Обновление пакетов десктоп-клиента на РЕД ОС при установке в закрытом контуре

Когда нет доступа к внешней сети или есть доступ только к определенным ресурсам, то можно обновить пакеты десктоп-клиента на РЕД ОС в закрытом контуре.

1. Создайте каталог на ПК с помощью команды:

```bash
$ mkdir /home/user/repo
```

2. Скачайте пакет `termit-desktop` в каталог `repo`.
3. Скачайте пакеты из репозитория в каталог `repo` с помощью команды:

```bash
$ dnf repoquery --requires --resolve --recursive ./termit-desktop-2.***.rpm
| grep -v "i686" | sort -u > packages.txt
$ dnf download $(cat packages.txt) --downloaddir .
```

4. Перед созданием репозитория установите пакет с помощью команды:

```bash
sudo dnf install createrepo_c
```

5. Создайте репозиторий с помощью команды:

```bash
$ createrepo_c .
```

6. Перенесите пакеты на другой ПК, на котором нет доступа к внешней сети.
7. Отключите репозитории на ПК с помощью команды:

```bash
$ sudo dnf config-manager --set-disabled kernels updates base
```

8. Добавьте локальный репозиторий с помощью команды:

```bash
$ sudo dnf config-manager --add-repo /home/user/repo
```

9. Отключите проверку ключей с помощью команды:

```bash
$ sudo dnf config-manager --save --setopt=home_user_repo.gpgcheck=0
```

10. Установите пакет с помощью команды:

```
BASH |
$ sudo dnf install termit-desktop-2.***.rpm
```

Установка шлюза удаленного доступа

В этом разделе описано, как установить шлюз удаленного доступа и настроить аутентификацию.

Установка шлюза

Установите компонент `openssh-server`:

▸ РЕД ОС

▸ Astra Linux

▸ Debian

▸ OpenSUSE

▸ ALT Linux

Настройка шлюза удаленного доступа

Настройка аутентификации шлюза по LDAP

Настройте шлюз удаленного доступа через ввод в домен. Подробнее о вводе в домен смотрите в статье на сайтах:

- РЕД ОС
- Astra Linux
- Debian
- OpenSUSE
- ALT Linux

По умолчанию при потере соединения сессия остается в статусе «Активная».

Чтобы изменить поведение сессии и перевести ее в статус «Отключена» в панели администрирования, необходимо в файле конфигурации шлюза удаленного доступа `/etc/ssh/sshd_config` изменить параметры `ClientAliveInterval` и `ClientAliveCountMax`.

Пример:

- `ClientAliveInterval` — 15
- `ClientAliveCountMax` — 3

Сессия переходит в статус «Отключена» через 45 секунд.

Настройка аутентификации десктоп-клиента по сертификату на шлюзе

Чтобы настроить аутентификацию десктоп-клиента по сертификату на шлюзе удаленного доступа:

1. Скачайте корневой сертификат в настройках внешнего подключения.
2. Добавьте в файл конфигурации `/etc/ssh/sshd_config` (для Alt Linux `/etc/openssh/sshd_config`) строки:

   ```
   BASH
   TrustedUserCAKeys /etc/ssh/key.pub
   ```

   (для Alt Linux `TrustedUserCAKeys /etc/openssh/key.pub`)

   Если не используете предыдущие версии СТД «Термит»:

   ```
   BASH
   AuthenticationMethods publickey
   ```

   Если используете предыдущие версии СТД «Термит»:

   ```
   BASH
   AuthenticationMethods publickey password
   ```

3. Назначьте права на файл с помощью команды:

   ```
   BASH
   sudo chmod 600 /etc/ssh/key.pub
   ```

4. Перезагрузите sshd с помощью команды:

   ```
   BASH
   sudo systemctl restart sshd
   ```

Сервер настроен на допуск всех пользователей, предоставляющих сертификат, выданный центром сертификации (root_ca), при подключении.

Установка и настройка HAProxy

HAProxy (High Availability Proxy) — универсальный балансировщик нагрузки с открытым исходным кодом и программное обеспечение прокси-сервера. Его задача — эффективно распределять входящий сетевой трафик между несколькими брокерами для обеспечения высокой доступности, надежности и оптимальной производительности системы.

> Для корректной работы системы выберите тип балансировки с привязкой пользователя к определенному серверу: balance source и hash-type consistent.
>
> Подробнее о типах балансировки.

В этом разделе описано, как установить и настроить балансировщик нагрузки HAProxy для операционных систем: РЕД ОС, Astra Linux, Debian, Alt Linux и OpenSUSE.

РЕД ОС

1. Установите ПО HAProxy с помощью команды:

   ```bash
   sudo dnf install haproxy
   ```

2. Включите службу haproxy и добавьте автозагрузку с помощью команды:

   ```bash
   sudo systemctl enable haproxy --now
   ```

3. Добавьте в конец файла конфигурации `/etc/haproxy/haproxy.cfg` настройку серверов для балансировки.

   ▸ Пример конфигурации для внутреннего балансировщика

   ▸ Пример конфигурации для внешнего балансировщика

4. Проверьте созданный файл конфигурации с помощью команды ниже. Вывод команды не должен содержать ошибок. Предупреждения (warnings) не являются ошибками.

   ```bash
   sudo haproxy -f /etc/haproxy/haproxy.cfg -c
   ```

5. Перезапустите службу с помощью команды:

```bash
sudo systemctl restart haproxy
```

6. Разрешите работу в SELinux с помощью команд:

▸ **для внутреннего балансировщика**

▸ **для внешнего балансировщика**

7. Проверьте доступность СТД «Термит» по URL адресу, указанному при установке брокер серверов.

Astra Linux

Для установки используется последняя доступная в репозитории Astra Linux версия HAProxy. Подробнее о конфигурации можно ознакомиться на официальном сайте Astra Linux.

Для корректной работы HAProxy на Astra Linux в режимах высокой или максимальной защищенности может понадобится отключение или дополнительная настройка ненулевых классификационных меток. Подробнее о настройке можно ознакомиться на официальном сайте Astra Linux.

1. Установите ПО HAProxy с помощью команды:

```bash
sudo apt install haproxy
```

2. Включите службу haproxy и добавьте автозагрузку с помощью команды:

```bash
sudo systemctl enable haproxy --now
```

3. Добавьте в конец файла конфигурации `/etc/haproxy/haproxy.cfg` настройку серверов для балансировки.

▸ **Пример конфигурации для внутреннего балансировщика**

▸ **Пример конфигурации для внешнего балансировщика**

4. Проверьте созданный файл конфигурации с помощью команды ниже. Вывод команды не должен содержать ошибок. Предупреждения (warnings) не являются ошибками.

```bash
sudo haproxy -f /etc/haproxy/haproxy.cfg -c
```

5. Перезапустите службу с помощью команды:

```bash
sudo systemctl restart haproxy
```

6. Проверьте доступность СТД «Термит» по URL адресу, указанному при установке брокера.

Debian

1. Установите ПО HAProxy с помощью команды:

```bash
sudo systemctl install haproxy
```

2. Включите службу haproxy и добавьте автозагрузку с помощью команды:

```bash
sudo systemctl enable haproxy --now
```

3. Добавьте в конец файла конфигурации `/etc/haproxy/haproxy.cfg` настройку серверов для балансировки:

▸ Пример конфигурации для внутреннего балансировщика

▸ Пример конфигурации для внешнего балансировщика

4. Проверьте файл конфигурации с помощью команды:

```bash
sudo haproxy -f /etc/haproxy/haproxy.cfg -c
```

5. Перезапустите службу с помощью команды:

```bash
sudo systemctl restart haproxy
```

6. Проверьте доступность СТД «Термит» по URL адресу, указанному при установке брокера.

ALT Linux

1. Установите ПО HAProxy с помощью команды:

```bash
sudo apt-get install haproxy
```

2. Включите службу haproxy и добавьте автозагрузку с помощью команды:

```bash
sudo systemctl enable haproxy --now
```

3. Добавьте в конец файла конфигурации `/etc/haproxy/haproxy.cfg` настройку серверов для балансировки.

   ▸ **Пример конфигурации для внутреннего балансировщика**

   ▸ **Пример конфигурации для внешнего балансировщика**

4. Проверьте созданный файл конфигурации с помощью команды ниже. Вывод команды не должен содержать ошибок. Предупреждения (warnings) не являются ошибками.

```bash
sudo haproxy -f /etc/haproxy/haproxy.cfg -c
```

5. Перезапустите службу с помощью команды:

```bash
sudo systemctl restart haproxy
```

6. Проверьте доступность СТД «Термит» по URL адресу, указанному при установке брокера.

OpenSUSE

1. Установите ПО HAProxy с помощью команды:

```bash
sudo zypper install haproxy
```

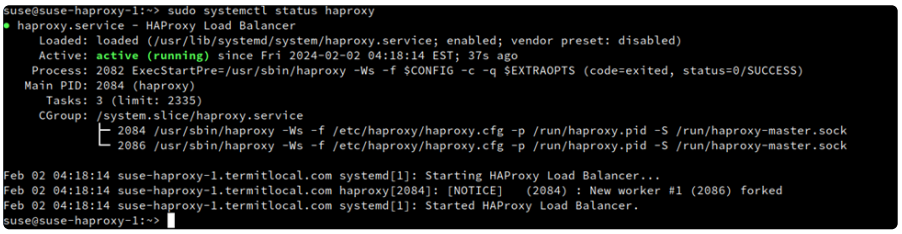
2. Запустите службу haproxy с помощью команды:

```bash
sudo systemctl enable haproxy --now
```

3. Проверьте статус службы haproxy с помощью команды:

```bash
sudo systemctl status haproxy
```

   Статус должен быть «Active».



4. Добавьте в конец файла конфигурации `/etc/haproxy/haproxy.cfg` настройку серверов для балансировки.

   ▸ Пример конфигурации для внутреннего балансировщика

   ▸ Пример конфигурации для внешнего балансировщика

5. Проверьте созданный файл конфигурации с помощью команды ниже. Вывод команды не должен содержать ошибок. Предупреждения (warnings) не являются ошибками.

   ```
   BASH
   sudo haproxy -f /etc/haproxy/haproxy.cfg -c
   ```

6. Перезапустите службу с помощью команды:

   ```
   BASH
   sudo systemctl restart haproxy
   ```

7. Проверьте статус службы с помощью команды:

   ```
   BASH
   sudo systemctl status haproxy
   ```

   Статус должен быть «Active».

8. Проверьте доступность СТД «Термит» по URL адресу, указанному при установке брокера.

Установка и настройка Keepalived

Keepalived — программный пакет с открытым исходным кодом, который обеспечивает высокую доступность (HA) и отказоустойчивость для систем на базе Linux. Он используется для:

- управления виртуальными IP-адресами (VIP);
- обеспечения бесперебойной доступности сервисов в сценариях, где могут произойти сбои системы или приложения.

В этом разделе описано, как установить и настроить Keepalived для операционных систем: РЕД ОС, Astra Linux, Debian, Alt Linux и OpenSUSE.

1. Установка Keepalived на серверах

Перед установкой проверьте, что Keepalived установлен на серверах:

- основном;
- резервном.

Его можно установить с помощью менеджера пакетов системы для:

РЕД ОС

```bash
sudo dnf install keepalived
```

Astra Linux

```bash
sudo apt install keepalived
```

Debian

```bash
sudo apt install keepalived
```

Alt Linux

```bash
sudo apt-get install keepalived
```

OpenSUSE

```bash
sudo zypper install keepalived
```

2. Настройка Keepalived на основном сервере

1. Отредактируйте файл конфигурации Keepalived, который по умолчанию находится по адресу `/etc/keepalived/keepalived.conf`, на основном сервере с помощью команды ниже:

   ```bash
   sudo nano /etc/keepalived/keepalived.conf
   ```

2. Добавьте следующую конфигурацию в файл:

   ```bash
   vrrp_instance VI_1 {

       state MASTER
       interface %Имя_сетевого_интерфейса%
       virtual_router_id %Уникальный_идентификатор_роутера_VRRP%
       priority %Приоритет_основного_сервера%
       advert_int %Интервал_обновления_состояния_сервера%
       authentication {
           auth_type PASS
           auth_pass %Пароль%
       }
       virtual_ipaddress {
           %Виртуальный_IP-адрес%
       }
   }
   ```

   Где:

   - **%Имя_сетевого_интерфейса%** — имя сетевого интерфейса, например «eth0».
   - **%Уникальный_идентификатор_роутера_VRRP%** — уникальный идентификатор роутера VRRP, например «51». Он должен совпадать на серверах.
   - **%Приоритет_основного_сервера%** — приоритет основного сервера, например «100». Он должен быть выше, чем на резервном.
   - **%Интервал_обновления_состояния_сервера%** — интервал обновления состояния сервера, например «1».
   - **%Пароль%** — пароль, например «password123». Он должен быть одинаковым на серверах.
   - **%Виртуальный_IP-адрес%** — VIP для службы, например «192.168.1.100».

3. Сохраните и закройте файл.

4. Чтобы применить изменения, перезапустите Keepalived с помощью команд:

```bash
sudo systemctl restart keepalived

sudo systemctl enable keepalived
```

3. Настройка Keepalived на резервном сервере

1. Отредактируйте файл конфигурации Keepalived, который по умолчанию находится по адресу `/etc/keepalived/keepalived.conf`, на резервном сервере с помощью команды ниже:

```bash
sudo nano /etc/keepalived/keepalived.conf
```

2. Добавьте следующую конфигурацию в файл:

```bash
vrrp_instance VI_1 {

    state BACKUP
    interface %Имя_сетевого_интерфейса%
    virtual_router_id %Уникальный_идентификатор_роутера_VRRP%
    priority %Приоритет_резервного_сервера%
    advert_int %Интервал_обновления_состояния_сервера%
    authentication {
        auth_type PASS
        auth_pass %Пароль%
    }
    virtual_ipaddress {
        %Виртуальный_IP-адрес%
    }
}
```

Где:

- **%Имя_сетевого_интерфейса%** — имя сетевого интерфейса, например «eth0».
- **%Уникальный_идентификатор_роутера_VRRP%** — уникальный идентификатор роутера VRRP, например «51». Он должен совпадать на серверах.
- **%Приоритет_резервного_сервера%** — приоритет резервного сервера, например «90». Он должен быть ниже, чем на основном.
- **%Интервал_обновления_состояния_сервера%** — интервал обновления состояния сервера, например «1».
- **%Пароль%** — пароль, например «password123». Он должен быть одинаковым на серверах.
- **%Виртуальный_IP-адрес%** — VIP для службы, например «192.168.1.100».

3. Сохраните и закройте файл.
4. Чтобы применить изменения, перезапустите Keepalived с помощью команд:

```bash
sudo systemctl restart keepalived

sudo systemctl enable keepalived
```

4. Проверка работоспособности

1. Проверьте, что Keepalived работает на серверах с помощью команды:

   ```bash
   sudo systemctl status keepalived
   ```

2. Проверьте, какой из серверов обрабатывает VIP.

   ```bash
   ip a | grep %Виртуальный_IP-адрес%
   ```

   На основном сервере этот IP-адрес должен быть привязан к сетевому интерфейсу.

3. Для тестирования отказоустойчивости отключите сетевой интерфейс на основном сервере с помощью команды:

   ```bash
   sudo ifdown %Имя_сетевого_интерфейса%
   ```

4. После отключения основного сервера VIP должен автоматически перейти на резервный сервер. Это можно проверить с помощью команды:

   ```bash
   ip a | grep %Виртуальный_IP-адрес%
   ```

5. Восстановите работу интерфейса на основном сервере:

   ```bash
   sudo ifup %Имя_сетевого_интерфейса%
   ```

Подготовка окружения

Перед развертыванием СТД «Термит» подготовьте окружение:

1. Проверьте, что подсети Docker Compose не пересекаются с сетями для серверов баз данных, LDAP-каталогов, терминалов и брокеров. Подробнее в статье Ошибки при пересечении IP-адресов между Docker и хост-системой.
2. Разверните отдельные серверы:
   - для брокеров.

     Количество брокеров зависит от типа развертывания:

     - для «Standalone» необходим один брокер;
     - для «High Availability» необходимо три брокера.
   - для терминальных серверов под управлением Windows или Linux в зависимости от требований;
   - (Опционально) для сервера баз данных. Можно использовать существующий сервер.
   - (Опционально) для RADIUS-сервера, если планируете использовать MFA.
3. Проверьте доступ к службе каталогов и PostgreSQL.
4. Добавьте терминальные серверы в домен службы каталогов.
5. В каталоге пользователей создайте сервисную учетную запись для синхронизации объектов из службы каталогов. Обязательно отключите опцию «User must change password at next logon» и включите «Password never expires».
6. В каталоге пользователей создайте группы безопасности для назначения ролей: администратора, технической поддержки и пользователя, в соответствии с требованиями вашей компании.
7. В каталоге пользователей добавьте в состав групп, созданных на предыдущем шаге, учетные записи пользователей для взаимодействия с СТД «Термит».