  --name="{name}" \
  --password=mypass \
  --subject="{subject}" \
  --keep-key \
  --san=DNS:"{ENGINE_FQDN}"
done
```

## 22.2. 엔진 구성 도구

### 22.2.1. 엔진 구성 도구

엔진 구성 도구는 **Red Hat Virtualization** 환경의 글로벌 설정을 구성하는 명령줄 유틸리티입니다. 이 툴은 엔진 데이터베이스에 저장된 키-값 매핑 목록과 상호 작용하며 개별 키 값을 검색하고 설정할 수 있으며 사용 가능한 모든 구성 키와 값 목록을 검색할 수 있습니다. 또한 **Red Hat Virtualization** 환경의 각 구성 수준에 대해 서로 다른 값을 저장할 수 있습니다.



#### 참고

구성 키의 값을 검색하거나 설정하려면 **Red Hat Virtualization Manager** 또는 **Red Hat JBoss Enterprise Application Platform**을 실행할 필요가 없습니다. 구성 키 값-키 매핑은 엔진 데이터베이스에 저장되므로 **postgresql** 서비스가 실행되는 동안 업데이트할 수 있습니다. 그러면 **ovirt-engine** 서비스가 다시 시작될 때 변경 사항이 적용됩니다.

### 22.2.2. engine-config 명령의 구문

**Red Hat Virtualization Manager**가 설치된 머신에서 엔진 구성 도구를 실행할 수 있습니다. 사용법에 대한 자세한 내용을 보려면 명령에 대한 도움말 출력을 출력합니다.

## # engine-config --help

일반적인 작업:

- 사용 가능한 구성 키 나열

```
# engine-config --list
```

- 사용 가능한 구성 값 나열

```
# engine-config --all
```

- 구성 키의 값 검색

```
# engine-config --get KEY_NAME
```

**KEY\_NAME** 을 지정된 키 버전의 값을 검색하기 위해 기본 키의 이름으로 바꿉니다. **--cver** 매개변수를 사용하여 검색할 값의 구성 버전을 지정합니다. 버전이 제공되지 않으면 기존 버전의 값이 반환됩니다.

- 설정 키의 값 설정

```
# engine-config --set KEY_NAME=KEY_VALUE --cver=VERSION
```

**KEY\_NAME** 을 설정할 특정 키의 이름으로 바꾸고 **KEY\_VALUE** 를 설정할 값으로 교체합니다. 구성 버전이 두 개 이상인 환경에서 **VERSION** 을 지정해야 합니다.

- **ovirt-engine** 서비스를 다시 시작하여 변경 사항을 로드합니다.

변경 사항을 적용하려면 **ovirt-engine** 서비스를 다시 시작해야 합니다.

```
# systemctl restart ovirt-engine.service
```

### 22.3. USB 필터 편집기

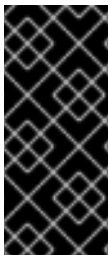
### 22.3.1. USB 필터 편집기 설치

**USB 필터 편집기**는 **usbfilter.txt** 정책 파일을 구성하는 데 사용되는 **Windows** 도구입니다. 이 파일에 정의된 정책 규칙에서는 **Red Hat Virtualization Manager**를 사용하여 관리되는 가상 머신으로 클라이언트 머신에서 특정 **USB** 장치의 자동 통과를 허용하거나 거부합니다. 정책 파일은 **Red Hat Virtualization Manager**의 **/etc/ovirt-engine/usbfilter.txt** **USB** 필터 정책에 대한 변경 사항은 **Red Hat Virtualization Manager**의 **ovirt-engine** 서비스가 재시작되지 않는 한 적용되지 않습니다.

"[Installers and Images for Red Hat Virtualization Manager](#)" 에서 **USB** 필터 편집기 설치 프로그램을 다운로드합니다.

#### USB 필터 편집기 설치

1. **Windows** 머신에서 **USB** 필터 편집기에 대해 다운로드한 **.msi** 파일을 실행합니다.
2. 설치 마법사의 단계를 따릅니다. 별도로 지정하지 않는 한 **USB** 필터 편집기는 기본적으로 **C:\Program Files\RedHat\ECDHE Filter Editor** 또는 **C:\Program Files(x86)\RedHat\ECDHE Filter Editor** 에 설치됩니다.
3. **USB** 필터 편집기 바로 가기 아이콘이 데스크탑에 생성됩니다.



#### 중요

**SCP(Secure Copy)** 클라이언트를 사용하여 **Red Hat Virtualization Manager**에서 필터 정책을 가져오고 내보냅니다. **Windows** 머신을 위한 보안 복사 도구는 **WinSCP** (<http://winscp.net>)입니다.

기본 **USB** 장치 정책은 가상 머신에 **USB** 장치에 대한 기본 액세스를 제공합니다. 추가 **USB** 장치를 사용할 수 있도록 정책을 업데이트합니다.

### 22.3.2. USB 필터 편집기 인터페이스

데스크탑에서 **USB** 필터 편집기 바로 가기 아이콘을 두 번 클릭합니다.

**Red Hat USB** 필터 편집기 인터페이스에는 각 **USB** 장치에 대한 클래스, 벤더, 제품, 버전 및 작업이 표시됩니다. 허용되는 **USB** 장치는 **Action** 열에서 허용으로 설정됩니다. 금지된 장치는 블록으로 설정됨



니다.

표 22.1. USB 편집기 필드

| 이름     | 설명                               |
|--------|----------------------------------|
| 클래스    | USB 장치 유형(예: 프린터, 대용량 스토리지 컨트롤러) |
| vendor | 선택한 유형의 장치 제조업체입니다.              |
| 제품     | 특정 USB 장치 모델입니다.                 |
| 버전     | 제품의 개정판입니다.                      |
| 동작     | 지정된 장치를 허용하거나 차단합니다.             |

**USB** 장치 정책 규칙은 나열된 순서로 처리됩니다. 위쪽 및 아래쪽 버튼을 사용하여 목록에서 규칙을 더 높이 또는 낮게 이동합니다. **USB** 필터 편집기에서 명시적으로 허용되지 않는 한 모든 **USB** 장치가 거부되도록 범용 블록 규칙은 가장 낮은 항목으로 남아 있어야 합니다.

### 22.3.3. USB 정책 추가

데스크탑에서 **USB** 필터 편집기 바로 가기 아이콘을 두 번 클릭하여 편집기를 엽니다.

#### USB 정책 추가

1. **추가**를 클릭합니다.
2. **USB** 클래스, 벤더 ID, 제품 ID 및 리버전 확인란을 사용하여 장치를 지정합니다.

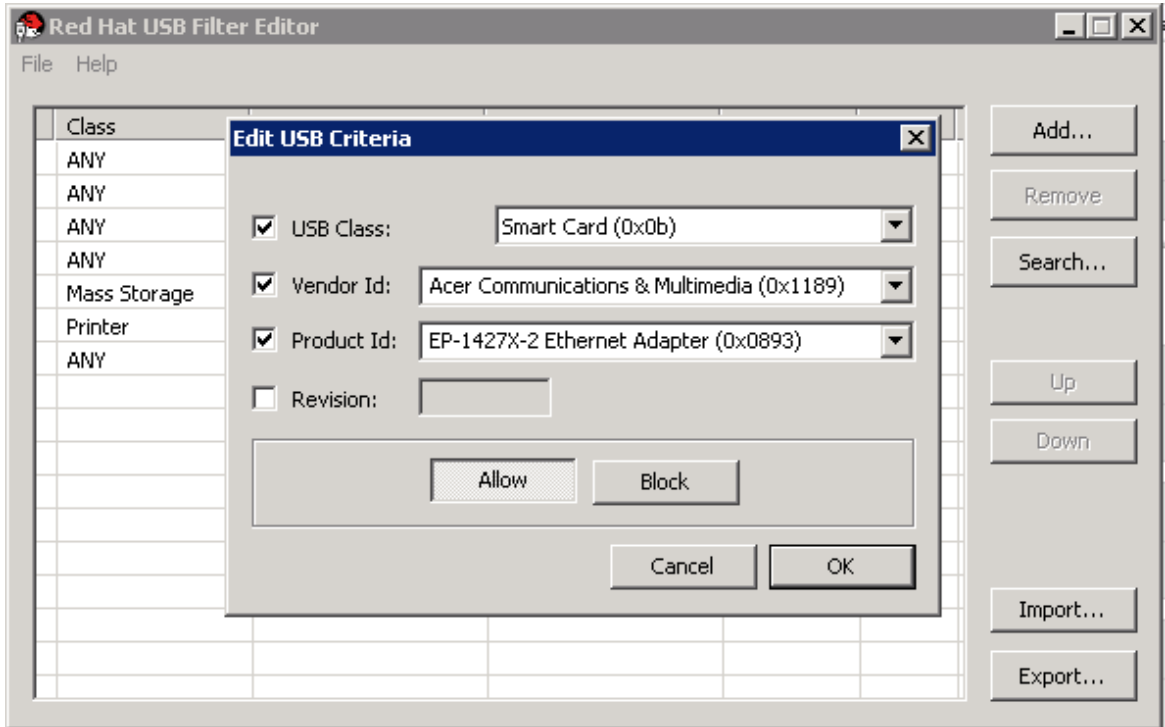
가상 머신에서 **USB** 장치의 사용을 허용하려면 허용 버튼을 클릭합니다. 블록 버튼을 클릭하여 **USB** 장치를 가상 머신에서 금지합니다.

확인 을 클릭하여 선택한 필터 규칙을 목록에 추가하고 창을 닫습니다.

#### 예 22.1. 장치 추가

다음은 제조업체의 **AcerECDHE & Multimedia** 에서 허용되는 장치 목록에 **USB** 클래스

스마트 카드, 장치 **EP-1427X-2** 이더넷 어댑터를 추가하는 방법의 예입니다.



3. 파일 → 저장을 클릭하여 변경 사항을 저장합니다.

**USB** 정책을 **USB** 필터 편집기에 추가했습니다. 효과를 적용하려면 **USB** 필터 정책을 **Red Hat Virtualization Manager**로 내보내야 합니다.

### 22.3.4. USB 정책 제거

데스크탑에서 **USB** 필터 편집기 바로 가기 아이콘을 두 번 클릭하여 편집기를 엽니다.

#### USB 정책 제거

1. 제거할 정책을 선택합니다.
2. 제거를 클릭합니다. 정책을 제거하려는지 확인하라는 메시지가 표시되면 **A message prompting you to confirm that you want to remove the policy.**

3. **Yes** 를 클릭하여 정책을 삭제할지 확인합니다.

4. 파일 → 저장을 클릭하여 변경 사항을 저장합니다.

**USB** 필터 편집기에서 **USB** 정책을 제거했습니다. 효과를 적용하려면 **USB** 필터 정책을 **Red Hat Virtualization Manager**로 내보내야 합니다.

### 22.3.5. USB 장치 정책 검색

연결된 **USB** 장치를 검색하여 **USB** 필터 편집기에서 허용하거나 차단합니다.

데스크탑에서 **USB** 필터 편집기 바로 가기 아이콘을 두 번 클릭하여 편집기를 엽니다.

#### **USB** 장치 정책 검색

1. **Search**를 클릭합니다 연결된 **USB** 장치 창에 연결된 모든 장치 목록이 표시됩니다.
2. 장치를 선택하고 필요에 따라 허용 또는 블록 을 클릭합니다. 선택한 장치를 두 번 클릭하여 창을 닫습니다. 장치에 대한 정책 규칙이 목록에 추가됩니다.
3. **Up** 및 **Down** 버튼을 사용하여 목록에서 새 정책 규칙의 위치를 변경합니다.
4. 파일 → 저장을 클릭하여 변경 사항을 저장합니다.

연결된 **USB** 장치를 검색했습니다. **USB** 필터 정책을 적용하려면 **Red Hat Virtualization Manager**로 내보내야 합니다.

### 22.3.6. USB 정책 내보내기

업데이트된 정책을 적용하려면 **USB** 장치 정책 변경 사항을 **Red Hat Virtualization Manager**에 내보내고 업로드해야 합니다. 정책을 업로드하고 **ovirt-engine** 서비스를 다시 시작합니다.

데스크탑에서 **USB** 필터 편집기 바로 가기 아이콘을 두 번 클릭하여 편집기를 엽니다.

### USB 정책 내보내기

1. **Export; Save As** 창이 열립니다.
2. **usbfilter.txt** 파일 이름으로 파일을 저장합니다.
3. **WinSCP**와 같은 보안 복사 클라이언트를 사용하여 **usbfilter.txt** 파일을 **Red Hat Virtualization Manager**를 실행하는 서버에 업로드합니다. 파일은 서버의 다음 디렉터리에 배치해야 합니다. **/etc/ovirt-engine/**
4. **Red Hat Virtualization Manager**를 실행하는 서버에서 **root** 사용자로 **ovirt-engine** 서비스를 다시 시작합니다.

```
# systemctl restart ovirt-engine.service
```

### 22.3.7. USB 정책 가져오기

기존 **USB** 장치 정책을 다운로드하여 **USB** 필터 편집기로 가져와야 편집할 수 있습니다.

#### USB 정책 가져오기

1. **WinSCP**와 같은 보안 복사 클라이언트를 사용하여 **Red Hat Virtualization Manager**를 실행하는 서버에서 **usbfilter.txt** 파일을 다운로드합니다. 파일은 서버의 다음 디렉터리에서 찾을 수 있습니다. **/etc/ovirt-engine/**
2. 데스크탑에서 **USB** 필터 편집기 바로 가기 아이콘을 두 번 클릭하여 편집기를 엽니다.
3. 가져오기를 클릭하여 열기 창을 엽니다.
4. 서버에서 다운로드한 **usbfilter.txt** 파일을 엽니다.

## 22.4. 로그 수집기 툴

### 22.4.1. 로그 수집기

**Red Hat Virtualization Manager**에 로그 수집 도구가 포함되어 있습니다. 이를 통해 지원을 요청할 때 **Red Hat Virtualization** 환경에서 관련 로그를 쉽게 수집할 수 있습니다.

로그 수집 명령은 **ovirt-log-collector** 입니다. **root** 사용자로 로그인하고 **Red Hat Virtualization** 환경에 대한 관리 자격 증명을 제공해야 합니다. **ovirt-log-collector -h** 명령은 **ovirt-log-collector** 명령에 유효한 모든 옵션 목록을 포함하여 사용법 정보를 표시합니다.

### 22.4.2. ovirt-log-collector 명령의 구문

로그 수집기 명령의 기본 구문은 다음과 같습니다.

```
# ovirt-log-collector options list all|clusters|datacenters
# ovirt-log-collector options collect
```

지원되는 두 가지 작업 모드는 **list** 및 **collect** 입니다.

- **list** 매개변수는 **Red Hat Virtualization Manager**에 연결된 호스트, 클러스터 또는 데이터 센터를 나열합니다. 나열된 오브젝트를 기반으로 로그 컬렉션을 필터링할 수 있습니다.
- **collect** 매개변수는 **Red Hat Virtualization Manager**에서 로그 컬렉션을 수행합니다. 수집된 로그는 **/tmp/logcollector** 디렉터리 아래의 아카이브 파일에 배치됩니다. **ovirt-log-collector** 명령은 각 로그에 특정 파일 이름을 할당합니다.

다른 매개변수를 지정하지 않으면 기본 작업은 사용 가능한 호스트를 데이터 센터와 함께 속하는 클러스터와 함께 나열하는 것입니다. 특정 로그를 검색하기 위해 사용자 이름과 암호를 입력하라는 메시지가 표시됩니다.

**ovirt-log-collector** 명령을 추가로 구체화하는 다양한 매개변수가 있습니다.

일반 옵션

**--version**

사용하는 명령의 버전 번호를 표시하고 프롬프트로 돌아갑니다.

**-h, --help**

명령 사용 정보를 표시하고 프롬프트로 돌아갑니다.

**--conf-file=PATH**

틀에서 사용할 구성 파일로 **PATH** 를 설정합니다.

**--local-tmp=PATH**

로그가 저장되는 디렉터리로 **PATH** 를 설정합니다. 기본 디렉터리는 **/tmp/logcollector** 입니다.

**--ticket-number=TICKET**

**TICKET** 을 티켓 또는 케이스 번호로 설정하여 **SOS** 보고서와 연결합니다.

**--upload=FTP\_SERVER**

**FTP\_SERVER** 를 **FTP** 를 사용하여 검색된 로그의 대상으로 설정합니다.

**Red Hat** 지원 담당자가 권장하지 않는 한 이 옵션을 사용하지 마십시오.

**--log-file=PATH**

**PATH** 를 로그 출력에 사용해야 하는 특정 파일 이름으로 설정합니다.

**--quiet**

자동 모드를 설정하여 콘솔 출력을 최소로 줄입니다. 자동 모드는 기본적으로 해제되어 있습니다.

**-v, --verbose**

자세한 정보 표시 모드를 설정하고 더 많은 콘솔 출력을 제공합니다. 세부 정보 표시 모드는 기본적으로 해제되어 있습니다.

**--time-only**

전체 **SOS** 보고서를 생성하지 않고 호스트 간의 시간 차이에 대한 정보만 표시합니다.

**Red Hat Virtualization Manager** 옵션

이러한 옵션은 로그 컬렉션을 필터링하고 **Red Hat Virtualization Manager**에 대한 인증 세부 정보를 지정합니다.

이러한 매개변수는 특정 명령에 대해 결합할 수 있습니다. 예를 들어 **ovirt-log-collector --user=admin@internal --cluster ClusterA,ClusterB --hosts "SalesHost"\*** 는 사용자를 **admin@internal** 로 지정하고 로그 컬렉션을 **A** 및 **B** 클러스터의 **SalesHost** 호스트로만 제한합니다.

### **--no-hypervisors**

로그 컬렉션에서 가상화 호스트를 생략합니다.

### **--one-hypervisor-per-cluster**

각 클러스터에서 하나의 호스트(하나의 경우)의 로그를 수집합니다.

### **-u USER, --user=USER**

로그인할 사용자 이름을 설정합니다. **USER** 는 형식 사용자@도메인 으로 지정됩니다. 여기서 **user** 는 사용자 이름이며 **domain** 은 사용 중인 디렉터리 서비스 도메인입니다. 사용자가 디렉터리 서비스에 있어야 하며 **Red Hat Virtualization Manager**에 대해 알고 있어야 합니다.

### **-r FQDN, --rhev=FQDN**

로그를 수집할 **Red Hat Virtualization Manager**의 정규화된 도메인 이름을 설정합니다. 여기서 **FQDN** 은 **Manager**의 정규화된 도메인 이름으로 교체됩니다. 로그 수집기가 **Red Hat Virtualization Manager**와 동일한 로컬 호스트에서 실행되고 있다고 가정합니다. 기본값은 **localhost** 입니다.

### **-c CLUSTER, --cluster=CLUSTER**

**Red Hat Virtualization Manager**에서 지정된 **CLUSTER**의 가상화 호스트에서 로그를 수집합니다. 포함할 클러스터는 쉼표로 구분된 클러스터 이름 목록 또는 일치 패턴으로 지정해야 합니다.

### **-d DATACENTER, --data-center=DATACENTER**

**Red Hat Virtualization Manager**에서 로그 외에도 후보 **DATACENTER**의 가상화 호스트에서 로그를 수집합니다. 포함할 데이터 센터를 쉼표로 구분된 데이터 센터 이름 목록 또는 일치 패턴으로 지정해야 합니다.

### **-H HOSTS\_LIST, --hosts=HOSTS\_LIST**

**Red Hat Virtualization Manager**에서 로그 외에도 지정된 **HOSTS\_LIST**의 가상화 호스트에서 로그를 수집합니다. 포함할 호스트는 쉼표로 구분된 호스트 이름, 정규화된 도메인 이름 또는 **IP** 주소로 지정해야 합니다. 일치 패턴도 유효합니다.

## SSH 구성

**--ssh-port=PORT**

가상화 호스트와 **SSH** 연결에 사용할 포트로 **PORT** 를 설정합니다.

**-k KEYFILE, --key-file=KEYFILE**

**KEYFILE** 을 가상화 호스트에 액세스하는 데 사용할 공용 **SSH** 키로 설정합니다.

**--max-connections=MAX\_CONNECTIONS**

**MAX\_CONNECTIONS** 를 가상화 호스트의 로그에 대한 최대 동시 **SSH** 연결로 설정합니다. 기본값은 **10**입니다.

**PostgreSQL** 데이터베이스 옵션

기본값에서 변경된 경우 **pg-user** 및 **dbname** 매개변수를 사용하여 데이터베이스 사용자 이름과 데이터베이스 이름을 지정해야 합니다.

데이터베이스가 로컬 호스트에 없는 경우 **pg-dbhost** 매개변수를 사용합니다. 선택적 **pg-host-key** 매개변수를 사용하여 원격 로그를 수집합니다. 원격 로그 수집에 성공하려면 **PostgreSQL SOS** 플러그인을 데이터베이스 서버에 설치해야 합니다.

**--no-postgresql**

데이터베이스 컬렉션을 비활성화합니다. 로그 수집기는 **Red Hat Virtualization Manager PostgreSQL** 데이터베이스에 연결하고 **--no-postgresql** 매개변수를 지정하지 않는 한 로그 보고서에 데이터를 포함합니다.

**--pg-user=USER**

**USER** 를 데이터베이스 서버와의 연결에 사용할 사용자 이름으로 설정합니다. 기본값은 **postgres** 입니다.

**--pg-database=DATABASE**

데이터베이스 서버와의 연결에 사용할 데이터베이스 이름으로 **DATABASE** 을 설정합니다. 기본값은 **postgres** 입니다.

**--pg-dbhost=DBHOST**

**DBHOST** 를 데이터베이스 서버의 호스트 이름으로 설정합니다. 기본값은 **localhost** 입니다.

**--pg-host-key=KEYFILE**



**KEYFILE** 을 데이터베이스 서버의 공개 **ID** 파일(개인 키)으로 설정합니다. 이 값은 기본적으로 설정되지 않습니다. 데이터베이스가 로컬 호스트에 없는 경우에만 필요합니다.

### 22.4.3. 기본 로그 수집기 사용량

추가 매개변수를 지정하지 않고 **ovirt-log-collector** 명령을 실행하면 기본 동작은 **Red Hat Virtualization Manager** 및 연결된 호스트에서 모든 로그를 수집하는 것입니다. **--no-postgresql** 매개변수가 추가되지 않는 한 데이터베이스 로그도 수집합니다. 다음 예제에서는 로그 수집기를 실행하여 **Red Hat Virtualization Manager**와 연결된 호스트의 모든 로그를 수집합니다.

#### 예 22.2. 로그 수집기 사용량

```
# ovirt-log-collector
INFO: Gathering oVirt Engine information...
INFO: Gathering PostgreSQL the oVirt Engine database and log files from localhost...
Please provide REST API password for the admin@internal oVirt Engine user (CTRL+D to abort):
About to collect information from 3 hypervisors. Continue? (Y/n):
INFO: Gathering information from selected hypervisors...
INFO: collecting information from 192.168.122.250
INFO: collecting information from 192.168.122.251
INFO: collecting information from 192.168.122.252
INFO: finished collecting information from 192.168.122.250
INFO: finished collecting information from 192.168.122.251
INFO: finished collecting information from 192.168.122.252
Creating compressed archive...
INFO Log files have been collected and placed in /tmp/logcollector/sosreport-rhn-account-
20110804121320-ce2a.tar.xz.
The MD5 for this file is 6d741b78925998caff29020df2b2ce2a and its size is 26.7M
```

## 22.5. ISO 업로더 툴

### 22.5.1. ISO 업로더 툴



참고

**ISO** 업로더 툴이 더 이상 사용되지 않습니다. **Red Hat**은 관리 포털 또는 **REST API**를 사용하여 **ISO** 이미지를 데이터 도메인에 업로드하는 것이 좋습니다. 자세한 내용은 **11.8.1 절**. “데이터 스토리지 도메인에 이미지 업로드” 을 참조하십시오.

**ISO** 업로더는 **ISO** 스토리지 도메인에 **ISO** 이미지를 업로드하는 툴입니다. 이는 **Red Hat Virtualization Manager**의 일부로 설치됩니다.

**ISO uploader** 명령은 **engine-iso-uploader** 입니다. **root** 사용자로 로그인하고 이 명령을 사용하려면

**Red Hat Virtualization** 환경에 대한 관리 자격 증명을 제공해야 합니다. **engine-iso-uploader -h** 명령은 **engine-iso-uploader** 명령에 유효한 모든 옵션 목록을 포함하여 사용 정보를 표시합니다.

## 22.5.2. engine-iso-uploader 명령의 구문

**ISO uploader** 명령의 기본 구문은 다음과 같습니다.

```
# engine-iso-uploader options list
# engine-iso-uploader options upload file file file
```

**ISO uploader** 명령은 목록 및 업로드 의 두 가지 작업을 지원합니다.

- list** 작업에는 **ISO** 파일을 업로드할 수 있는 **ISO** 스토리지 도메인이 나열됩니다. **Red Hat Virtualization Manager**는 설치 프로세스 중에 **Manager**가 설치된 시스템에 이 목록을 생성합니다.
- 업로드 작업에서는 단일 **ISO** 파일 또는 공백으로 구분된 여러 **ISO** 파일을 지정된 **ISO** 스토리지 도메인에 업로드합니다. **NFS**는 기본적으로 사용되지만 **SSH**도 사용할 수 있습니다.



### 참고

**SSH**를 사용하는 경우 **SSH** 사용자 암호를 입력하라는 메시지가 여러 번 표시됩니다. **ISO** 파일을 업로드하기 전에 공개 **SSH** 키를 **iso** 도메인 서버에 업로드한 다음 **--key-file=KEYFILE** 옵션을 사용하여 이러한 프롬프트를 방지할 수 있습니다. 공용 **SSH** 키를 업로드하는 한 가지 방법은 **ssh-copy-id -i ~/.ssh/mykey user@host** 명령을 사용하는 것입니다.

**ISO uploader** 명령을 사용할 때 위의 작업 중 하나를 지정해야 합니다. 또한 업로드 작업을 사용하려면 하나 이상의 로컬 파일을 지정해야 합니다.

**engine-iso-uploader** 명령을 추가로 구체화하는 몇 가지 매개 변수가 있습니다.

### 일반 옵션

**--version**

**ISO uploader** 명령의 버전을 표시합니다.

**-h, --help**

**ISO uploader** 명령 사용 방법에 대한 정보를 표시합니다.

**--conf-file=PATH**

명령에서 사용할 구성 파일로 **PATH** 를 설정합니다. 기본값은 **/etc/ovirt-engine/isouploader.conf** 입니다.

**--log-file=PATH**

명령이 로그 출력을 작성하는 데 사용할 특정 파일 이름으로 **PATH** 를 설정합니다. 기본값은 **/var/log/ovirt-engine/ovirt-iso-uploader/ovirt-iso-uploader\_date.log** 입니다.

**--cert-file=PATH**

엔진의 유효성을 검증하기 위한 인증서로 **PATH** 를 설정합니다. 기본값은 **/etc/pki/ovirt-engine/ca.pem** 입니다.

**--insecure**

엔진을 확인하려고 시도하지 않도록 지정합니다.

**--nossl**

**SSL** 이 엔진에 연결하는 데 사용되지 않도록 지정합니다.

**--quiet**

자동 모드를 설정하여 콘솔 출력을 최소로 줄입니다.

**-v, --verbose**

자세한 정보 표시 모드를 설정하고 더 많은 콘솔 출력을 제공합니다.

**-f, --force**

업로드 중인 소스 파일의 파일 이름이 대상 **ISO** 도메인에 있는 기존 파일과 동일한 경우 강제 모드가 필요합니다. 이 옵션을 사용하면 기존 파일을 덮어쓰게 됩니다.

**Red Hat Virtualization Manager** 옵션

**-u USER, --user=USER**

명령을 실행하는 데 자격 증명이 있는 사용자를 지정합니다. **USER** 는 사용자 이름@domain 형식으로 지정됩니다. 사용자가 지정된 도메인에 있어야 하며 **Red Hat Virtualization Manager**에 대해 알고 있어야 합니다.

### **-r FQDN, --engine=FQDN**

이미지를 업로드할 **Red Hat Virtualization Manager**의 IP 주소 또는 정규화된 도메인 이름을 지정합니다. 이미지 업로드기가 **Red Hat Virtualization Manager**가 설치된 동일한 시스템에서 실행 중인 것으로 가정합니다. 기본값은 **localhost:443** 입니다.

### **ISO** 스토리지 도메인 옵션

다음 옵션은 이미지를 업로드할 **ISO** 도메인을 지정합니다. 이러한 옵션은 함께 사용할 수 없습니다. **-i** 옵션 또는 **-n** 옵션을 사용해야 합니다.

### **-i, --iso-domain=ISODOMAIN**

스토리지 도메인 **ISODOMAIN** 을 업로드 대상으로 설정합니다.

### **-n, --nfs-server=NFSSERVER**

**NFS** 경로 **NFSSERVER** 을 업로드할 대상으로 설정합니다.

### 연결 옵션

**ISO** 업로더는 **NFS**를 기본값으로 사용하여 파일을 업로드합니다. 이러한 옵션은 대신 **SSH** 파일 전송을 지정합니다.

### **--ssh-user=USER**

**USER** 를 업로드에 사용할 **SSH** 사용자 이름으로 설정합니다. 기본값은 **root** 입니다.

### **--ssh-port=PORT**

**SSH**에 연결할 때 사용할 포트로 **PORT** 를 설정합니다.

### **-k KEYFILE, --key-file=KEYFILE**

**KEYFILE** 을 **SSH** 인증에 사용할 개인 키로 설정합니다. 키를 설정하지 않으면 **--ssh-user=USER** 로 지정된 사용자의 암호를 입력하라는 메시지가 표시됩니다. **ISO** 파일을 업로드하기 전에 공용 **SSH** 키를 **ISO** 도메인 서버에 업로드해야 합니다. 이 작업을 수행하는 한 가지 방법은 **ssh-copy-id -i ~/.ssh/mykey user@host** 명령을 사용하는 것입니다.

### 22.5.3. NFS 서버 지정

예 22.3. NFS 서버에 업로드

```
# engine-iso-uploader --nfs-server=storage.demo.redhat.com:/iso/path upload RHEL6.0.iso
```

### 22.5.4. 기본 ISO 업로더 사용

아래 예제에서는 **ISO uploader** 및 **list** 매개변수를 보여줍니다. 첫 번째 명령은 사용 가능한 **ISO** 스토리지 도메인을 나열합니다. **admin@internal** 사용자는 명령에 지정된 사용자가 없으므로 사용됩니다. 두 번째 명령은 **NFS**에서 **ISO** 파일을 지정된 **ISO** 도메인에 업로드합니다.

예 22.4. 도메인 나열 및 이미지 업로드

```
# engine-iso-uploader list
Please provide the REST API password for the admin@internal oVirt Engine user (CTRL+D to abort):
ISO Storage Domain Name | Datacenter      | ISO Domain Status
ISODomain                | Default         | active

# engine-iso-uploader --iso-domain=[ISODomain] upload [RHEL6.iso]
Please provide the REST API password for the admin@internal oVirt Engine user (CTRL+D to abort):
```

### 22.5.5. ISO 스토리지 도메인에 VirtIO 및 게스트 툴 이미지 파일 업로드



참고

**ISO** 도메인은 더 이상 사용되지 않는 스토리지 도메인 유형입니다. **ISO** 업로더 툴이 더 이상 사용되지 않습니다. **Red Hat**은 관리 포털 또는 **REST API**를 사용하여 **ISO** 이미지를 데이터 도메인에 업로드하는 것이 좋습니다. 자세한 내용은 [11.8.1절. “데이터 스토리지 도메인에 이미지 업로드”](#) 을 참조하십시오.

**virtio-win.iso** 이미지 및 **VFD(Virtual Floppy Drive)** 이미지에는 **Windows** 가상 머신용 **VirtIO** 드라이버가 포함되어 있으며 **RHV-toolsSetup\_version.iso**에는 **Windows** 가상 머신용 **Red Hat Virtualization** 게스트 도구가 포함되어 있습니다. 이러한 이미지 파일은 가상 머신에 설치할 수 있는 소프트웨어를 제공하여 성능과 가용성을 향상시킵니다.

**virtio-win** 및 **RHV-toolsSetup\_version.iso** 파일의 최신 버전을 설치하고 업로드하려면 다음을 수행합니다.

1.

**Manager** 시스템에 이미지 파일을 설치합니다.

```
# yum -y install virtio-win rhv-guest-tools-iso*
```

**Manager** 시스템에 설치한 후 다음 이미지 파일은 다음과 같습니다.

- `/usr/share/virtio-win/virtio-win_amd64.vfd (symbolic link)`
- `/usr/share/virtio-win/virtio-win_servers_amd64.vfd (symbolic link)`
- `/usr/share/virtio-win/virtio-win_servers_x86.vfd (symbolic link)`
- `/usr/share/virtio-win/virtio-win_x86.vfd (symbolic link)`
- `/usr/share/virtio-win/virtio-win.iso (symbolic 링크)`
- `/usr/share/rhv-guest-tools-iso/RHV-toolsSetup_version.iso`



참고

위의 심볼릭 링크는 이름에 버전이 있는 파일을 나타냅니다. 복사하거나 **rsync**와 같은 도구를 사용할 때 심볼릭 링크의 대상을 복사하십시오.

2.

설치 중에 로컬로 생성되지 않은 **ISO** 스토리지 도메인에 이미지 파일을 업로드합니다.

```
# yum -y install virtio-win rhv-guest-tools-iso*
# engine-iso-uploader --iso-domain=ISODomain upload \
  /usr/share/virtio-win/virtio-win_amd64.vfd \
  /usr/share/virtio-win/virtio-win_servers_amd64.vfd \
  /usr/share/virtio-win/virtio-win_servers_x86.vfd \
  /usr/share/virtio-win/virtio-win_x86.vfd \
  /usr/share/virtio-win/virtio-win.iso \
  /usr/share/rhv-guest-tools-iso/RHV-toolsSetup_version.iso
```

3.

이미지 파일을 가상 머신에 연결합니다.

이제 가상 머신에서 **virtio** 드라이버 및 게스트 툴을 사용할 수 있습니다.

이미지 파일을 가상 머신에 연결하는 방법에 대한 자세한 내용은 가상 머신 관리 가이드 의 [Windows 에 게스트 에이전트, 툴 및 드라이버 설치를 참조하십시오.](#)

## 22.6. 엔진 VACUUM 툴

### 22.6.1. 엔진 Vacuum 툴

**Engine Vacuum** 툴은 테이블을 업데이트하고 종료된 행을 제거하여 **PostgreSQL** 데이터베이스를 유지 관리하므로 디스크 공간을 재사용할 수 있습니다. **VACUUM** 명령 및 해당 매개변수에 대한 정보는 [PostgreSQL 설명서를 참조하십시오.](#)

**Engine Vacuum** 명령은 **engine-vacuum** 입니다. **root** 사용자로 로그인하고 **Red Hat Virtualization** 환경에 대한 관리 자격 증명을 제공해야 합니다.

또는 **engine-setup** 명령을 사용하여 기존 설치를 사용자 지정하는 동안 **Engine Vacuum** 툴을 실행할 수 있습니다.

```
$ engine-setup
...
[ INFO ] Stage: Environment customization
...
Perform full vacuum on the engine database engine@localhost?
This operation may take a while depending on this setup health and the
configuration of the db vacuum process.
See https://www.postgresql.org/docs/10/static/sql-vacuum.html
(Yes, No) [No]:
```

**Yes** 옵션은 **Engine Vacuum** 툴을 전체 제공 정보 표시 모드로 실행합니다.

### 22.6.2. 엔진 Vacuum 모드

엔진 **Vacuum**에는 두 가지 모드가 있습니다.

#### 표준 Vacuum

자주 사용되는 표준이 권장됩니다.

표준은 테이블 및 인덱스에서 잘못된 행 버전을 제거하고 나중에 재사용할 수 있도록 공간을 표시합니다. 자주 업데이트되는 테이블은 정기적으로 유지되어야 합니다. 그러나 표준은 운영 체제로 공간을 반환하지 않습니다.

표준은 매개 변수가 없는 상태에서 현재 데이터베이스의 모든 테이블을 처리합니다.

## 전체 Vacuum

전체 준비는 일상적인 사용을 위해 권장되지 않지만 상당한 양의 공간을 테이블 내에서 회수해야 할 때만 실행해야 합니다.

전체 성공률은 테이블이 정지되지 않은 상태에서 테이블 파일의 새 복사본을 작성하여 테이블을 압축하여 운영 체제에서 공간을 회수할 수 있도록 합니다. 가득 차있는 것은 오랜 시간이 걸릴 수 있습니다.

전체 성공하려면 작업이 완료되고 이전 복사가 삭제될 때까지 테이블의 새 복사본을 위한 추가 디스크 공간이 필요합니다. 전체 설정에는 테이블에 배타적 잠금이 필요하므로 테이블의 다른 사용과 병렬로 실행할 수 없습니다.

### 22.6.3. engine-vacuum 명령의 구문

**engine-vacuum** 명령의 기본 구문은 다음과 같습니다.

```
# engine-vacuum
```

```
# engine-vacuum option
```

옵션 없이 **engine-vacuum** 명령을 실행하면 표준이 수행됩니다.

**engine-vacuum** 명령을 더 구체화하는 몇 가지 매개 변수가 있습니다.

일반 옵션

**-h --help**



**engine-vacuum** 명령을 사용하는 방법에 대한 정보를 표시합니다.

**-a**

표준 성공률을 실행하고, 데이터베이스를 분석하며, **IRQ** 통계를 업데이트합니다.

**-A**

데이터베이스를 분석하고 **IRQ** 통계를 업데이트합니다.

**-f**

가득 차있는 번거 로움을 실행합니다.

**-v**

자세한 정보 표시 모드로 실행되며 더 많은 콘솔 출력을 제공합니다.

**-t table\_name**

특정 테이블 또는 테이블입니다.

```
# engine-vacuum -f -v -t vm_dynamic -t vds_dynamic
```

## 22.7. VDSM에서 네트워크 이름 매핑 툴

### 22.7.1. VDSM 이름을 논리 네트워크 이름에 매핑

논리 네트워크의 이름이 15자를 초과하거나 비ASCII 문자를 포함하는 경우 시스템은 자동으로 호스트 식별자(**vds\_name**) 이름을 생성합니다. 네트워크 고유 식별자의 처음 13자(예: **on a1b2c3d4e5f6g**)로 구성됩니다. 이 이름은 호스트의 로그 파일에 표시됩니다. 논리적 네트워크 이름 및 자동 생성된 네트워크 이름 목록을 보려면 **/usr/share/ovirt-engine/bin/**에 있는 **VDSM-to-Network-Name** 매핑 도구를 사용합니다.

#### 절차

1.

툴을 처음 실행할 때 **Manager** 데이터베이스에 대한 읽기 권한이 있는 데이터베이스 사용자의 암호인 **PASSWORD** 환경 변수를 정의합니다. 예를 들어 다음을 실행합니다.

```
# export PASSWORD=DatabaseUserPassword
```

2.

**VDSM-to-Network-Name** 매핑 툴을 실행합니다.

```
# vdsd_to_network_name_map --user USER
```

여기서 **USER** 는 **Manager** 데이터베이스에 대한 읽기 액세스 권한이 있는 데이터베이스 사용자이며 암호는 **PASSWORD** 환경 변수에 할당됩니다.

툴은 동등한 호스트 식별자에 매핑되는 논리적 네트워크 이름 목록을 표시합니다.

### 추가 **ECDHE**

다음 플래그를 사용하여 도구를 실행할 수 있습니다.

**--host** 는 데이터베이스 서버의 호스트 이름/IP 주소입니다. 기본값은 **localhost** 입니다.

**--port** 는 데이터베이스 서버의 포트 번호입니다. 기본값은 **5432** 입니다. **--database** 는 데이터베이스의 이름입니다. 기본값은 **engine** 이며, 이는 **Manager** 데이터베이스입니다.

**--secure** 는 데이터베이스와의 보안 연결을 활성화합니다. 기본적으로 도구는 보안 연결 없이 실행됩니다.

**IV 부. 환경에 대한 정보 수집**

## 23장. RED HAT VIRTUALIZATION MANAGER에 INSIGHTS 배포

Red Hat Virtualization Manager가 설치된 기존 RHEL(Red Hat Enterprise Linux) 시스템에 Red Hat Insights를 배포하려면 다음 작업을 완료합니다.

- 시스템을 Red Hat Insights 애플리케이션에 등록합니다.
- Red Hat Virtualization 환경에서 데이터 수집을 활성화합니다.

### Red Hat Insights에 시스템 등록

Red Hat Insights 서비스와 통신하고 Red Hat Insights 콘솔에 표시되는 결과를 보려면 시스템을 등록합니다.

```
[root@server ~]# insights-client --register
```

### Red Hat Virtualization 환경에서 데이터 수집 활성화

다음 행을 포함하도록 /etc/ovirt-engine/rhv-log-collector-analyzer/rhv-log-collector-analyzer.conf 파일을 수정합니다.

```
upload-json=True
```

### Insights 콘솔에서 Insights 결과 보기

시스템 및 인프라 결과는 Insights 콘솔에서 볼 수 있습니다. 개요 탭에는 인프라에 대한 현재 위협의 대시보드 보기가 있습니다. 이 시작점에서 특정 규칙이 시스템에 어떤 영향을 미치는지 조사하거나 시스템 기반 접근 방식을 사용하여 시스템에 위협을 초래하는 모든 규칙과 일치하는지 확인할 수 있습니다.

1. 연결에 영향을 미치는 총 위협 (Critical, Important, Moderate 또는 Low)으로 규칙을 보려면 **Rule hits by severity**를 선택합니다.

또는

2. **Rule hits by category**를 선택하여 인프라에 미치는 위협 유형(사용 가능, 안정성, 성능 또는 보안)을 확인합니다.
3. 이름별 특정 규칙을 검색하거나 규칙 목록을 스크롤하여 노출된 위협, 노출된 시스템 및

**Ansible Playbook**의 가용성에 대한 고급 정보를 확인하여 수정을 자동화합니다.

4. 규칙을 클릭하여 규칙에 대한 설명을 확인하고, 관련 기술 자료 문서에서 자세히 알아보고, 영향을 받는 시스템 목록을 확인합니다.
  
5. 시스템에서 탐지된 문제 및 문제 해결 단계에 대한 특정 정보를 보려면 시스템을 클릭합니다.

24장. 로그 파일

24.1. 관리자 설치 로그 파일

표 24.1. 설치

| 로그 파일                                                                             | 설명                                                                                                                                                                                   |
|-----------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>/var/log/ovirt-engine/engine-cleanup-yyyy_mm_dd_hh_mm_ss.log</code>         | <b>engine-cleanup</b> 명령에서 로그아웃합니다. 이는 Red Hat Virtualization Manager 설치를 재설정하는 데 사용되는 명령입니다. 명령이 실행될 때마다 로그가 생성됩니다. 여러 개의 로그가 존재할 수 있도록 파일 이름에 실행 날짜 및 시간을 사용합니다.                   |
| <code>/var/log/ovirt-engine/engine-db-install-yyyy_mm_dd_hh_mm_ss.log</code>      | 엔진 데이터베이스의 생성 및 구성을 자세히 설명하는 <b>engine-setup</b> 명령에서 로그아웃합니다.                                                                                                                       |
| <code>/var/log/ovirt-engine/ovirt-engine-dwh-setup-yyyy_mm_dd_hh_mm_ss.log</code> | <b>ovirt-engine-dwh-setup</b> 명령에서 로그인합니다. 이는 보고를 위한 <b>ovirt_engine_history</b> 데이터베이스를 생성하는 데 사용되는 명령입니다. 명령이 실행될 때마다 로그가 생성됩니다. 여러 개의 로그가 동시에 존재할 수 있도록 파일 이름에 실행 날짜 및 시간을 사용합니다. |
| <code>/var/log/ovirt-engine/setup/ovirt-engine-setup-yyyymmddhhmmss.log</code>    | <b>engine-setup</b> 명령에서 로그아웃합니다. 명령이 실행될 때마다 로그가 생성됩니다. 여러 개의 로그가 동시에 존재할 수 있도록 파일 이름에 실행 날짜 및 시간을 사용합니다.                                                                           |

24.2. RED HAT VIRTUALIZATION MANAGER 로그 파일

표 24.2. 서비스 활동

| 로그 파일                                                | 설명                                                                                        |
|------------------------------------------------------|-------------------------------------------------------------------------------------------|
| <code>/var/log/ovirt-engine/engine.log</code>        | 모든 Red Hat Virtualization Manager GUI 충돌, Active Directory 조회, 데이터베이스 문제 및 기타 이벤트를 반영합니다. |
| <code>/var/log/ovirt-engine/host-deploy</code>       | Red Hat Virtualization Manager에서 배포한 호스트의 로그 파일입니다.                                       |
| <code>/var/lib/ovirt-engine/setup-history.txt</code> | Red Hat Virtualization Manager와 관련된 패키지의 설치 및 업그레이드를 추적합니다.                               |

| 로그 파일                                        | 설명                                                                                                                                                                                         |
|----------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| /var/log/httpd/ovirt-requests-log            | <p>각 요청이 걸린 시간을 포함하여 HTTPS를 통해 Red Hat Virtualization Manager에 대한 요청의 파일을 기록합니다.</p> <p>로그 파일을 /var/log/ovirt-engine/engine.log 와 비교할 때 요청을 비교할 수 있도록 <b>Correlation-Id</b> 헤더가 포함됩니다.</p> |
| /var/log/ovn-provider/ovirt-provider-ovn.log | OVN 공급자의 활동을 기록합니다. Open vSwitch 로그에 대한 자세한 내용은 <a href="#">Open vSwitch 설명서</a> 를 참조하십시오.                                                                                                 |

### 24.3. SPICE 로그 파일

**SPICE** 로그 파일은 **SPICE** 연결 문제를 해결할 때 유용합니다. **SPICE** 디버깅을 시작하려면 로그 수준을 디버깅으로 변경합니다. 그런 다음 로그 위치를 식별합니다.

게스트 시스템과 게스트 시스템 자체에 액세스하는 데 사용되는 클라이언트 모두 **SPICE** 로그 파일이 있습니다. 클라이언트 측 로그의 경우 **console.vv** 파일이 다운로드된 기본 클라이언트를 사용하여 **SPICE** 클라이언트가 시작된 경우 **remote-viewer** 명령을 사용하여 디버깅을 활성화하고 로그 출력을 생성합니다.

#### 24.3.1. Hypervisor SPICE Server를 위한 SPICE 로그

표 24.3. Hypervisor SPICE Server를 위한 SPICE 로그

| 로그 유형                   | 로그 위치                                  | 로그 수준을 변경하려면 다음을 수행합니다.                                                                                                                                                                                                |
|-------------------------|----------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 호스트/Hypervisor SPICE 서버 | /var/log/libvirt/qemu/(guest_name).log | <p>게스트를 시작하기 전에 호스트/하이퍼바이저에서 내보내기 <b>SPICE_DEBUG_LEVEL=5</b>를 실행합니다. 이 변수는 QEMU에서 구문 분석하며 시스템 전체에서 시스템 전체에 있는 모든 가상 머신의 디버깅 정보를 출력합니다. 이 명령은 클러스터의 각 호스트에서 실행해야 합니다. 이 명령은 클러스터별로 사용하지 않고 호스트별/하이퍼바이저 기준으로만 작동합니다.</p> |

#### 24.3.2. 게스트 머신용 SPICE 로그

표 24.4. 게스트 머신의 **Spice-vdagent** 로그

| 로그 유형                          | 로그 위치                                                        | 로그 수준을 변경하려면 다음을 수행합니다.                                                                                                                                                                                                                                                                                                                        |
|--------------------------------|--------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Windows 게스트                    | C:\Windows\Temp\vdagent.log<br>C:\Windows\Temp\vdservice.log | 해당 없음                                                                                                                                                                                                                                                                                                                                          |
| Red Hat Enterprise Linux Guest | <b>journalctl</b> 을 root 사용자로 사용합니다.                         | <p><b>spice-vdagentd</b> 서비스를 디버그 모드에서 실행하려면 root 사용자가 이 항목을 사용하여 <b>/etc/sysconfig/spice-vdagentd</b> 파일을 만듭니다.</p> <p><b>SPICE_VDAGENTD_EXTRA_ARGS="-d -d"</b></p> <p>디버그 모드에서 <b>spice-vdagent</b> 를 실행하려면 명령줄에서 다음을 수행합니다.</p> <pre>\$ killall -u \$USER spice-vdagent \$ spice-vdagent -x -d [-d] [   &amp; tee spice-vdagent.log ]</pre> |

**24.3.3. console.vv** 파일을 사용하여 시작된 **SPICE** 클라이언트용 **SPICE** 로그

**Linux** 클라이언트 머신의 경우:

1. **--spice-debug** 옵션을 사용하여 **remote-viewer** 명령을 실행하여 **SPICE** 디버깅을 활성화합니다. 메시지가 표시되면 연결 **URL** 을 입력합니다(예: **spice://virtual\_machine\_IP:포트** ).

```
# remote-viewer --spice-debug
```

2. **debug** 매개 변수를 사용하여 **SPICE** 클라이언트를 실행하고 **.vv** 파일을 전달하려면 **console.vv** 파일을 다운로드하고 **--spice-debug** 옵션을 사용하여 **remote-viewer** 명령을 실행하고 **console.vv** 파일의 전체 경로를 지정합니다.

```
# remote-viewer --spice-debug /path/to/console.vv
```

**Windows** 클라이언트 머신의 경우:

1. **virt-viewer 2.0-11.el7ev** 이상 버전에서 **virt-viewer.msi** 는 **virt-viewer** 및 **debug-**



**viewer.exe** 를 설치합니다.

2. **spice-debug** 인수를 사용하여 **remote-viewer** 명령을 실행하고 경로에서 콘솔로 명령을 보냅니다.

```
remote-viewer --spice-debug path\to\console.vv
```

3. 로그를 보려면 가상 머신에 연결하고 **GDB**를 실행하여 **remote-viewer**의 표준 출력 및 표준 오류를 출력하는 명령 프롬프트가 표시됩니다.

#### 24.4. 호스트 로그 파일

| 로그 파일                                     | 설명                                                                                                                                                           |
|-------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| /var/log/ECDHE                            | <b>libvirt</b> 에서 사용하는 로그 파일입니다. <b>journalctl</b> 을 사용하여 로그를 봅니다. 로그를 보려면 <i>adm</i> , <i>systemd-journal</i> 또는 <i>hammer</i> 그룹의 멤버여야 합니다.                |
| /var/log/vdsm/spm-lock.log                | 스토리지 풀 관리자 역할에 대한 리스를 가져올 수 있는 호스트 기능을 자세히 설명하는 로그 파일입니다. 호스트가 리스를 취득, 릴리스, 업데이트 또는 업데이트하지 못한 경우 로그 세부 정보입니다.                                                |
| /var/log/vdsm/vdsm.log                    | 호스트의 Manager 에이전트인 VDSM에 대한 로그 파일입니다.                                                                                                                        |
| /tmp/ovirt-host-deploy-Date.log           | 호스트가 성공적으로 배포된 후 Manager에 <b>/var/log/ovirt-engine/host-deploy/ovirt-Date-Host-Correlation_ID.log</b> 로 복사되는 호스트 배포 로그입니다.                                   |
| /var/log/vdsm/import/import-UUID-Date.log | 가져오기 실패 정보를 포함하여 KVM 호스트, VMWare 공급자 또는 RHEL 5ECDHE 호스트에서 가상 머신 가져오기를 자세히 설명하는 로그 파일입니다. <i>UUID</i> 는 가져온 가상 머신의 UUID이고 <i>Date</i> 는 가져오기가 시작된 날짜 및 시간입니다. |
| /var/log/vdsm/supervdsm.log               | 슈퍼 유저 권한으로 실행된 VDSM 작업을 기록합니다.                                                                                                                               |
| /var/log/vdsm/upgrade.log                 | VDSM은 호스트 업그레이드 중에 이 로그 파일을 사용하여 구성 변경 사항을 기록합니다.                                                                                                            |
| /var/log/vdsm/mom.log                     | VDSM의 메모리 과다 할당 관리자 활동을 기록합니다.                                                                                                                               |

#### 24.5. 호스트 로깅 서버 설정

호스트는 로그 파일을 생성 및 업데이트하고 작업 및 문제를 기록합니다. 이러한 로그 파일을 중앙에서 수집하면 디버깅이 간소화됩니다.

이 절차는 중앙 집중식 로그 서버에서 사용해야 합니다. 별도의 로깅 서버를 사용하거나 이 절차를 사용하여 **Red Hat Virtualization Manager**에서 호스트 로깅을 활성화할 수 있습니다.

### 호스트 로깅 서버 설정

1.

방화벽이 **UDP 514** 포트에서 트래픽을 허용하고 **syslog** 서비스 트래픽에 열려 있는지 확인합니다.

```
# firewall-cmd --query-service=syslog
```

출력이 밝힌 경우 다음을 사용하여 **UDP 514** 포트에서 트래픽을 허용합니다.

```
# firewall-cmd --add-service=syslog --permanent
# firewall-cmd --reload
```

2.

**syslog** 서버에 새 **.conf** 파일을 만듭니다(예: **/etc/ECDHE.d/from\_remote.conf**).

```
template(name="DynFile" type="string"
string="/var/log/%HOSTNAME%/%PROGRAMNAME%.log")
RuleSet(name="RemoteMachine"){ action(type="omfile" dynaFile="DynFile") }
Module(load="imudp")
Input(type="imudp" port="514" ruleset="RemoteMachine")
```

3.

**rsyslog** 서비스를 다시 시작하십시오.

```
# systemctl restart rsyslog.service
```

4.

하이퍼바이저에 로그인하고 **/etc/ECDHE.conf** 에서 다음 행을 추가합니다.

```
*.info;mail.none;authpriv.none;cron.none @<syslog-FQDN>:514
```

5.

하이퍼바이저에서 **rsyslog** 서비스를 다시 시작합니다.

```
# systemctl restart rsyslog.service
```

이제 가상화 호스트에서 메시지를 수신 및 저장하고 메시지를 보호하도록 중앙 집중식 로그 서버가 구성되어 있습니다.

## 24.6. OVIRT ENGINE EXTENSION LOGGER LOG4J 활성화

로거 구현에는 **ovirt-engine-extension-logger-log4j** 패키지가 필요합니다. **Red Hat Virtualization Manager**는 구현을 통해 레코드를 **log4j**에 위임합니다. **Log4j**는 **SNMP** 및 **syslog**를 포함한 다양한 기술을 위한 **appender**를 제공하는 사용자 정의 프레임워크입니다.

**oVirt Engine Extension Logger log4j**는 **engine.log** 파일을 기존 **syslog** 서버에 전달합니다. 구성 절차는 [호스트 로깅 서버 설정](#) 과 중복됩니다.

중앙 **syslog** 서버에서 이 절차를 사용하십시오. 별도의 로깅 서버를 사용하거나 이 절차를 사용하여 **Manager**에서 **syslog** 서버로 **engine.log** 파일을 전달할 수 있습니다.

### 참고

이 확장에 사용할 **syslog** 서버를 정의하려면 **/etc/ovirt-engine/extensions.d** 디렉터리로 이동하여 **Log4jLogger.properties** 파일에서 **log4j.appender.myappender.SyslogHost** 값을 편집합니다.

**syslog** 기능을 정의하려면 **/etc/ovirt-engine/extensions.d** 디렉터리로 이동하여 **Log4jLogger.properties** 파일에서 **log4j.appender.myappender.Facility** 값을 편집합니다. 예를 들면 **log4j.appender.myappender.Facility=local1** 입니다.

### oVirt Engine Extension Logger log4j 구성

1. 확장을 설치합니다.

```
# yum install ovirt-engine-extension-logger-log4j
```

2. **/etc/ovirt-engine/extensions.d/** 디렉터리에 **Log4jLogger.properties** 파일을 만들고 다음 콘텐츠를 포함합니다.

```
ovirt.engine.extension.name = log4jlogger
ovirt.engine.extension.provides = org.ovirt.engine.api.extensions.logger.Logger
ovirt.engine.extension.bindings.method = jbossmodule
ovirt.engine.extension.binding.jbossmodule.module = org.ovirt.engine-extensions.logger.log4j
ovirt.engine.extension.binding.jbossmodule.class =
org.ovirt.engineextensions.logger.log4j.Log4jLogger
log4j.rootLogger=DEBUG, myappender
log4j.appender.myappender=org.apache.log4j.net.SyslogAppender
log4j.appender.myappender.SyslogHost=localhost
log4j.appender.myappender.layout=org.apache.log4j.PatternLayout
log4j.appender.myappender.layout.ConversionPattern=[%c] %m%n
```

3.

**rsyslog**를 설치하고 구성합니다.

```
# yum install rsyslog
```

4.

**rsyslog** 트래픽을 허용하도록 **SELinux**를 구성합니다.

```
# semanage port -a -t syslogd_port_t -p udp 514
```

5.

**/etc/ECDHE.conf**를 편집하고 다음 행을 추가합니다.

```
$template TmplAuth, "/var/log/%fromhost%/secure"
$template TmplMsg, "/var/log/%fromhost%/messages"

$RuleSet remote
authpriv.* ?TmplAuth
*.info,mail.none;authpriv.none,cron.none ?TmplMsg
$RuleSet RSYSLOG_DefaultRuleset
$InputUDPServerBindRuleset remote
```

6.

다음 두 줄의 주석을 제거합니다.

```
#$ModLoad imudp
#$UDPServerRun 514
```

7.

**rsyslog** 서비스를 다시 시작하십시오.

```
# systemctl restart rsyslog.service
```

8.

방화벽이 활성화되어 활성 상태인 경우 다음 명령을 실행하여 **Firewalld**에서 **rsyslog** 포트를 여는 데 필요한 규칙을 추가합니다.

```
# firewall-cmd --permanent --add-port=514/udp  
# firewall-cmd --reload
```

9.

**Red Hat Virtualization Manager**를 다시 시작하십시오.

```
# restart ovirt-engine
```

이제 기존 **syslog** 서버에서 **engine.log** 파일을 수신하고 저장할 수 있습니다.

## 부록 A. VDSM 및 후크

## A.1. VDSM

VDSM 서비스는 **Red Hat Virtualization Manager**에서 **RHVH(Red Hat Virtualization Host)** 및 **Red Hat Enterprise Linux** 호스트를 관리하는 데 사용됩니다. VDSM은 호스트의 스토리지, 메모리 및 네트워크 리소스를 관리하고 모니터링합니다. 또한 가상 머신 생성, 통계 수집, 로그 수집 및 기타 호스트 관리 작업을 공동으로 조정합니다. VDSM은 **Red Hat Virtualization Manager**에서 관리하는 각 호스트에서 데몬으로 실행됩니다. 클라이언트의 **XML-RPC** 호출에 응답합니다. **Red Hat Virtualization Manager**는 VDSM 클라이언트로 작동합니다.

## A.2. VDSM 후크

VDSM은 후크를 통해 확장할 수 있습니다. 후크는 주요 이벤트가 발생할 때 호스트에서 실행되는 스크립트입니다. 지원되는 이벤트가 발생하면 VDSM이 `/usr/libexec/vdsm/hooks/nn_event-name/` 호스트에서 영숫자 순서로 실행 가능한 후크 스크립트를 실행합니다. 규칙에 따라 각 후크 스크립트는 파일 이름 앞에 포함된 두 자리 숫자가 할당되어 스크립트를 실행할 순서가 명확하지 않습니다. 모든 프로그래밍 언어로 후크 스크립트를 생성할 수 있지만 이 장에 포함된 예제에는 **Python**이 사용됩니다.

이벤트의 호스트에 정의된 모든 스크립트가 실행됩니다. 지정된 후크가 호스트에서 실행되는 가상 머신의 하위 집합에 대해서만 실행되어야 하는 경우, 가상 시스템과 관련된 **Custom Properties** 를 평가하여 후크 스크립트 자체에서 이 요구 사항을 처리해야 합니다.



## 주의

VDSM 후크는 **Red Hat Virtualization**의 작동을 방해할 수 있습니다. VDSM 후크의 버그로 인해 가상 머신이 충돌하고 데이터 손실이 발생할 수 있습니다. VDSM 후크는 주의해서 구현해야 하며 엄격하게 테스트해야 합니다. **Hooks API**는 새로운 기능이며 향후 상당한 변경될 수 있습니다.

## A.3. HOOKS를 사용하여 VDSM 확장

이 장에서는 이벤트 중심 후크를 사용하여 VDSM을 확장하는 방법을 설명합니다. 후크를 사용한 VDSM 확장은 실험적 기술이며 이 장에서는 숙련된 개발자를 위한 것입니다. 가상 머신에 사용자 지정 속성을 설정하면 지정된 가상 머신과 관련된 추가 매개변수를 후크 스크립트에 전달할 수 있습니다.

## A.4. 지원되는 VDSM 이벤트

표 A.1. 지원되는 VDSM 이벤트

| 이름                            | 설명                                                                                                    |
|-------------------------------|-------------------------------------------------------------------------------------------------------|
| before_vm_start               | 가상 머신이 시작되기 전에                                                                                        |
| after_vm_start                | 가상 머신이 시작된 후                                                                                          |
| before_vm_cont                | 가상 머신이 계속 진행되기 전입니다.                                                                                  |
| after_vm_cont                 | 가상 머신이 계속된 후                                                                                          |
| before_vm_pause               | 가상 머신이 일시 중지되기 전입니다.                                                                                  |
| after_vm_pause                | 가상 머신이 일시 정지된 후.                                                                                      |
| before_vm_hibernate           | 가상 머신보다 먼저.                                                                                           |
| after_vm_hibernate            | 가상 머신 후.                                                                                              |
| before_vm_dehibernate         | 가상 머신이 종료되기 전입니다.                                                                                     |
| after_vm_dehibernate          | 가상 머신 제거 후.                                                                                           |
| before_vm_migrate_source      | 가상 머신 마이그레이션 전에 마이그레이션이 발생하는 소스 호스트에서 실행합니다.                                                          |
| after_vm_migrate_source       | 가상 머신 마이그레이션 후 마이그레이션이 발생하는 소스 호스트에서 실행합니다.                                                           |
| before_vm_migrate_destination | 가상 머신 마이그레이션 전에 마이그레이션이 발생하는 대상 호스트에서 실행합니다.                                                          |
| after_vm_migrate_destination  | 가상 머신 마이그레이션 후 마이그레이션이 발생하는 대상 호스트에서 실행합니다.                                                           |
| after_vm_destroy              | 가상 머신이 종료된 후                                                                                          |
| before_vdsm_start             | 호스트에서 VDSM을 시작하기 전에 다음을 수행하십시오. <b>before_vdsm_start</b> 후크는 root 사용자로 실행되고 VDSM 프로세스의 환경을 상속하지 않습니다. |
| after_vdsm_stop               | 호스트에서 VDSM이 중지되면 다음을 수행하십시오. <b>after_vdsm_stop</b> 후크는 root 사용자로 실행되며 VDSM 프로세스의 환경을 상속하지 않습니다.      |
| before_nic_hotplug            | NIC가 가상 머신에 핫플러그되기 전에                                                                                 |
| after_nic_hotplug             | NIC가 가상 머신에 핫플러그됩니다.                                                                                  |

| 이름 | 설명 |
|----|----|
|----|----|

|                                   |                                           |
|-----------------------------------|-------------------------------------------|
| before_nic_hotunplug              | 가상 머신에서 NIC가 핫 플러그 해제되기 전에                |
| after_nic_hotunplug               | NIC가 가상 머신에서 핫 플러그 해제된 후                  |
| after_nic_hotplug_fail            | NIC를 가상 머신에 핫플러그하면 실패합니다.                 |
| after_nic_hotunplug_fail          | 가상 머신에서 NIC를 핫 플러그 해제한 후 실패합니다.           |
| before_disk_hotplug               | 디스크가 가상 머신에 핫플러그되기 전                      |
| after_disk_hotplug                | 디스크가 가상 머신에 핫플러그됩니다.                      |
| before_disk_hotunplug             | 가상 머신에서 디스크를 핫 플러그 해제하기 전                 |
| after_disk_hotunplug              | 디스크가 가상 머신에서 핫 플러그 해제된 후                  |
| after_disk_hotplug_fail           | 디스크를 가상 머신에 핫플러그하면 실패합니다.                 |
| after_disk_hotunplug_fail         | 가상 머신에서 디스크를 핫 플러그 해제한 후 실패합니다.           |
| before_device_create              | 사용자 지정 속성을 지원하는 장치를 생성하기 전에 다음을 수행합니다.    |
| after_device_create               | 사용자 지정 속성을 지원하는 장치를 생성한 후                 |
| before_update_device              | 사용자 지정 속성을 지원하는 장치를 업데이트하기 전에 다음을 수행합니다.  |
| after_update_device               | 사용자 지정 속성을 지원하는 장치를 업데이트한 후               |
| before_device_destroy             | 사용자 지정 속성을 지원하는 장치를 제거하기 전에 다음을 수행합니다.    |
| after_device_destroy              | 사용자 지정 속성을 지원하는 장치를 삭제한 후                 |
| before_device_migrate_destination | 장치 마이그레이션 전에 마이그레이션이 발생하는 대상 호스트에서 실행합니다. |
| after_device_migrate_destination  | 장치 마이그레이션 후 마이그레이션이 발생하는 대상 호스트에서 실행합니다.  |
| before_device_migrate_source      | 장치 마이그레이션 전에 마이그레이션이 발생하는 소스 호스트에서 실행합니다. |



| 이름                          | 설명                                       |
|-----------------------------|------------------------------------------|
| after_device_migrate_source | 장치 마이그레이션 후 마이그레이션이 발생하는 소스 호스트에서 실행합니다. |
| after_network_setup         | 호스트 시스템을 시작할 때 네트워크를 설정한 후               |
| before_network_setup        | 호스트 시스템을 시작할 때 네트워크를 설정하기 전에             |

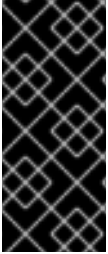
### A.5. VDSM 후크 환경

대부분의 후크 스크립트는 **vdsmd** 사용자로 실행되며 **VDSM** 프로세스의 환경을 상속합니다. 예외적으로 **before\_vdsm\_start** 및 **after\_vdsm\_stop** 이벤트에 의해 트리거되는 후크 스크립트입니다. 이러한 이벤트에 의해 트리거된 후크 스크립트는 **root** 사용자로 실행되며 **VDSM** 프로세스의 환경을 상속하지 않습니다.

### A.6. VDSM 후크 도메인 XML 오브젝트

후크 스크립트가 시작되면 **\_hook\_domxml** 변수가 환경에 추가됩니다. 이 변수는 관련 가상 시스템의 **libvirt** 도메인 XML 표현의 경로를 포함합니다. 아래에 설명된 대로 여러 후크는 이 규칙의 예외입니다. 다음 후크의 **\_hook\_domxml** 변수에는 가상 머신이 아닌 **NIC**의 XML 표현이 포함되어 있습니다.

- **\*\_nic\_hotplug\_\***
- **\*\_nic\_hotunplug\_\***
- **\*\_update\_device**
- **\*\_device\_create**
- **\*\_device\_migrate\_\***



## 중요

**before\_migration\_destination** 및 **before\_dehibernation** 후크는 현재 소스 호스트에서 도메인의 XML을 수신합니다. 대상에 있는 도메인의 XML에는 다양한 차이점이 있습니다.

**libvirt** 도메인 XML 형식은 **VDSM**에서 가상 머신을 정의하는 데 사용됩니다. **libvirt** 도메인 XML 형식에 대한 자세한 내용은 <http://libvirt.org/formatdomain.html>에서 확인할 수 있습니다. 가상 시스템의 **UUID**는 도메인 XML에서 추론할 수 있지만 환경 변수 **vmld**로도 사용할 수 있습니다.

## A.7. 사용자 정의 속성 정의

**Red Hat Virtualization Manager**에서 승인하고 사용자 정의 후크로 전달되는 사용자 정의 속성은 **engine-config** 명령을 사용하여 정의합니다. **Red Hat Virtualization Manager**가 설치된 호스트에서 이 명령을 **root** 사용자로 실행합니다.

**UserDefinedVMProperties** 및 **CustomDeviceProperties** 구성 키는 지원되는 사용자 지정 속성의 이름을 저장하는 데 사용됩니다. 이름이 지정된 각 사용자 지정 속성에 유효한 값을 정의하는 정규식도 이러한 구성 키에 포함됩니다.

여러 사용자 지정 속성은 **Semi-colon**으로 구분됩니다. 구성 키를 설정할 때 포함된 기존 값을 덮어씁니다. 새 사용자 지정 속성과 기존 사용자 지정 속성을 결합하는 경우 키 값을 설정하는 데 사용되는 명령의 모든 사용자 지정 속성을 포함해야 합니다.

구성 키가 업데이트되면 새 값을 적용하려면 **ovirt-engine** 서비스를 다시 시작해야 합니다.

## 예 A.1. 가상 머신 속성 - 스마트 카드 사용자 정의 속성 정의

1.

다음 명령을 사용하여 **UserDefinedVMProperties** 구성 키로 정의된 기존 사용자 지정 속성을 확인합니다.

```
# engine-config -g UserDefinedVMProperties
```

아래 출력에서 볼 수 있듯이 사용자 지정 속성 메모리는 이미 정의되어 있습니다. 정규식 **^[0-9]+\$**를 사용하면 사용자 지정 속성에 숫자 문자만 포함됩니다.

```
# engine-config -g UserDefinedVMProperties
UserDefinedVMProperties: version: 3.6
UserDefinedVMProperties: version: 4.0
```

```
UserDefinedVMProperties : memory=^[0-9]+$ version: 4.0
```

2.

메모리 사용자 지정 속성은 이미 **UserDefinedVMProperties** 구성 키에 정의되어 있으므로 새 사용자 지정 속성을 추가해야 합니다. 추가 사용자 지정 속성인 **smartcard** 가 설정 키 값에 추가됩니다. 새로운 사용자 지정 속성은 **true** 또는 **false** 값을 보유할 수 있습니다.

```
# engine-config -s UserDefinedVMProperties='memory=^[0-9]+$;smartcard=^(true|false)$'
--cver=4.0
```

3.

**UserDefinedVMProperties** 구성 키로 정의한 사용자 정의 속성이 올바르게 업데이트되었는지 확인합니다.

```
# engine-config -g UserDefinedVMProperties
UserDefinedVMProperties: version: 3.6
UserDefinedVMProperties: version: 4.0
UserDefinedVMProperties : memory=^[0-9]+$;smartcard=^(true|false)$ version: 4.0
```

4.

마지막으로 구성 변경을 적용하려면 **ovirt-engine** 서비스를 다시 시작해야 합니다.

```
# systemctl restart ovirt-engine.service
```

#### 예 A.2. 장치 속성 - 인터페이스 사용자 정의 속성 정의

1.

다음 명령을 사용하여 **CustomDeviceProperties** 구성 키로 정의한 기존 사용자 지정 속성을 확인합니다.

```
# engine-config -g CustomDeviceProperties
```

아래 출력에서 볼 수 있듯이 사용자 지정 속성은 아직 정의되지 않았습니다.

```
# engine-config -g CustomDeviceProperties
CustomDeviceProperties: version: 3.6
CustomDeviceProperties: version: 4.0
```

2.

인터페이스 사용자 지정 속성이 아직 존재하지 않으므로 그대로 추가할 수 있습니다. 이 예에서 **speed** 하위 속성의 값은 **0**에서 **99999** 범위로 설정되고 **duplex** 하위 속성의 값은 전체 또는 반의 선택으로 설정됩니다.

```
# engine-config -s CustomDeviceProperties="{type=interface;prop={speed=^[0-9]{1,5}$;duplex=^(full|half)$}}" --cver=4.0
```

3.

**CustomDeviceProperties** 설정 키로 정의한 사용자 정의 속성이 올바르게 업데이트되었는지 확인합니다.

```
# engine-config -g CustomDeviceProperties
UserDefinedVMProperties: version: 3.6
UserDefinedVMProperties: version: 4.0
UserDefinedVMProperties : {type=interface;prop={speed=^[0-9]{1,5}$;duplex=^(full|half)$}} version: 4.0
```

4.

마지막으로 구성 변경을 적용하려면 **ovirt-engine** 서비스를 다시 시작해야 합니다.

```
# systemctl restart ovirt-engine.service
```

### A.8. 가상 머신 사용자 정의 속성 설정

**Red Hat Virtualization Manager**에 사용자 정의 속성이 정의되면 가상 머신에서 사용자 지정 속성을 설정할 수 있습니다. 사용자 지정 속성은 새 가상 머신의 사용자 지정 속성 탭에서 설정되고 관리 포털의 가상 머신 편집 창에 설정됩니다.

**Run Virtual Machine(s)** 대화 상자에서 사용자 지정 속성을 설정할 수도 있습니다. **Run Virtual Machine(s)** (가상 머신 실행) 대화 상자에서 설정된 사용자 지정 속성은 다음에 종료될 때까지 가상 머신에만 적용됩니다.

사용자 정의 속성 탭에서는 정의된 사용자 지정 속성 목록에서 선택할 수 있는 기능을 제공합니다. 사용자 지정 속성 키를 선택하면 해당 키의 값을 입력할 수 있는 추가 필드가 표시됩니다. **+** 버튼을 클릭하고 **-** 버튼을 클릭하여 키/값 쌍을 추가합니다.

### A.9. VDSM 후크에서 가상 머신 사용자 정의 속성 평가

후크 스크립트를 호출할 때 가상 머신의 **Custom Properties** 필드에 설정된 각 키는 환경 변수로 추가됩니다. **Custom Properties** 필드의 유효성을 확인하는 데 사용되는 정규식에서는 몇 가지 보호 기능을 제공하지만 제공된 입력이 예상과 일치하는지 확인해야 합니다.

#### 예 A.3. 사용자 정의 속성 평가

이 짧은 **Python** 예제에서는 사용자 지정 속성 **key1** 이 있는지 확인합니다. 사용자 지정 속성이 설정되면 값이 표준 오류로 인쇄됩니다. 사용자 지정 속성이 설정되지 않으면 작업이 수행되지 않습니다.

```
#!/usr/bin/python

import os
import sys

if os.environ.has_key('key1'):
    sys.stderr.write('key1 value was : %s\n' % os.environ['key1'])
else:
    sys.exit(0)
```

## A.10. VDSM 후크 모듈 사용

VDSM에는 VDSM 후크 스크립트에 대한 도움말 기능을 제공하는 **Python** 후크 모듈과 함께 제공됩니다. 이 모듈은 예제로 제공되며 **Python**으로 작성된 **VDSM** 후크에만 관련이 있습니다.

후크 모듈은 가상 머신의 **libvirt XML**을 **ECDHE** 오브젝트로 읽을 수 있도록 지원합니다. 후크 스크립트는 **Python**의 빌드된 **xml.dom** 라이브러리(<http://docs.python.org/release/2.6/library/xml.dom.html>)를 사용하여 오브젝트를 조작할 수 있습니다.

그런 다음, 수정된 오브젝트를 **hooking** 모듈을 사용하여 **libvirt XML**에 다시 저장할 수 있습니다. **hooking** 모듈은 후크 개발을 지원하기 위해 다음 기능을 제공합니다.

표 A.2. 후크 모듈 함수

| 이름                  | 인수              | 설명                                         |
|---------------------|-----------------|--------------------------------------------|
| <b>tobool</b>       | string          | 문자열 "true" 또는 "false"를 부울 값으로 변환합니다.       |
| <b>read_domxml</b>  | -               | 가상 머신의 libvirt XML을 ECDHE 개체로 읽습니다.        |
| <b>write_domxml</b> | FlexVolume 오브젝트 | DestinationRule 개체에서 가상 머신의 libvirt XML 쓰기 |

## A.11. VDSM 후크 실행

**before\_vm\_start** 스크립트는 **libvirt**에 도달하기 전에 가상 머신의 **VDSM** 정의를 변경하기 위해 도메인 **XML**을 편집할 수 있습니다. 그렇게 할 때 주의를 기울여야 합니다. 후크 스크립트는 **VDSM** 작업을 중단할 가능성이 있으며 버그 스크립트로 인해 **Red Hat Virtualization** 환경이 중단될 수 있습니다. 특히 도메인의 **UUID**를 변경하지 말고 충분한 배경 지식이 없는 도메인에서 장치를 제거하지 않도록 합니다.

**before\_vdsm\_start** 및 **after\_vdsm\_stop** 후크 스크립트는 모두 **root** 사용자로 실행됩니다. 권한 에스컬레이션에 **sudo** 명령을 사용하려면 시스템에 대한 루트 액세스 권한이 필요한 기타 후크 스크립트를 작성해야 합니다. 이를 지원하려면 **vdsm** 사용자가 암호를 다시 입력하지 않고 **sudo** 를 사용할 수 있도록 **/etc/sudoers** 를 업데이트해야 합니다. 후크 스크립트가 대화형으로 실행되므로 이 작업이 필요합니다.

#### 예 A.4. VDSM 후크의 **sudo** 설정

이 예제에서는 **vdsm** 사용자가 **/bin/chown** 명령을 **root** 로 실행할 수 있도록 **sudo** 명령이 구성됩니다.

1. 가상화 호스트에 **root** 로 로그인합니다.
2. 텍스트 편집기에서 **/etc/sudoers** 파일을 엽니다.
3. 다음 줄을 파일에 추가합니다.

```
vdsm ALL=(ALL) NOPASSWD: /bin/chown
```

이는 **vdsm** 사용자가 **/bin/chown** 명령을 **root** 사용자로 실행할 수 있음을 나타냅니다. **NOW D** 매개변수는 **sudo** 를 호출할 때 암호를 입력하라는 메시지가 표시되지 않음을 나타냅니다.

이 설정을 변경하면 **VDSM** 후크가 **sudo** 명령을 사용하여 **/bin/chown** 을 **root** 로 실행할 수 있습니다. 이 **Python** 코드는 **sudo** 를 사용하여 **/bin/chown** 파일을 **/my\_file** 파일에서 **root** 로 실행합니다.

```
retcode = subprocess.call(["/usr/bin/sudo", "/bin/chown", "root", "/my_file"])
```

후크 스크립트의 표준 오류 스트림은 **VDSM**의 로그에 수집됩니다. 이 정보는 후크 스크립트를 디버깅하는 데 사용됩니다.

#### A.12. VDSM 후크 반환 코드

후크 스크립트는 표 A.3. “후크 반환 코드”에 표시된 반환 코드 중 하나를 반환해야 합니다. 반환 코드는 **VDSM**에서 추가 후크 스크립트를 처리하는지 여부를 결정합니다.

##### 표 A.3. 후크 반환 코드

| 코드 | 설명                                |
|----|-----------------------------------|
| 0  | 후크 스크립트가 성공적으로 종료되었습니다.           |
| 1  | 후크 스크립트가 실패하여 다른 후크를 처리해야 함       |
| 2  | 후크 스크립트가 실패하여 추가 후크를 처리해서는 안 됩니다. |
| >2 | reserved                          |

### A.13. VDSM 후크 예

이 섹션에 제공된 후크 스크립트는 **Red Hat**에서 엄격하게 지원하지 않습니다. 소스에 관계없이 시스템에 설치하는 모든 후크 스크립트가 환경에 대해 철저하게 테스트되었는지 확인해야 합니다.

#### 예 A.5. NUMA 노드 튜닝

제품 이름:

이 후크 스크립트를 사용하면 **numaset** 사용자 지정 속성을 기반으로 **NUMA** 호스트에서 메모리 할당을 조정할 수 있습니다. 사용자 지정 속성이 설정되지 않은 경우 작업이 수행되지 않습니다.

구성 문자열:

```
numaset=^(interleave|strict|preferred):[^\?]\d+(-\d+)?(,[^\?]\d+(-\d+)?)*$
```

사용되는 정규식을 사용하면 지정된 가상 시스템의 **numaset** 사용자 지정 속성을 사용하여 할당 모드(연장, 엄격한 기본설정) 및 사용할 노드를 모두 지정할 수 있습니다. 두 값은 콜론(:)으로 구분됩니다. 정규식을 사용하면 다음과 같이 노드 세트를 지정할 수 있습니다.

- 특정 노드(**numaset=strict:1**, 노드 1만 사용됨)를 지정하거나
- 노드 범위가 사용됨(**numaset=strict:1-4**, 노드 1~4가 사용됨)
- 특정 노드를 사용하지 않는 경우(**numaset=strict:^3**, 노드 3이 사용되지 않음을 지정함) 또는

- 위의 쉽표로 구분된 조합(**numaset=strict:1-4,6**)은 노드 **1~4** 및 **6**을 사용하도록 지정합니다.

스크립트:

**/usr/libexec/vdsm/hooks/before\_vm\_start/50\_numa**

```
#!/usr/bin/python

import os
import sys
import hooking
import traceback

'''
numa hook
=====
add numa support for domain xml:

<numatune>
  <memory mode="strict" nodeset="1-4,^3" />
</numatune>

memory=interleave|strict|preferred

numaset="1" (use one NUMA node)
numaset="1-4" (use 1-4 NUMA nodes)
numaset="^3" (don't use NUMA node 3)
numaset="1-4,^3,6" (or combinations)

syntax:
  numa=strict:1-4
'''

if os.environ.has_key('numa'):
    try:
        mode, nodeset = os.environ['numa'].split(':')

        domxml = hooking.read_domxml()

        domain = domxml.getElementsByTagName('domain')[0]
        numas = domxml.getElementsByTagName('numatune')

        if not len(numas) > 0:
            numatune = domxml.createElement('numatune')
            domain.appendChild(numatune)

            memory = domxml.createElement('memory')
```



```
memory.setAttribute('mode', mode)
memory.setAttribute('nodeset', nodeset)
numatune.appendChild(memory)

hooking.write_domxml(domxml)
else:
    sys.stderr.write('numa: numa already exists in domain xml')
    sys.exit(2)
except:
    sys.stderr.write('numa: [unexpected error]: %s\n' % traceback.format_exc())
    sys.exit(2)
```

부록 B. 사용자 정의 네트워크 속성

B.1. BRIDGE\_OPTS 매개변수 설명

표 B.1. bridge\_opts 매개변수

| 매개변수                        | 설명                                                                                                                                                                                          |
|-----------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| forward_delay               | 브릿지가 청취 및 학습 상태에 소비되는 시간을 설정합니다. 이 시점에서 전환 루프가 발견되지 않으면 브릿지가 전달 상태로 들어갑니다. 이를 통해 정상적인 네트워크 작동 전에 네트워크의 트래픽 및 레이아웃을 검사할 수 있습니다.                                                              |
| gc_timer                    | 전달 데이터베이스가 시간 초과 항목이 확인되고 지워지는 가비지 컬렉션 시간을 역초 단위로 설정합니다. Sets the garbage collection time, in deciseconds, after which the forwarding database is checked and cleared of timed-out entries. |
| group_addr                  | 일반 쿼리를 보낼 때 0으로 설정합니다. 그룹별 쿼리 또는 그룹 및 소스별 쿼리를 보낼 때 IP 멀티캐스트 주소로 설정합니다.                                                                                                                      |
| group_fwd_mask              | 브릿지가 링크 로컬 그룹 주소를 전달하도록 활성화합니다. 이 값을 기본값에서 변경하면 비표준 브리징 동작이 허용됩니다.                                                                                                                          |
| hash_elasticity             | 해시 테이블에서 허용되는 최대 체인 길이입니다. 다음 새 멀티캐스트 그룹이 추가될 때까지 적용되지 않습니다. 재충전 후 이를 충족할 수 없는 경우 해시 충돌이 발생하고 스누핑이 비활성화됩니다.                                                                                 |
| hash_max                    | 해시 테이블의 최대 버킷 양입니다. 이는 즉시 적용되며 현재 멀티캐스트 그룹 항목 수보다 작은 값으로 설정할 수 없습니다. 값은 두 가지의 힘이어야 합니다.                                                                                                     |
| hello_time                  | 네트워크 토폴로지에서 'hello' 메시지를 보내는 사이에 시간 간격을 초 단위로 설정합니다. 이 브릿지가 Spanning Tree 루트 브릿지인 경우에만 적용됩니다.                                                                                               |
| hello_timer                 | time, in deciseconds, 마지막 'hello' 메시지가 전송되었기 때문입니다.                                                                                                                                         |
| max_age                     | 해당 브릿지가 dead 것으로 간주되고 takeover가 시작되기 전에 다른 루트 브리지에서 'hello' 메시지를 수신하도록 최대 시간을 deciseconds로 설정합니다.                                                                                           |
| multicast_last_member_count | 호스트에서 'leave group' 메시지를 수신한 후 멀티캐스트 그룹에 전송된 '최신 멤버' 쿼리 수를 설정합니다.                                                                                                                           |

| 매개변수                              | 설명                                                                                                                                                                                                                                                                                                                                                                                                            |
|-----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| multicast_last_member_interval    | 'last member' 쿼리 사이에 시간(종료)을 설정합니다.                                                                                                                                                                                                                                                                                                                                                                           |
| multicast_membership_interval     | 멀티 캐스트 트래픽 전송을 중지하기 전에 브리지가 멀티 캐스트 그룹의 멤버로부터 청취하기 위해 대기할 시간을 초 단위로 설정합니다.                                                                                                                                                                                                                                                                                                                                     |
| multicast_querier                 | 브리지가 멀티 캐스트 쿼리어를 적극적으로 실행할지 여부를 설정합니다. 브리지가 다른 네트워크 호스트에서 'multicast 호스트 멤버십' 쿼리를 수신하면 쿼리가 수신되는 시간과 멀티 캐스트 쿼리 간격 시간을 기준으로 해당 호스트가 추적됩니다. 나중에 브리지에서 해당 멀티 캐스트 멤버십에 대한 트래픽을 전달하거나 쿼리 멀티 캐스트 라우터와 통신하는 경우 이 타이머는 대기열의 유효성을 확인합니다. 유효한 경우, 멀티 캐스트 트래픽이 브리지의 기존 멀티 캐스트 멤버십 테이블을 통해 전달됩니다. 더 이상 유효하지 않은 경우 트래픽이 모든 브리지 포트를 통해 전송됩니다. Broadcast 도메인을 통해 트래픽이 전송되며 멀티 캐스트 멤버십은 성능 향상을 위해 하나 이상의 멀티 캐스트 쿼리어를 실행해야 합니다. |
| multicast_querier_interval        | 호스트에서 수신된 마지막 'multicast 호스트 멤버십' 쿼리 사이에 해당 시간이 계속 유효한지 확인하기 위해 밀리초 단위로 최대 시간을 설정합니다.                                                                                                                                                                                                                                                                                                                         |
| multicast_query_use_ifaddr        | 부울. 기본값은 '0'이며, 이 경우 querier는 IPv4 메시지의 소스 주소로 0.0.0.0을 사용합니다. 이 값을 변경하면 브리지 IP가 소스 주소로 설정됩니다.                                                                                                                                                                                                                                                                                                                |
| multicast_query_interval          | 브리지에서 보낸 쿼리 메시지 간에 멀티 캐스트 멤버십의 유효성을 보장하기 위해 시간(초)을 설정합니다. 현재 또는 브릿지에서 해당 멤버십에 대한 멀티 캐스트 쿼리를 전송하도록 요청되는 경우, 브리지는 점검 및 multicast_query_interval을 요청하는 시간에 따라 자체 멀티 캐스트 쿼리어 상태를 확인합니다. 이 멤버십에 대한 멀티 캐스트 쿼리가 마지막 multicast_query_interval 내에서 전송된 경우 다시 전송되지 않습니다.                                                                                                                                                |
| multicast_query_response_interval | 시간 간격(/ciseconds)에서 호스트는 쿼리가 전송된 후 쿼리에 응답할 수 있습니다.Must는 multicast_query_interval 값보다 작거나 동일합니다.                                                                                                                                                                                                                                                                                                               |

| 매개변수                             | 설명                                                                                                                                                                                                   |
|----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| multicast_router                 | 멀티 캐스트 라우터가 연결된 포트를 활성화하거나 비활성화할 수 있습니다. 하나 이상의 멀티 캐스트 라우터가 있는 포트는 모든 멀티 캐스트 트래픽을 수신합니다. 값이 0이면 완전히 비활성화되고 값이 1이면 시스템이 쿼리를 기반으로 라우터의 존재를 자동으로 감지할 수 있으며, 2의 값은 포트에서 모든 멀티 캐스트 트래픽을 항상 수신할 수 있습니다.    |
| multicast_snooping               | 스누핑이 활성화되었는지 여부를 전환합니다. 스누핑을 사용하면 브리지가 라우터와 호스트 간의 네트워크 트래픽을 수신 대기하여 적절한 링크로 멀티 캐스트 트래픽을 필터링할 수 있습니다. 이 옵션을 사용하면 해시 충돌로 인해 자동으로 비활성화된 경우 사용자가 스누핑을 다시 활성화할 수 있지만 해시 충돌이 해결되지 않으면 스노핑이 다시 활성화되지 않습니다. |
| multicast_startup_query_count    | 멤버십 정보를 결정하기 위해 시작시 전송되는 쿼리 수를 설정합니다.                                                                                                                                                                |
| multicast_startup_query_interval | 시작 시 전송된 쿼리 간에 멤버십 정보를 확인하기 위해 시간을 설정합니다.                                                                                                                                                            |

**B.2. RED HAT VIRTUALIZATION MANAGER를 ETHTOOL을 사용하도록 설정하는 방법**

관리 포털에서 호스트 네트워크 인터페이스 카드에 대한 **ethtool** 속성을 구성할 수 있습니다. **ethtool\_opts** 키는 기본적으로 사용할 수 없으며 엔진 구성 도구를 사용하여 **Manager**에 추가해야 합니다. 호스트에 필요한 **VDSM** 후크 패키지도 설치해야 합니다.

**Manager에 ethtool\_opts 키 추가**

1. **Manager**에서 다음 명령을 실행하여 키를 추가합니다.
 

```
# engine-config -s UserDefinedNetworkCustomProperties=ethtool_opts=. * --cver=4.0
```
2. **ovirt-engine** 서비스를 다시 시작합니다.
 

```
# systemctl restart ovirt-engine.service
```
3. **ethtool** 속성을 구성하려는 호스트에서 **VDSM** 후크 패키지를 설치합니다. 패키지는 **Red Hat Virtualization Host**에서 기본적으로 사용할 수 있지만 **Red Hat Enterprise Linux** 호스트에

설치해야 합니다.

```
# yum install vdsm-hook-ethtool-options
```

이제 관리 포털에서 **ethtool\_opts** 키를 사용할 수 있습니다. 논리 네트워크에 **ethtool** 속성을 적용하려면 **9.4.2절. “호스트 네트워크 인터페이스 편집 및 호스트에 논리 네트워크 할당”** 를 참조하십시오.

### B.3. FCoE를 사용하도록 RED HAT VIRTUALIZATION MANAGER 설정 방법

관리 포털에서 호스트 네트워크 인터페이스 카드에 대한 **FCoE(Fibre Channel over Ethernet)** 속성을 구성할 수 있습니다. **fcoe** 키는 기본적으로 사용할 수 없으며 엔진 구성 도구를 사용하여 **Manager**에 추가해야 합니다. 다음 명령을 실행하여 **fcoe** 가 이미 활성화되어 있는지 확인할 수 있습니다.

```
# engine-config -g UserDefinedNetworkCustomProperties
```

호스트에 필요한 **VDSM** 후크 패키지도 설치해야 합니다. 호스트의 **FCoE** 카드에 따라 특수 구성이 필요할 수도 있습니다. **Red Hat Enterprise Linux** 스토리지 관리 가이드에서 **파이버 채널 over 이더넷 인터페이스 구성** 을 참조하십시오.

관리자에 **fcoe** 키 추가

1.

**Manager**에서 다음 명령을 실행하여 키를 추가합니다.

```
# engine-config -s UserDefinedNetworkCustomProperties='fcoe=^((enable|dcb|auto_vlan)=(yes|no),?)*$'
```

2.

**ovirt-engine** 서비스를 다시 시작합니다.

```
# systemctl restart ovirt-engine.service
```

3.

**FCoE** 속성을 구성하려는 각 **Red Hat Enterprise Linux** 호스트에 **VDSM** 후크 패키지를 설치합니다. 이 패키지는 **RHVH(Red Hat Virtualization Host)**에서 기본적으로 사용할 수 있습니다.

```
# yum install vdsm-hook-fcoe
```

이제 **fcoe** 키가 관리 포털에서 사용할 수 있습니다. 논리 네트워크에 **FCoE** 속성을 적용하려면 **9.4.2**

절. “호스트 네트워크 인터페이스 편집 및 호스트에 논리 네트워크 할당” 를 참조하십시오.

## 부록 C. RED HAT VIRTUALIZATION 사용자 인터페이스 플러그인

### C.1. RED HAT VIRTUALIZATION 사용자 인터페이스 플러그인

**Red Hat Virtualization**은 비표준 기능을 노출하는 플러그인을 지원합니다. 따라서 **Red Hat Virtualization** 관리 포털을 사용하여 다른 시스템과 쉽게 통합할 수 있습니다. 각 인터페이스 플러그인은 **Red Hat Virtualization**과 함께 사용하기 위해 패키징화 및 배포할 수 있는 일련의 사용자 인터페이스 확장 기능을 나타냅니다.

**Red Hat Virtualization**의 사용자 인터페이스 플러그인은 **JavaScript** 프로그래밍 언어를 사용하여 클라이언트에서 직접 관리 포털과 통합됩니다. 플러그인은 관리 포털에서 호출하고 웹 브라우저의 **JavaScript** 런타임에서 실행됩니다. 사용자 인터페이스 플러그인은 **JavaScript** 언어 및 해당 라이브러리를 사용할 수 있습니다.

런타임 중 주요 이벤트에서 관리 포털은 **Administration-Portal-to-plugin-in** 통신을 나타내는 이벤트 처리기 함수를 통해 개별 플러그인을 호출합니다. 관리 포털에서 여러 이벤트-**handler** 기능을 지원하지만 플러그인은 구현에만 관심이 있는 함수를 선언합니다. 각 플러그인은 관리 포털에서 플러그인을 사용하기 전에 관련 이벤트 처리기 기능을 플러그인 부트스트랩 시퀀스의 일부로 등록해야 합니다.

관리 포털에서는 개별 플러그인이 사용할 수 있는 글로벌(**top-level**) **pluginApi JavaScript** 오브젝트로 플러그인 **API**를 노출합니다. 각 플러그인은 별도의 **pluginApi** 인스턴스를 가져와서 관리 포털에서 플러그인의 라이프사이클과 관련하여 각 플러그인의 **API** 기능 호출을 제어할 수 있습니다.

### C.2. RED HAT VIRTUALIZATION USER INTERFACE PLUGIN LIFECYCLE

#### C.2.1. Red Hat Virtualization 사용자 인터페이스 플러그인 라이프 사이클

사용자 인터페이스 플러그인의 기본 라이프 사이클은 다음 세 단계로 나뉩니다.

- 플러그인 검색.
- 플러그인 로드.
- 플러그인 부트스트랩.

#### C.2.2. Red Hat Virtualization 사용자 인터페이스 플러그인 검색

플러그인 설명자 생성은 플러그인 검색 프로세스의 첫 번째 단계입니다. 플러그인 설명자에는 중요한 플러그인 메타데이터와 선택적 기본 플러그인별 구성이 포함되어 있습니다.

관리 포털 **HTML** 페이지 요청(**HTTP GET**)을 처리하는 과정의 일환으로 사용자 인터페이스 플러그인 인프라는 로컬 파일 시스템에서 플러그인 설명자를 검색하고 로드하려고 합니다. 각 플러그인 설명자에 대해 인프라는 기본 플러그인 특정 구성(있는 경우) 및 플러그인 런타임 동작을 재정의하는 데 사용되는 해당 플러그인 사용자 구성을 로드하려고 합니다. 플러그인 사용자 구성은 선택 사항입니다. 설명자 및 해당 사용자 구성 파일을 로드한 후 **oVirt Engine**은 사용자 인터페이스 플러그인 데이터를 집계하여 런타임 평가를 위해 관리 포털 **HTML** 페이지에 삽입합니다.

기본적으로 플러그인 설명자는 **\$ENGINE\_USR/ui-plug-ins**에 있으며, **oVirt Engine** 로컬 구성에 정의된 대로 **ENGINE\_USR=/usr/share/ovirt-engine**의 기본 매핑이 사용됩니다. 플러그인 설명자는 **JSON** 형식 사양을 준수할 것으로 예상되지만 플러그인 설명자는 **JSON** 형식 사양 외에도 **Java/C++** 스타일 주석(**\*** 및 **//**의 둘 다)을 허용합니다.

기본적으로 플러그인 사용자 구성 파일은 **\$ENGINE\_ETC/ui-plug-ins**에 있으며, **oVirt Engine** 로컬 구성에 정의된 대로 **ENGINE\_ETC=/etc/ovirt-engine**의 기본 매핑을 사용합니다. 플러그인 사용자 구성 파일은 플러그인 설명자와 동일한 콘텐츠 형식 규칙을 준수해야 합니다.



#### 참고

플러그인 사용자 구성 파일은 일반적으로 **<descriptorFileName>-config.json** 이름 지정 규칙을 따릅니다.

### C.2.3. Red Hat Virtualization 사용자 인터페이스 플러그인 로드

플러그인이 검색되고 해당 데이터가 관리 포털 **HTML** 페이지에 내장된 후 관리 포털의 일부로 플러그인을 로드하려고 합니다(애플리케이션 시작의 일부로 로드하지 않는 경우).

검색된 각 플러그인의 경우 관리 포털은 호스트 페이지를 로드하는 데 사용되는 **HTML iframe** 요소를 생성합니다. 플러그인 호스트 페이지는 플러그인 부트스트랩 프로세스를 시작하는 데 필요합니다. 이 부트스트랩 프로세스는 플러그인의 **iframe** 요소 컨텍스트에서 플러그인 코드를 평가하는 데 사용됩니다. 사용자 인터페이스 플러그인 인프라는 로컬 파일 시스템의 플러그인 리소스 파일(예: 플러그인 호스트 페이지) 제공을 지원합니다. 플러그인 호스트 페이지는 **iframe** 요소로 로드되고 플러그인 코드가 평가됩니다. 플러그인 코드를 평가한 후 플러그인은 플러그인 **API**를 통해 관리 포털과 통신합니다.

### C.2.4. Red Hat Virtualization 사용자 인터페이스 플러그인 부트 스트랩

일반적인 플러그인 부트스트랩 시퀀스는 다음 단계로 구성됩니다.



플러그인 **BootstrapECDHE**

1. 지정된 플러그인의 **pluginApi** 인스턴스 가져오기
2. 런타임 플러그인 구성 오브젝트 가져오기 (선택 사항)
3. 관련 이벤트 처리기 함수 등록
4. 플러그인 초기화를 진행하도록 **UI** 플러그인 인프라에 알립니다.

다음 코드 조각은 위에서 언급한 단계를 보여줍니다.

```
// Access plug-in API using 'parent' due to this code being evaluated within the context of an iframe
// element.
// As 'parent.pluginApi' is subject to Same-Origin Policy, this will only work when WebAdmin HTML
// page and plug-in
// host page are served from same origin. WebAdmin HTML page and plug-in host page will always
// be on same origin
// when using UI plug-in infrastructure support to serve plug-in resource files.
var api = parent.pluginApi('MyPlugin');

// Runtime configuration object associated with the plug-in (or an empty object).
var config = api.configObject();

// Register event handler function(s) for later invocation by UI plug-in infrastructure.
api.register({
  // Uilnit event handler function.
  Uilnit: function() {
    // Handle Uilnit event.
    window.alert('Favorite music band is ' + config.band);
  }
});

// Notify UI plug-in infrastructure to proceed with plug-in initialization.
api.ready();
```

**C.3.** 사용자 인터페이스 플러그인 관련 파일 및 해당 위치표 **C.1.** UI 플러그인 관련 파일 및 해당 위치

| 파일                     | 위치                                                                                       | marks                                      |
|------------------------|------------------------------------------------------------------------------------------|--------------------------------------------|
| 플러그인 설명자 파일(meta-data) | <code>/usr/share/ovirt-engine/ui-plugins/my-plugin.json</code>                           |                                            |
| 플러그인 사용자 구성 파일         | <code>/etc/ovirt-engine/ui-plugins/my-plugin-config.json</code>                          |                                            |
| 플러그인 리소스 파일            | <code>/usr/share/ovirt-engine/ui-plugins/&lt;resourcePath&gt;/PluginHostPage.html</code> | <resourcePath>는 플러그인 설명자의 해당 특성에 의해 정의됩니다. |

#### C.4. 사용자 인터페이스 플러그인 배포 예

다음 지침에 따라 **Red Hat Virtualization Manager** 관리 포털에 로그인할 때 **Hello World!** 프로그램을 실행하는 사용자 인터페이스 플러그인을 생성합니다.

#### **Hello World**를 배포하십시오. **Plug-in**

1. `/usr/share/ovirt-engine/ui-plugins/helloECDHE.json`의 **Manager**에 다음 파일을 생성하여 플러그인 설명자를 생성합니다.

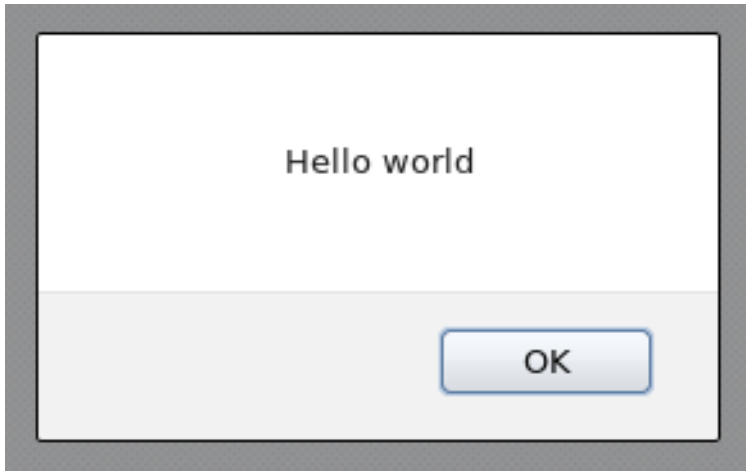
```
{
  "name": "HelloWorld",
  "url": "/ovirt-engine/webadmin/plugin/HelloWorld/start.html",
  "resourcePath": "hello-files"
}
```

2. `/usr/share/ovirt-engine/ui-plugins/hello-files/start.html`의 **Manager**에 다음 파일을 생성하여 플러그인 호스트 페이지를 생성합니다.

```
<!DOCTYPE html><html><head>
<script>
  var api = parent.pluginApi("HelloWorld");
  api.register({
    Uilnit: function() { window.alert('Hello world'); }
  });
  api.ready();
</script>
</head><body></body></html>
```

**Hello World!** 플러그인을 성공적으로 구현한 경우 관리 포털에 로그인하면 이 화면이 표시됩니다.

그림 C.1. Hello World의 성공적인 구현! Plug-in



## 부록 D. RED HAT VIRTUALIZATION 및 암호화된 통신

## D.1. RED HAT VIRTUALIZATION MANAGER CA 인증서 교체



## 주의

`/etc/pki` 디렉터리 또는 하위 디렉토리에 대한 권한 및 소유권을 변경하지 마십시오. `/etc/pki` 및 `/etc/pki/ovirt-engine` 디렉터리에 대한 권한은 기본값인 **755**로 유지되어야 합니다.

조직의 타사 **CA** 인증서를 구성하여 **HTTPS**를 통해 연결하는 사용자와 **Red Hat Virtualization Manager**를 식별할 수 있습니다.



## 참고

**HTTPS** 연결에 타사 **CA** 인증서를 사용하는 경우 관리자와 호스트 간의 인증에 사용되는 인증서에는 영향을 미치지 않습니다. **Manager**에서 생성한 자체 서명된 인증서를 계속 사용합니다.

## 사전 요구 사항

- 타사 **CA** 인증서입니다. 사용하려는 인증서를 발급한 **CA(인증 기관)**의 인증서입니다. **PEM** 파일로 제공됩니다. 인증서 체인은 루트 인증서까지 완료해야 합니다. 체인의 순서가 중요하며 마지막 중간 인증서에서 루트 인증서로 이동해야 합니다. 이 절차에서는 타사 **CA** 인증서가 `/tmp/3rd-party-ca-cert.pem`에 제공되는 것으로 가정합니다.
- **Apache httpd**에 사용할 개인 키입니다. 암호가 없어야 합니다. 이 절차에서는 `/tmp/apache.key`에 있다고 가정합니다.
- **CA**에서 발급한 인증서입니다. 이 절차에서는 `/tmp/apache.cer`에 있다고 가정합니다.

**P12** 파일에서 **CA**에서 개인 키와 인증서를 받은 경우 다음 절차를 사용하여 압축을 풉니다. 다른 파일 형식에 대해서는 **CA**에 문의하십시오. 개인 키 및 인증서를 추출한 후 **Red Hat Virtualization Manager**

## Apache CA 인증서 교체를 진행합니다.

### P12 번들에서 인증서 및 개인 키 추출

내부 CA는 내부적으로 생성된 키와 인증서를 `/etc/pki/ovirt-engine/keys/apache.p12` 에 P12 파일에 저장합니다. Red Hat은 새 파일을 동일한 위치에 저장할 것을 권장합니다. 다음 절차에서는 새로운 P12 파일이 `/tmp/apache.p12` 에 있다고 가정합니다.

1. 현재 `apache.p12` 파일을 백업하십시오.

```
# cp -p /etc/pki/ovirt-engine/keys/apache.p12 /etc/pki/ovirt-engine/keys/apache.p12.bck
```

2. 현재 파일을 새 파일로 바꿉니다.

```
# cp /tmp/apache.p12 /etc/pki/ovirt-engine/keys/apache.p12
```

3. 필요한 위치에 개인 키와 인증서를 추출합니다. 파일이 암호로 보호된 경우 `-passin pass:_password_` 를 추가하고 암호를 필수 암호로 교체해야 합니다.

```
# openssl pkcs12 -in /etc/pki/ovirt-engine/keys/apache.p12 -nocerts -nodes > /tmp/apache.key
# openssl pkcs12 -in /etc/pki/ovirt-engine/keys/apache.p12 -nokeys > /tmp/apache.cer
```

#### 중요

새로운 Red Hat Virtualization 설치의 경우 이 절차의 모든 단계를 완료해야 합니다. 상용 인증서가 이미 구성된 Red Hat Enterprise Virtualization 3.6 환경에서 업그레이드 한 경우 1, 8 및 9 단계만 필요합니다.

### Red Hat Virtualization Manager Apache CA 인증서 교체

1. 셀프 호스트 엔진을 사용하는 경우 환경을 글로벌 유지 관리 모드로 설정합니다.

```
# hosted-engine --set-maintenance --mode=global
```

자세한 내용은 [15.1절](#). “[셀프 호스트 엔진 유지 관리](#)”의 내용을 참조하십시오.

2.

호스트 전체 신뢰 저장소에 **CA** 인증서를 추가합니다.

```
# cp /tmp/3rd-party-ca-cert.pem /etc/pki/ca-trust/source/anchors
# update-ca-trust
```

3.

**Manager**는 `/etc/pki/ovirt-engine/apache-ca.pem` 을 사용하도록 구성되어 있으며, 이는 `/etc/pki/ovirt-engine/ca.pem` 에 심볼릭 링크입니다. 심볼릭 링크를 제거합니다.

```
# rm /etc/pki/ovirt-engine/apache-ca.pem
```

4.

**CA** 인증서를 `/etc/pki/ovirt-engine/apache-ca.pem` 으로 저장합니다.

```
# cp /tmp/3rd-party-ca-cert.pem /etc/pki/ovirt-engine/apache-ca.pem
```

5.

기존 개인 키 및 인증서를 백업합니다.

```
# cp /etc/pki/ovirt-engine/keys/apache.key.nopass /etc/pki/ovirt-engine/keys/apache.key.nopass.bck
# cp /etc/pki/ovirt-engine/certs/apache.cer /etc/pki/ovirt-engine/certs/apache.cer.bck
```

6.

개인 키를 필요한 위치에 복사합니다.

```
# cp /tmp/apache.key /etc/pki/ovirt-engine/keys/apache.key.nopass
```

7.

개인 키 소유자를 **root**로 설정하고 권한을 **0640** 으로 설정합니다.

```
# chown root:ovirt /etc/pki/ovirt-engine/keys/apache.key.nopass
# chmod 640 /etc/pki/ovirt-engine/keys/apache.key.nopass
```

8.

필요한 위치에 인증서를 복사합니다.

```
# cp /tmp/apache.cer /etc/pki/ovirt-engine/certs/apache.cer
```

9.

**Apache** 서버를 다시 시작합니다.

```
# systemctl restart httpd.service
```

10.

다음 매개변수를 사용하여 새 신뢰 저장소 구성 파일 **/etc/ovirt-engine/engine.d/99-custom-truststore.conf** 를 만듭니다.

```
ENGINE_HTTPS_PKI_TRUST_STORE="/etc/pki/java/cacerts"
ENGINE_HTTPS_PKI_TRUST_STORE_PASSWORD=""
```

11.

**/etc/ovirt-engine/ovirt-websocket-proxy.conf.d/10-setup.conf** 파일을 복사하고 **10**보다 큰 인덱스 번호(예: **99-setup.conf**)로 이름을 바꿉니다. 새 파일에 다음 매개변수를 추가합니다.

```
SSL_CERTIFICATE=/etc/pki/ovirt-engine/certs/apache.cer
SSL_KEY=/etc/pki/ovirt-engine/keys/apache.key.nopass
```

12.

**websocket-proxy** 서비스를 다시 시작합니다.

```
# systemctl restart ovirt-websocket-proxy.service
```

13.

**/etc/ovirt-provider-ovn/conf.d/10-setup-provider-ovn.conf** 파일을 수동으로 변경하거나 이전 설치의 구성 파일을 사용하는 경우 **Manager**가 여전히 **/etc/pki/ovirt-engine/apache-ca.pem** 을 인증서 소스로 사용하도록 구성되어 있는지 확인합니다.

14.

다음 콘텐츠를 사용하여 새 파일 **/etc/ovirt-engine-backup/engine-backup-config.d/update-system-wide-pki.sh**를 생성하여 복원 시 시스템을 업데이트하도록 **engine-backup**을 활성화합니다.

```
BACKUP_PATHS="${BACKUP_PATHS}
/etc/ovirt-engine-backup"
cp -f /etc/pki/ovirt-engine/apache-ca.pem
/etc/pki/ca-trust/source/anchors/3rd-party-ca-cert.pem
update-ca-trust
```

15.

**ovirt-provider-ovn** 서비스를 다시 시작합니다.

```
# systemctl restart ovirt-provider-ovn.service
```

16.

**ovirt-engine** 서비스를 다시 시작합니다.

```
# systemctl restart ovirt-engine.service
```

17.

셀프 호스트 엔진을 사용하는 경우 글로벌 유지 관리 모드를 끕니다.

```
# hosted-engine --set-maintenance --mode=none
```

이제 사용자가 **HTTPS** 트래픽을 암호화하는 데 사용되는 인증서의 진위 여부를 확인하지 않고도 관리 포털 및 **VM** 포털에 연결할 수 있습니다.

## D.2. 관리자와 LDAP 서버 간 암호화 통신 설정

**Red Hat Virtualization Manager**와 **LDAP** 서버 간에 암호화된 통신을 설정하려면 **LDAP** 서버의 루트 **CA** 인증서를 가져와서 관리자에 루트 **CA** 인증서를 복사하고 **PEM** 인코딩 **CA** 인증서를 생성합니다. 키 저장소 유형은 모든 **Java** 지원 유형일 수 있습니다. 다음 절차에서는 **JDK(Java KeyStore)** 형식을 사용합니다.



참고

**PEM** 인코딩 **CA** 인증서 생성 및 인증서 가져오기에 대한 자세한 내용은 **README** 파일의 **X.509CERTIFICATESTORE** 섹션을 참조하십시오. **/usr/share/doc/ovirt-engine-extension-aaa-ldap-버전**.

### PEM 인코딩 CA 인증서 생성

1.

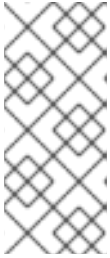
**Red Hat Virtualization Manager**에서 **LDAP** 서버의 루트 **CA** 인증서를 **/tmp** 디렉터리에 복사하고 **keytool** 을 사용하여 **PEM** 인코딩 **CA** 인증서를 생성합니다. 다음 명령은 **/tmp/myrootca.pem** 에서 루트 **CA** 인증서를 가져오고 **/etc/ovirt-engine/aaa/** 아래에 **PEM** 인코딩 **CA** 인증서 **myrootca.jks** 를 생성합니다. 인증서의 위치와 암호를 기록해 둡니다. 대화형 설정 도구를 사용하는 경우 필요한 모든 정보입니다. **LDAP** 서버를 수동으로 구성하는 경우 나머지 절차에 따라 구성 파일을 업데이트합니다.

```
$ keytool -importcert -noprompt -trustcacerts -alias myrootca -file /tmp/myrootca.pem -keystore /etc/ovirt-engine/aaa/myrootca.jks -storepass password
```

2.

**/etc/ovirt-engine/aaa/profile1.properties** 파일을 인증서 정보로 업데이트합니다.





## 참고

**`${local:_basedir}`** 는 **LDAP** 속성 구성 파일이 있고 **`/etc/ovirt-engine/aaa`** 디렉토리를 가리키는 디렉터리입니다. **PEM** 인코딩 **CA** 인증서를 다른 디렉터리에 생성한 경우 **`${local:_basedir}`** 를 인증서의 전체 경로로 교체합니다.

- **startTLS(권장됨)**를 사용하려면 다음을 수행합니다.

```
# Create keystore, import certificate chain and uncomment
pool.default.ssl.startTLS = true
pool.default.ssl.truststore.file = ${local:_basedir}/myrootca.jks
pool.default.ssl.truststore.password = password
```

- **SSL**을 사용하려면 다음을 수행합니다.

```
# Create keystore, import certificate chain and uncomment
pool.default.serverset.single.port = 636
pool.default.ssl.enable = true
pool.default.ssl.truststore.file = ${local:_basedir}/myrootca.jks
pool.default.ssl.truststore.password = password
```

외부 **LDAP** 공급자 구성을 계속 보려면 외부 **LDAP** 공급자 구성을 참조하십시오. **LDAP** 및 **Kerberos for Single Sign-on**을 계속 구성하려면 **SSO( Single Sign-On)용 LDAP** 및 **Kerberos** 구성 을 참조하십시오.

### D.3. VDSM을 위해 수동으로 암호화된 통신 설정

**Manager** 및 기타 **VDSM** 인스턴스와 함께 **VDSM**에 대해 암호화된 통신을 수동으로 설정할 수 있습니다.

클러스터 수준 **3.6**, **4.0** 및 **4.1**이 있는 클러스터의 호스트만 수동 구성이 필요합니다. 레벨 **4.2**가 있는 클러스터의 호스트는 호스트 재설치 중에 강력한 암호화를 위해 자동으로 재구성됩니다.



## 참고

**RHVH 3.6, 4.0** 및 **4.1** 호스트에서는 강력한 암호화를 지원하지 않습니다. **RHVH 4.2** 및 **RHEL** 호스트가 이를 지원합니다.

**RHVH 4.2** 호스트가 있는 **3.6, 4.0** 또는 **4.1** 클러스터가 있는 경우 강력한 암호화를 사용할 수 있습니다.

## 절차

1. **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.
2. **Management** → **Maintenance** (유지 관리)를 클릭하여 **Maintenance Host(s)** 확인 창을 엽니다.
3. 확인 을 클릭하여 유지보수 모드를 시작합니다.
4. 호스트에서 다음 설정을 사용하여 `/etc/vdsm/vdsm.conf.d/99-custom-ciphers.conf` 를 만듭니다.

```
[vars]
ssl_ciphers = HIGH
```

자세한 내용은 [OpenSSL Cipher 문자열](#) 을 참조하십시오.

5. **VDSM** 재시작:
 

```
# systemctl restart vdsm
```
6. **Compute** → **Hosts** (컴퓨팅 호스트)를 클릭하고 호스트를 선택합니다.
7. **Management** → **Activate** (활성화)를 클릭하여 호스트를 다시 활성화합니다.

## 부록 E. 프록시

## E.1. SPICE 프록시

## E.1.1. SPICE 프록시 개요

SPICE 프록시는 SPICE 클라이언트가 하이퍼바이저를 연결하는 네트워크 외부에 있을 때 SPICE 클라이언트를 가상 머신에 연결하는 데 사용되는 툴입니다. SPICE 프록시 설정은 시스템에 Squid 를 설치하고 프록시 트래픽을 허용하도록 방화벽을 구성하여 구성됩니다. SPICE 프록시를 켜면 Manager에서 engine-config 를 사용하여 키 SpiceProxyDefault 를 프록시 이름 및 포트로 구성된 값으로 설정합니다. SPICE 프록시를 끄는 것은 Manager에서 engine-config 를 사용하여 SpiceProxyDefault 키가 설정된 값을 제거하는 것으로 구성됩니다.



## 중요

SPICE 프록시는 독립형 SPICE 클라이언트와만 사용할 수 있으며 noVNC를 사용하여 가상 머신에 연결하는 데 사용할 수 없습니다.

## E.1.2. SPICE 프록시 머신 설정

다음 절차에서는 머신을 SPICE 프록시로 설정하는 방법을 설명합니다. SPICE 프록시를 사용하면 네트워크 외부에서 Red Hat Virtualization 네트워크에 연결할 수 있습니다. 이 절차에서는 프록시 서비스를 제공하기 위해 Squid 를 사용합니다.

## Red Hat Enterprise Linux에 Squid 설치

1. 프록시 시스템에 Squid 를 설치합니다.

```
# yum install squid
```

2. /etc/squid/squid.conf 를 엽니다. 변경 사항:

```
http_access deny CONNECT !SSL_ports
```

다음으로 변경합니다.

```
http_access deny CONNECT !Safe_ports
```

3.

**squid** 서비스를 시작하고 재부팅 후 자동으로 실행되도록 활성화합니다.

```
# systemctl enable squid.service --now
```

4.

기본 **firewalld** 영역에서 **squid** 서비스에 대한 들어오는 요청을 활성화합니다.

```
# firewall-cmd --permanent --add-service=squid
```

5.

런타임 구성에서 이 방화벽 규칙을 영구적으로 설정합니다.

```
# firewall-cmd --reload
```

6.

**squid** 서비스가 방화벽 서비스 목록에 표시되는지 확인합니다.

```
# firewall-cmd --list-services  
ssh dhcpv6-client squid
```

이제 머신을 **SPICE** 프록시로 설정했습니다. 네트워크 외부에서 **Red Hat Virtualization** 네트워크에 연결하기 전에 **SPICE** 프록시를 활성화합니다.

### E.1.3. SPICE 프록시 켜기

다음 절차에서는 **SPICE** 프록시를 활성화(또는 켜기)하는 방법을 설명합니다.

#### SPICE 프록시 활성화

1.

**Manager**에서 **engine-config** 도구를 사용하여 프록시를 설정합니다.

```
# engine-config -s SpiceProxyDefault=someProxy
```

2.

**ovirt-engine** 서비스를 다시 시작합니다.

```
# systemctl restart ovirt-engine.service
```

프록시에는 다음이 있어야 합니다.

`protocol://[host]:[port]`



참고

**Red Hat Enterprise Linux 6.7, Red Hat Enterprise Linux 7.2 이상에서** 제공되는 **SPICE** 클라이언트만 **HTTPS** 프록시를 지원합니다. 이전 클라이언트는 **HTTP**만 지원합니다. 이전 클라이언트에 대해 **HTTPS**를 지정하면 클라이언트는 프록시 설정을 무시하고 호스트에 대한 직접 연결을 시도합니다.

**SPICE** 프록시가 활성화되었습니다(반환). 이제 **SPICE** 프록시를 통해 **Red Hat Virtualization** 네트워크에 연결할 수 있습니다.

#### E.1.4. SPICE 프록시 끄기

다음 절차에서는 **SPICE** 프록시를 끄는(비활성화) 방법을 설명합니다.

##### SPICE 프록시 끄기

1. **Manager**에 로그인합니다.

```
$ ssh root@[IP of Manager]
```

2. 다음 명령을 실행하여 **SPICE** 프록시를 지웁니다.

```
# engine-config -s SpiceProxyDefault=""
```

3. 관리자를 다시 시작하십시오.

```
# systemctl restart ovirt-engine.service
```

**SPICE** 프록시가 비활성화되었습니다(중단됨). 더 이상 **SPICE** 프록시를 통해 **Red Hat Virtualization** 네트워크에 연결할 수 없습니다.

## E.2. SQUID PROXY

### E.2.1. Squid 프록시 설치 및 구성

이 섹션에서는 VM 포털에 Squid 프록시를 설치하고 구성하는 방법을 설명합니다. Squid 프록시 서버는 콘텐츠 가속기로 사용됩니다. 자주 보는 콘텐츠를 캐시하고 대역폭을 줄이고 응답 시간을 개선합니다.

#### Squid 프록시 구성

1.

Squid 프록시 서버의 HTTPS 포트에 대한 키 쌍 및 인증서를 가져옵니다. 이 키 쌍은 다른 SSL/TLS 서비스에 대한 키 쌍을 가져오는 것과 동일한 방식으로 가져올 수 있습니다. 키 쌍은 개인 키와 서명된 인증서를 포함하는 두 개의 PEM 파일 형식으로 되어 있습니다. 이 절차에서는 이름이 proxy.key 및 proxy.cer 인 것으로 가정합니다.



#### 참고

키 쌍 및 인증서는 엔진의 인증 기관을 사용하여 생성할 수도 있습니다. 프록시의 개인 키와 인증서가 있고 엔진 인증 기관을 사용하여 생성하지 않으려면 다음 단계로 건너뛵니다.

2.

프록시의 호스트 이름을 선택합니다. 그런 다음 프록시에 대한 인증서 고유 이름의 다른 구성 요소를 선택합니다.

## 참고

엔진 자체에서 사용하는 것과 동일한 국가와 동일한 조직 이름을 사용하는 것이 좋습니다. **Manager**가 설치된 머신에 로그인하여 다음 명령을 실행하여 이 정보를 찾습니다.

```
# openssl x509 -in /etc/pki/ovirt-engine/ca.pem -noout -subject
```

이 명령은 다음과 같이 출력됩니다.

```
subject= /C=US/O=Example Inc./CN=engine.example.com.81108
```

관련 부분은 **/C=US/O=ECDHE Inc.**입니다. 이를 사용하여 프록시의 인증서에 대한 완전한 고유 이름을 빌드합니다.

```
/C=US/O=Example Inc./CN=proxy.example.com
```

## 3.

프록시 시스템에 로그인하고 인증서 서명 요청을 생성합니다.

```
# openssl req -newkey rsa:2048 -subj '/C=US/O=Example Inc./CN=proxy.example.com' -nodes -keyout proxy.key -out proxy.req
```

## 중요

인증서의 고유 이름에 따옴표를 포함해야 합니다. **-nodes** 옵션을 사용하면 개인 키가 암호화되지 않도록 합니다. 즉, 프록시 서버를 시작하기 위해 암호를 입력할 필요가 없습니다.

이 명령은 **proxy.key** 및 **proxy.req** 파일 두 개를 생성합니다. **proxy.key**는 개인 키입니다. 이 파일을 안전하게 보관하십시오. **proxy.req**는 인증서 서명 요청입니다. **proxy.req**에는 특별한 보호 기능이 필요하지 않습니다.

## 4.

서명된 인증서를 생성하려면 프록시 시스템에서 **Manager** 시스템으로 인증서 서명 요청 파일을 복사합니다.

```
# scp proxy.req engine.example.com:/etc/pki/ovirt-engine/requests/
```

5.

**Manager** 시스템에 로그인하고 인증서에 서명합니다.

```
# /usr/share/ovirt-engine/bin/pki-enroll-request.sh --name=proxy --days=3650 --
subject='/C=US/O=Example Inc./CN=proxy.example.com'
```

이는 인증서에 서명하고 **10년 (3650일)** 동안 유효합니다. 원하는 경우 이전에 만료되도록 인증서를 설정합니다.

6.

생성된 인증서 파일은 **/etc/pki/ovirt-engine/certs** 디렉터리에서 사용할 수 있으며 이름이 **proxy.cer** 여야 합니다. 프록시 시스템에서 **Manager** 시스템에서 현재 디렉터리로 이 파일을 복사합니다.

```
# scp engine.example.com:/etc/pki/ovirt-engine/certs/proxy.cer .
```

7.

프록시 시스템에 **proxy.key** 및 **proxy.cer** 이 모두 있는지 확인합니다.

```
# ls -l proxy.key proxy.cer
```

8.

프록시 시스템에 **Squid** 프록시 서버 패키지를 설치합니다.

```
# yum install squid
```

9.

개인 키와 서명된 인증서를 프록시가 액세스할 수 있는 위치로 이동합니다(예: **/etc/squid** 디렉터리).

```
# cp proxy.key proxy.cer /etc/squid/.
```

10.

**squid** 사용자가 다음 파일을 읽을 수 있도록 권한을 설정합니다.

```
# chgrp squid /etc/squid/proxy.*
# chmod 640 /etc/squid/proxy.*
```

11.

**Squid** 프록시는 엔진에서 사용하는 인증서를 확인해야 합니다. **Manager** 인증서를 프록시 시스템에 복사합니다. 이 예에서는 파일 경로 **/etc/squid:**를 사용합니다.

```
# scp engine.example.com:/etc/pki/ovirt-engine/ca.pem /etc/squid/.
```





참고

기본 **CA** 인증서는 **Manager** 시스템의 `/etc/pki/ovirt-engine/ca.pem` 에 있습니다.

12.

**squid** 사용자가 인증서 파일을 읽을 수 있도록 권한을 설정합니다.

```
# chgrp squid /etc/squid/ca.pem
# chmod 640 /etc/squid/ca.pem
```

13.

**SELinux**가 강제 모드인 경우 **semanage** 툴을 사용하여 포트 **443** 컨텍스트를 변경하여 **Squid**가 포트 **443**을 사용하도록 허용합니다.

```
# yum install polycoreutils-python
# semanage port -m -p tcp -t http_cache_port_t 443
```

14.

기존 **Squid** 구성 파일을 다음으로 교체합니다.

```
https_port 443 key=/etc/squid/proxy.key cert=/etc/squid/proxy.cer ssl-bump
defaultsite=engine.example.com
cache_peer engine.example.com parent 443 0 no-query originserver ssl
sslcafile=/etc/squid/ca.pem name=engine login=PASSTHRU
cache_peer_access engine allow all
ssl_bump allow all
http_access allow all
```

15.

**Squid** 프록시 서버를 다시 시작합니다.

```
# systemctl restart squid.service
```



참고

기본 구성의 **squid Proxy**는 15개의 유틸리티 분 후에 연결을 종료합니다. **Squid Proxy**가 유틸리티 연결을 종료하기 전의 시간을 늘리려면 **squid.conf** (예: `read_timeout 10` 시간)에서 `read_timeout` 옵션을 조정합니다.

## E.3. WEBSOCKET 프록시

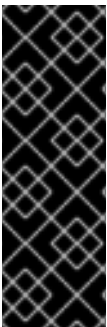
### E.3.1. WebSocket 프록시 개요

**Websocket** 프록시를 사용하면 사용자는 **noVNC** 콘솔을 통해 가상 머신에 연결할 수 있습니다.

**Websocket** 프록시는 초기 구성 (**Red Hat Virtualization Manager** 구성 참조) 또는 별도의 시스템에 **Red Hat Virtualization Manager** 시스템에 설치 및 구성할 수 있습니다( **Separate Machine**에 **Websocket** 프록시 설치참조).

**Websocket** 프록시는 **Manager** 시스템에서 별도의 시스템으로 마이그레이션할 수도 있습니다. **E.3.2** 절. “**Websocket** 프록시를 분리 머신으로 마이그레이션” 을 참조하십시오.

### E.3.2. **Websocket** 프록시를 분리 머신으로 마이그레이션



#### 중요

**Websocket** 프록시와 **noVNC**는 기술 프리뷰 기능 전용입니다. 기술 프리뷰 기능은 **Red Hat** 프로덕션 서비스 수준 계약(SLA)에서 지원되지 않으며 기능적으로 완전하지 않을 수 있으며 프로덕션에는 사용하지 않는 것이 좋습니다. 이러한 기능을 사용하면 향후 제품 기능을 조기에 이용할 수 있어 개발 과정에서 고객이 기능을 테스트하고 피드백을 제공할 수 있습니다. 자세한 내용은 **Red Hat** 기술 프리뷰 기능 지원 범위를 참조하십시오.

보안 또는 성능상의 이유로 **Websocket** 프록시는 **Red Hat Virtualization Manager**를 실행하지 않는 별도의 머신에서 실행할 수 있습니다. **Manager** 머신에서 별도의 시스템으로 **Websocket** 프록시를 마이그레이션하는 프로시저에는 **Manager** 머신에서 **websocket** 프록시 구성을 제거한 다음 별도의 머신에 프록시를 설치해야 합니다.

**engine-cleanup** 명령을 사용하여 **Manager** 머신에서 **Websocket** 프록시를 제거할 수 있습니다.

#### **Manager** 시스템에서 **Websocket** 프록시 제거

1.

**Manager** 시스템에서 **engine-cleanup** 을 실행하여 필요한 구성을 제거합니다.

```
# engine-cleanup
```

2.

모든 구성 요소를 제거하라는 메시지가 표시되면 **No** 를 입력하고 **Enter** 를 누릅니다.

```
Do you want to remove all components? (Yes, No) [Yes]: No
```

3.

엔진을 제거하라는 메시지가 표시되면 **No** 를 입력하고 **Enter** 를 누릅니다.

Do you want to remove the engine? (Yes, No) [Yes]: No

4.

**websocket** 프록시를 제거하라는 메시지가 표시되면 **Yes** 를 입력하고 **Enter** 를 누릅니다.

Do you want to remove the WebSocket proxy? (Yes, No) [No]: Yes

다른 구성 요소를 제거하라는 메시지가 표시되면 **No** 를 선택합니다.

### 별도의 컴퓨터에 **Websocket** 프록시 설치



#### 중요

**Websocket** 프록시와 **noVNC**는 기술 프리뷰 기능 전용입니다. 기술 프리뷰 기능은 **Red Hat** 프로덕션 서비스 수준 계약(SLA)에서 지원되지 않으며 기능적으로 완전하지 않을 수 있으며 프로덕션에는 사용하지 않는 것이 좋습니다. 이러한 기능을 사용하면 향후 제품 기능을 조기에 이용할 수 있어 개발 과정에서 고객이 기능을 테스트하고 피드백을 제공할 수 있습니다. 자세한 내용은 [Red Hat 기술 프리뷰 기능 지원 범위를 참조하십시오](#).

**Websocket** 프록시를 사용하면 사용자는 **noVNC** 콘솔을 통해 가상 머신에 연결할 수 있습니다. **noVNC** 클라이언트는 **Websockets**를 사용하여 **VNC** 데이터를 전달합니다. 그러나 **QEMU**의 **VNC** 서버는 **websocket**을 지원하지 않으므로 클라이언트와 **VNC** 서버 간에 **Websocket** 프록시를 배치해야 합니다. 프록시는 **Manager** 머신을 포함하여 네트워크에 액세스할 수 있는 모든 머신에서 실행할 수 있습니다.

보안 및 성능상의 이유로 사용자는 별도의 시스템에서 **Websocket** 프록시를 구성해야 할 수 있습니다.

#### 절차

1.

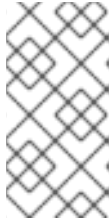
**Websocket** 프록시를 설치합니다.

```
# yum install ovirt-engine-websocket-proxy
```

2.

**engine-setup** 명령을 실행하여 **Websocket** 프록시를 구성합니다.

```
# engine-setup
```



## 참고

**ECDHE** 패키지 도 설치된 경우 이 호스트에서 **Manager (Engine)**를 구성하라는 메시지가 표시될 때 **No** 를 선택합니다.

3. **Enter** 를 눌러 **engine-setup** 을 눌러 시스템에서 **Websocket** 프록시 서버를 구성할 수 있습니다.

Configure WebSocket Proxy on this machine? (Yes, No) [Yes]:

4. **Enter** 를 눌러 자동으로 감지된 호스트 이름을 수락하거나 대체 호스트 이름을 입력하고 **Enter** 를 누릅니다. 가상 호스트를 사용하는 경우 자동으로 탐지된 호스트 이름이 올바르지 않을 수 있습니다.

Host fully qualified DNS name of this server [host.example.com]:

5. **Enter** 를 눌러 **engine-setup** 을 눌러 방화벽을 구성하고 외부 통신에 필요한 포트를 열 수 있습니다. **engine-setup** 이 방화벽 구성을 변경할 수 없는 경우 필요한 포트를 수동으로 열어야 합니다.

Setup can automatically configure the firewall on this system.

Note: automatic configuration of the firewall may overwrite current settings.

Do you want Setup to configure the firewall? (Yes, No) [Yes]:

6. **Manager** 시스템의 **FQDN**을 입력하고 **Enter** 를 누릅니다.

Host fully qualified DNS name of the engine server []: manager.example.com

7. **Enter** 를 눌러 **Engine-setup** 이 **Manager** 시스템에서 작업을 수행하거나 **2** 를 눌러 작업을 수동으로 수행할 수 있습니다.

Setup will need to do some actions on the remote engine server. Either automatically, using ssh as root to access it, or you will be prompted to manually perform each such action.

Please choose one of the following:

1 - Access remote engine server using ssh as root

2 - Perform each action manually, use files to copy content around

(1, 2) [1]:

- a. **Enter** 를 눌러 기본 **SSH** 포트 번호를 수락하거나 **Manager** 시스템의 포트 번호를 입력합니다.

ssh port on remote engine server [22]:

b.

**Manager** 시스템에 로그인할 루트 암호를 입력하고 **Enter** 를 누릅니다.

root password on remote engine server engine\_host.example.com:

8.

현재 설정과 다른 **iptables** 규칙을 검토할지 여부를 선택합니다.

Generated iptables rules are different from current ones.  
Do you want to review them? (Yes, No) [No]:

9.

**Enter** 를 눌러 구성 설정을 확인합니다.

==== CONFIGURATION PREVIEW ====

```
Firewall manager           : iptables
Update Firewall            : True
Host FQDN                  : host.example.com
Configure WebSocket Proxy  : True
Engine Host FQDN           : engine_host.example.com
```

Please confirm installation settings (OK, Cancel) [OK]:

구성된 **Websocket** 프록시를 사용하도록 **Manager** 시스템을 구성하기 위한 지침이 제공됩니다.

Manual actions are required on the engine host  
in order to enroll certs for this host and configure the engine about it.

Please execute this command on the engine host:  
engine-config -s WebSocketProxy=host.example.com:6100  
and than restart the engine service to make it effective

10.

**Manager** 시스템에 로그인하고 제공된 지침을 실행합니다.

```
# engine-config -s WebSocketProxy=host.example.com:6100
# systemctl restart ovirt-engine.service
```

## 부록 F. 브랜딩

### F.1. 브랜딩

#### F.1.1. 관리자의 다시 시작

**Red Hat Virtualization Manager**의 다양한 측면(예: 에서 사용하는 아이콘 및 팝업 창에 표시되는 텍스트) 및 **welcome** 페이지에 표시되는 링크와 같이 사용자 지정할 수 있습니다. 이를 통해 **Manager**를 다시 브랜드화하고 최종 모양을 세밀하게 제어할 수 있으며 관리자와 사용자에게 표시되는 것처럼 느낄 수 있습니다.

관리자를 사용자 지정하는 데 필요한 파일은 **Manager**가 설치된 시스템의 **/etc/ovirt-engine/branding/** 디렉터리에 있습니다. 파일은 그래픽 사용자 인터페이스의 다양한 측면과 **Manager**의 다양한 구성 요소에 통합되는 메시지 및 링크를 포함하는 속성 파일 세트를 구성하는 데 사용됩니다.

구성 요소를 사용자 지정하려면 해당 구성 요소의 파일을 편집하고 변경 사항을 저장합니다. 다음에 해당 구성 요소를 열거나 새로 고침하면 변경 사항이 적용됩니다.

#### F.1.2. 로그인 화면

로그인 화면은 관리 포털과 **VM** 포털 모두에서 사용하는 로그인 화면입니다. 사용자 정의할 수 있는 로그인 화면의 요소는 다음과 같습니다.

- 테두리
- 왼쪽의 헤더 이미지
- 오른쪽의 헤더 이미지
- 헤더 텍스트

로그인 화면의 클래스는 **common.css** 에 있습니다.

#### F.1.3. 관리 포털 화면

관리 포털에 로그인할 때 표시되는 주요 화면은 관리 포털 화면입니다. 사용자 정의할 수 있는 관리 포털 화면의 요소는 다음과 같습니다.

- 로고
- 왼쪽 배경 이미지
- 센터 배경 이미지
- 오른쪽 배경 이미지
- 로고 오른쪽에 있는 텍스트

관리 포털 화면의 클래스는 `web_admin.css` 에 있습니다.

#### F.1.4. VM 포털 화면

VM 포털 화면은 VM 포털에 로그인할 때 표시되는 화면입니다. 사용자 정의할 수 있는 VM 포털 화면의 요소는 다음과 같습니다.

- 로고
- 센터 배경 이미지
- 오른쪽 배경 이미지
- 주요 **Grid** 주변의 경계
- 사용자 레이블 위의 텍스트

VM 포털 화면의 클래스는 `user_portal.css` 에 있습니다.

### F.1.5. 팝업 Windows

팝업 창은 **Manager**의 모든 창으로 호스트 또는 가상 머신과 같은 엔터티를 생성, 편집 또는 업데이트 할 수 있습니다. 사용자 정의할 수 있는 팝업 창의 요소는 다음과 같습니다.

- 테두리
- 왼쪽의 헤더 이미지
- 헤더 센터 이미지 (**repeated**)

팝업 창에 대한 클래스는 `common.css` 에 있습니다.

### F.1.6. 탭

관리 포털의 많은 팝업 창에 탭이 포함되어 있습니다. 사용자 지정할 수 있는 이러한 탭의 요소는 다음과 같습니다.

- 활성 상태
- **inactive**

탭의 클래스는 `common.css` 및 `user_portal.css` 에 있습니다.

### F.1.7. The welcome page

**welcome** 페이지는 **Manager**의 홈페이지를 방문할 때 처음 표시되는 페이지입니다. 전체 모양 및 느낌 사용자 지정 외에도 템플릿 파일을 편집하여 추가 문서 또는 내부 웹 사이트를 위해 페이지에 링크를 추가하는 등의 다른 변경을 할 수 있습니다. 사용자 정의할 수 있는 **welcome** 페이지의 요소는 다음과 같습니다.



- 페이지 제목
- 헤더(**left, center, right**)
- 오류 메시지
- 해당 링크에 대한 전달 및 관련 메시지 링크

**Restic Page**의 클래스는 **welcome\_style.css**에 있습니다.

템플릿 파일

**welcome\_page.template** 이름의 템플릿 파일은 **HTML, HEAD** 또는 **BODY** 태그가 포함되지 않은 일반 **HTML** 파일입니다. 이 파일은 **welcome** 페이지 자체에 직접 삽입되고, **welcome** 페이지에 표시되는 콘텐츠의 컨테이너 역할을 합니다. 따라서 이 파일을 편집하여 새 링크를 추가하거나 콘텐츠 자체를 변경해야 합니다. 템플릿 파일의 또 다른 기능은 **subscription** 페이지가 처리될 때 **messages.properties** 파일의 해당 텍스트로 대체되는 **{user\_portal}**와 같은 자리 표시자 텍스트가 포함되어 있다는 것입니다.

#### F.1.8. 페이지를 찾을 수 없음 페이지

페이지 찾을 수 없음 페이지는 **Red Hat Virtualization Manager**에서 찾을 수 없는 페이지에 대한 링크를 열 때 표시되는 페이지입니다. 사용자 정의할 수 있는 페이지 **Not Found** 페이지의 요소는 다음과 같습니다.

- 페이지 제목
- 헤더(**left, center, right**)
- 오류 메시지
- 해당 링크에 대한 전달 및 관련 메시지 링크

페이지 **Not Found** 페이지의 클래스는 **welcome\_style.css** 에 있습니다.

## 부록 G. 시스템 계정

### G.1. 시스템 계정

#### G.1.1. Red Hat Virtualization Manager 사용자 계정

**FlexVolume** 패키지가 설치되면 **Red Hat Virtualization**을 지원하도록 여러 시스템 사용자 계정이 생성됩니다. 각 시스템 사용자에게는 기본 **UID(사용자 ID)**가 있습니다. 생성된 시스템 사용자 계정은 다음과 같습니다.

- **vdsm 사용자(UID 36).** NFS 스토리지 도메인을 마운트 및 액세스하는 데 필요합니다.
- **ovirt 사용자(UID 108).** ovirt-engine Red Hat JBoss Enterprise Application Platform 인스턴스의 소유자.
- **ovirt-vmconsole 사용자(UID 498).** 게스트 직렬 콘솔에 필요합니다.

#### G.1.2. Red Hat Virtualization Manager 그룹

**FlexVolume** 패키지가 설치되면 **Red Hat Virtualization**을 지원하도록 여러 시스템 사용자 그룹이 생성됩니다. 각 시스템 사용자 그룹에는 기본 그룹 식별자(**GID**)가 있습니다. 생성된 시스템 사용자 그룹은 다음과 같습니다.

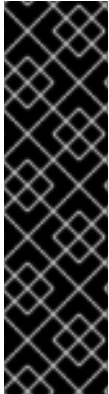
- **kvm 그룹(GID 36)**입니다. 그룹 멤버는 다음과 같습니다.
- **vdsm 사용자**입니다.
- **ovirt 그룹(GID 108).** 그룹 멤버는 다음과 같습니다.
- **ovirt 사용자.**
- **ovirt-vmconsole 그룹(GID 498).** 그룹 멤버는 다음과 같습니다.

- **ovirt-vmconsole** 사용자.

### G.1.3. 가상화 호스트 사용자 계정

**vds**m 및 **qemu-kvm-rhev** 패키지가 설치되면 가상화 호스트에 여러 시스템 사용자 계정이 생성됩니다. 각 시스템 사용자에게는 기본 **UID(사용자 ID)**가 있습니다. 생성된 시스템 사용자 계정은 다음과 같습니다.

- **vds**m 사용자(**UID 36**).
- **qemu** 사용자(**UID 107**).
- **sanlock** 사용자(**UID 179**).
- **ovirt-vmconsole** 사용자(**UID 498**).



#### 중요

할당된 사용자 식별자(**UID**) 및 그룹 식별자(**GID**)는 시스템마다 다를 수 있습니다. **vds**m 사용자는 **UID 36** 으로 고정되어 **kvm** 그룹은 **GID 36** 으로 고정되어 있습니다.

시스템의 다른 계정에서 **UID 36** 또는 **GID 36**을 이미 사용하고 있는 경우 **vds**m 및 **qemu-kvm-rhev** 패키지를 설치하는 동안 충돌이 발생합니다.

### G.1.4. 가상화 호스트 그룹

**vds**m 및 **qemu-kvm-rhev** 패키지가 설치되면 가상화 호스트에 여러 시스템 사용자 그룹이 생성됩니다. 각 시스템 사용자 그룹에는 기본 그룹 식별자(**GID**)가 있습니다. 생성된 시스템 사용자 그룹은 다음과 같습니다.

- **kvm** 그룹(**GID 36**)입니다. 그룹 멤버는 다음과 같습니다.
- **qemu** 사용자입니다.

- **sanlock** 사용자입니다.
- **qemu** 그룹(**GID 107**). 그룹 멤버는 다음과 같습니다.
- **vdsm** 사용자입니다.
- **sanlock** 사용자입니다.
- **ovirt-vmconsole** 그룹(**GID 498**). 그룹 멤버는 다음과 같습니다.
- **ovirt-vmconsole** 사용자.



#### 중요

할당된 사용자 식별자(**UID**) 및 그룹 식별자(**GID**)는 시스템마다 다를 수 있습니다. **vdsm** 사용자는 **UID 36** 으로 고정되어 **kvm** 그룹은 **GID 36** 으로 고정되어 있습니다.

시스템의 다른 계정에서 **UID 36** 또는 **GID 36**을 이미 사용하고 있는 경우 **vdsm** 및 **qemu-kvm-rhev** 패키지를 설치하는 동안 충돌이 발생합니다.