RuBackup
Система резервного копирования и восстановления данных

Создание сертификатов и ключей SSL

RuBackup

Версия 2.3.0

25.10.2024

Содержание

Введение....3
Процедура создания ключей и сертификатов....4
Создание серверного сертификата:....4
Создание клиентского сертификата:....5
Создание сертификата для оконного менеджера:....6
Использование цепочки сертификатов....7
Подготовка сертификатов для сервера....7
Подготовка сертификатов для клиента....8
Проверка созданных ключей и сертификатов....9
Размещение сертификатов и ключей....10

Введение

В комплекте поставки RuBackup есть необходимые для работы SSL-сертификаты клиента и сервера. В этом руководстве описан процесс создания собственных ключей и сертификатов вместо тех, которые входят в стандартную поставку.

Сертификаты, необходимые для работы RuBackup, располагаются в каталоге /opt/rubackup/keys и предоставляются в составе пакета rubackup-common.

В процессе подключения к серверу клиент отправляет свой сертификат /opt/rubackup/keys/client/clientCert.crt для проверки подлинности клиента сервером. Также клиент принимает от сервера его сертификат /opt/rubackup/keys/server/serverCert.crt и проверяет его подлинность с использованием серверного корневого сертификата /opt/rubackup/keys/rootCA/serverRootCACert.crt. Сервер проверяет подлинность полученного клиентского сертификата с помощью клиентского корневого сертификата /opt/rubackup/keys/rootCA/clientRootCACert.crt.

При подключении к серверу оконный менеджер отправляет свой сертификат /opt/rubackup/keys/rbm/rbmCert.crt на проверку. Также он принимает от сервера его сертификат /opt/rubackup/keys/server/serverCert.crt и проверяет его подлинность с использованием серверного корневого сертификата /opt/rubackup/keys/rootCA/serverRootCACert.crt. Сервер проверяет подлинность полученного сертификата оконного менеджера с помощью клиентского корневого сертификата /opt/rubackup/keys/rootCA/clientRootCACert.crt.

Для взаимодействия с сервером лицензий и проверки его на подлинность используется корневой сертификат сервера лицензий /opt/rubackup/keys/rootCA/licenseServerRootCACert.crt.

Процедура создания ключей и сертификатов

Создание серверного сертификата:

Чтобы создать серверный сертификат, выполните следующие шаги:

1. Создайте приватный ключ для серверного корневого сертификата командой:

```
# openssl genrsa -out serverRootCAKey.key 2048
```

Примечание: храните этот ключ в надежном месте!

2. Создайте серверный корневой сертификат. В представленном примере сертификат действует 20000 дней:

```
# openssl req -x509 -new -nodes -key serverRootCAKey.key -days 20000 -out /opt/rubackup/keys/rootCA/serverRootCACert.crt
```

3. В интерактивном меню введите двухбуквенный код страны, провинцию, город, организацию, подразделение, Common Name и e-mail адрес.

4. Создайте приватный ключ сервера:

```
# openssl genrsa -out /opt/rubackup/keys/server/serverKey.key 2048
```

5. Создайте запрос на подпись:

```
# openssl req -new -key /opt/rubackup/keys/server/serverKey.key -out /opt/rubackup/keys/server/serverCert.csr
```

6. В интерактивном меню впишите ответ на те же вопросы, что и при создании корневого сертификата. Введенный Common Name должен отличаться от Common Name у корневого сертификата.

7. Создайте серверный сертификат и подпишите его серверным корневым сертификатом. В представленном примере сертификат действует 20000 дней:

```
# openssl x509 -req -in /opt/rubackup/keys/server/serverCert.csr -CA /opt/rubackup/keys/rootCA/serverRootCACert.crt -CAkey
```

serverRootCAKey.key -CAcreateserial -out /opt/rubackup/keys/server/serverCert.crt -days 20000

8. При необходимости пересоздайте файл, используемый в алгоритме Диффи-Хеллмана, для обмена сессионными ключами с клиентом:

```
# openssl dhparam -out /opt/rubackup/keys/server/dh_2048.pem 2048
```

Создание клиентского сертификата:

Чтобы создать клиентский сертификат, выполните следующие шаги:

1. Создайте приватный ключ для клиентского корневого сертификата командой:

```
# openssl genrsa -out clientRootCAKey.key 2048
```

Примечание: храните этот ключ в надежном месте!

2. Создайте клиентский корневой сертификат. В представленном примере сертификат действует 20000 дней:

 # openssl req -x509 -new -nodes -key serverRootCAKey.key -days 20000 -out /opt/rubackup/keys/rootCA/clientRootCACert.crt

3. В интерактивном меню введите двухбуквенный код страны, провинцию, город, организацию, подразделение, Common Name и e-mail адрес.

4. Создайте приватный ключ клиента:

```
# openssl genrsa -out /opt/rubackup/keys/client/clientKey.key 2048
```

5. Создайте запрос на подпись:

 # openssl req -new -key /opt/rubackup/keys/client/clientKey.key -out /opt/rubackup/keys/client/clientCert.csr

6. В интерактивном меню впишите ответ на те же вопросы, что и при создании корневого сертификата. Введенный Common Name должен отличаться от Common Name у корневого сертификата.

7. Создайте клиентский сертификат и подписать его клиентским корневым сертификатом. В представленном примере сертификат действует 20000 дней:

```
# openssl x509 -req -in /opt/rubackup/keys/client/clientCert.csr -CA /opt/rubackup/keys/rootCA/clientRootCACert.crt -CAkey clientRootCAKey.key -CAcreateserial -out /opt/rubackup/keys/client/clientCert.crt -days 20000
```

Создание сертификата для оконного менеджера:

Чтобы создать сертификат для оконного менеджера, выполните следующие шаги:

1.Создайте приватный ключ оконного менеджера:

```
# openssl genrsa -out /opt/rubackup/keys/rbm/rbmKey.key 2048
```

3. Создайте запрос на подпись:

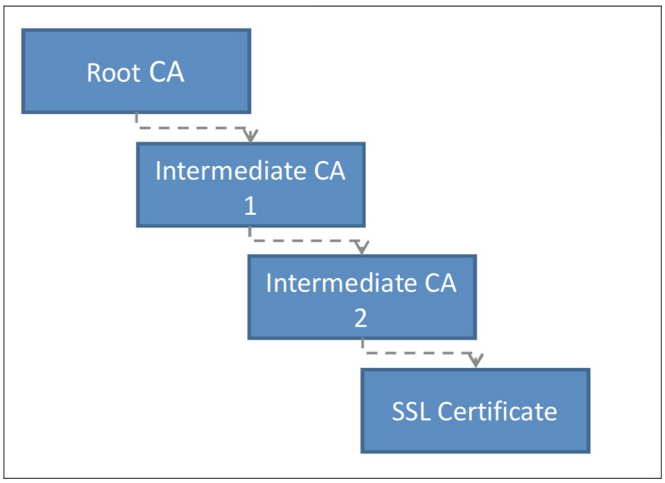
```
# openssl req -new -key /opt/rubackup/keys/rbm/rbmKey.key -out /opt/rubackup/keys/rbm/rbmCert.csr
```

3. В интерактивном меню впишите ответ на те же вопросы, что и при создании корневого сертификата. Введенный Common Name должен отличаться от Common Name у корневого сертификата.

4. Создайте сертификат оконного менеджера и подпишите его клиентским корневым сертификатом. В представленном примере сертификат действует 20000 дней:

```
# openssl x509 -req -in /opt/rubackup/keys/rbm/rbmCert.csr -CA /opt/rubackup/keys/rootCA/clientRootCACert.crt -CAkey clientRootCAKey.key -CAcreateserial -out /opt/rubackup/keys/rbm/rbmCert.crt -days 20000
```

Использование цепочки сертификатов



Иногда клиентский или серверный сертификат подписывается не корневым клиентским или серверным сертификатом, а промежуточным сертификатом, который, в свою очередь, подписан корневым или следующим промежуточным сертификатом. Это называется цепочкой сертификатов.

Чтобы RuBackup мог работать с такой цепочкой сертификатов, необходимо объединить все промежуточные и корневой сертификаты в единый корневой клиентский или серверный сертификат.

Подготовка сертификатов для сервера

Чтобы подготовить сертификаты для сервера, выполните следующие шаги:

1. Разместите в отдельной папке промежуточные сертификаты и корневой сертификат.

2. Если некоторые из промежуточных или корневой сертификат имеют расширение .cer или .pem, конвертируйте их в формат .crt с помощью одной из следующих команд:

 # openssl x509 -in '<имя сертификата>.pem' -out '<имя сертификата>.crt' -outform DER

 # openssl x509 -inform PEM -in '<имя сертификата>.cer' -out '<имя сертификата>.crt'

3. Объедините промежуточные сертификаты и корневой сертификаты в единый корневой серверный сертификат:

```
# cat <путь к промежуточному сертификату 1> <путь к промежуточному сертификату 2> <путь к корневому сертификату> /opt/rubackup/keys/rootCA/serverRootCACert.crt
```

Подготовка сертификатов для клиента

Чтобы подготовить сертификаты для клиента, выполните следующие шаги:

1. 1. Разместите в отдельной папке промежуточные сертификаты и корневой сертификат.

2. Если некоторые из промежуточных или корневой сертификат имеют расширение .cer или .pem, конвертируйте их в формат .crt с помощью одной из следующих команд:

 # openssl x509 -in '<имя сертификата>.pem' -out '<имя сертификата>.crt' -outform DER

 # openssl x509 -inform PEM -in '<имя сертификата>.cer' -out '<имя сертификата>.crt'

3. Объедините промежуточные сертификаты и корневой сертификат в единый корневой клиентский сертификат:

```
# cat <путь к промежуточному сертификату 1> <путь к промежуточному сертификату 2> <путь к корневому сертификату> /opt/rubackup/keys/rootCA/clientRootCACert.crt
```

Проверка созданных ключей и сертификатов

Для проверки созданных ключей и сертификатов выполните следующие шаги:

2. Проверьте сертификат сервера:

```
# openssl verify -no-CApath -CAfile
/opt/rubackup/keys/rootCA/serverRootCACert.crt
/opt/rubackup/keys/server/serverCert.crt
```

Вывод команды должен содержать: **OK**.

3. Проверьте сертификат клиента:

```
# openssl verify -no-CApath -CAfile
/opt/rubackup/keys/rootCA/clientRootCACert.crt
/opt/rubackup/keys/client/clientCert.crt
```

Вывод команды должен содержать: **OK**.

4. Проверьте сертификат оконного менеджера:

```
# openssl verify -no-CApath -CAfile
/opt/rubackup/keys/rootCA/clientRootCACert .crt
/opt/rubackup/keys/rbm/rbmCert.crt
```

Вывод команды должен содержать: **OK**.

Размещение сертификатов и ключей

Файлы приватных ключей следует хранить в надёжном месте, недоступном ни с сервера, ни с клиента RuBackup.

При замене сертификатов на собственные необходимо убедиться, что все сертификаты обновлены на всех узлах, где установлены компоненты RuBackup: клиент, сервер, медиасервер, резервный сервер, оконный менеджер, REST API сервис и другие.