Red Hat Virtualization 4.4

데이터 웨어하우스 가이드

Red Hat Virtualization의 데이터 웨어하우스 기능을 사용하는 방법

Last Updated: 2022-06-26

Red Hat Virtualization 4.4 데이터 웨어하우스 가이드

Red Hat Virtualization의 데이터 웨어하우스 기능을 사용하는 방법

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/Data_Warehouse_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 ® is the registered trademark of Linus Torvalds in the United States and other countries.

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

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

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

Node.js ® 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 ® 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 Data 웨어하우스와 관련된 정보 및 절차를 설명합니다.

차례

PREFACE

Red Hat Virtualization Manager에는 호스트, 가상 시스템 및 스토리지에 대한 모니터링 데이터를 수집하는 데이터 웨어하우스가 포함되어 있습니다. 데이터베이스와 서비스가 포함된 데이터 웨어하우스는 동일한 시스템 또는 별도의 서버에 관리자 설정과 함께 설치하고 구성해야 합니다.

Red Hat Virtualization 설치는 다음 두 가지 데이터베이스를 생성합니다.

- Manager 데이터베이스(**engine**)는 Red Hat Virtualization Manager에서 사용하는 기본 데이터 저장소입니다. 상태, 구성 및 성능과 같은 가상화 환경에 대한 정보는 이 데이터베이스에 저장됩니다.

- 데이터 웨어하우스 데이터베이스(**ovirt_engine_history**)에는 Manager 데이터베이스에서 시간 경과에 따라 수집된 구성 정보 및 통계 데이터가 포함되어 있습니다. Manager 데이터베이스의 구성 데이터는 1분마다 검사되며 변경 사항은 데이터 웨어하우스 데이터베이스에 복제됩니다. 데이터베이스의 변경 사항을 추적하면 데이터베이스의 오브젝트에 대한 정보가 제공됩니다. 이를 통해 Red Hat Virtualization 환경의 성능을 분석 및 개선하고 문제를 해결할 수 있습니다.

ovirt_engine_history 데이터베이스에서 사용할 공간과 리소스의 추정치를 계산하려면 RHV Manager History Database Size Calculator 툴을 사용합니다. 추정치는 엔터티 수와 기록 레코드를 유지하기 위해 선택한 시간을 기반으로 합니다.

1장. 데이터 웨어하우스 설치 및 구성

1.1. 데이터 웨어하우스 구성 개요

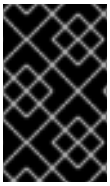
Manager와 동일한 머신 또는 Manager에 액세스할 수 있는 별도의 머신에 데이터 웨어하우스를 설치하고 구성할 수 있습니다.

Manager 시스템에 데이터 웨어하우스 설치 및 구성

이 구성에는 등록된 단일 시스템만 필요하며 구성하기가 가장 간편하지만 Manager 시스템에 대한 수요가 증가합니다. 데이터 웨어하우스 서비스에 액세스해야 하는 사용자는 관리자 시스템 자체에 액세스해야 합니다. *로컬 데이터베이스를 사용하여 독립 실행형 관리자로 Red Hat Virtualization 설치에서 Red Hat Virtualization Manager 구성*을 참조하십시오.

별도의 머신에 데이터 웨어하우스 설치 및 구성

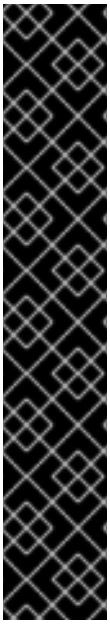
이 구성에는 등록된 두 개의 시스템이 필요합니다. Manager 시스템의 부하를 줄이고 해당 시스템에서 잠재적인 CPU 및 메모리 공유 충돌을 방지합니다. 관리자는 Manager 시스템에 대한 액세스 권한을 부여하지 않고도 사용자가 데이터 웨어하우스 시스템에 대한 액세스 권한을 허용할 수도 있습니다. 이 구성에 대한 자세한 내용은 Machine에서 데이터 3.4 설치 및 구성을 참조하십시오.



중요

데이터 웨어하우스 배포의 모든 시스템의 시스템 시간대를 UTC로 설정하는 것이 좋습니다. 이렇게 하면 데이터 수집이 현지 시간대의 변형으로 인해 중단되지 않습니다(예: 여름이 아닌 시간에서 후보 시간으로 변경됨).

ovirt_engine_history 데이터베이스에서 사용할 공간과 리소스의 추정치를 계산하려면 RHV Manager History Database Size Calculator 툴을 사용합니다. 추정치는 엔터티 수와 기록 레코드를 유지하기 위해 선택한 시간을 기반으로 합니다.



중요

engine-setup 에서 다음과 같은 동작이 예상됩니다.

- 데이터 웨어하우스 패키지를 설치하고 **engine-setup** 을 실행하고 **No** 로 응답하여 데이터 웨어하우스를 구성합니다.

```
Configure Data Warehouse on this host (Yes, No) [Yes]: No
```

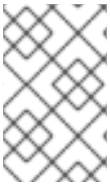
- **engine-setup** 을 다시 실행합니다. 설정해도 더 이상 Data 웨어하우스를 구성할 수 있는 옵션이 표시되지 않습니다.

engine-setup 을 강제 적용하여 옵션을 다시 제공하려면 **engine-setup --reconfigure-optional-components** 를 실행합니다.

현재 설치된 데이터 웨어하우스 패키지만 구성하고 활성화된 리포지토리에 있는 패키지 업데이트를 적용하지 못하도록 하려면 **--offline** 옵션을 추가합니다.

1.2. 별도의 머신에 데이터 웨어하우스 설치 및 구성

이 섹션에서는 Red Hat Virtualization Manager와 별도의 시스템에서 데이터 웨어하우스 서비스를 설치하고 구성하는 방법에 대해 설명합니다. 별도의 시스템에 데이터 웨어하우스를 설치하면 Manager 시스템의 부하를 줄이는 데 도움이 됩니다.



참고

Red Hat은 이러한 구성 요소를 서로 별도의 시스템에 설치할 수 있더라도 데이터 웨어하우스 데이터베이스, 데이터 웨어하우스 서비스 및 Grafana 모두에만 설치할 수 있도록 지원합니다.

사전 요구 사항

- Red Hat Virtualization Manager는 별도의 시스템에 설치됩니다.
- Red Hat Enterprise Linux 8을 실행하는 물리적 서버 또는 가상 시스템.
- Manager 데이터베이스 암호입니다.
- 데이터 웨어하우스 시스템에서 관리자 데이터베이스 시스템의 TCP 포트 5432로 액세스합니다.
- 데이터 웨어하우스 서비스와 별도로 데이터 웨어하우스 데이터베이스를 설치하도록 선택하는 경우 데이터베이스를 먼저 설정해야 합니다. 원격 데이터 웨어하우스 데이터베이스를 수동으로 설치하려면 원격 PostgreSQL 데이터베이스 준비를 참조하십시오. 데이터베이스 시스템에 대한 다음 정보가 있어야 합니다.
  - FQDN
  - 데이터베이스에 도달할 수 있는 포트(기본적으로5432)
  - 데이터베이스 이름
  - 데이터베이스 사용자
  - 데이터베이스 암호
  - **postgresql.conf** 파일을 편집하여 액세스 권한을 수동으로 부여해야 합니다. **/var/lib/pgsql/data/postgresql.conf** 파일을 편집하고 다음과 일치 **하도록 listen_addresses** 행을 수정합니다.

    ```
    listen_addresses = '*'
    ```

    행이 없거나 주석 처리된 경우 수동으로 추가합니다.

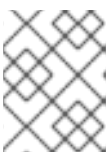
Red Hat Virtualization Manager 리포지토리 활성화

Red Hat Subscription Manager에 데이터 웨어하우스 시스템에 로그인하고 등록하고 **Red Hat Virtualization Manager 서브스크립션을 연결한 다음 Manager** 리포지토리를 활성화해야 합니다.

절차

1. 시스템을 Content Delivery Network에 등록하고 메시지가 표시되면 고객 포털 사용자 이름과 암호를 입력합니다.

   ```
   # subscription-manager register
   ```



참고

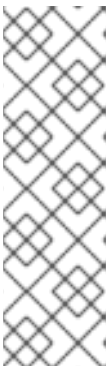
IPv6 네트워크를 사용하는 경우 IPv6 전환 메커니즘을 사용하여 콘텐츠 전달 네트워크 및 서브스크립션 관리자에게 액세스합니다.

2. **Red Hat Virtualization Manager** 서브스크립션 풀을 찾아 풀 ID를 기록합니다.

```
# subscription-manager list --available
```

3. 풀 ID를 사용하여 서브스크립션을 시스템에 연결합니다.

```
# subscription-manager attach --pool=pool_id
```



참고

현재 첨부된 서브스크립션을 보려면 다음을 수행합니다.

```
# subscription-manager list --consumed
```

활성화된 리포지토리를 모두 나열하려면 다음을 수행합니다.

```
# dnf repolist
```

4. 리포지토리를 구성합니다.

```
# subscription-manager repos \
    --disable='*' \
    --enable=rhel-8-for-x86_64-baseos-rpms \
    --enable=rhel-8-for-x86_64-appstream-rpms \
    --enable=rhv-4.4-manager-for-rhel-8-x86_64-rpms
```

5. **pki-deps** 모듈을 활성화합니다.

```
# dnf module -y enable pki-deps
```

6. **postgresql** 모듈 버전 12를 활성화합니다.

```
# dnf module -y enable postgresql:12
```

7. **nodejs** 모듈의 버전 14를 활성화합니다.

```
# dnf module -y enable nodejs:14
```

8. 설치된 패키지를 동기화하여 사용 가능한 최신 버전으로 업데이트합니다.

```
# dnf distro-sync --nobest
```

추가 리소스

모듈 및 모듈 스트림에 대한 자세한 내용은 *사용자 공간 구성 요소 설치, 관리 및 제거* 의 다음 섹션을 참조하십시오.

- 모듈 스트림
- 패키지 설치 전에 스트림 선택
- 모듈 스트림 재설정

- 이후 스트림으로 전환

별도의 머신에 데이터 웨어하우스 설치

절차

1. 데이터베이스를 설치할 시스템에 로그인합니다.
2. 모든 패키지가 최신 상태인지 확인합니다.

```
# dnf upgrade --nobest
```

3. **ovirt-engine-dwh-setup** 패키지를 설치합니다.

```
# dnf install ovirt-engine-dwh-setup
```

4. **engine-setup** 명령을 실행하여 설치를 시작합니다.

```
# engine-setup
```

5. 답변 이 시스템에 데이터 웨어하우스를 설치하려면 **예**:

```
Configure Data Warehouse on this host (Yes, No) [Yes]:
```

6. 답변 **예**: 이 시스템에 Grafana를 설치합니다.

```
Configure Grafana on this host (Yes, No) [Yes]:
```

7. **Enter** 를 눌러 자동으로 감지된 호스트 이름을 승인하거나 대체 호스트 이름을 입력한 후 **Enter** 키를 누릅니다.

```
Host fully qualified DNS name of this server [autodetected hostname]:
```

8. **Enter** 를 눌러 방화벽을 자동으로 구성하거나 **No** 를 **입력하고 Enter** 를 눌러 기존 설정을 유지 관리합니다.

```
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 활성화되지 않은 경우 지원되는 옵션 목록에서 선택한 방화벽 관리자를 선택하라는 메시지가 표시됩니다. 방화벽 관리자 이름을 **입력하고 Enter** 키를 누릅니다. 이는 하나의 옵션만 나열되는 경우에도 적용됩니다.

9. Manager 시스템의 정규화된 도메인 이름을 입력한 다음 **Enter** 키를 누릅니다.

```
Host fully qualified DNS name of the engine server []:
```

10. **Enter** 를 눌러 설정에서 SSH를 통해 Manager의 인증서에 서명할 수 있도록 합니다.

```
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]:
```

11. **Enter** 를 눌러 기본 SSH 포트를 수락하거나 대체 포트 번호를 입력한 다음 **Enter** 키를 누릅니다.

```
ssh port on remote engine server [22]:
```

12. Manager 시스템의 루트 암호를 입력합니다.

```
root password on remote engine server manager.example.com:
```

13. 이 시스템(로컬) 또는 다른 시스템(원격)에서 데이터 웨어하우스 데이터베이스를 호스트할지 여부를 지정합니다.

참고

Red Hat은 이러한 구성 요소를 서로 별도의 시스템에 설치할 수 있더라도 데이터 웨어하우스 데이터베이스, 데이터 웨어하우스 서비스 및 Grafana 모두에만 설치할 수 있도록 지원합니다.

```
Where is the DWH database located? (Local, Remote) [Local]:
```

- **Local** 을 선택하면 **engine-setup** 스크립트가 사용자 및 데이터베이스 추가를 포함하여 데이터베이스를 자동으로 구성하거나 미리 구성된 로컬 데이터베이스에 연결할 수 있습니다.

```
Setup can configure the local postgresql server automatically for the DWH to run. This
may conflict with existing applications.
Would you like Setup to automatically configure postgresql and create DWH database, or
prefer to perform that manually? (Automatic, Manual) [Automatic]:
```

  - **Enter** 를 눌러 **자동** 을 선택하면 여기서 추가 작업이 필요하지 않습니다.
  - **Manual** 을 선택하는 경우 수동으로 구성된 로컬 데이터베이스에 대해 다음 값을 입력합니다.

```
DWH database secured connection (Yes, No) [No]:
DWH database name [ovirt_engine_history]:
DWH database user [ovirt_engine_history]:
DWH database password:
```

14. Manager 데이터베이스 시스템의 정규화된 도메인 이름과 암호를 입력합니다. Manager 데이터베이스가 설치된 동일한 시스템에 데이터 웨어하우스 데이터베이스를 설치하는 경우 동일한 FQDN을 사용합니다. **Enter** 를 눌러 서로 필드의 기본값을 적용합니다.

```
Engine database host []: engine-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
```

15. 수집된 데이터를 보관할 데이터 웨어하우스를 선택합니다.

```
Please choose Data Warehouse sampling scale:
(1) Basic
(2) Full
(1, 2)[1]:
```

full 는 ovirt-engine-dwhd.conf(데이터 웨어하우스가 원격 호스트에 설치될 때 권장)의 데이터 웨어하우스 서비스에 대해 애플리케이션 스토리지 설정에 대한 기본값을 사용합니다.

기본 설정은 **DWH_TABLES_KEEP_HOURLY** 값을 **720** 으로 줄이고 **DWH_TABLES_KEEP_DAILY를 0** 으로 줄여 Manager 시스템의 부하를 줄입니다. Manager 및 Data 웨어하우스가 동일한 시스템에 설치된 경우 **Basic** 을 사용합니다.

16. 설치 설정을 확인합니다.

```
Please confirm installation settings (OK, Cancel) [OK]:
```

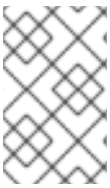
17. 데이터 웨어하우스 구성이 완료되면 Red Hat Virtualization Manager에서 **ovirt-engine** 서비스를 다시 시작하십시오.

```
# systemctl restart ovirt-engine
```

18. 선택적으로 링크의 지침을 사용하여 데이터베이스 연결을 보호하도록 SSL을 설정합니다 . https://www.postgresql.org/docs/12/ssl-tcp.html#SSL-FILE-USAGE.

2장. 별도의 머신으로 데이터 SOFTWARE SOFTWARE TO A SEPARATE MACHINE으로 마이그레이션

이 섹션에서는 데이터 웨어하우스 데이터베이스 및 서비스를 Red Hat Virtualization Manager 시스템에서 별도의 시스템으로 마이그레이션하는 방법에 대해 설명합니다. 별도의 시스템에서 데이터 웨어하우스 서비스를 호스팅하면 각 개별 시스템의 부하가 줄어들고 CPU 및 메모리 리소스를 다른 프로세스와 공유하여 발생할 수 있는 충돌을 방지합니다.



참고

Red Hat은 이러한 구성 요소를 서로 별도의 시스템에 설치할 수 있더라도 데이터 웨어하우스 데이터베이스, 데이터 웨어하우스 서비스 및 Grafana 모두에만 설치할 수 있도록 지원합니다.

다음과 같은 마이그레이션 옵션이 있습니다.

- Manager 시스템에서 데이터 웨어하우스 서비스를 마이그레이션하고 기존 데이터 웨어하우스 데이터베이스(**ovirt_engine_history**)와 연결할 수 있습니다.
- Manager 시스템에서 데이터 웨어하우스 데이터베이스를 마이그레이션한 다음 데이터 웨어하우스 서비스를 마이그레이션할 수 있습니다.

2.1. 별도의 머신으로 데이터 웨어하우스 데이터베이스 마이그레이션

데이터 웨어하우스 서비스를 마이그레이션하기 전에 데이터 웨어하우스 데이터베이스(**ovirt_engine_history**)를 마이그레이션합니다. **engine-backup** 을 사용하여 데이터베이스 백업을 생성하고 새 데이터베이스 시스템에서 복원합니다. engine **-backup** 에 대한 자세한 내용을 보려면 **engine-backup --help** 를 실행합니다.

참고

Red Hat은 이러한 구성 요소를 서로 별도의 시스템에 설치할 수 있더라도 데이터 웨어하우스 데이터베이스, 데이터 웨어하우스 서비스 및 Grafana 모두에만 설치할 수 있도록 지원합니다.

새 데이터베이스 서버에 Red Hat Enterprise Linux 8이 설치되어 있어야 합니다.

새 데이터베이스 서버에서 필요한 리포지토리를 활성화합니다.

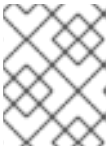
2.1.1. Red Hat Virtualization Manager 리포지토리 활성화

Red Hat Subscription Manager에 데이터 웨어하우스 시스템에 로그인하고 등록하고 **Red Hat Virtualization Manager 서브스크립션을 연결한 다음 Manager** 리포지토리를 활성화해야 합니다.

절차

1. 시스템을 Content Delivery Network에 등록하고 메시지가 표시되면 고객 포털 사용자 이름과 암호를 입력합니다.

```
# subscription-manager register
```



참고

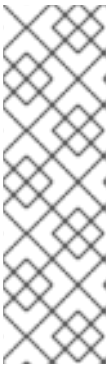
IPv6 네트워크를 사용하는 경우 IPv6 전환 메커니즘을 사용하여 콘텐츠 전달 네트워크 및 서브스크립션 관리자에게 액세스합니다.

2. **Red Hat Virtualization Manager** 서브스크립션 풀을 찾아 풀 ID를 기록합니다.

```
# subscription-manager list --available
```

3. 풀 ID를 사용하여 서브스크립션을 시스템에 연결합니다.

```
# subscription-manager attach --pool=pool_id
```



참고

현재 첨부된 서브스크립션을 보려면 다음을 수행합니다.

```
# subscription-manager list --consumed
```

활성화된 리포지토리를 모두 나열하려면 다음을 수행합니다.

```
# dnf repolist
```

4. 리포지토리를 구성합니다.

```
# subscription-manager repos \
    --disable='*' \
    --enable=rhel-8-for-x86_64-baseos-rpms \
    --enable=rhel-8-for-x86_64-appstream-rpms \
    --enable=rhv-4.4-manager-for-rhel-8-x86_64-rpms
```

5. **postgresql** 모듈 버전 12를 활성화합니다.

```
# dnf module -y enable postgresql:12
```

6. **nodejs** 모듈의 버전 14를 활성화합니다.

```
# dnf module -y enable nodejs:14
```

7. 설치된 패키지를 동기화하여 사용 가능한 최신 버전으로 업데이트합니다.

```
# dnf distro-sync --nobest
```

추가 리소스

모듈 및 모듈 스트림에 대한 자세한 내용은 *사용자 공간 구성 요소 설치, 관리 및 제거* 의 다음 섹션을 참조하십시오.

- 모듈 스트림
- 패키지 설치 전에 스트림 선택

- 모듈 스트림 재설정
- 이후 스트림으로 전환

2.1.2. 별도의 머신으로 데이터 웨어하우스 데이터베이스 마이그레이션

절차

1. Manager에서 데이터 웨어하우스 데이터베이스 및 구성 파일의 백업을 생성합니다.

```
# engine-backup --mode=backup --scope=dwhdb --scope=files --file=file_name --
log=log_file_name
```

2. Manager의 백업 파일을 새 시스템으로 복사합니다.

```
# scp /tmp/file_name root@new.dwh.server.com:/tmp
```

3. 새 시스템에 **engine-backup** 을 설치합니다.

```
# dnf install ovirt-engine-tools-backup
```

4. PostgreSQL 서버 패키지를 설치합니다.

```
# dnf install postgresql-server postgresql-contrib
```

5. PostgreSQL 데이터베이스를 초기화하고, **postgresql** 서비스를 시작한 다음, 이 서비스가 부팅 시 시작되는지 확인합니다.

```
# su - postgres -c 'initdb'
# systemctl enable postgresql
# systemctl start postgresql
```

6. 새 시스템에서 데이터 웨어하우스 데이터베이스를 복원합니다. *file_name* 은 Manager에서 복사한 백업 파일입니다.

```
# engine-backup --mode=restore --scope=files --scope=dwhdb --file=file_name --
log=log_file_name --provision-dwh-db --restore-permissions
```

데이터 웨어하우스 데이터베이스는 이제 Manager가 호스팅되는 별도의 시스템에서 호스팅됩니다. 데이터 웨어하우스 데이터베이스를 복원한 후 프롬프트에서 **engine-setup** 명령을 실행하도록 지시합니다. 이 명령을 실행하기 전에 데이터 웨어하우스 서비스를 마이그레이션합니다.

2.2. 별도의 머신으로 데이터 웨어하우스 서비스 마이그레이션

Red Hat Virtualization Manager에 설치 및 구성된 데이터 웨어하우스 서비스를 별도의 시스템으로 마이그레이션할 수 있습니다. 별도의 시스템에서 데이터 웨어하우스 서비스를 호스팅하면 Manager 시스템의 부하를 줄이는 데 도움이 됩니다.

이 절차에서는 데이터 웨어하우스 서비스만 마이그레이션합니다.

데이터 웨어하우스 서비스를 마이그레이션하기 전에 데이터 웨어하우스 데이터베이스 (**ovirt_engine_history**)를 마이그레이션하려면 데이터장 데이터베이스를 분리 머신으로 마이그레이션을 참조하십시오.

참고

Red Hat은 이러한 구성 요소를 서로 별도의 시스템에 설치할 수 있더라도 데이터 웨어하우스 데이터베이스, 데이터 웨어하우스 서비스 및 Grafana 모두에만 설치할 수 있도록 지원합니다.

사전 요구 사항

- 동일한 시스템에 Manager 및 Data 웨어하우스를 설치하고 구성해야 합니다.
- 새 데이터 웨어하우스 시스템을 설정하려면 다음이 있어야 합니다.
  - Manager의 **/etc/ovirt-engine/engine.conf.d/10-setup-database.conf** 파일의 암호입니다.
  - 데이터 웨어하우스 시스템에서 관리자 데이터베이스 시스템의 TCP 포트 5432로 액세스할 수 있습니다.
  - Manager의 **/etc/ovirt-engine-dwh/ovirt-engine-dwhd.conf.d/10-setup-database.conf** 파일에서 데이터 웨어하우스 데이터베이스의 사용자 이름과 암호.
    데이터 웨어하우스 데이터베이스를 분리 머신으로 마이그레이션하는 데 설명된 절차를 사용하여 **ovirt_engine_history** 데이터베이스 를 마이그레이션하면 해당 시스템에 데이터베이스 설정 중에 정의한 이러한 자격 증명이 포함됩니다.

이 시나리오를 설치하려면 다음 네 가지 단계가 필요합니다.

1. 새 데이터 웨어하우스 머신 설정
2. Manager 시스템에서 데이터 installer 서비스 중지
3. 새 데이터 저장소 머신 구성
4. Manager 시스템에서 데이터 installer 패키지 비활성화

2.2.1. 새 데이터 웨어하우스 머신 설정

Red Hat Virtualization 리포지토리를 활성화하고 Red Hat Enterprise Linux 8 시스템에 데이터 웨어하우스 설치 패키지를 설치합니다.

1. 필요한 리포지토리를 활성화합니다.
   a. 시스템을 Content Delivery Network에 등록하고 메시지가 표시되면 고객 포털 사용자 이름과 암호를 입력합니다.

   ```
   # subscription-manager register
   ```

   b. **Red Hat Virtualization Manager** 서브스크립션 풀을 찾아 풀 ID를 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-8-for-x86_64-baseos-rpms \
    --enable=rhel-8-for-x86_64-appstream-rpms \
    --enable=rhv-4.4-manager-for-rhel-8-x86_64-rpms \
    --enable=fast-datapath-for-rhel-8-x86_64-rpms \
    --enable=jb-eap-7.4-for-rhel-8-x86_64-rpms
```

2. **pki-deps** 모듈을 활성화합니다.

```
# dnf module -y enable pki-deps
```

3. 현재 설치된 모든 패키지가 최신 상태인지 확인합니다.

```
# dnf upgrade --nobest
```

4. **ovirt-engine-dwh-setup** 패키지를 설치합니다.

```
# dnf install ovirt-engine-dwh-setup
```

2.2.2. 관리자 시스템에서 데이터 웨어하우스 서비스 중지

절차

1. 데이터 웨어하우스 서비스를 중지합니다.

```
# systemctl stop ovirt-engine-dwhd.service
```

2. 데이터베이스가 원격 시스템에서 호스팅되는 경우 postgres.conf 파일을 편집하여 액세스 권한을 수동으로 부여해야 합니다. **/var/lib/pgsql/data/postgresql.conf** 파일을 편집하고 다음과 일치하도록 listen_addresses 행을 수정합니다.

```
listen_addresses = '*'
```

행이 없거나 주석 처리된 경우 수동으로 추가합니다.

데이터베이스가 Manager 시스템에서 호스팅되고 Red Hat Virtualization Manager를 새로 설정하는 동안 구성된 경우 기본적으로 액세스 권한이 부여됩니다.

3. postgresql 서비스를 다시 시작하십시오.

```
# systemctl restart postgresql
```

2.2.3. 새 데이터 웨어하우스 머신 구성

이 섹션에 표시된 옵션 또는 설정의 순서는 환경에 따라 다를 수 있습니다.

1. **ovirt_engine_history** 데이터베이스와 데이터 웨어하우스 서비스를 모두 **동일한** 시스템으로 마이그레이션하는 경우 다음을 실행하십시오. 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 데이터 웨어하우스 구성을 시작합니다.

```
# engine-setup
```

3. **Enter** 를 눌러 자동으로 감지된 호스트 이름을 승인하거나 대체 호스트 이름을 입력한 후 **Enter** 키를 누릅니다.

```
Host fully qualified DNS name of this server [autodetected host name]:
```

4. **Enter** 를 눌러 방화벽을 자동으로 구성하거나 **No** 를 **입력하고 Enter** 를 눌러 기존 설정을 유지 관리합니다.

```
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 활성화되지 않은 경우 지원되는 옵션 목록에서 선택한 방화벽 관리자를 선택하라는 메시지가 표시됩니다. 방화벽 관리자 이름을 **입력하고 Enter** 키를 누릅니다. 이는 하나의 옵션만 나열되는 경우에도 적용됩니다.

5. 관리자에 대해 정규화된 도메인 이름과 암호를 입력합니다. **Enter** 를 눌러 서로 필드의 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
```

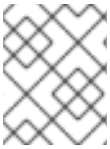
6. Manager 데이터베이스 시스템의 FQDN 및 암호를 입력합니다. **Enter** 를 눌러 서로 필드의 기본값을 적용합니다.

```
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
```

7. 설치 설정을 확인합니다.

```
Please confirm installation settings (OK, Cancel) [OK]:
```

데이터 웨어하우스 서비스가 이제 원격 시스템에 구성되어 있습니다. Manager 시스템에서 데이터 웨어하우스 서비스를 비활성화합니다.



참고

원격 서버의 권장되는 스케일로 데이터 웨어하우스 샘플링 스케일링을 변경하려면 데이터 검사기 샘플링 확장 변경을 참조하십시오.

2.2.4. 관리자 시스템에서 데이터 웨어하우스 서비스 비활성화

절차

1. Manager 시스템에서 Manager를 다시 시작합니다.

```
# service ovirt-engine restart
```

2. 다음 명령을 실행하여 **/etc/ovirt-engine-setup.conf.d/20-setup-ovirt-post.conf** 파일을 수정하고 옵션을 **False로 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

# sed -i \
    -e 's;^\(OVESETUP_GRAFANA_CORE/enable=bool\):True;\1:False;' \
    /etc/ovirt-engine-setup.conf.d/20-setup-ovirt-post.conf
```

3. 데이터 웨어하우스 서비스를 비활성화합니다.

```
# systemctl disable ovirt-engine-dwhd.service
```

4. 데이터 웨어하우스 파일을 제거합니다.

```
# rm -f /etc/ovirt-engine-dwh/ovirt-engine-dwhd.conf.d/*.conf /var/lib/ovirt-engine-dwh/backups/*
```

데이터 웨어하우스 서비스는 이제 Manager와 별도의 시스템에서 호스팅됩니다.

2.3. 데이터 웨어하우스 샘플링 스케일 변경

Red Hat Virtualization에는 데이터 웨어하우스가 필요합니다. Manager와 동일한 시스템 또는 Manager에 액세스할 수 있는 별도의 시스템에 설치하고 구성할 수 있습니다. 모든 설정에는 기본 데이터 보존 설정이 필요하지 않을 수 있으므로 **engine-setup** 은 두 가지 데이터 샘플링 스케일링을 제공합니다. **기본** 및 **전체**.

- **Full** 은 데이터 웨어하우스 서비스에 대해 애플리케이션 설정에 나열된 데이터 보존 설정의 기본값을 사용합니다(데이터 웨어하우스가 원격 호스트에 설치된 경우 권장됨).

- **기본** 설정은 **DWH_TABLES_KEEP_HOURLY** 값을 **720** 으로 줄이고 **DWH_TABLES_KEEP_DAILY를 0** 으로 줄여 Manager 시스템의 부하를 줄입니다. Manager 및 Data 웨어하우스가 동일한 시스템에 설치된 경우 **Basic** 을 사용합니다.

샘플링 스케일링은 설치 중에 **engine-setup** 에 의해 구성됩니다.

```
--== MISC CONFIGURATION ==--

Please choose Data Warehouse sampling scale:
(1) Basic
(2) Full
(1, 2)[1]:
```

나중에 **--reconfigure-dwh-scale** 옵션을 사용하여 **engine-setup** 을 다시 실행하여 샘플링 스케일링을 변경할 수 있습니다.

데이터 웨어하우스 샘플링 스케일 변경

```
# engine-setup --reconfigure-dwh-scale
[...]
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]:
[...]
Perform full vacuum on the oVirt engine history
database ovirt_engine_history@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/12/static/sql-vacuum.html
(Yes, No) [No]:
[...]
Setup can backup the existing database. The time and space required for the database backup
depend on its size. This process takes time, and in some cases (for instance, when the size is few
GBs) may take several hours to complete.
If you choose to not back up the database, and Setup later fails for some reason, it will not be able to
restore the database and all DWH data will be lost.
Would you like to backup the existing database before upgrading it? (Yes, No) [Yes]:
[...]
Please choose Data Warehouse sampling scale:
(1) Basic
(2) Full
(1, 2)[1]: 2
[...]
During execution engine service will be stopped (OK, Cancel) [OK]:
[...]
Please confirm installation settings (OK, Cancel) [OK]:
```

필요한 경우 데이터 웨어하우스 서비스의 애플리케이션 설정에 설명된 대로 개별 데이터 보존 설정을 조정할 수도 있습니다.

3장. 기록 데이터베이스 정보

3.1. 기록 데이터베이스 개요

Red Hat Virtualization에는 애플리케이션을 보고하여 데이터 센터, 클러스터 및 호스트 수준에서 보고서를 생성하는 데 사용할 수 있는 포괄적인 관리 이력 데이터베이스가 포함되어 있습니다. 이 장에서는 기록 데이터베이스에 대한 쿼리를 설정할 수 있는 정보를 제공합니다.

Red Hat Virtualization Manager는 **PostgreSQL 12.x** 를 데이터베이스 플랫폼으로 사용하여 가상화 환경 상태, 구성 및 성능에 대한 정보를 저장합니다. 설치 시 Red Hat Virtualization Manager는 **engine** 이라는 PostgreSQL 데이터베이스를 만듭니다.

ovirt-engine-dwh 패키지를 설치하면 **ovirt_engine_history** 라는 두 번째 데이터베이스가 생성됩니다. 이 데이터베이스에는 **엔진** 운영 데이터베이스에서 1분마다 수집한 기록 구성 정보와 통계 지표가 포함됩니다. 데이터베이스 변경 사항을 추적하면 데이터베이스의 오브젝트에 대한 정보가 제공되므로 사용자가 활동을 분석하고 성능을 향상하며 문제를 해결할 수 있습니다.

주의

ovirt_engine_history 데이터베이스의 데이터 복제는 Red Hat Virtualization Manager Extract Transform Load Service, **ovirt-engine-dwhd** 를 통해 수행됩니다. 이 서비스는 데이터 통합 도구인 Talend Open Studio를 기반으로 합니다. 이 서비스는 데이터 웨어하우스 패키지 설정 중에 자동으로 시작되도록 구성됩니다. **엔진** 데이터베이스에서 데이터를 추출하여 데이터를 기록 데이터베이스 표준으로 변환하고 **ovirt_engine_history** 데이터베이스로 로드하는 Java 프로그램입니다.

ovirt-engine-dwhd 서비스를 중지하면 안 됩니다.

ovirt_engine_history 데이터베이스 스키마는 시간이 지남에 따라 변경됩니다. 데이터베이스에는 일관된 구조와 함께 지원되는 버전 API를 제공하는 데이터베이스 뷰 세트가 포함되어 있습니다. 보기는 데이터베이스 쿼리의 결과 집합으로 구성된 가상 테이블입니다. 데이터베이스는 뷰의 정의를 **SELECT** 문으로 저장합니다. **SELECT** 문의 결과는 보기에서 반환하는 가상 테이블을 채웁니다. 사용자가 표를 참조하는 방식과 동일한 **방식으로 SM/PGSQL** 문의 보기 이름을 참조합니다.

3.2. 구성 내역 추적

Red Hat Virtualization History Database(ovirt **_engine_history**라고 함)의 데이터를 사용하여 **엔진** 데이터베이스를 추적할 수 있습니다.

ETL 서비스 **ovirt-engine-dwhd는** 다음 세 가지 유형의 변경 사항을 추적합니다.

- 새 엔터티가 **engine** 데이터베이스에 추가됩니다. ETL 서비스는 **ovirt_engine_history** 데이터베이스의 변경 사항을 새 항목으로 복제합니다.
- 기존 엔터티가 업데이트됨 - ETL 서비스는 **ovirt_engine_history** 데이터베이스에 대한 변경 사항을 새 항목으로 복제합니다.
- 엔터티가 **엔진** 데이터베이스에서 제거됩니다. **ovirt_engine_history** 데이터베이스의 새 항목은 제거된 대로 해당 엔터티를 플래그로 지정합니다. 제거된 엔터티는 제거된 항목으로만 플래그가 지정됩니다.

ovirt_engine_history 데이터베이스의 구성 테이블은 여러 가지 면에서 **engine** 데이터베이스의 해당 테이블과 다릅니다. 가장 분명한 차이점은 **구성** 열이 적은 수입니다. 이는 특정 구성 항목이 다른 구성 항목보다 덜 관심을 갖고 데이터베이스 크기 고려 사항으로 인해 유지되지 않기 때문입니다. 또한 **엔진** 데이터베이스의 몇 가지 테이블의 열이 **ovirt_engine_history** 의 단일 테이블에 표시되고 다른 열 이름이 있어 데이터를 보다 쉽고 이해하기 쉽게 확인할 수 있습니다. 모든 구성 테이블에는 다음이 포함됩니다.

- 엔터티의 구성 버전을 나타내는 **history_id**;
- 엔터티가 시스템에 추가된 시기를 나타내는 **create_date** 필드
- 엔터티가 변경된 시기를 나타내는 **update_date** 필드
- 엔터티가 시스템에서 제거된 날짜를 나타내는 **delete_date** 필드입니다.

3.3. 통계 기록 기록

ETL 서비스는 1분마다 통계를 표로 수집합니다. 지난 24시간 동안 최소 24시간마다 데이터가 저장되지만 삭제 작업이 마지막으로 실행된 시간에 따라 48시간 동안 저장할 수 있습니다. 2시간 이상 경과한 분별 데이터는 매시간 데이터로 집계되어 2개월 동안 저장됩니다. 2일 이상 경과한 시간별 데이터가 일별 데이터로 수집되어 5년 동안 저장됩니다.

시간별 데이터 및 일일 데이터는 시간별 및 일별 테이블에서 확인할 수 있습니다.

각 통계 점들은 샘플, 시간별, 일별 기록 등 각 집계 수준 테이블에 유지됩니다. 모든 기록 테이블에는 행을 고유하게 식별하는 **history_id** 열도 포함되어 있습니다. 테이블은 과거 구성과 관련된 엔터티의 통계에 대한 보고서를 활성화하기 위해 호스트의 구성 버전을 참조합니다.

3.4. OVIRT-ENGINE-DWHD.CONF의 데이터 웨어하우스 서비스의 애플리케이션 설정

다음은 데이터 웨어하우스 서비스에 대한 애플리케이션 설정을 구성하는 옵션 목록입니다. 이러한 옵션은 **/usr/share/ovirt-engine-dwh/services/ovirt-engine-dwhd/ovirt-engine-dwhd.conf** 파일에서 사용할 수 있습니다. **/etc/ovirt-engine-dwh/ovirt-engine-dwhd.conf.d/ 아래의 덮어쓰기 파일의 기본값을 구성합니다.** 변경 사항을 저장한 후 데이터 웨어하우스 서비스를 다시 시작합니다.

표 3.1. ovirt-engine-dwhd.conf 애플리케이션 설정 변수

변수 이름	기본값	비고
DWH_DELETE_JOB_HOUR	**3**	삭제 작업이 실행되는 시간입니다. **0** 에서 **23** 사이의 값을 지정합니다. 여기서 **0** 은 자정입니다.
DWH_SAMPLING	**60**	통계 테이블에 데이터가 수집된 간격(초)입니다.
DWH_TABLES_KEEP_SAMPLES	**24**	**DWH_SAMPLING** 의 데이터가 저장된 시간입니다. 2시간 이상의 데이터가 시간별 데이터로 집계됩니다.

변수 이름	기본값	비고
DWH_TABLES_KEEP_HOURLY	**1440**	매시간 데이터가 저장되는 시간 수입니다. 기본값은 60일입니다. 2일 이상 경과한 시간별 데이터가 일별 데이터로 집계됩니다.
DWH_TABLES_KEEP_DAILY	**43800**	매일 데이터가 저장되는 시간 수입니다. 기본값은 5년입니다.
DWH_ERROR_EVENT_INTERVAL	**300000**	오류가 Manager의 audit. **log** 로 푸시되는 최소 간격(밀리초)입니다.

3.5. 추적 태그 기록

ETL 서비스는 1분마다 관리 포털에 표시된 태그 정보를 수집하고 이 데이터를 기록 테이블에 저장합니다. ETL 서비스는 5가지 유형의 변경을 추적합니다.

- 태그는 관리 포털에 생성됩니다. ETL 서비스는 태그 세부 정보, 태그 트리의 위치 및 태그 트리의 다른 개체와 관계됩니다.
- 엔터티는 관리 포털의 태그 트리에 연결되어 있습니다. ETL 서비스는 **ovirt_engine_history** 데이터베이스에 추가된 내용을 새 항목으로 복제합니다.
- 태그가 업데이트됨 - ETL 서비스는 **ovirt_engine_history** 데이터베이스에 태그 세부 정보 변경 사항을 새 항목으로 복제합니다.
- 엔터티 또는 태그 분기는 관리 포털에서 제거됩니다. **ovirt_engine_history** 데이터베이스는 새 항목에서 제거된 해당 태그 및 관계를 플래그로 지정합니다. 제거된 태그와 관계는 제거 또는 분리로만 플래그가 지정됩니다.
- 태그 분기가 이동됨 - 해당 태그와 관계가 새 항목으로 업데이트됩니다. 이동 태그와 홍보는 업데이트로만 플래그가 지정됩니다.

3.6. 기록 데이터베이스에 대한 읽기 전용 액세스 허용

편집을 허용하지 않고 기록 데이터베이스에 액세스할 수 있도록 하려면 **ovirt_engine_history** 데이터베이스에 로그인하고 읽을 수 있는 읽기 전용 PostgreSQL 사용자를 만들어야 합니다. 이 절차는 히스토리 데이터베이스가 설치된 시스템에서 실행해야 합니다.

기록 데이터베이스에 대한 읽기 전용 액세스 허용

1. postgres 사용자 쉘에 root로 로그인합니다.

```
# su - postgres
```

2. 히스토리 데이터베이스에 대한 읽기 전용 액세스 권한을 부여할 사용자를 생성합니다.

```
# psql -U postgres -c "CREATE ROLE username WITH LOGIN ENCRYPTED PASSWORD
'password';" -d ovirt_engine_history
```

3. 새로 생성된 사용자에게 기록 데이터베이스에 연결할 수 있는 권한을 부여합니다.

```
# psql -U postgres -c "GRANT CONNECT ON DATABASE ovirt_engine_history TO
username;"
```

4. 새로 생성된 사용자 사용을 **공개** 스키마에 부여합니다.

```
# psql -U postgres -c "GRANT USAGE ON SCHEMA public TO username;"
ovirt_engine_history
```

5. 새로 생성된 사용자에게 부여할 나머지 권한을 생성하여 파일에 저장합니다.

```
# psql -U postgres -c "SELECT 'GRANT SELECT ON ' || relname || ' TO username;' FROM
pg_class JOIN pg_namespace ON pg_namespace.oid = pg_class.relnamespace WHERE
nspname = 'public' AND relkind IN ('r', 'v');" --pset=tuples_only=on  ovirt_engine_history >
grant.sql
```

6. 이전 단계에서 만든 파일을 사용하여 새로 만든 사용자에게 권한을 부여합니다.

```
# psql -U postgres -f grant.sql ovirt_engine_history
```

7. 새로 생성된 사용자에게 권한을 부여하는 데 사용한 파일을 제거합니다.

```
# rm grant.sql
```

8. **Ctrld**를 눌러 postgres 사용자 쉘을 종료합니다.

9. 새로 생성된 사용자에 대해 **/var/lib/pgsql/data/pg_hba.conf에 대해 다음 행을 모두 로컬로 시작하는 행 앞의 /var/lib/pgsql/data/pg_hba.conf** 에 추가합니다.

```
# TYPE  DATABASE              USER         ADDRESS              METHOD
host    ovirt_engine_history  username     0.0.0.0/0              md5
host    ovirt_engine_history  username     ::0/0                  md5
local   all           all                                peer
```

10. PostgreSQL 서비스를 다시 로드합니다.

```
# systemctl reload postgresql
```

11. 읽기 전용 사용자의 액세스 권한을 테스트하려면 다음을 수행합니다.

```
# su - postgres -c 'psql -U username ovirt_engine_history -h localhost'
Password for user username:
psql (9.2.23)
Type "help" for help.

ovirt_engine_history=>
```

12. **ovirt_engine_history** 데이터베이스를 종료하려면 **\q** 를 입력합니다.

수정이 실패하는 동안 **ovirt_engine_history** 데이터베이스의 테이블 및 뷰에 대한 읽기 전용 사용자의 **SELECT** 문은 성공합니다.

3.7. 통계 이력 보기

통계 데이터는 **시간별, 일별 , 샘플 보기에서 사용할 수 있습니다.**

통계 뷰를 쿼리하려면 SELECT * FROM *view_name*_[hourly|daily|samples]; 을 실행합니다. 예를 들면 다음과 같습니다.

```
# SELECT * FROM v4_4_statistics_hosts_resources_usage_daily;
```

사용 가능한 모든 보기를 나열하려면 다음을 실행합니다.

```
# \dv
```

3.7.1. 디버그 모드 활성화

디버그 모드를 활성화하여 /var/log/ovirt-engine-dwh/ovirt-engine-dwh/ovirt-engine-dwhd.log 파일에 로그 샘플링, 시간별, 일상 작업 시간을 기록할 수 있습니다. 이는 ETL 프로세스를 확인하는 데 유용합니다. 디버그 모드는 기본적으로 비활성화되어 있습니다.

1. **Manager 시스템에 로그인하고 구성 파일(예: /etc/ovirt-engine-dwh/ovirt-engine-dwhd.conf.d/logging.conf)을 만듭니다.**
2. **구성 파일에 다음 행을 추가합니다.**

   ```
   DWH_AGGREGATION_DEBUG=true
   ```

3. **ovirt-engine-dwhd 서비스를 다시 시작합니다.**

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

   ```
   To disable debug mode, delete the configuration file and restart the service.
   // removed note
   ```

3.7.2. 스토리지 도메인 통계 보기

표 3.2. 시스템의 각 스토리지 도메인에 대한 기록 통계

이름	유형	설명	인덱싱됨
history_id	BigInt	표에서 이 행의 고유 ID입니다.	없음

이름	유형	설명	인덱싱됨
history_datetime	date	이 기록 행의 타임스탬프(분, 시간, 일로 집계 수준에 따라).	있음
storage_domain_id	UUID	시스템의 스토리지 도메인의 고유 ID입니다.	있음
storage_domain_status	smallint	스토리지 도메인 상태입니다.	없음
seconds_in_status	integer	스토리지 도메인이 집계 기간의 상태 열에 표시된 대로 상태가 표시된 총 시간(초)입니다. 예를 들어 스토리지 도메인이 55초 동안 "Active"이고 1분 동안 5초 동안 "비활성"이면 동일한 분 동안 테이블에 두 개의 행이 보고됩니다. 한 행은 seconds_in_status가 55인 Active(활성) 상태가 되고 다른 행은 Inactive(비활성)이고 seconds_in_status는 5입니다.	없음
minutes_in_status	숫자(7,2)	집계 기간의 상태 열에 표시된 대로 스토리지 도메인이 상태에 있는 총 분(분)입니다. 예를 들어 스토리지 도메인이 55분 동안 "Active"이고 한 시간 동안 5분 동안 "비활성"인 경우 동일한 시간 동안 테이블에 두 개의 행이 보고됩니다. 한 행의 상태가 minutes_in_status 55이고 다른 행은 Inactive(비활성)이고 minutes_in_status는 5입니다.	없음
available_disk_size_gb	integer	디스크에서 사용 가능한 총 용량(GB)으로 표시됩니다.	없음
used_disk_size_gb	integer	디스크에 사용된 총 용량(GB) 단위로 표현됩니다.	없음

이름	유형	설명	인덱싱됨
storage_configuration_version	integer	샘플 시 스토리지 도메인 구성 버전입니다. 이는 **v4_4_configuration_history_storage_domains 보기의 history_ id** 값과 동일하며 결합하는 데 사용할 수 있습니다.	있음

3.7.3. 호스트 통계 뷰

표 3.3. 시스템의 각 호스트에 대한 기록 통계

이름	유형	설명	인덱싱됨
history_id	BigInt	표에서 이 행의 고유 ID입니다.	없음
history_datetime	date	이 기록 행의 타임스탬프(분, 시간, 일로 집계 수준에 따라).	있음
host_id	UUID	시스템에서 호스트의 고유 ID입니다.	있음
host_status	smallint	• -1 - **알 수 없는 상태** (ETL 문제를 나타내는 데만 사용됩니다. Red Hat 지원부에 알려주십시오.) • 1 - **위로** • 2 - **유지 관리** • 3 - **문제 발생**	없음

이름	유형	설명	인덱싱됨
seconds_in_status	integer	호스트가 집계 기간의 상태 열에 표시된 총 시간(초)입니다. 예를 들어, 1분 동안 5초 동안 호스트가 55초 동안 가동된 경우 이 분 동안 두 개의 행이 표시됩니다. 한 개는 **Up** 및 seconds_in_status 55의 상태가 되고 다른 하나는 **Down** 으로, seconds_in_status는 5가 됩니다.	없음
minutes_in_status	숫자(7,2)	호스트가 집계 기간의 상태 열에 표시된 상태에 있는 총 분(분)입니다. 예를 들어 한 호스트가 한 시간 동안 5분 동안 55분 동안 가동된 경우 이 시간 동안 두 개의 행이 표시됩니다. 한 개는 **Up** 및 minutes_in_status 55의 상태가 되고 다른 하나는 **Down** 상태이고 minutes_in_status는 5입니다.	없음
memory_usage_percent	smallint	호스트에서 사용된 메모리의 백분율입니다.	없음
max_memory_usage	smallint	집계 기간의 최대 메모리 사용량은 백분율로 표시됩니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
ksm_shared_memory_mb	BigInt	호스트가 사용 중인 커널 공유 메모리 크기(MB)입니다.	없음
max_ksm_shared_memory_mb	BigInt	집계 기간의 최대 KSM 메모리 사용량은 메가바이트(MB)로 표현됩니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음

이름	유형	설명	인덱싱됨
cpu_usage_percent	smallint	호스트에서 사용된 CPU 백분율입니다.	없음
max_cpu_usage	smallint	집계 기간의 최대 CPU 사용량은 백분율로 표시됩니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
ksm_cpu_percent	smallint	호스트의 CPU 백분율 ksm은.	없음
max_ksm_cpu_percent	smallint	집계 기간의 최대 KSM 사용량이 백분율로 표시됩니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
active_vms	smallint	이 집계의 평균 활성 가상 시스템 수입니다.	없음
max_active_vms	smallint	집계 기간 동안 최대 활성 가상 시스템 수입니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
total_vms	smallint	이 집계의 호스트에 있는 모든 가상 시스템의 평균 수입니다.	없음
max_total_vms	smallint	집계 기간 동안 최대 총 가상 머신 수입니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
total_vms_vcpus	integer	호스트에 할당된 총 vCPU 수.	없음

이름	유형	설명	인덱싱됨
max_total_vms_vcpus	integer	집계 기간 동안 최대 총 가상 머신 vCPU 수입니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
cpu_load	integer	호스트의 CPU 로드입니다.	없음
max_cpu_load	integer	집계 기간의 최대 CPU 로드입니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
system_cpu_usage_percent	smallint	호스트에서 사용된 CPU 백분율입니다.	없음
max_system_cpu_usage_percent	smallint	집계 기간의 최대 시스템 CPU 사용량은 백분율로 표시됩니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
user_cpu_usage_percent	smallint	호스트에서 사용된 사용자 CPU 백분율.	없음
max_user_cpu_usage_percent	smallint	집계 기간의 최대 사용자 CPU 사용량은 백분율로 표시됩니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
swap_used_mb	integer	사용된 호스트의 스왑 크기 사용량(MB).	없음

이름	유형	설명	인덱싱됨
max_swap_used_mb	integer	집계 기간(MB) 단위로 호스트의 최대 스왑 크기 사용량이 백분율로 표시됩니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
host_configuration_version	integer	샘플 시 호스트 구성 버전입니다. 샘플 시 호스트 구성 버전입니다. 이는 **v4_4_configuration_history_hosts 보기의 history_ id** 값과 동일하며 이를 결합하는 데 사용할 수 있습니다.	있음

3.7.4. 호스트 인터페이스 통계 보기

표 3.4. 시스템의 각 호스트 네트워크 인터페이스에 대한 기록 통계

이름	유형	설명	인덱싱됨
history_id	BigInt	표에서 이 행의 고유 ID입니다.	없음
history_datetime	date	이 기록 뷰의 타임스탬프(분, 시간, 일)는 집계 수준에 따라 다릅니다.	있음
host_interface_id	UUID	시스템에서 인터페이스의 고유 식별자.	있음
receive_rate_percent	smallint	호스트에서 사용된 수신 비율.	없음
max_receive_rate_percent	smallint	집계 기간의 최대 수신 속도(백분율)입니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
transmit_rate_percent	smallint	호스트에서 사용된 전송 속도 비율.	없음

이름	유형	설명	인덱싱됨
max_transmit_rate_percent	smallint	집계 기간의 최대 전송 속도는 백분율로 표시됩니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
received_total_byte	BigInt	호스트에서 수신한 총 바이트 수입니다.	없음
transmitted_total_byte	BigInt	호스트에서 전송되는 총 바이트 수입니다.	없음
host_interface_configuration_version	integer	샘플 시 호스트 인터페이스 구성 버전입니다. 이는 **v4_4_configuration_history_hosts_interfaces 보기의 history_ id** 값과 동일하며 이를 결합하는 데 사용할 수 있습니다.	있음

3.7.5. 가상 머신 통계 보기

표 3.5. 시스템의 각 가상 머신에 대한 기록 통계

이름	유형	설명	인덱싱됨
history_id	BigInt	표에서 이 행의 고유 ID입니다.	없음
history_datetime	date	이 기록 행의 타임스탬프(분, 시간, 일로 집계 수준에 따라).	있음
vm_id	UUID	시스템에서 가상 머신의 고유 ID입니다.	있음

이름	유형	설명	인덱싱됨
vm_status	smallint	• -1 - 알 수 없는 상태 (ETL과 관련된 문제를 나타내는 데만 사용됩니다. Red Hat 지원부에 알려주십시오.) • 0 - 아래로 • 1 - 위로 • 2 - 일시 중지됨 • 3 - 문제 발생	없음
seconds_in_status	integer	가상 시스템이 집계 기간의 상태 열에 표시된 총 시간(초)입니다. 예를 들어 1분 동안 5초 동안 가상 시스템이 55초 동안 가동된 경우 이 분 동안 두 개의 행이 표시됩니다. 한 개는 Up 및 seconds_in_status 상태가 되고 다른 하나는 Down으로, seconds_in_status는 5가 됩니다.	없음
minutes_in_status	숫자(7,2)	가상 시스템이 집계 기간의 상태 열에 표시된 상태에 있는 총 분(분)입니다. 예를 들어 한 시간 동안 5분 동안 가상 시스템이 55분 동안 가동된 경우 이 시간 동안 두 개의 행이 표시됩니다. 하나는 Up 및 minutes_in_status 상태가 되고 다른 하나는 Down 상태이고 minutes_in_status는 5입니다.	없음
cpu_usage_percent	smallint	가상 시스템에서 사용 중인 CPU의 백분율입니다.	없음

이름	유형	설명	인덱싱됨
max_cpu_usage	smallint	집계 기간의 최대 CPU 사용량은 백분율로 표시됩니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
memory_usage_percent	smallint	가상 시스템에서 사용된 메모리의 백분율입니다. 메모리 사용량을 기록하려면 가상 머신에 게스트 툴을 설치해야 합니다.	없음
max_memory_usage	smallint	집계 기간의 최대 메모리 사용량은 백분율로 표시됩니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다. 메모리 사용량을 기록하려면 가상 머신에 게스트 툴을 설치해야 합니다.	없음
user_cpu_usage_percent	smallint	호스트에서 사용된 사용자 CPU 백분율.	없음
max_user_cpu_usage_percent	smallint	집계 기간의 최대 사용자 CPU 사용량은 백분율로 표시됩니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
system_cpu_usage_percent	smallint	호스트에서 사용된 시스템 CPU 백분율.	없음
max_system_cpu_usage_percent	smallint	집계 기간의 최대 시스템 CPU 사용량은 백분율로 표시됩니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음

이름	유형	설명	인덱싱됨
vm_ip	text	첫 번째 NIC의 IP 주소입니다. 게스트 에이전트가 설치된 경우에만 표시됩니다.	없음
currently_running_on_host	UUID	가상 시스템이 실행 중인 호스트의 고유 ID입니다.	없음
current_user_id	UUID	게스트 에이전트가 설치된 경우 가상 머신 콘솔에 로그인한 사용자의 고유 ID입니다.	없음
disks_usage	text	디스크 설명입니다. 파일 시스템 유형, 마운트 지점, 총 크기 및 사용된 크기.	없음
vm_configuration_version	integer	샘플 시 가상 머신 구성 버전입니다. **v4_4_configuration_history _vms 보기의 history_ id** 값과 동일합니다.	있음
current_host_configuration_version	integer	샘플 시 호스트 구성 버전입니다. 이는 **v4_4_configuration_history_hosts 보기의 history_ id** 값과 동일하며 이를 결합하는 데 사용할 수 있습니다.	있음
memory_buffered_kb	BigInt	가상 머신의 버퍼링된 메모리 크기(KB)입니다.	없음
memory_cached_kb	BigInt	가상 시스템의 캐시된 메모리 양(KB)입니다.	없음
max_memory_buffered_kb	BigInt	집계 기간(KB) 단위의 최대 버퍼링된 메모리입니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음

이름	유형	설명	인덱싱됨
max_memory_cached_kb	BigInt	집계 기간(KB) 단위로 최대 캐시된 메모리입니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음

3.7.6. 가상 머신 인터페이스 통계 보기

표 3.6. 시스템의 가상 머신 네트워크 인터페이스에 대한 이전 통계

이름	유형	설명	인덱싱됨
history_id	integer	표에서 이 행의 고유 ID입니다.	없음
history_datetime	date	이 기록 행의 타임스탬프(분, 시간, 일로 집계 수준에 따라).	있음
vm_interface_id	UUID	시스템에서 인터페이스의 고유 ID입니다.	있음
receive_rate_percent	smallint	호스트에서 사용된 수신 비율.	없음
max_receive_rate_percent	smallint	집계 기간의 최대 수신 속도(백분율)입니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
transmit_rate_percent	smallint	호스트에서 사용된 전송 속도 비율.	없음
max_transmit_rate_percent	smallint	집계 기간의 최대 전송 속도는 백분율로 표시됩니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일일 집계의 경우 최대 시간별 평균 속도입니다.	없음
received_total_byte	BigInt	가상 시스템에서 수신한 총 바이트 수입니다.	없음
transmitted_total_byte	BigInt	가상 시스템에서 전송되는 총 바이트 수입니다.	없음

이름	유형	설명	인덱싱됨
vm_interface_configuration_version	integer	샘플 시 가상 머신 인터페이스 구성 버전입니다. 이는 **v4_4_configuration_history_vms_interfaces 보기의 history_ id** 값과 동일하며 이를 결합하는 데 사용할 수 있습니다.	있음

3.7.7. 가상 디스크 통계 보기

표 3.7. 시스템의 가상 디스크에 대한 기록 통계

이름	유형	설명	인덱싱됨
history_id	BigInt	표에서 이 행의 고유 ID입니다.	없음
history_datetime	date	이 기록 행의 타임스탬프(분, 시간, 일로 집계 수준에 따라).	있음
vm_disk_id	UUID	시스템의 디스크 고유 ID입니다.	있음
vm_disk_status	smallint	• 0 - 할당되지 않음 • 1 - 확인 • 2 - 잠김 • 3 - 유효하지 않음 • 4 - 불법	없음

이름	유형	설명	인덱싱됨
seconds_in_status	integer	가상 디스크가 집계 기간의 상태 열에 표시된 총 시간(초)입니다. 예를 들어 가상 디스크가 55초 동안 잠겨 있고 1분 동안 5초 동안 OK(확인)가 이 분 동안 두 개의 행이 표시됩니다. 한 개는 **Locked(잠김)** 및 seconds_in_status(55) 상태가 되고 다른 하나는 OK(확인)로, seconds_in_status는 5가 됩니다.	없음
minutes_in_status	숫자(7,2)	가상 디스크가 집계 기간의 상태 열에 표시된 상태에 있는 총 분(분)입니다. 예를 들어, 가상 디스크가 55분 동안 잠겨 있고 한 시간 동안 5분 동안 OK(확인)하면 이 시간 동안 두 개의 행이 표시됩니다. 한 개는 **Locked(잠김)** 상태로, minutes_in_status는 55이고 다른 하나는 OK(확인)이고 minutes_in_status는 5입니다.	없음
vm_disk_actual_size_mb	integer	디스크에 할당된 실제 크기입니다.	없음
read_rate_bytes_per_second	integer	초당 바이트 단위의 디스크 속도(바이트)입니다.	없음
max_read_rate_bytes_per_second	integer	집계 기간의 최대 읽기 속도입니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
read_ops_total_count	numeric(20,0)	vm start 이후의 I/O 작업을 디스크에 읽습니다.	없음
read_latency_seconds	숫자(18,9)	가상 디스크 읽기 대기 시간(초)입니다.	없음

이름	유형	설명	인덱싱됨
write_rate_bytes_per_second	integer	초당 쓰기 속도(바이트)입니다.	없음
max_read_latency_seconds	숫자(18,9)	집계 기간에 대한 최대 읽기 대기 시간(초)입니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
max_write_rate_bytes_per_second	integer	집계 기간의 최대 쓰기 속도입니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
write_ops_total_count	numeric(20,0)	vm start 이후 디스크에 I/O 작업을 씁니다.	없음
write_latency_seconds	숫자(18,9)	가상 디스크 쓰기 대기 시간(초)입니다.	없음
max_write_latency_seconds	숫자(18,9)	집계 기간에 대한 최대 쓰기 대기 시간(초)입니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
flush_latency_seconds	숫자(18,9)	가상 디스크 플러시 대기 시간(초)입니다.	없음
max_flush_latency_seconds	숫자(18,9)	집계 기간에 대한 최대 플러시 대기 시간(초 단위)입니다. 매시간 집계의 경우 수집된 최대 샘플 값입니다. 일별 집계의 경우 최대 시간별 평균 값입니다.	없음
vm_disk_configuration_version	integer	샘플 시 가상 디스크 구성 버전입니다. 이는 **v4_4_configuration_history_vms_disks 보기의 history_ id** 값과 동일하며 결합하는 데 사용할 수 있습니다.	있음

3.8. 구성 내역 보기

구성 뷰를 쿼리하려면 **SELECT * FROM *view_name*;** 을 실행합니다. 예를 들면 다음과 같습니다.

```
# SELECT * FROM v4_4_configuration_history_datacenters;
```

사용 가능한 모든 보기를 나열하려면 다음을 실행합니다.

```
# \dv
```

참고

이러한 보기는 정의에 따라 삭제되지 않은 최신 엔터티 구성을 제공하므로 **delete_date** 는 최신 뷰에 표시되지 않습니다.

3.8.1. 데이터 센터 설정

다음 표는 시스템에 있는 데이터 센터의 구성 이력 매개 변수를 보여줍니다.

표 3.8. v4_4_configuration_history_datacenters

이름	유형	설명	인덱싱됨
history_id	integer	기록 데이터베이스에 있는 구성 버전의 ID입니다. 이는 **v4_4_configuration_history_clusters** 보기의 **datacenter_configuration_ version** 값과 동일하며 이를 결합하는 데 사용할 수 있습니다.	없음
datacenter_id	UUID	시스템의 데이터 센터의 고유 ID입니다.	있음
datacenter_name	문자 변경(40)	편집 대화 상자에 표시되는 데이터 센터의 이름입니다.	없음
datacenter_description	문자 변경()	편집 대화 상자에 표시된 데이터 센터에 대한 설명입니다.	없음

이름	유형	설명	인덱싱됨
is_local_storage	boolean	데이터 센터에서 로컬 스토리지를 사용하는지 여부를 나타내는 플래그입니다.	없음
create_date	시간대를 사용한 타임스탬프	이 엔터티를 시스템에 추가한 날짜입니다.	없음
update_date	시간대를 사용한 타임스탬프	시스템에서 이 엔터티가 변경된 날짜입니다.	없음
delete_date	시간대를 사용한 타임스탬프	이 엔터티가 시스템에서 삭제된 날짜입니다.	없음

3.8.2. 데이터 센터 스토리지 도메인 맵

다음 표는 시스템의 스토리지 도메인과 데이터 센터 간의 관계를 보여줍니다.

표 3.9. v4_4_map_history_datacenters_storage_domains

이름	유형	설명	인덱싱됨
history_id	integer	기록 데이터베이스에 있는 구성 버전의 ID입니다. 이는 **v4_4_configuration_history_clusters** 보기의 **datacenter_configuration_ version** 값과 동일하며 이를 결합하는 데 사용할 수 있습니다.	없음
storage_domain_id	UUID	시스템에서 이 스토리지 도메인의 고유 ID입니다.	있음
datacenter_id	UUID	시스템의 데이터 센터의 고유 ID입니다.	없음
attach_date	시간대를 사용한 타임스탬프	스토리지 도메인을 데이터 센터에 연결된 날짜입니다.	없음

이름	유형	설명	인덱싱됨
detach_date	시간대를 사용한 타임스탬프	스토리지 도메인이 데이터 센터에서 분리된 날짜입니다.	없음

3.8.3. 스토리지 도메인 구성

다음 표는 시스템에 있는 스토리지 도메인의 구성 이력 매개 변수를 보여줍니다.

표 3.10. v4_4_configuration_history_storage_domains

이름	유형	설명	인덱싱됨
history_id	integer	기록 데이터베이스에 있는 구성 버전의 ID입니다. 스토리지 도메인 통계 뷰의 **storage_configuration_version** 값과 동일하며 이를 결합하는 데 사용할 수 있습니다.	없음
storage_domain_id	UUID	시스템에서 이 스토리지 도메인의 고유 ID입니다.	있음
storage_domain_name	문자 변경(250)	스토리지 도메인 이름.	없음
storage_domain_type	smallint	* 0 - 데이터(마스터) * 1 - 데이터 * 2 - ISO * 3 - 내보내기	없음
storage_type	smallint	* 0 - 알 수 없음 * 1 - NFS * 2 - FCP * 3 - iSCSI * 4 - 로컬 * 6 - 모두	없음

이름	유형	설명	인덱싱됨
create_date	시간대를 사용한 타임스탬프	이 엔터티를 시스템에 추가한 날짜입니다.	없음
update_date	시간대를 사용한 타임스탬프	시스템에서 이 엔터티가 변경된 날짜입니다.	없음
delete_date	시간대를 사용한 타임스탬프	이 엔터티가 시스템에서 삭제된 날짜입니다.	없음

3.8.4. 클러스터 구성

다음 표는 시스템에 있는 클러스터의 구성 이력 매개 변수를 보여줍니다.

표 3.11. v4_4_configuration_history_clusters

이름	유형	설명	인덱싱됨
history_id	integer	기록 데이터베이스에 있는 구성 버전의 ID입니다. v4_4_**configuration_history_ hosts** 및 **v4_configuration_history_ vms** 보기의 **cluster_configuration_history_ version** 값과 동일하며 결합하는 데 사용할 수 있습니다.	없음
cluster_id	UUID	이 클러스터가 상주하는 데이터센터의 고유 식별자입니다.	있음
cluster_name	문자 변경(40)	편집 대화 상자에 표시되는 클러스터의 이름입니다.	없음
cluster_description	문자 변경()	편집 대화 상자에서 정의된 대로.	없음
datacenter_id	UUID	이 클러스터가 상주하는 데이터센터의 고유 식별자입니다.	있음
cpu_name	문자 변경(255)	편집 대화 상자에 표시된 대로.	없음

이름	유형	설명	인덱싱됨
count_threads_as_cores	boolean	호스트가 총 프로세서 **코어 수가 호스트의 코어 수보다 큰 가상 시스템을 실행할 수 있는 경우 Count Threads** (코어로 개수 스레드) 열은 **true** 로 표시됩니다.	없음
compatibility_version	문자 변경(40)	편집 대화 상자에 표시된 대로.	없음
datacenter_configuration_version	integer	생성 또는 업데이트 시 데이터 센터 구성 버전입니다. 생성 또는 업데이트 시 데이터 센터 구성 버전입니다. 이는 **v4_4_configuration_history_datacenters 보기의 history_ id** 값과 동일하며 결합하는 데 사용할 수 있습니다.	없음
create_date	시간대를 사용한 타임스탬프	이 엔터티를 시스템에 추가한 날짜입니다.	없음
update_date	시간대를 사용한 타임스탬프	시스템에서 이 엔터티가 변경된 날짜입니다.	없음
delete_date	시간대를 사용한 타임스탬프	이 엔터티가 시스템에서 삭제된 날짜입니다.	없음

3.8.5. 호스트 설정

다음 표는 시스템에 있는 호스트의 구성 이력 매개 변수를 보여줍니다.

표 3.12. v4_4_configuration_history_hosts

이름	유형	설명	인덱싱됨
history_id	integer	기록 데이터베이스에 있는 구성 버전의 ID입니다. 이는 호스트 통계 보기의 **host_configuration_version** 값과 동일하며 이를 결합하는 데 사용할 수 있습니다.	없음

이름	유형	설명	인덱싱됨
host_id	UUID	시스템에서 호스트의 고유 ID입니다.	있음
host_unique_id	문자 변경(128)	이 필드는 호스트의 물리적 UUID와 MAC 주소 중 하나의 조합으로, 시스템에 이미 등록된 호스트를 탐지하는 데 사용됩니다.	없음
host_name	문자 변경(255)	호스트 이름(편집 대화 상자와 동일).	없음
cluster_id	UUID	이 호스트가 속하는 클러스터의 고유 ID입니다.	있음
host_type	smallint	* 0 - RHEL 호스트 * 2 - RHEV Hypervisor 노드	없음
fqdn_or_ip	문자 변경(255)	Red Hat Virtualization Manager가 통신할 호스트의 DNS 이름 또는 IP 주소(편집 대화 상자에서 표시됨).	없음
memory_size_mb	integer	호스트의 실제 메모리 용량은 메가바이트(MB)로 표현됩니다.	없음
swap_size_mb	integer	호스트 스왑 파티션 크기입니다.	없음
cpu_model	문자 변경(255)	호스트의 CPU 모델.	없음
number_of_cores	smallint	호스트의 총 CPU 코어 수.	없음
number_of_sockets	smallint	총 CPU 소켓 수.	없음
cpu_speed_mh	숫자(18,0)	메가헤르츠(Multihertz)로 표현된 호스트의 CPU 속도.	없음
host_os	문자 변경(255)	호스트의 운영 체제 버전.	없음
kernel_version	문자 변경(255)	호스트의 커널 버전.	없음

이름	유형	설명	인덱싱됨
kvm_version	문자 변경(255)	호스트의 KVM 버전.	없음
vdsm_version	문자 변경	호스트의 VDSM 버전.	없음
vdsm_port	integer	편집 대화 상자에 표시된 대로.	없음
threads_per_core	smallint	코어당 총 스레드 수.	없음
hardware_manufacturer	문자 변경(255)	호스트의 하드웨어 제조업체.	없음
hardware_product_name	문자 변경(255)	호스트 하드웨어의 제품 이름입니다.	없음
hardware_version	문자 변경(255)	호스트 하드웨어의 버전입니다.	없음
hardware_serial_number	문자 변경(255)	호스트 하드웨어의 일련 번호입니다.	없음
cluster_configuration_version	integer	생성 또는 업데이트 시 클러스터 구성 버전입니다. 이는 **v4_4_configuration_history_clusters 보기의 history_ id** 값과 동일하며 결합하는 데 사용할 수 있습니다.	없음
create_date	시간대를 사용한 타임스탬프	이 엔터티를 시스템에 추가한 날짜입니다.	없음
update_date	시간대를 사용한 타임스탬프	시스템에서 이 엔터티가 변경된 날짜입니다.	없음
delete_date	시간대를 사용한 타임스탬프	이 엔터티가 시스템에서 삭제된 날짜입니다.	없음

3.8.6. 호스트 인터페이스 구성

다음 표는 시스템에 있는 호스트 인터페이스의 구성 이력 매개 변수를 보여줍니다.

표 3.13. v4_4_configuration_history_hosts_interfaces

이름	유형	설명	인덱싱됨
history_id	integer	기록 데이터베이스에 있는 구성 버전의 ID입니다. 이는 호스트 인터페이스 통계 보기에서 **host_interface_configuration_version** 값과 동일하며 이를 결합하는 데 사용할 수 있습니다.	없음
host_interface_id	UUID	시스템에서 이 인터페이스의 고유 ID입니다.	있음
host_interface_name	문자 변경(50)	호스트에서 보고한 인터페이스 이름입니다.	없음
host_id	UUID	이 인터페이스가 속하는 호스트의 고유 ID입니다.	있음
host_interface_type	smallint	* 0 - rt18139_pv * 1 - rt18139 * 2 - e1000 * 3 - pv	없음
host_interface_speed_bps	integer	인터페이스 속도(초당 비트 수).	없음
mac_address	문자 변경(59)	인터페이스 MAC 주소입니다.	없음
logical_network_name	문자 변경(50)	인터페이스와 연결된 논리적 네트워크입니다.	없음
ip_address	문자 변경(20)	편집 대화 상자에 표시된 대로.	없음
gateway	문자 변경(20)	편집 대화 상자에 표시된 대로.	없음
본딩	boolean	이 인터페이스가 본딩 인터페이스인지 나타내는 플래그입니다.	없음
bond_name	문자 변경(50)	본딩의 이름은 이 인터페이스의 일부입니다(본딩의 일부인 경우).	없음

이름	유형	설명	인덱싱됨
vlan_id	integer	편집 대화 상자에 표시된 대로.	없음
host_configuration_version	integer	생성 또는 업데이트 시 호스트 구성 버전입니다. 이는 **v4_4_configuration_history_hosts 보기의 history_ id** 값과 동일하며 이를 결합하는 데 사용할 수 있습니다.	없음
create_date	시간대를 사용한 타임스탬프	이 엔터티를 시스템에 추가한 날짜입니다.	없음
update_date	시간대를 사용한 타임스탬프	시스템에서 이 엔터티가 변경된 날짜입니다.	없음
delete_date	시간대를 사용한 타임스탬프	이 엔터티가 시스템에서 삭제된 날짜입니다.	없음

3.8.7. 가상 머신 구성

다음 표는 시스템에 있는 가상 시스템의 구성 이력 매개 변수를 보여줍니다.

표 3.14. v4_4_configuration_history_vms

이름	유형	설명	인덱싱됨
history_id	integer	기록 데이터베이스에 있는 구성 버전의 ID입니다. 이는 가상 시스템 통계 보기의 **vm_configuration_version** 값과 동일하며 이를 결합하는 데 사용할 수 있습니다.	없음
vm_id	UUID	시스템에서 이 가상 시스템의 고유 ID입니다.	있음
vm_name	문자 변경(255)	가상 시스템의 이름입니다.	없음
vm_description	문자 변경()	편집 대화 상자에 표시된 대로.	없음

이름	유형	설명	인덱싱됨
vm_type	smallint	* 0 - 데스크탑 * 1 - 서버	없음
cluster_id	UUID	이 가상 시스템이 속하는 클러스터의 고유 ID입니다.	있음
template_id	UUID	이 가상 머신의 고유 ID는 에서 파생됩니다. 템플릿은 이 버전의 Red Hat Virtualization에서 기록 데이터베이스에 동기화되지 않습니다.	없음
template_name	문자 변경(40)	이 가상 머신이 파생되는 템플릿의 이름입니다.	없음
cpu_per_socket	smallint	소켓당 가상 CPU 수.	없음
number_of_sockets	smallint	총 가상 CPU 소켓 수.	없음
memory_size_mb	integer	가상 시스템에 할당된 총 메모리(MB)로 표현됩니다.	없음
operating_system	smallint	* 0 - 기타 OS * 1 - Windows XP * 3 - Windows 2003 * 4 - Windows 2008 * 5 - Linux * 7 - Red Hat Enterprise Linux 5.x * 8 - Red Hat Enterprise Linux 4.x * 9 - Red Hat Enterprise Linux 3.x * 10 - Windows 2003 x64 * 11 - Windows 7 * 12 - Windows 7 x64	없음

이름	유형	설명	인덱싱됨
		* 13 - Red Hat Enterprise Linux 5.x x64 * 14 - Red Hat Enterprise Linux 4.x x64 * 15 - Red Hat Enterprise Linux 3.x x64 * 16 - Windows 2008 x64 * 17 - Windows 2008 R2 x64 * 18 - Red Hat Enterprise Linux 6.x * 19 - Red Hat Enterprise Linux 6.x x64 * 20 - Windows 8 * 21 - Windows 8 x64 * 23 - Windows 2012 x64 * 1001 - 기타 * 1002 - Linux * 1003 - Red Hat Enterprise Linux 6.x * 1004 - SUSE Linux Enterprise Server 11 * 1193 - SUSE Linux Enterprise Server 11 * 1252 - Ubuntu Precise Pangolin LTS * 1253 - Ubuntu 수량 Quetzal * 1254 - Ubuntu Raring Ringtails * 1255 - Ubuntu Saucy Salamander	
default_host	UUID	편집 대화 상자에 표시된 대로 시스템의 기본 호스트 ID입니다.	없음

이름	유형	설명	인덱싱됨
고가용성(_A)	boolean	편집 대화 상자에 표시된 대로.	없음
초기화됨	boolean	이 가상 머신이 Sysprep 초기화 목적으로 한 번 이상 시작되었는지 나타내는 플래그입니다.	없음
상태 비저장	boolean	편집 대화 상자에 표시된 대로.	없음
fail_back	boolean	편집 대화 상자에 표시된 대로.	없음
usb_policy	smallint	편집 대화 상자에 표시된 대로.	없음
time_zone	문자 변경(40)	편집 대화 상자에 표시된 대로.	없음
vm_pool_id	UUID	이 가상 시스템이 속한 풀의 ID입니다.	없음
vm_pool_name	문자 변경(255)	가상 시스템 풀의 이름입니다.	없음
created_by_user_id	UUID	이 가상 시스템을 생성한 사용자의 ID입니다.	없음
cluster_configuration_version	integer	생성 또는 업데이트 시 클러스터 구성 버전입니다. 이는 **v4_4_configuration_history_clusters 보기의 history_ id** 값과 동일하며 결합하는 데 사용할 수 있습니다.	없음
default_host_configuration_version	integer	생성 또는 업데이트 시 호스트 구성 버전입니다. 이는 **v4_4_configuration_history_hosts 보기의 history_ id** 값과 동일하며 이를 결합하는 데 사용할 수 있습니다.	없음

이름	유형	설명	인덱싱됨
create_date	시간대를 사용한 타임스탬프	이 엔터티를 시스템에 추가한 날짜입니다.	없음
update_date	시간대를 사용한 타임스탬프	시스템에서 이 엔터티가 변경된 날짜입니다.	없음
delete_date	시간대를 사용한 타임스탬프	이 엔터티가 시스템에서 삭제된 날짜입니다.	없음

3.8.8. 가상 머신 인터페이스 구성

다음 표는 시스템에 있는 가상 인터페이스의 구성 이력 매개 변수를 보여줍니다.

표 3.15. v4_4_configuration_history_vms_interfaces

이름	유형	설명	인덱싱됨
history_id	integer	기록 데이터베이스에 있는 구성 버전의 ID입니다. 이는 가상 시스템 인터페이스 통계 보기에서 **vm_interface_configuration_version** 값과 동일하며 이를 결합하는 데 사용할 수 있습니다.	없음
vm_id	UUID	시스템에서 가상 머신의 고유 ID입니다.	있음
vm_interface_id	UUID	시스템에서 이 인터페이스의 고유 ID입니다.	있음
vm_interface_name	문자 변경(50)	편집 대화 상자에 표시된 대로.	없음
vm_interface_type	smallint	가상 인터페이스의 유형입니다. * 0 - rt18139_pv * 1 - rt18139 * 2 - e1000 * 3 - pv	없음

이름	유형	설명	인덱싱됨
vm_interface_speed_bps	integer	인터페이스의 평균 속도(초당 비트 수)입니다.	없음
mac_address	문자 변경(20)	편집 대화 상자에 표시된 대로.	없음
logical_network_name	문자 변경(50)	편집 대화 상자에 표시된 대로.	없음
vm_configuration_version	integer	생성 또는 업데이트 시 가상 머신 구성 버전입니다. 이는 **v4_4_configuration_history_vms 보기의 history_ id** 값과 동일하며 결합하는 데 사용할 수 있습니다.	없음
create_date	시간대를 사용한 타임스탬프	이 엔터티를 시스템에 추가한 날짜입니다.	없음
update_date	시간대를 사용한 타임스탬프	시스템에서 이 엔터티가 변경된 날짜입니다.	없음
delete_date	시간대를 사용한 타임스탬프	이 엔터티가 시스템에서 삭제된 날짜입니다.	없음

3.8.9. 가상 머신 장치 설정

다음 표는 디스크 및 가상 인터페이스를 포함하여 가상 시스템과 관련 장치 간의 관계를 보여줍니다.

표 3.16. v4_4_configuration_history_vms_devices

이름	유형	설명	인덱싱됨
history_id	integer	기록 데이터베이스에 있는 구성 버전의 ID입니다.	없음
vm_id	UUID	시스템의 가상 머신 고유 ID입니다.	있음
device_id	UUID	시스템에서 장치의 고유 ID입니다.	없음

이름	유형	설명	인덱싱됨
type	문자 변경(30)	가상 시스템 장치 유형입니다. "디스크" 또는 "인터페이스"일 수 있습니다.	있음
address	문자 변경(255)	장치의 물리적 주소입니다.	없음
is_managed	boolean	Manager에서 장치를 관리하는지 나타내는 플래그입니다.	없음
is_plugged	boolean	장치가 가상 머신에 연결되어 있는지를 나타내는 플래그입니다.	없음
is_readonly	boolean	장치가 읽기 전용인지를 나타내는 플래그입니다.	없음
vm_configuration_version	integer	샘플을 만들 때 가상 머신 구성 버전입니다.	없음
device_configuration_version	integer	샘플이 수행될 때 장치 구성 버전을 사용합니다. - **type** 필드의 값이 **interface** 로 설정된 경우 이 필드는 **v4_4_configuration_history_vms_interfaces 보기의 history_ id** 필드와 조인됩니다. - **type** 필드의 값이 disk 로 설정된 경우 이 필드는 **v4_4_configuration_history_vms_disks_disks 보기의 history_id 필드와** 결합됩니다.	없음
create_date	시간대를 사용한 타임스탬프	이 엔터티를 시스템에 추가한 날짜입니다.	없음
update_date	시간대를 사용한 타임스탬프	이 엔터티를 시스템에 추가한 날짜입니다.	없음

이름	유형	설명	인덱싱됨
delete_date	시간대를 사용한 타임스탬프	이 엔터티를 시스템에 추가한 날짜입니다.	없음

3.8.10. 가상 디스크 구성

다음 표는 시스템에 있는 가상 디스크의 구성 이력 매개 변수를 보여줍니다.

표 3.17. v4_4_configuration_history_vms_disks

이름	유형	설명	인덱싱됨
history_id	integer	기록 데이터베이스에 있는 구성 버전의 ID입니다. 이는 가상 디스크 통계 보기의 **vm_disk_configuration_version** 값과 동일하며 이를 결합하는 데 사용할 수 있습니다.	없음
vm_disk_id	UUID	시스템에서 이 디스크의 고유 ID입니다.	있음
vm_disk_name	text	편집 대화 상자에 표시되는 가상 디스크의 이름입니다.	없음
vm_disk_description	문자 변경(500)	편집 대화 상자에 표시된 대로.	없음
image_id	UUID	시스템에 있는 이미지의 고유 ID입니다.	없음
storage_domain_id	UUID	이 디스크 이미지가 속하는 스토리지 도메인의 ID입니다.	있음
vm_disk_size_mb	integer	정의된 디스크 크기(MB)입니다.	없음

이름	유형	설명	인덱싱됨
vm_disk_type	smallint	편집 대화 상자에 표시된 대로. 현재 시스템 및 데이터만 사용되고 있습니다. * 0 - 할당되지 않음 * 1 - 시스템 * 2 - data * 3 - 공유 * 4 - 스왑 * 5 - 임시	없음
vm_disk_format	smallint	편집 대화 상자에 표시된 대로. * 3 - 할당되지 않음 * 4 - COW * 5 - 원시	없음
is_shared	boolean	가상 시스템의 디스크가 공유되었는지를 나타내는 플래그입니다.	없음
create_date	시간대를 사용한 타임스탬프	이 엔터티를 시스템에 추가한 날짜입니다.	없음
update_date	시간대를 사용한 타임스탬프	시스템에서 이 엔터티가 변경된 날짜입니다.	없음
delete_date	시간대를 사용한 타임스탬프	이 엔터티가 시스템에서 삭제된 날짜입니다.	없음

3.8.11. 사용자 정보 기록

다음 표는 시스템에 있는 사용자의 구성 이력 매개 변수를 보여줍니다.

표 3.18. v4_4_users_details_history

이름	유형	설명
user_id	UUID	Manager가 생성한 대로 시스템에 있는 사용자의 고유 ID.

이름	유형	설명
first_name	문자 변경(255)	사용자의 이름.
last_name	문자 변경(255)	사용자의 성.
domain	문자 변경(255)	권한 확장의 이름입니다.
사용자 이름	문자 변경(255)	계정 이름입니다.
부서	문자 변경(255)	사용자가 속한 조직 부서.
user_role_title	문자 변경(255)	조직 내에서 사용자의 제목 또는 역할.
email	문자 변경(255)	조직에 있는 사용자의 이메일입니다.
external_id	text	외부 시스템의 사용자 고유 식별자입니다.
활성	boolean	사용자가 활성 상태인지 여부를 나타내는 플래그입니다. 매시간 확인됩니다. 사용자를 권한 부여 확장자에서 찾을 수 있으면 활성 상태로 유지됩니다. 로그인에 성공하면 사용자가 활성화됩니다.
create_date	시간대를 사용한 타임스탬프	이 엔터티를 시스템에 추가한 날짜입니다.
update_date	시간대를 사용한 타임스탬프	시스템에서 이 엔터티가 변경된 날짜입니다.
delete_date	시간대를 사용한 타임스탬프	이 엔터티가 시스템에서 삭제된 날짜입니다.

부록 A. 법률 알림

Copyright © 2022 Red Hat, Inc.

라이센스 라이센스 (Creative Commons Attribution-ShareAlike 4.0 International License). (oVirt Project)에 대한 문서에서 파생됩니다. 이 문서 또는 문서의 수정을 배포하는 경우 원래 버전에 대한 URL을 제공해야 합니다.

수정된 버전에서는 모든 Red Hat 상표를 제거해야 합니다.

Red Hat, Red Hat Enterprise Linux, Red Hat 로고, Shadowman 로고, JBoss, OpenShift, Fedora, Infinity 로고 및 RHCE는 미국 및 기타 국가에 등록된 Red Hat, Inc.의 상표입니다.

Linux®는 미국 및 기타 국가에서 Linus Torvalds의 등록 상표입니다.

Java®는 Oracle 및/또는 그 계열사의 등록 상표입니다.

XFS®는 미국 및/또는 기타 국가에 있는 Silicon Graphics International Corp. 또는 해당 자회사의 상표입니다.

MySQL®은 미국, 유럽 연합 및 기타 국가에서 MySQL AB의 등록 상표입니다.

Node.js®는 Joyent의 공식 상표입니다. Red Hat Software Collections는 공식 Joyent Node.js 오픈 소스 또는 상용 프로젝트의 보증 대상이 아니며 공식적인 관계도 없습니다.

OpenStack® Word Mark 및 OpenStack 로고는 미국 및 기타 국가에서 OpenStack Foundation의 등록 상표/서비스표 또는 상표/서비스표이며 OpenStack Foundation의 권한에 사용됩니다. 당사는 OpenStack Foundation 또는 OpenStack 커뮤니티와 제휴 관계가 아니며 보증 또는 후원을 받지 않습니다.

기타 모든 상표는 각각 해당 소유자의 자산입니다.