



Резервное копирование и  
восстановление системы управления  
«РЕД Виртуализация»

# Оглавление

<b>1</b>	<b>Standalone HostedEngine .....</b>	<b>3</b>
1.1	Резервное копирование HostedEngine.....	3
1.2	Параметры базы данных Engine.....	4
1.3	Создание резервной копии с помощью engine-backup.....	4
1.4	Восстановление резервной копии .....	6
<b>2</b>	<b>Резервное копирование self-hosted Engine .....</b>	<b>8</b>
2.1	Резервное копирование исходного Engine.....	9
<b>3</b>	<b>Восстановление self-hosted Engine .....</b>	<b>11</b>
3.1	Восстановление резервной копии на новом self-hosted Engine .....	12
3.2	Переустановка хостов.....	15
3.3	Удаление домена хранения .....	16
3.4	Восстановление self-hosted Engine из существующей резервной копии.....	16
3.5	Включение режима глобального обслуживания .....	17
3.6	Восстановление резервной копии для перезаписи существующей установки .....	17
3.7	Отключение режима глобального обслуживания .....	18

# 1. Standalone HostedEngine

## 1.1 Резервное копирование HostedEngine

Инструмент `engine-backup` используется для регулярного создания резервных копий Engine РЕД Виртуализации. Инструмент создает резервную копию базы данных Engine и файлов конфигурации в один файл, который можно запускать, не прерывая работу службы `ovirt-engine`.

Команда `engine-backup` работает в одном из двух основных режимов:

```
engine-backup --mode=backup
engine-backup --mode=restore
```

Эти два режима дополнительно расширены набором параметров, которые позволяют указать область резервного копирования и различные учетные данные для базы данных Engine. Воспользуйтесь командой `engine-backup --help` для получения полного списка параметров и их функций.

Основные параметры:

- `--mode` - указывает, будет команда выполнять операцию резервного копирования или операцию восстановления. Доступны два варианта — `backup` и `restore`. Это обязательный параметр.
- `--file` - указывает путь и имя файла, в который должны быть сохранены резервные копии в режиме резервного копирования, а также путь и имя файла, из которого считываются данные резервной копии в режиме восстановления. Это обязательный параметр как в режиме резервного копирования, так и в режиме восстановления.
- `--log` - указывает путь и имя файла, в который должны быть записаны журналы операции резервного копирования или восстановления. Этот параметр необходим как в режиме резервного копирования, так и в режиме

восстановления.

- `--score` - указывает область операции резервного копирования или восстановления. Существует четыре варианта:
  - `all` - выполняет резервное копирование или восстановление всех баз данных и данных конфигурации;
  - `files` - создает резервные копии или восстанавливает только файлы в системе;
  - `db` - выполняет резервное копирование или восстановление только базы данных Engine;
  - `dwhdb` - выполняет резервное копирование или восстановление только базы данных хранилища данных.

Область действия по умолчанию — `all`.

Параметр `--score` может быть указан несколько раз в одной и той же команде `engine-backup`.

## 1.2 Параметры базы данных Engine

Данные параметры доступны только при использовании команды `engine-backup` в режиме `restore`. Приведенный ниже синтаксис параметра применяется для восстановления базы данных Engine. Те же параметры существуют для восстановления базы данных хранилища данных:

- `--provision-db` - создает базу данных PostgreSQL для восстановления резервной копии базы данных Engine. Это обязательный параметр при восстановлении резервной копии на удаленном хосте или новой установке, в которой еще не настроена база данных PostgreSQL.
- `--change-db-credentials` - позволяет указать альтернативные учетные данные для восстановления базы данных Engine с использованием учетных данных, отличных от тех, которые хранятся в самой резервной копии. См. дополнительные возможности, необходимые для этого параметра, используя команду `engine-backup --help`.
- `--restore-permissions` или `--no-restore-permissions` - восстанавливает (или не восстанавливает) разрешения пользователей базы данных. Один из этих параметров требуется при восстановлении резервной копии.

**Примечание.** Если резервная копия содержит данные для дополнительных пользователей базы данных, восстановление резервной копии с помощью параметров `--restore-permissions` и `--provision-db` (или `--provision-dwh-db`) создаст дополнительных пользователей со случайными паролями. Необходимо будет изменить такие пароли вручную, если дополнительным пользователям потребуется доступ к восстанавливаемой системе.

## 1.3 Создание резервной копии с помощью `engine-backup`

Вы можете создать резервную копию HostedEngine с помощью команды `engine-backup`, когда HostedEngine активен. Добавьте одно из следующих значений к параметру `--score`, чтобы указать, что вы хотите создать резервную

копию:

- all - полная резервная копия всех баз данных и файлов конфигурации на HostedEngine;
- files - резервное копирование только файлов в системе;
- db - резервная копия только базы данных Engine;
- dwhdb - резервная копия только базы данных хранилища данных;
- cinderlibdb - резервная копия только базы данных Cinderlib;
- grafanadb - резервная копия только базы данных Grafana.

Можно указывать параметр `--scope` более одного раза.

Также можно настроить команду `engine-backup` для резервного копирования дополнительных файлов. Она восстанавливает все, что резервирует.

**Важно!** Для восстановления базы данных на новой установке HostedEngine одной резервной копии базы данных недостаточно. Для Engine также требуется доступ к файлам конфигурации. Если указывается область, отличная от `all`, необходимо также включить `--scope=files` или создать резервную копию файловой системы. ■

Для получения подробной информации о команде `engine-backup` выполните `engine-backup --help` на машине Engine.

Для создания резервной копии выполните следующие действия:

1. Войдите в систему Engine.
2. Создайте резервную копию с помощью команды:

```
engine-backup --scope=all --mode=backup --file=<file_name>
--log=<log_file_name>
```

Команда создает резервную копию в формате `<file_name>.tar` и файл журнала в формате `<log_file_name>`.

Используйте `<file_name>.tar` для восстановления резервной копии.

Следующие примеры демонстрируют несколько различных сценариев резервного копирования.

Пример 1. Полная резервная копия:

```
engine-backup --scope=all --mode=backup --file=<file_name>
--log=<log_file_name>
```

Пример 2. Резервное копирование базы данных HostedEngine:

```
engine-backup --scope=files --scope=db --mode=backup --file=
<file_name> --log=<log_file_name>
```

Пример 3. Резервное копирование базы данных хранилища данных:

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

Пример 4. Добавление определенных файлов в резервную копию:

Создайте каталог для хранения настроек конфигурации для команды engine-backup:

```
mkdir -p /etc/ovirt-engine-backup/engine-backup-config.d
```

Создайте текстовый файл в новом каталоге ntp-chrony.sh со следующим содержимым:

```
BACKUP_PATHS="$BACKUP_PATHS  
/etc/chrony.conf  
/etc/ntp.conf  
/etc/ovirt-engine-backup"
```

Когда в следующий раз будет запущена команда engine-backup с использованием параметра --scope=files, резервное копирование и восстановление будет включать в себя /etc/chrony.conf, /etc/ntp.conf и /etc/ovirt-engine-backup.

## 1.4 Восстановление резервной копии

Команда engine-backup может восстановить резервную копию на машине, на которой уже установлен и настроен HostedEngine. Это полезно, когда вы сделали резервную копию среды, внесли изменения в эту среду, а затем хотите отменить изменения, восстановив среду из резервной копии.

Изменения, внесенные в среду после создания резервной копии, такие как добавление или удаление хоста, не будут отображаться в восстановленной среде. Вам необходимо будет повторить эти изменения.

Для восстановления резервной копии выполните следующий алгоритм действий:

1. Войдите в систему HostedEngine.

2. Удалите файлы конфигурации и очистите базу данных, связанную с HostedEngine:

```
engine-cleanup
```

Команда engine-cleanup очищает только базу данных HostedEngine; она не удаляет базу данных и не удаляет пользователя, которому принадлежит эта база данных.

3. Восстановите полную резервную копию или резервную копию только базы данных. Вам не нужно создавать новую базу данных или указывать учетные данные базы данных, поскольку пользователь и база данных уже существуют.

- Восстановить полную резервную копию:

```
engine-backup --mode=restore --file=<file_name> --log=<log_  
file_name> --restore-permissions
```

- Восстановить резервную копию только базы данных, восстановив файлы конфигурации и резервную копию базы данных:

```
engine-backup --mode=restore --scope=files --scope=db
--scope=dwhdb --file=<file_name> --log=<log_file_name> --restore
-permissions
```

**Примечание.** Чтобы восстановить только базу данных HostedEngine (например, если база данных хранилища данных находится на другом компьютере), параметр `--scope=dwhdb` можно опустить.

В случае успеха отображается следующий вывод:

```
You should now run engine-setup.
Done.
```

4. Переустановите HostedEngine командой:

```
engine-setup
```

После завершения команды, если в системе не восстановилась работа хоста и хранилища, необходимо перезагрузить машину HostedEngine.

## 2. Резервное копирование self-hosted Engine

Вы можете создать резервную копию self-hosted engine и восстановить его в новой среде. Используйте эту процедуру для таких задач, как миграция среды в новый домен хранения self-hosted engine с другим типом хранилища.

Когда вы указываете файл резервной копии во время развертывания, резервная копия восстанавливается на новой виртуальной машине Engine с новым доменом хранилища self-hosted engine. Старый Engine удаляется, а старый домен хранилища self-hosted engine переименовывается и может быть удален вручную после подтверждения правильности работы новой среды.

Настоятельно рекомендуется развертывание на новом хосте; если хост, используемый для развертывания, существовал в среде резервного копирования, он будет удален из восстановленной базы данных, чтобы избежать конфликтов в новой среде. При развертывании на новом хосте необходимо присвоить этому хосту уникальное имя. Повторное использование имени существующего хоста, включенного в резервную копию, может вызвать конфликты в новой среде.

Операция резервного копирования и восстановления включает следующие ключевые действия:

1. Создание резервной копии оригинального Engine с помощью инструмента engine-backup.
2. Развертывание нового self-hosted engine и восстановление резервной копии.
3. Включение репозитория Engine на новой виртуальной машине Engine.
4. Переустановка узлов self-hosted engine для обновления их конфигурации.
5. Удаление старого домена хранилища self-hosted engine.

Данная процедура предполагает, что у вас есть доступ и вы можете вносить изменения в исходный Engine.

Требования:

- Полное доменное имя, подготовленное для вашего Engine и хоста. Записи прямой и обратной записи должны быть установлены в DNS. Новый Engine



должен иметь то же полное доменное имя, что и исходный Engine.

- Уровень совместимости центра обработки данных должен быть установлен на последнюю версию, чтобы обеспечить совместимость с обновленной версией хранилища.

**Примечание.** Если необходимо восстановить резервную копию, но нет нового устройства, процесс восстановления будет приостановлен, вы сможете войти во временную машину Engine через SSH, зарегистрироваться, подписаться или настроить каналы по мере необходимости, а также обновить пакеты Engine до возобновления процесса восстановления.

- В среде должен быть по крайней мере один обычный хост. Этот хост (и любые другие обычные хосты) останутся активными для размещения роли SPM и любых запущенных виртуальных машин. Если обычный хост еще не является SPM, переместите роль SPM перед созданием резервной копии, выбрав обычный хост и нажав «Управление» - «Выбрать как SPM».

## 2.1 Резервное копирование исходного Engine

Создайте резервную копию с помощью команды `engine-backup` и скопируйте файл резервной копии в отдельное место, чтобы к нему можно было получить доступ в любой момент во время процесса восстановления.

Дополнительные сведения о параметрах `engine-backup --mode=backup` см. в п. 1.1 «Резервное копирование и восстановление HostedEngine».

Для создания резервной копии исходного Engine выполните следующие действия:

1. Войдите в self-hosted Engine и переведите среду в режим глобального обслуживания:

```
hosted-engine --set-maintenance --mode=global
```

2. Войдите в исходный Engine и остановите службу `ovirt-engine`:

```
systemctl stop ovirt-engine
systemctl disable ovirt-engine
```

**Примечание.** Хотя остановка работы исходного Engine не является обязательной, данное действие рекомендуется сделать, поскольку это гарантирует отсутствие изменений в среде после создания резервной копии. Кроме того, остановка службы предотвращает одновременное управление существующими ресурсами исходным и новым Engine.

3. Запустите команду `engine-backup`, указав имя создаваемого файла резервной копии и имя файла журнала, который необходимо создать для хранения журнала резервного копирования:

```
engine-backup --mode=backup --file=<file_name> --log=<log_file_name>
```

4. Скопируйте файлы на внешний сервер. В следующем примере **storage.example.com** — это полное доменное имя сервера сетевого хранилища, на котором будет храниться резервная копия до тех пор, пока она не понадобится, а **/backup/** — любая назначенная папка или путь.

```
scp -p <file_name> <log_file_name> storage.example.com:/backup/
```

5. Войдите в self-hosted Engine и выключите исходную виртуальную машину Engine:

```
hosted-engine --vm-shutdown
```

После резервного копирования Engine разверните новый хост и восстановите резервную копию на новой виртуальной машине.

### 3. Восстановление self-hosted Engine

Если self-hosted Engine недоступен из-за неполадок, которые невозможно исправить, вы можете восстановить его в новой среде self-hosted Engine, используя резервную копию, сделанную до возникновения проблемы, если она доступна.

Когда вы указываете файл резервной копии во время развертывания, резервная копия восстанавливается на новой виртуальной машине Engine с новым доменом хранилища self-hosted Engine. Старый Engine удаляется, а старый домен хранилища self-hosted Engine переименовывается и может быть удален вручную после того, как вы подтвердите, что новая среда работает правильно.

Настоятельно рекомендуется развертывание на новом хосте; если хост, используемый для развертывания, существовал в среде резервного копирования, он будет удален из восстановленной базы данных, чтобы избежать конфликтов в новой среде. При развертывании на новом хосте необходимо присвоить этому хосту уникальное имя. Повторное использование имени существующего хоста, включенного в резервную копию, может вызвать конфликты в новой среде.

Требования:

- Полное доменное имя, подготовленное для вашего Engine и хоста. Записи прямого и обратного просмотра должны быть установлены в DNS. Новый Engine должен иметь то же полное доменное имя, что и исходный Engine.

Восстановление self-hosted Engine включает в себя следующие ключевые действия:

1. Разворачивание нового self-hosted Engine и восстановление резервной копии.
2. Включение репозитория Engine на новой виртуальной машине Engine.
3. Переустановка хостов self-hosted Engine для обновления их конфигурации.
4. Удаление старого домена хранилища self-hosted Engine.

Данная процедура предполагает, что у вас нет доступа к исходному Engine и что новый хост может получить доступ к файлу резервной копии.

### 3.1 Восстановление резервной копии на новом self-hosted Engine

Запустите службу `hosted-engine` на новом хосте и используйте параметр `--restore-from-file=<path/to/file_name>` для восстановления резервной копии Engine во время развертывания.

Выполните следующий алгоритм действий:

1. Скопируйте файл резервной копии на новый хост. В следующем примере `host.example.com` - это полное доменное имя хоста, а `/backup/` - любая назначенная папка или путь.

```
scp -p <file_name> host.example.com:/backup/
```

2. Войдите на новый хост РЕД Виртуализации.

3. Если восстановление происходит в системе, где не работает хранилище, на котором был развернут старый Engine, необходимо добавить ключ `-he-remove-storage-vm` в файл `/usr/share/ansible/roles/ovirt.hosted_engine_setup/tasks/restore_backup.yml` (для РЕД Виртуализации 7.2) или в файл `/usr/share/ansible/collections/ansible_collections/ovirt/ovirt/roles/hosted_engine_setup/tasks/restore_backup.yml` (для РЕД Виртуализации 7.3).

Допишите его в конец строки:

```
--file=/root/engine_backup --provision-all-databases --restore-permissions
```

4. Запустите скрипт `hosted-engine`, указав путь к файлу резервной копии:

```
hosted-engine --deploy --restore-from-file=backup/<file_name>
```

Чтобы выйти из скрипта в любое время, используйте комбинацию клавиш «CTRL+D», чтобы прервать развертывание.

5. Выберите «Да», чтобы начать развертывание.

6. Настройте сеть. Сценарий определяет возможные сетевые адаптеры для использования в качестве моста управления для среды.

7. Если вы хотите использовать специальное устройство для установки виртуальной машины, введите путь к архиву OVA. В противном случае оставьте это поле пустым, чтобы использовать Engine Appliance.

8. Будет задан вопрос о приостановке выполнения сценариев восстановления, на него необходимо ответить «Да»:

```
Pause the execution after adding this host to the engine?
```

9. Введите пароль `root` для Engine.

10. Введите конфигурацию ЦП и памяти виртуальной машины.

11. Введите открытый ключ SSH, который позволит вам войти в Engine как пользователь `root`, и укажите, следует ли разрешить доступ SSH для пользователя `root`.

12. Введите MAC-адрес виртуальной машины Engine или примите случайно сгенерированный адрес. Если вы хотите предоставить виртуальной машине Engine IP-адрес через DHCP, убедитесь, что у вас действителен резерв DHCP для данного MAC-адреса. Скрипт развертывания не настраивает DHCP-сервер автоматически.

13. Введите сетевые данные виртуальной машины. Если вы укажете «Статический», введите IP-адрес Engine.

**Важно!** Статический IP-адрес должен принадлежать той же подсети, что и хост. Например, если узел находится в сети 10.1.1.0/24, IP-адрес виртуальной машины Engine должен находиться в том же диапазоне подсети (10.1.1.1-254/24). ■

14. Укажите, следует ли добавлять записи для виртуальной машины Engine и базового хоста в файл /etc/hosts виртуальной машины. Вы должны убедиться, что имена хостов разрешимы.

15. Укажите имя и номер TCP-порта SMTP-сервера, адрес электронной почты, используемый для отправки уведомлений по электронной почте, и список адресов электронной почты, разделенных запятыми, для получения этих уведомлений.

16. Введите пароль для доступа пользователя admin@internal к portalу администрирования.

Скрипт создает виртуальную машину. Это может занять некоторое время, если необходимо установить Engine Appliance.

**Примечание.** Если узел перестает работать из-за отсутствия требуемой сети или аналогичной проблемы, развертывание приостанавливается и отображается следующее сообщение:

```
[ INFO ] You can now connect to https://<host_name>:6900/ovirt
-engine/ and check the status of this host and eventually remedia-
te it, please continue only when the host is listed as 'up'
[ INFO ] TASK [ovirt.ovirt.hosted_engine_setup : include_tasks]
[ INFO ] ok: [localhost]
[ INFO ] TASK [ovirt.ovirt.hosted_engine_setup : Create temporary
lock file]
[ INFO ] changed: [localhost]
[ INFO ] TASK [ovirt.ovirt.hosted_engine_setup : Pause execution
until /tmp/ansible.<random>_he_setup_lock is removed, delete it once
ready to proceed]
```

Приостановка процесса позволяет:

- Подключиться к portalу администрирования, используя предоставленный URL-адрес.
- Оценить ситуацию и выяснить, почему хост не работает, а также исправить все необходимое. Например, если развертывание было восстановле-

но из резервной копии, и данная резервная копия включала необходимые сети для кластера хостов, можно настроить сети, подключив к этим сетям соответствующие сетевые адаптеры хоста.

- Когда все необходимые настройки будут установлены корректно, а состояние хоста будет «Up», можно удалить файл блокировки, представленный в сообщении выше. Развертывание продолжится.

17. Когда установка дойдет до этапа остановки выполнения, перейдите по указанной в вашем терминале ссылке.

Войдите на портал администрирования.

Если у нового хоста отличается семейство сри от старого, необходимо изменить свойства кластера.

18. Откройте страницу «Виртуализация» — «Узлы» и на старом хосте подтвердите, что узел был перезагружен.

19. Далее переведите старый хост в режим обслуживания.

20. Дождитесь, пока новый хост перейдет в состояние «Up».

21. Удалите файл, блокирующий дальнейшее выполнение восстановления.

Путь к файлу указан в терминале.

Дальнейшая установка продолжится автоматически.

22. Выберите тип используемого хранилища:

- Для **NFS** введите версию, полный адрес и путь к хранилищу, а также любые параметры монтирования.

**Важно!** Не используйте точку подключения старого локального домена хранилища для нового домена хранения, так как вы рискуете потерять данные виртуальной машины. ■

- Для **iSCSI** введите данные портала и выберите объект и LUN из автоматически обнаруженных списков. Вы можете выбрать только один объект iSCSI во время развертывания, но поддерживается несколько объектов для подключения всех порталов одной группы порталов.

**Примечание.** Чтобы указать более одного целевого объекта iSCSI, необходимо включить многопоточное подключение перед развертыванием self-hosted engine.

- Для хранилища **Gluster** введите полный адрес и путь к хранилищу, а также любые параметры подключения.

**Важно!** Не используйте точку подключения старого локального домена хранилища для нового домена хранения, так как вы рискуете потерять данные виртуальной машины. ■

**Важно!** Поддерживаются только реплика 1 и реплика 3 хранилища Gluster. Убедитесь, что вы настроили том следующим образом: ■

```
gluster volume set VOLUME_NAME group virt
gluster volume set VOLUME_NAME performance.strict-o-direct on
gluster volume set VOLUME_NAME network.remote-dio off
gluster volume set VOLUME_NAME storage.owner-uid 36
gluster volume set VOLUME_NAME storage.owner-gid 36
gluster volume set VOLUME_NAME network.ping-timeout 30
```

- Для **Fibre Channel** выберите LUN из автоматически определяемого списка. Адаптеры главной шины должны быть настроены и подключены, а LUN не должен содержать никаких существующих данных.

23. Введите размер диска Engine.

Сценарий продолжается до тех пор, пока развертывание не будет завершено.

24. Процесс развертывания изменяет SSH-ключи Engine. Чтобы клиентские компьютеры могли получить доступ к новому Engine без ошибок SSH, удалите запись исходного Engine из файла `.ssh/known_hosts` на всех клиентских компьютерах, которые обращались к исходному Engine.

## 3.2 Переустановка хостов

**Важно!** При установке или переустановке операционной системы хоста РЕД Виртуализации настоятельно рекомендуем сначала отключить любое существующее хранилище, не относящееся к ОС, которое подключено к хосту, чтобы избежать случайной инициализации этих дисков и потенциальной потери данных. ■

Требования:

- Если в кластере включена миграция, виртуальные машины могут автоматически мигрировать на другой хост в этом кластере. Поэтому переустановку узла следует осуществлять, пока его использование относительно невелико.
- Убедитесь, что в кластере хостам будет достаточно памяти для выполнения обслуживания. Если в кластере будет не хватать памяти, миграция виртуальных машин зависнет, а затем завершится сбоем. Чтобы уменьшить использование памяти, выключите некоторые или все виртуальные машины, прежде чем переводить узел на техническое обслуживание.
- Перед выполнением переустановки убедитесь, что кластер содержит более одного хоста. Не пытайтесь переустановить все хосты одновременно. Один хост должен оставаться доступным для выполнения задач Storage Pool Manager (SPM).

Для переустановки хоста выполните следующие действия:

1. Нажмите «**Виртуализация**» - «**Узлы**» и выберите хост.
2. Нажмите «**Управление**» - «**Обслуживание**» и «**ОК**».

3. Нажмите «**Установка**» - «**Переустановить**». Откроется окно «**Установить хост**».

4. Перейдите на вкладку **Hosted Engine** и выберите «**DEPLOY**» в раскрывающемся списке.

5. Нажмите «**ОК**», чтобы запустить переустановку хоста.

После того, как хост будет переустановлен и его статус вернется к «**Up**», можно перенести виртуальные машины обратно на узел.

**Важно!** После того, как узел РЕД Виртуализации будет зарегистрирован в HostedEngine и переустановлен, портал администрирования может ошибочно отобразить его статус как «**Install Failed**».

Нажмите «**Управление**» - «**Активировать**», после этого хост перейдет в состояние «**Up**» и будет готов к использованию. ■

После переустановки узлов self-hosted Engine вы можете проверить состояние новой среды, выполнив следующую команду на одном из узлов:

```
hosted-engine --vm-status
```

Во время восстановления старый домен хранилища с собственным хостингом будет переименован, но не удален из новой среды на случай, если восстановление было ошибочным. Убедившись, что среда работает нормально, можно удалить старый домен хранилища self-hosted Engine.

### 3.3 Удаление домена хранения

Если существует домен хранения в центре обработки данных, который необходимо удалить из виртуализированной среды, выполните следующий алгоритм действий:

1. Нажмите «**Хранилище**» - «**Домены**».
2. Переведите домен хранения в режим обслуживания и отсоедините его:
  - Нажмите на имя домена хранения. Откроется подробное описание.
  - Перейдите на вкладку «**Дата-центр**».
  - Нажмите «**Обслуживание**», затем нажмите «**ОК**».
  - Нажмите «**Отсоединить**», затем нажмите «**ОК**».

3. Щелкните «**Удалить**».

4. При необходимости выберите «**Формат домена**» (содержимое хранилища будет потеряно!), установите флажок, чтобы стереть содержимое домена.

5. Нажмите «**ОК**».

Домен хранения будет навсегда удален из среды.

### 3.4 Восстановление self-hosted Engine из существующей резервной копии

Если self-hosted Engine доступен, но возникла такая проблема, как повреждение базы данных или ошибка конфигурации, которую трудно откатить, вы



можете восстановить среду до предыдущего состояния, используя резервную копию, созданную до возникновения проблемы, если таковая доступна.

Восстановление предыдущего состояния self-hosted Engine включает следующие шаги:

1. Перевод среды в режим глобального обслуживания.
2. Восстановление резервной копии на виртуальной машине Engine.
3. Отключение режима глобального обслуживания.

Дополнительные сведения о параметрах `engine-backup --mode=restore` см. в разд. 1 Резервное копирование и восстановление HostedEngine.

### 3.5 Включение режима глобального обслуживания

Перед выполнением любых задач по настройке или обновлению на виртуальной машине Engine необходимо перевести среду self-hosted Engine в режим глобального обслуживания, для этого необходимо выполнить следующие действия:

1. Войдите в один из узлов self-hosted Engine и включите режим глобального обслуживания:

```
hosted-engine --set-maintenance --mode=global
```

2. Прежде чем продолжить, убедитесь, что среда находится в режиме глобального обслуживания:

```
hosted-engine --vm-status
```

Вы должны увидеть сообщение о том, что кластер находится в глобальном режиме обслуживания.

### 3.6 Восстановление резервной копии для перезаписи существующей установки

Команда `engine-backup` может восстановить резервную копию на машине, на которой уже установлен и настроен HostedEngine. Это полезно, когда вы сделали резервную копию среды, внесли изменения в эту среду, а затем хотите отменить изменения, восстановив среду из резервной копии.

Изменения, внесенные в среду после создания резервной копии, такие как добавление или удаление хоста, не будут отображаться в восстановленной среде. Необходимо будет повторить эти изменения.

Для восстановления резервной копии выполните следующие действия:

1. Войдите в систему Engine.
2. Удалите файлы конфигурации и очистите базу данных, связанную с Engine:

```
engine-cleanup
```

Команда `engine-cleanup` очищает только базу данных Engine; она не удаляет базу данных и не удаляет пользователя, которому принадлежит эта база данных.

3. Восстановите полную резервную копию или резервную копию только базы данных. Вам не нужно создавать новую базу данных или указывать учетные данные базы данных, поскольку пользователь и база данных уже существуют.

- Восстановить полную резервную копию:

```
engine-backup --mode=restore --file=<file_name> --log=<log_file_name> --restore-permissions
```

- Восстановить резервную копию только базы данных, восстановив файлы конфигурации и резервную копию базы данных:

```
engine-backup --mode=restore --scope=files --scope=db --scope=dwhdb --file=<file_name> --log=<log_file_name> --restore-permissions
```

**Примечание.** Чтобы восстановить только базу данных Engine (например, если база данных хранилища данных находится на другом компьютере), параметр `--scope=dwhdb` можно опустить.

В случае успеха будет отображен следующий вывод:

```
You should now run engine-setup.  
Done.
```

4. Переустановите Engine:

```
engine-setup
```

### 3.7 Отключение режима глобального обслуживания

Для отключения режима глобального обслуживания выполните следующий алгоритм действий:

1. Войдите в виртуальную машину Engine и выключите ее.
2. Войдите в один из узлов self-hosted Engine и отключите режим глобального обслуживания:

```
hosted-engine --set-maintenance --mode=none
```

Когда вы выходите из режима глобального обслуживания, `ovirt-ha-agent` запускает виртуальную машину Engine, после чего Engine автоматически запускается. Запуск Engine может занять до десяти минут.

3. Убедитесь, что среда работает:

```
hosted-engine --vm-status
```

Перечисленная информация включает статус Engine. Значение статуса Engine должно быть:

```
"health": "good", "vm": "up", "detail": "Up"
```

**Примечание.** Когда виртуальная машина все еще загружается, а Engine еще не запущен, статус Engine будет следующим:

```
"reason": "bad vm status", "health": "bad", "vm": "up", "detail":  
"Powering up"
```

В этом случае подождите несколько минут и повторите попытку.

Когда среда снова заработает, вы можете запустить все виртуальные машины, которые были остановлены, и проверить, что ресурсы в среде ведут себя должным образом.