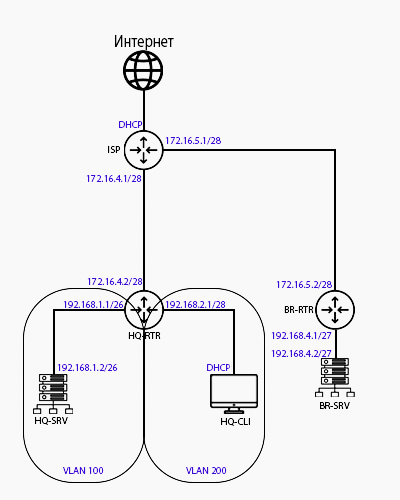
**Топология сети**

****

**Таблица адресации**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Устройство** | **Интерфейс** | **IP-адрес** | **Маска** | **VLAN** | **Подсеть** | **Шлюз** |
| **ISP** | eth0 (к интернету) | DHCP | DHCP | - | DHCP | DHCP |
|  | eth1 (к HQ-RTR) | 172.16.4.1 | 255.255.255.240 | - | 172.16.4.0/28 | - |
|  | eth2 (к BR-RTR) | 172.16.5.1 | 255.255.255.240 | - | 172.16.5.0/28 | - |
| **HQ-RTR** | eth0 (к ISP) | 172.16.4.2 | 255.255.255.240 | - | 172.16.4.0/28 | 172.16.4.1 |
|  | eth1 (Trunk) | - | - | Trunk | - | - |
|  | eth1.100 | 192.168.1.1 | 255.255.255.192 | 100 | 192.168.1.0/26 | - |
|  | eth1.200 | 192.168.2.1 | 255.255.255.240 | 200 | 192.168.2.0/28 | - |
|  | eth1.999 | 192.168.3.1 | 255.255.255.248 | 999 | 192.168.3.0/29 | - |
|  | gre1 (IP туннель) | 10.10.10.1 | 255.255.255.252 | - | 10.10.10.0/30 | - |
| **HQ-SRV** | enp0s3 (Trunk) | - | - | Trunk | - | - |
| enp0s3.100 | 192.168.1.2 | 255.255.255.192 | 100 | 192.168.1.0/26 | 192.168.1.1 |
| **HQ-CLI** | enp0s3.200 | 192.168.2.2 | 255.255.255.240 | 200 | 192.168.2.0/28 | 192.168.2.1 |
| **BR-RTR** | eth0 (к ISP) | 172.16.5.2 | 255.255.255.240 | - | 172.16.5.0/28 | 172.16.5.1 |
|  | eth1 (к BR-SRV) | 192.168.4.1 | 255.255.255.224 | - | 192.168.4.0/27 | - |
| gre1 (IP туннель) | 10.10.10.2 | 255.255.255.252 | - | 10.10.10.0/30 | - |
| **BR-SRV** | enp0s3 (к BR-RTR) | 192.168.4.2 | 255.255.255.224 | - | 192.168.4.0/27 | 192.168.4.1 |

**Версии дистрибутивов к соответствующим устройствам (ссылки):**

**ISP, HQ-RTR, BR-RTR –**

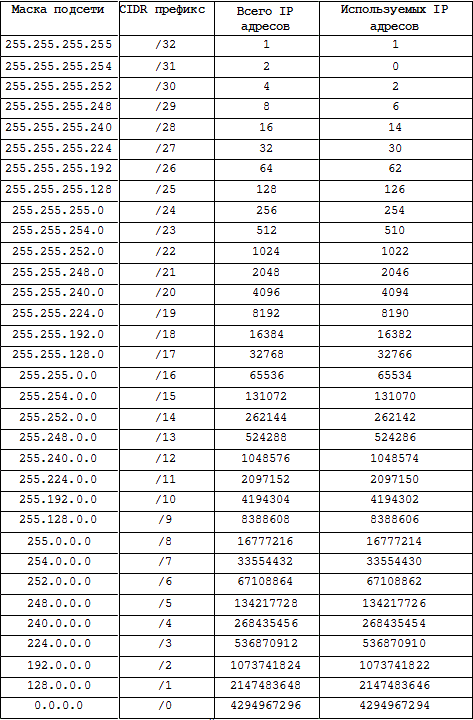
[**https://dl.astralinux.ru/astra/stable/2.12\_x86-64/iso/alce-2.12.46.6-17.04.2023\_15.09.iso**](https://dl.astralinux.ru/astra/stable/2.12_x86-64/iso/alce-2.12.46.6-17.04.2023_15.09.iso)

**HQ-SRV, BR-SRV –**

[**https://download.basealt.ru/pub/distributions/ALTLinux/p10/images/server/x86\_64/alt-server-10.2-x86\_64.iso**](https://download.basealt.ru/pub/distributions/ALTLinux/p10/images/server/x86_64/alt-server-10.2-x86_64.iso)

**HQ-CLI –**

[**https://download.basealt.ru/pub/distributions/ALTLinux/p10/images/workstation/x86\_64/alt-workstation-10.1-x86\_64.iso**](https://download.basealt.ru/pub/distributions/ALTLinux/p10/images/workstation/x86_64/alt-workstation-10.1-x86_64.iso)Таблица масок

****

**МОДУЛЬ №1**

1. **Произведите базовую настройку устройств**

**ВСЕ СЛЕДУЮЩИЕ НАСТРОЙКИ ПРОИЗВОДЯТСЯ ОТ root!!!**

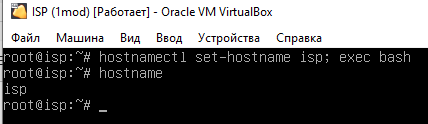
**Команда для перехода в режим суперпользователя:**

**su - (ALT Linux)**

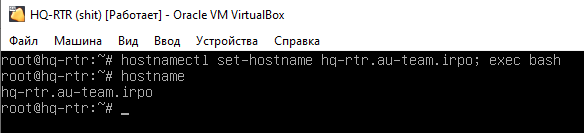
**sudo -i (ASTRA Linux)**

* 1. **Настройте имена устройств согласно топологии. Используйте полное доменное имя.**

Настроим имя на **ISP**:



Настроим им на **HQ-RTR**:



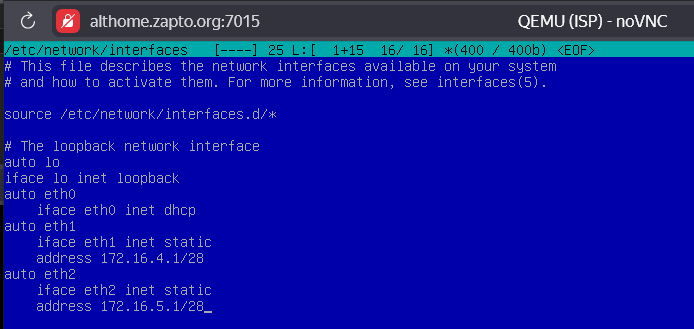
АНАЛОГИЧНО НА ДРУГИХ УСТРОЙСТВАХ!

* 1. **На всех устройствах необходимо сконфигурировать IPv4**

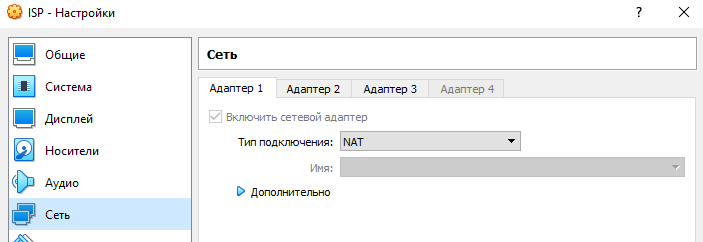
**Адресация на ISP:**

Настраивать будем через следующую команду:

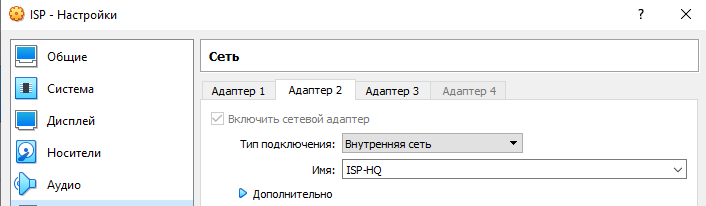
**mcedit /etc/network/interfaces**



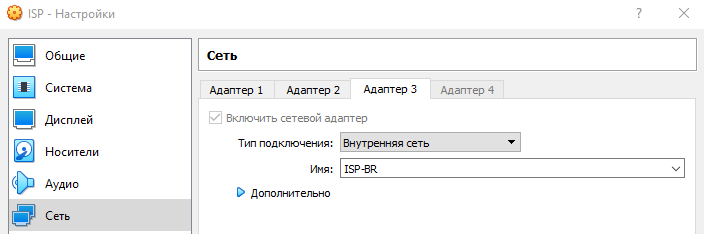
**eth0** – интерфейс, подключенный к провайдеру (ИнтЫрнет), должен быть по dhcp



**eth1** – интерфейс, подключенный к ISP-HQ, должна быть настроена static (см. [Таблица адресации](#_heading=h.l9s1roti7rez))



**eth2** – интерфейс, подключенный к ISP-BR, должна быть настроена static (см. [Таблица адресации](#_heading=h.l9s1roti7rez)).



Маска 255.255.255.240 (/28) была выбрана с условием, что сеть должна вмещать не более 32 хостов (см. [Таблица масок](#_heading=h.ey0kpzprx7z5))

Из **mcedit** выходим нажатием **F2** для сохранения изменений и **F10** для выхода из него.

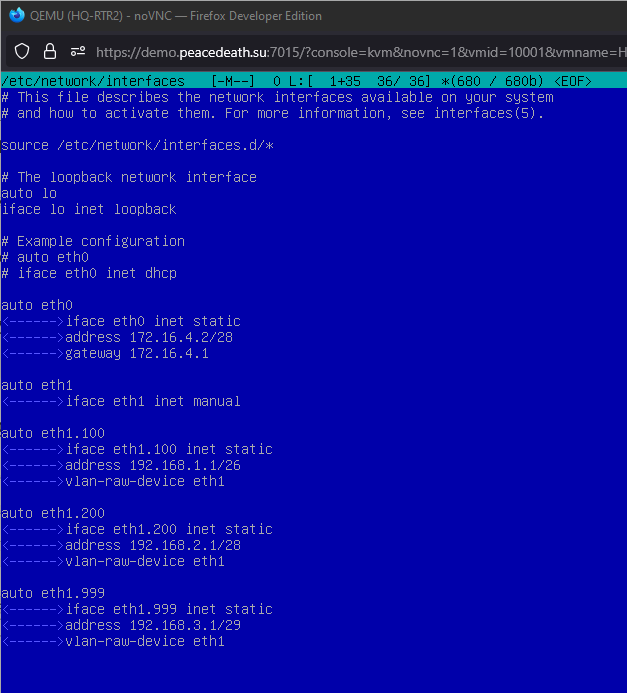
**systemctl restart networking** (перезапуск службы сети для применения изменений на Astra Linux)

**systemctl restart network** (тоже самое, только на Alt Linux)

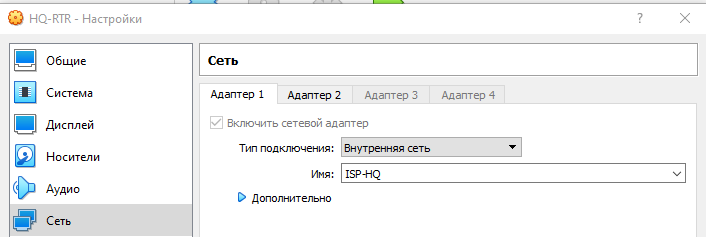
**Адресация на HQ-RTR:**

Настраивать будем через следующую команду:

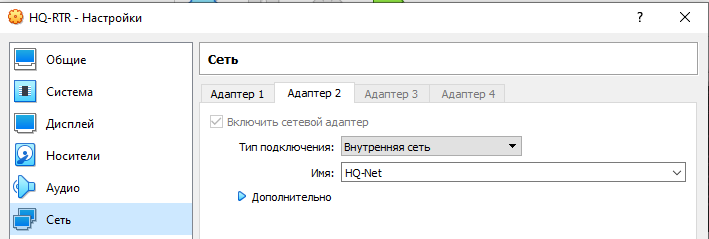
**mcedit /etc/network/interfaces**



**eth0** – интерфейс, подключенный к ISP-HQ, должна быть настроена static (см. [Таблица адресации](#_heading=h.l9s1roti7rez)).



**eth1 –** интерфейс, подключенный к HQ-Net, должен быть настроен на manual, так как далее мы на нём будем настраивать VLAN`ы.



**eth1.100** – интерфейс, подключенный к HQ-Net, должен быть настроен на static и настроен на VLAN 100 с маской /26. Локальная сеть в сторону HQ-SRV(VLAN100) должна вмещать не более 64 адресов.

**eth1.200** – интерфейс, подключенный к HQ-Net, должен быть настроен на static и настроен на VLAN 200 с маской /28. Локальная сеть в сторону HQ-CLI(VLAN200) должна вмещать не более 16 адресов.

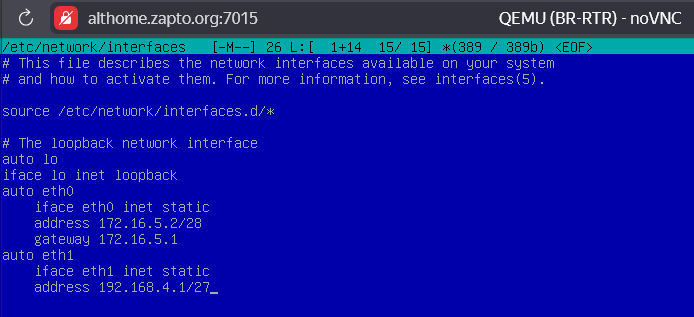
**eth1.999** – интерфейс, подключенный к HQ-Net, должен быть настроен на static и настроен на VLAN 999 с маской /29. Локальная сеть для управления(VLAN999) должна вмещать не более 8 адресов.

**systemctl restart networking** (перезапуск службы сети для применения изменений)

**Адресация на BR-RTR:**

Настраивать будем через следующую команду:

**mcedit /etc/network/interfaces**

****

**eth0 -** интерфейс, подключенный к ISP-BR, должна быть настроена static (см. [Таблица адресации](#_heading=h.l9s1roti7rez)).

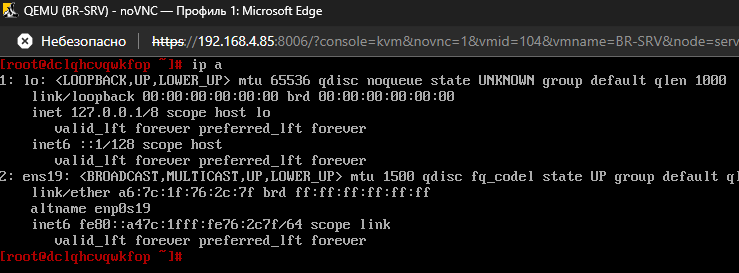
**eth1 -** интерфейс, подключенный к BR-Net, должна быть настроена static (см. [Таблица адресации](#_heading=h.l9s1roti7rez)). Локальная сеть в сторону BR-SRV должна вмещать не более 32 адресов.

**systemctl restart networking** (перезапуск службы сети для применения изменений)!

**Адресация на BR-SRV:**

Альт отличается настройкой, как минимум тем, что в нём для настройки интерфейса нужно использовать отдельный каталог и внутри ещё каталоги, сейчас всё увидите, перейдём в каталог нужного нам интерфейса, но для начала посмотрим наши интерфейсы через команду:

**ip a**

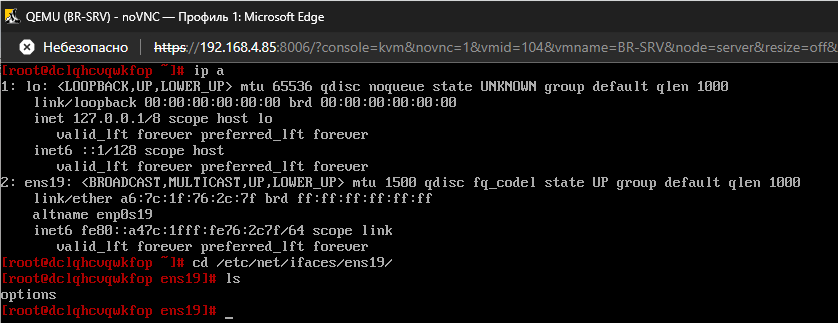


Видим, что нужный нам интерфейс имеет название **ens19** (У вас может отличаться, смотрите внимательно)

Переходим в каталог этого интерфейса:

**cd /etc/net/ifaces/ens19**

**ls (выводим содержимое этого каталога)**



Теперь будем настраивать файл, который здесь лежит, остальные создадим сами, приступаем.

Первым делом настраивать будем options через следующую команду:

**mcedit /etc/net/ifaces/ens19/options**

Если вы уже в каталоге, и делали всё по нашим шагам, то просто:

**mcedit options**

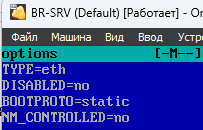
Приведём строки в файле к следующему виду:

**TYPE=eth**

**DISABLED=no**

**BOOTPROTO=static**

**NM\_CONTROLLED=no**

****

**СОХРАНЯЕМ ИЗМЕНЕНИЯ НАЖАТИЕМ КЛАВИШИ F2 И ЗАКРОЕМ КЛАВИШЕЙ F10, ЗАПОМНИТЕ, РЕБЯТИШКИ!!!**

Далее настроим файл ipv4address (если его нет, то он создастся):

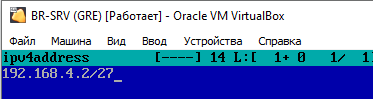
**mcedit /etc/net/ifaces/ens19/ipv4address**

Если вы уже в каталоге, и делали всё по нашим шагам, то просто:

**mcedit ipv4address**

Внесём туда следующую строку:

**192.168.4.2/27** (см. [Таблица адресации](#_heading=h.l9s1roti7rez))



Далее настроим файл ipv4route:

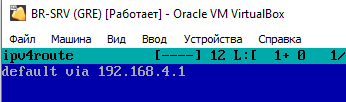
**mcedit /etc/net/ifaces/ens19/ipv4route**

Если вы уже в каталоге, и делали всё по нашим шагам, то просто:

**mcedit ipv4route**

Внесём туда следующую строку:

**default via 192.168.4.1** (см. [Таблица адресации](#_heading=h.l9s1roti7rez))



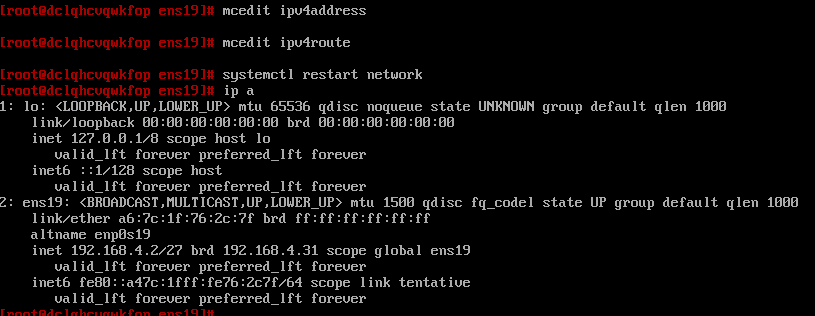
Настройка адресации на **BR-SRV** завершена!

Перезапускаем службу network командой:

**systemctl restart network**

И смотрим ещё раз данные об интерфейсах командой:

**ip a**

****

Всё успешно! Но если у вас на Альт установлена служба **systemd-networkd**, то придётся вносить запись о перезагрузке службы **network** в **crontab**, как у нас. В Альт существует по умолчанию своя служба управлению сетью – это **Etcnet** (никто не знает зачем, из-за неё как раз и приходится создавать отдельный каталог для интерфейсов), а вместе **Etcnet** и **systemd-networkd** работать не могут, поэтому вы можете просто отключить службу **systemd-networkd** командой:

**systemctl disable --now systemd-networkd**

***!ВАЖНО!*** *Это касается ТОЛЬКО СЕРВЕРОВ!!!*

В ином случае, когда обе службы работают, нужно будет взаимодействовать с **crontab**, как это сделано далее, чтобы адрес не пропадал с интерфейса.

Делаем это следующим образом, пишем команду:

**export EDITOR=mcedit**

А затем:

**crontab -e**

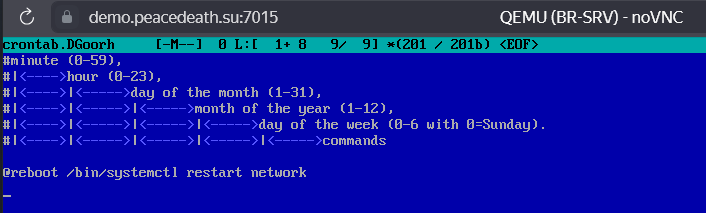
****

Мы попадаем в конфиг, где указываются различные задачи, которые выполняются в назначенное время. В нашем случае нужно перезагружать службу **network** каждый раз после перезапуска системы.

Для этого мы в конце файла пишем следующее:

**@reboot /bin/systemctl restart network**

***!ВАЖНО!*** *Оставляем пустую строку после введённой строки выше, иначе не будет сохранения! В этом файле всегда нужно оставлять снизу* ***ПУСТУЮ СТРОКУ****!*



Если всё сделано успешно, то появится следующее сообщение в консоли:



И теперь вы можете перезагружать спокойно машину, не боясь, что адрес с интерфейса может пропасть. (ПО РФ 😀)

**Настройка HQ-SRV и HQ-CLI производится по заданию позже!**

1. **Настройте часовой пояс на всех устройствах, согласно месту проведения экзамена.**

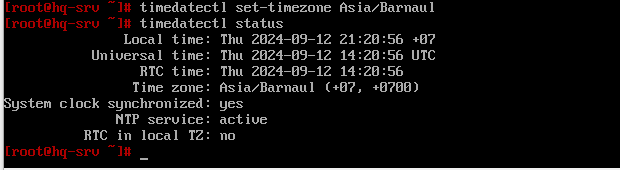
Настройка производится встроенной службой, настроим зону на **HQ-SRV** следующей командой:

**timedatectl set-timezone Asia/Barnaul**

****

И проверим правильность настройки:

**timedatectl status**



Аналогично на других устройствах

Настройка часового пояса завершена завершена.

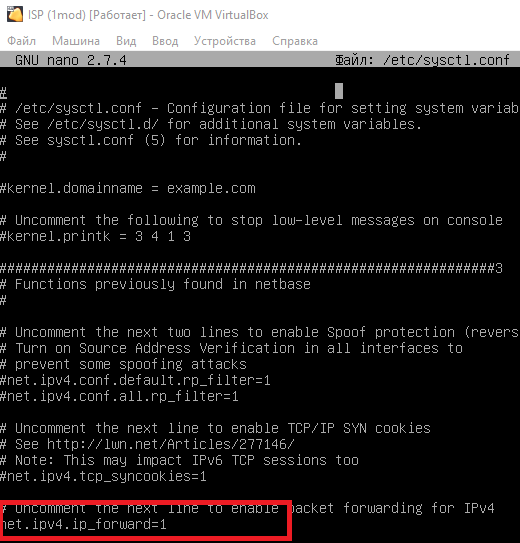
1. **Настройка forward пакетов:**

**ISP:**

**mcedit /etc/sysctl.conf**

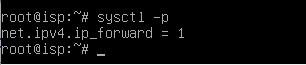
Убрать знак комментария на этой строке:

**net.ipv4.ip\_forward=1**



И применить изменения:

**sysctl –p**



АНАЛОГИЧНО НА ДРУГИХ РОУТЕРАХ!

1. **Настройка NAT:**

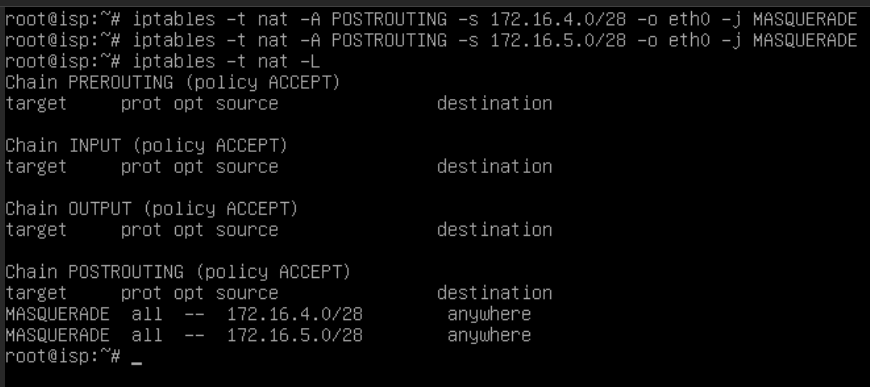
**ISP:**

Пишем в консоль следующие команды:

**iptables -t nat -A POSTROUTING –s 172.16.4.0/28 –o eth0 -j MASQUERADE** (Правило для доступа в интернет для устройств сети HQ)

**iptables -t nat -A POSTROUTING –s 172.16.5.0/28 –o eth0 -j MASQUERADE** (Правило для доступа в интернет для устройств сети BR)

**iptables -t nat -L** (Вывод прописанных правил для **nat**)



Сохраним наши правила, пишем в консоль следующую команду:

**iptables-save > /root/rules**

****

Запишем в **crontab** одну команду, чтобы при старте системы, правила загружались из файла, в котором они хранятся.

Ппишем в консоль следующие команды:

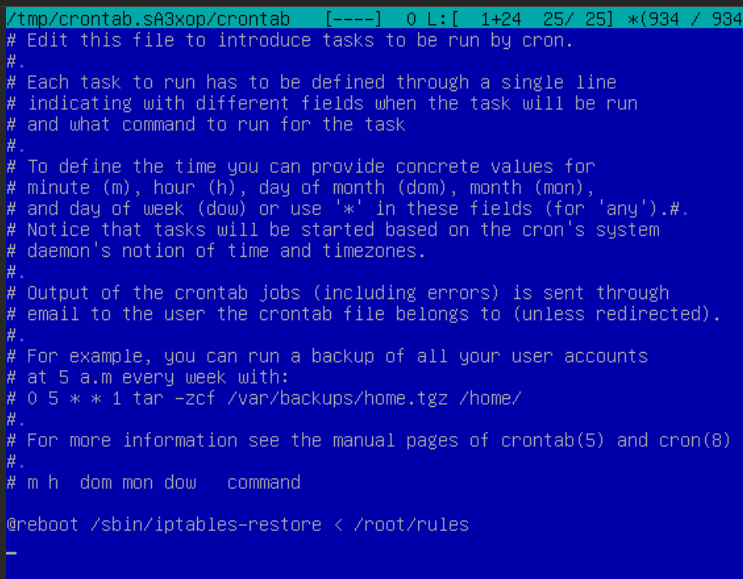
**export EDITOR=mcedit** (Команда одноразовая, для комфортной работы с crontab её нужно писать каждый раз)

**crontab -e**

Добавляем в конец файла следующие строки:

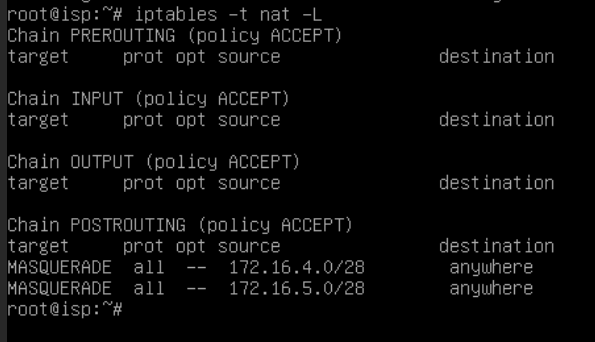
**@reboot /sbin/iptables-restore < /root/rules**

***!ВАЖНО!*** *Оставляем пустую строку после введённой строки выше, иначе не будет сохранения! В этом файле всегда нужно оставлять снизу* ***ПУСТУЮ СТРОКУ****!*

****

Перезагружаем машину и смотрим список правил, применяются ли они при запуске системы:

**iptables –t nat -L**



**HQ-RTR:**

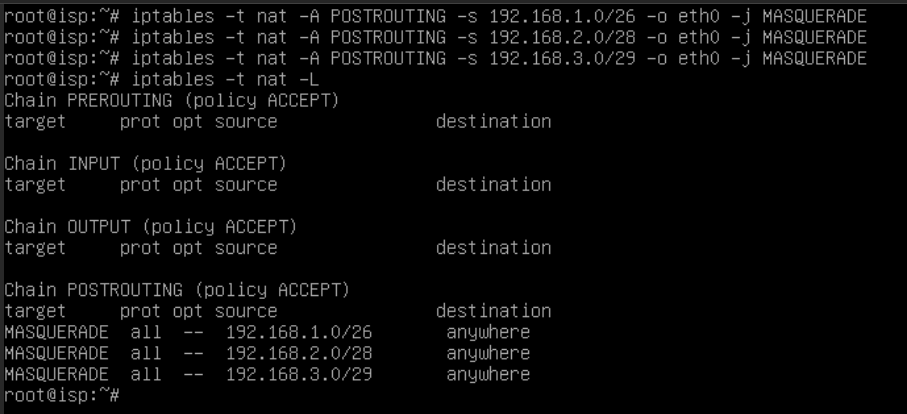
Пишем в консоль следующие команды:

**iptables -t nat -A POSTROUTING -s 192.168.1.0/26 -o eth0 -j MASQUERADE**

**iptables -t nat -A POSTROUTING -s 192.168.2.0/28 -o eth0 -j MASQUERADE**

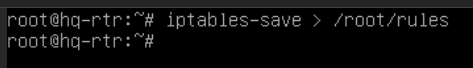
**iptables -t nat -A POSTROUTING -s 192.168.3.0/29 -o eth0 -j MASQUERADE**

**iptables -t nat -L**

****

Сохраним наши правила, пишем в консоль следующую команду:

**iptables-save > /root/rules**

****

Запишем в **crontab** одну команду, чтобы при старте системы, правила загружались из файла, в котором они хранятся.

Ппишем в консоль следующие команды:

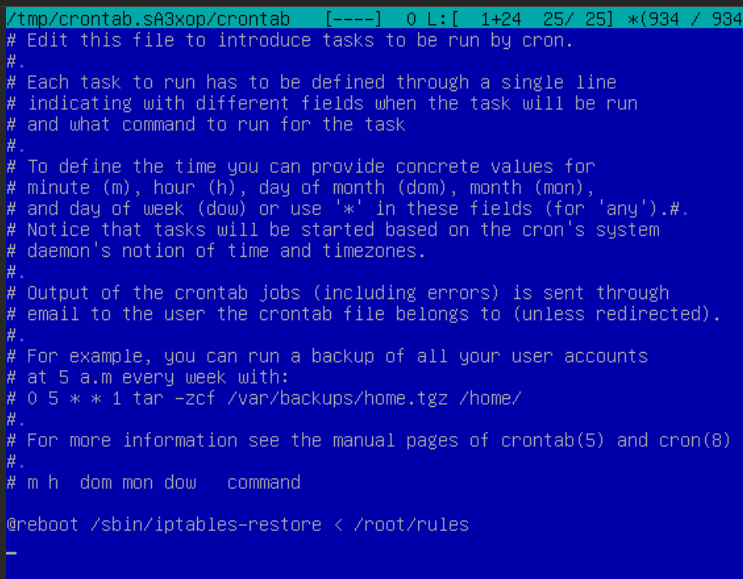
**export EDITOR=mcedit**

**crontab -e**

Добавляем в конец файла следующие строки:

**@reboot /sbin/iptables-restore < /root/rules**

***!ВАЖНО!*** *Оставляем пустую строку после введённой строки выше, иначе не будет сохранения! В этом файле всегда нужно оставлять снизу* ***ПУСТУЮ СТРОКУ****!*

****

Перезагружаем машину и смотрим список правил, применяются ли они при запуске системы:

**iptables –t nat -L**



**BR-RTR:**

Пишем в консоль следующие команды:

**iptables -t nat -A POSTROUTING -s 192.168.4.0/27 -o eth0 -j MASQUERADE**

**iptables -t nat -L**



Сохраним наши правила, пишем в консоль следующую команду:

**iptables-save > /root/rules**

****

Запишем в **crontab** одну команду, чтобы при старте системы, правила загружались из файла, в котором они хранятся.

Ппишем в консоль следующие команды:

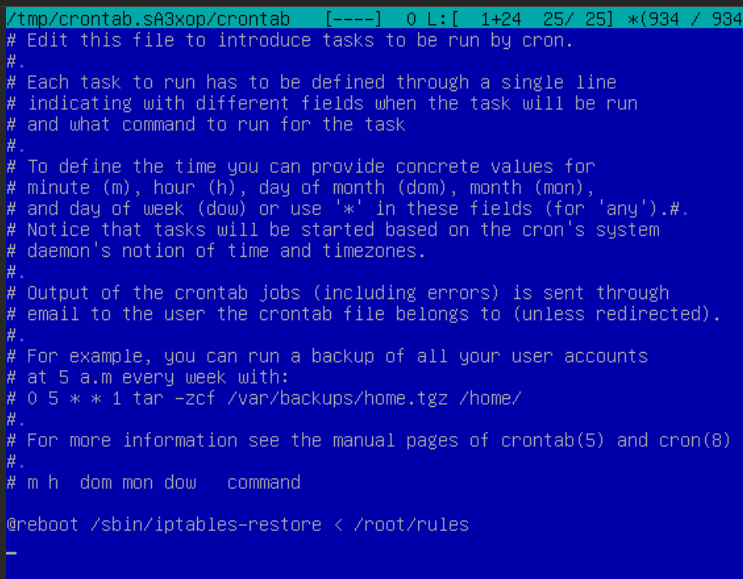
**export EDITOR=mcedit**

**crontab -e**

Добавляем в конец файла следующие строки:

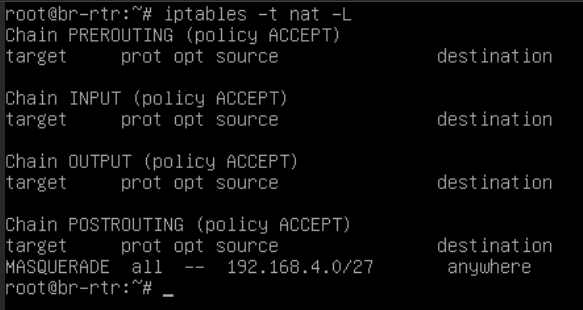
**@reboot /sbin/iptables-restore < /root/rules**

***!ВАЖНО!*** *Оставляем пустую строку после введённой строки выше, иначе не будет сохранения! В этом файле всегда нужно оставлять снизу* ***ПУСТУЮ СТРОКУ****!*

****

Перезагружаем машину и смотрим список правил, применяются ли они при запуске системы:

**iptables –t nat -L**

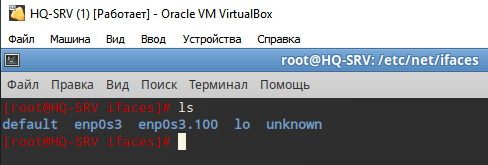


1. **Настройка VLAN для HQ-SRV и HQ-CLI:**

**HQ-SRV:**

Каталог **enp0s3** (у вас может быть своё название интерфейса, будьте внимательны) оставлять без изменений и перейти к настройке VLAN:

**mkdir** /etc/net/ifaces/enp0s3.100 (создание каталога под VLAN интерфейс)



Создадим файл options и откроем его командой

**mcedit /etc/net/ifaces/enp0s3.100/options**

Запишем в него следующее содержимое**:**

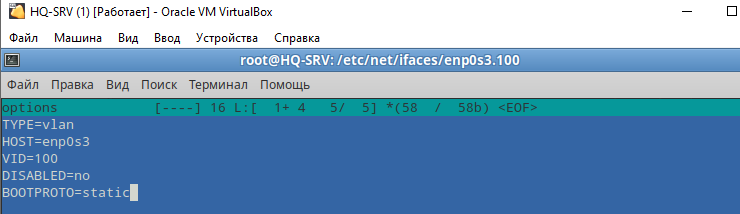
**TYPE=vlan**

**HOST= enp0s3** (основной интерфейс, но у вас может быть иное название)

**VID=100** (id VLAN’а)

**DISABLED=no**

**BOOTPROTO=static**

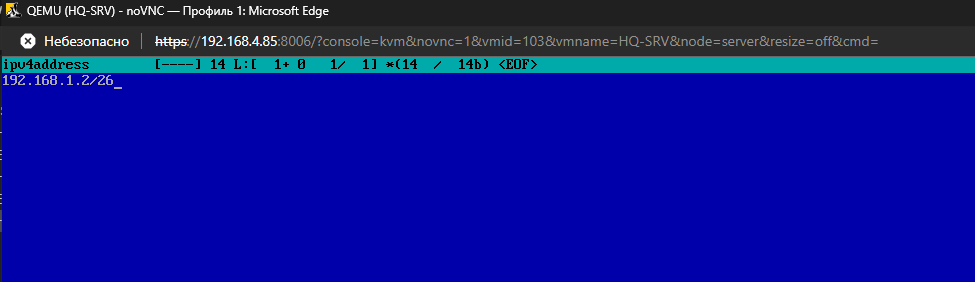


Создадим файлы **ipv4address** и **ipv4route** и откроем их командой:

**mcedit /etc/net/ifaces/enp0s3.100/ipv4address**

Записать туда следующую строку:

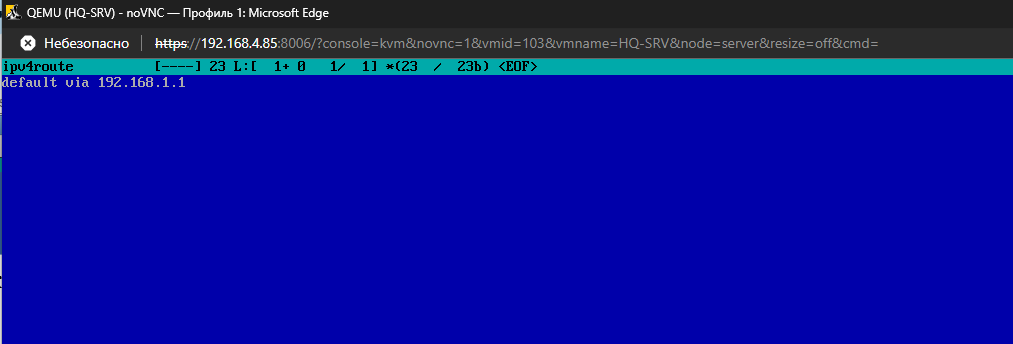
**192.168.1.2/26**



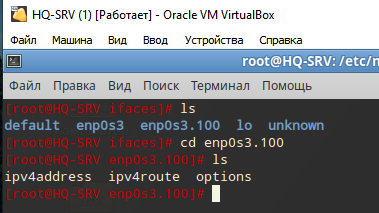
**mcedit /etc/net/ifaces/enp0s3.100/ipv4route**

Записать туда следующую строку:

**default via 192.168.1.1**



В итоге должен получится такой набор файлов в каталоге интерфейса:



Обязательно после всех настроек интерфейсов ввести:

**systemctl restart network**

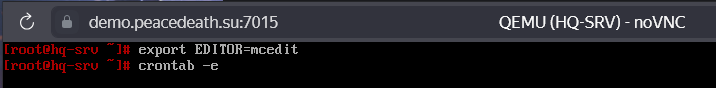
Также добавим запись о перезагрузке службы **network** в **crontab**.

Делаем это следующим образом, пишем команду:

**export EDITOR=mcedit**

А затем:

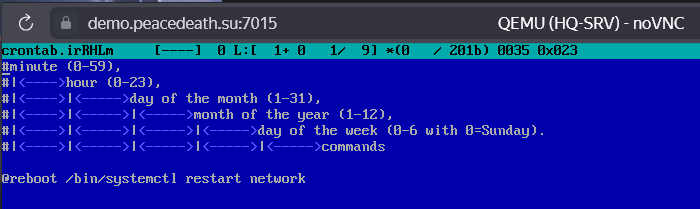
**crontab -e**

****

И в конце файла пишем следующее:

**@reboot /bin/systemctl restart network**

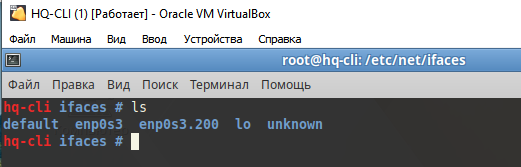
***!ВАЖНО!*** *Оставляем пустую строку после введённой строки выше, иначе не будет сохранения! В этом файле всегда нужно оставлять снизу* ***ПУСТУЮ СТРОКУ****!*

****

**HQ-CLI:**

Каталог enp0s3 оставлять без изменений и перейти к настройке VLAN:

**mkdir** /etc/net/ifaces/enp0s3.200 (создание каталога под VLAN интерфейс)



Создадим файл options и откроем его командой:

**mcedit /etc/net/ifaces/enp0s3.200/options**

Запишем в него следующее содержимое:

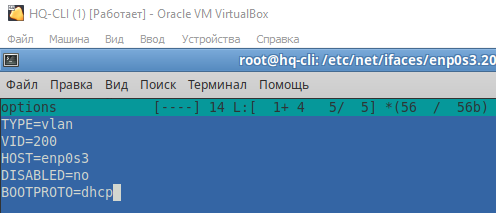
**TYPE=vlan**

**VID=200** (id VLAN’а)

**HOST= enp0s3** (основной интерфейс)

**DISABLED=no**

**BOOTPROTO=dhcp**



Создавать файлы **ipv4address** и **ipv4route** не нужно, т.к. мы получаем на **HQ-CLI** настройки по **DHCP**, который далее будет настроен на роутере.

Обязательно после всех настроек интерфейсов ввести:

**systemctl restart network** (Альт)

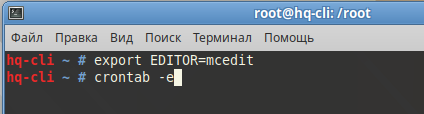
Также добавим запись о перезагрузке службы **network** в **crontab**.

Делаем это следующим образом, пишем команду:

**export EDITOR=mcedit**

А затем:

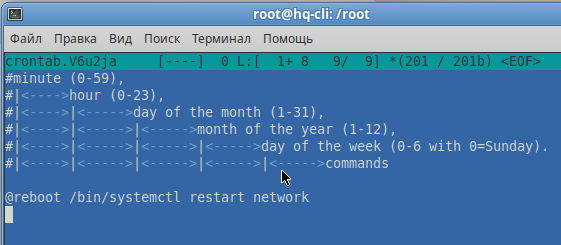
**crontab -e**

****

И в конце файла пишем следующее:

**@reboot /bin/systemctl restart network**

***!ВАЖНО!*** *Оставляем пустую строку после введённой строки выше, иначе не будет сохранения! В этом файле всегда нужно оставлять снизу* ***ПУСТУЮ СТРОКУ****!*

****

1. **Настройка IP-туннеля между офисами HQ и BR:**

Создание туннеля производится на маршрутизаторах **HQ-RTR** и **BR-RTR**.

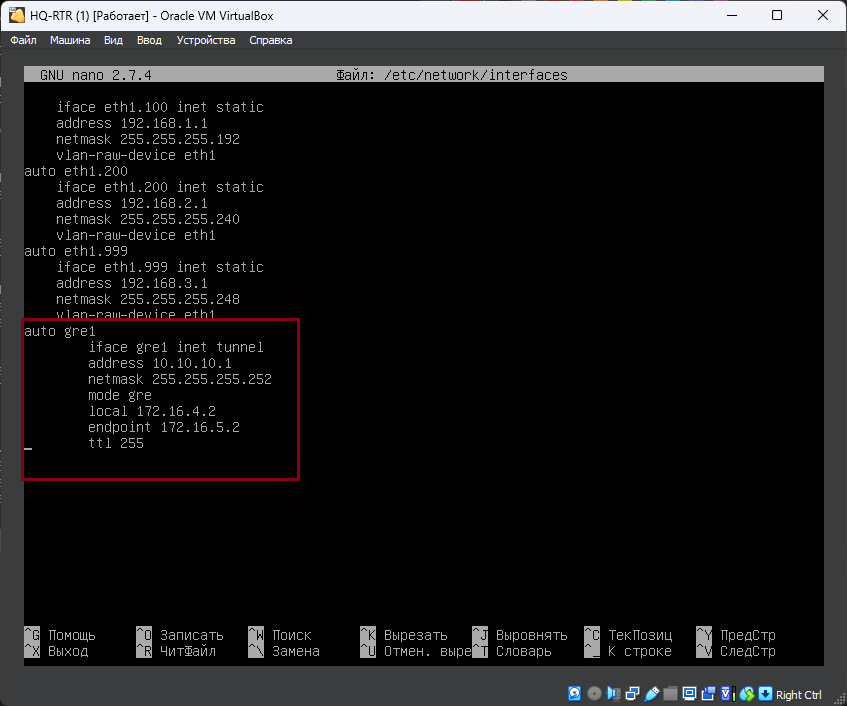
**HQ-RTR:**

Для создания туннеля необходимо добавить новый интерфейс в файл **/etc/network/interfaces**

Откроем этот файл текстовым редактором следующей командой:

**mcedit /etc/network/interfaces**

Добавляем в конец файла то, что выделено на скриншоте:



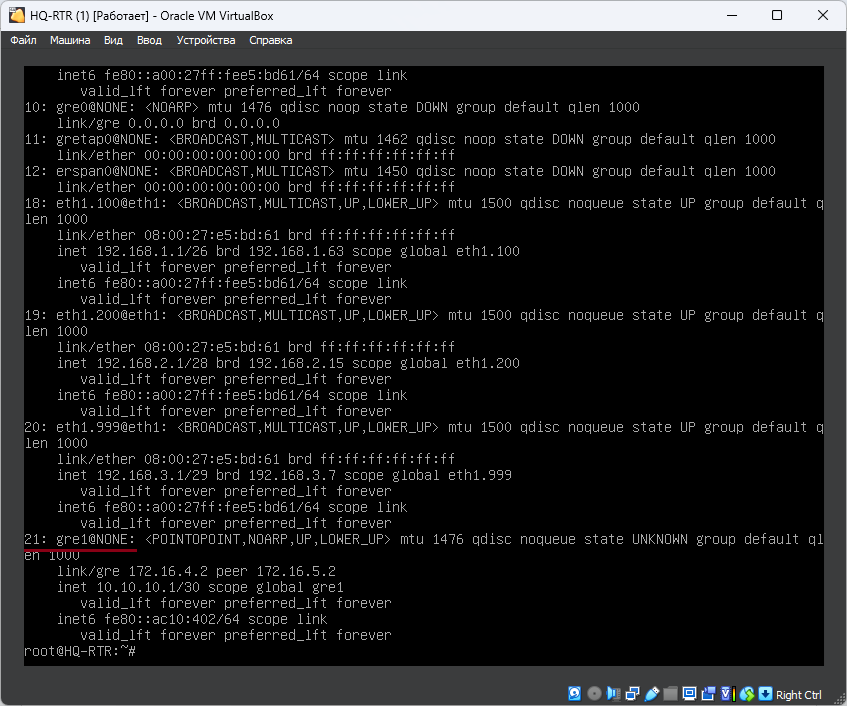
Сохраняем файл, выходим из редактора.

Перезапускаем службу networking для применения изменений:

**systemctl restart networking**

Проверяем наличие IP-туннеля:

**ip a**



Туннель появился.

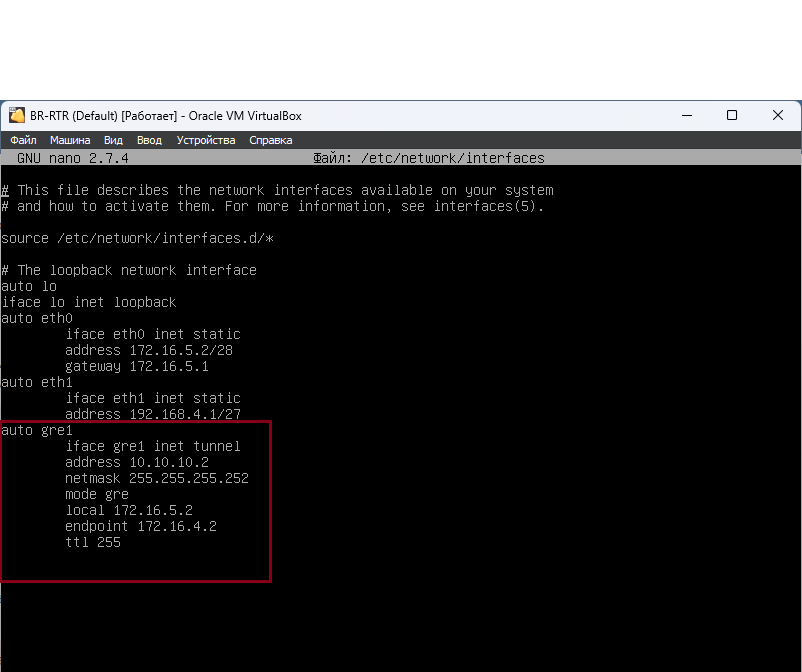
**BR-RTR:**

На этом роутере тоже самое, только нужно поменять IP-адрес туннеля и IP-адреса local и endpoint.

Открываем файл текстовым редактором **mcedit** следующей командой:

**mcedit /etc/network/interfaces**

Прописываем в конец файла следующее содержимое:



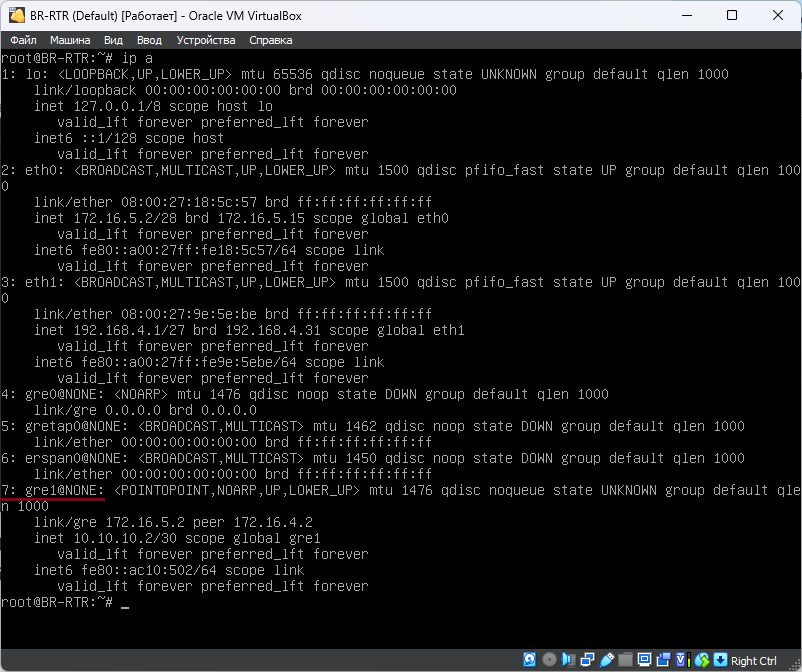
Сохраняем файл, выходим из редактора.

Также перезапускаем службу networking для применения изменений:

**systemctl restart networking**

Проверяем наличие IP-туннеля:

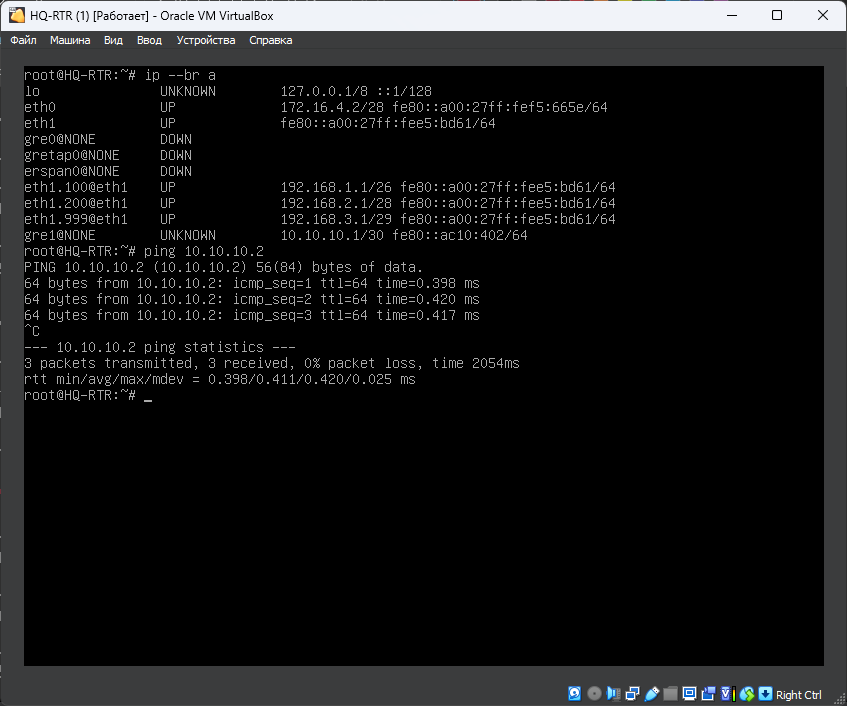
**ip a**



Туннель между офисами настроен, полностью проверить его работу можно после настройки **OSPF**. Но пинги между **10.10.10.1** и **10.10.10.2** уже должны доходить.

Отправим с **HQ-RTR** эхо-запрос до **BR-RTR** по туннелю:

**ping 10.10.10.2**



Работает, приступаем к следующему этапу для полной работы туннеля

1. **Настройка динамической маршрутизации с помощью link-state протокола OSPF.**

Для работы OSPF нам нужна служба frr, которой по умолчанию нет на наших маршрутизаторах HQ-RTR и BR-RTR, поэтому проделаем следующие шаги.

**HQ-RTR:**

Нужно закомментировать в /etc/apt/sources.list первую строку с репозиторием АСТРЫ, т.к. он не имеет пакета frr даже после обновлений репозиториев, вместо него мы будем использовать debian репозиторий.

Для начала зайдём туда следующей комадной:

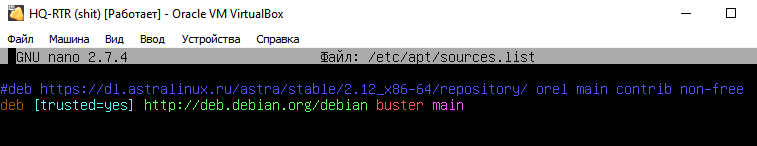
**mcedit /etc/apt/sources.list**

Комментируем первую строку знаком #:

**#deb** [**https://dl.astralinux.ru/astra/stables/2.12\_x86-64/repository/**](https://dl.astralinux.ru/astra/stables/2.12_x86-64/repository/) **orel main contrib non-free**

Ниже пишем следующую строку:

**deb [trusted=yes]** [**http://deb.debian.org/debian**](http://deb.debian.org/debian) **buster main**

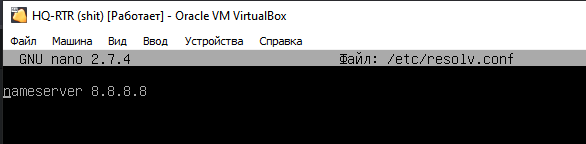


Ещё нам нужно добавить в /etc/resolv.conf сервер Google, иначе мы не сможем обновить репозитории, поэтому идём его редактировать следующей командой:

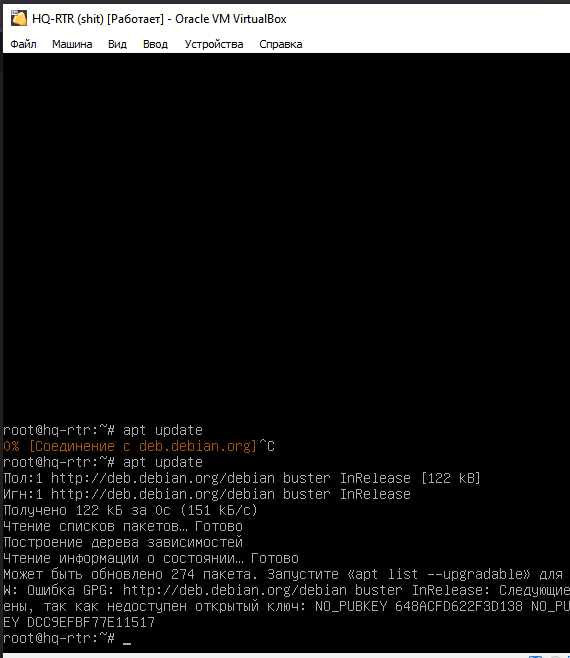
**mcedit /etc/resolv.conf**

И добавляем следующую строку в него:

**nameserver 8.8.8.8**

Сохраняем и идём теперь обновлять список пакетов:

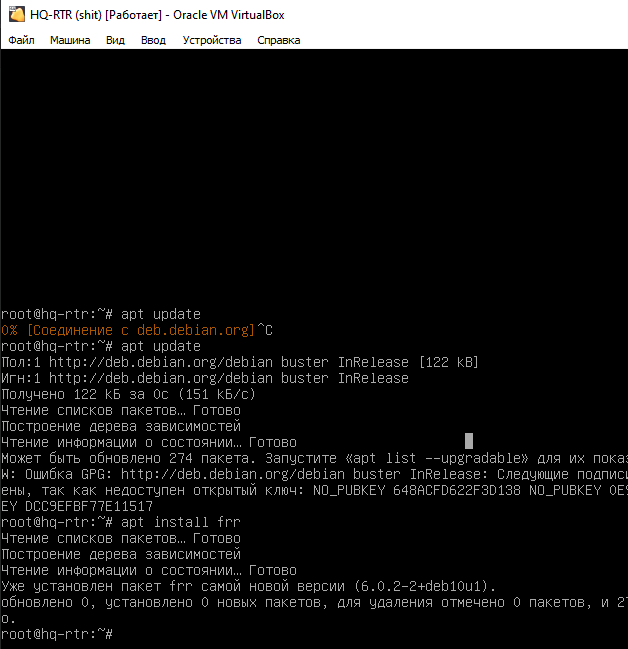
**apt update**



То, что он может ругаться на недоступный открытый ключ, это нормально, идём дальше!

Теперь качаем сам пакет frr:

**apt install frr**



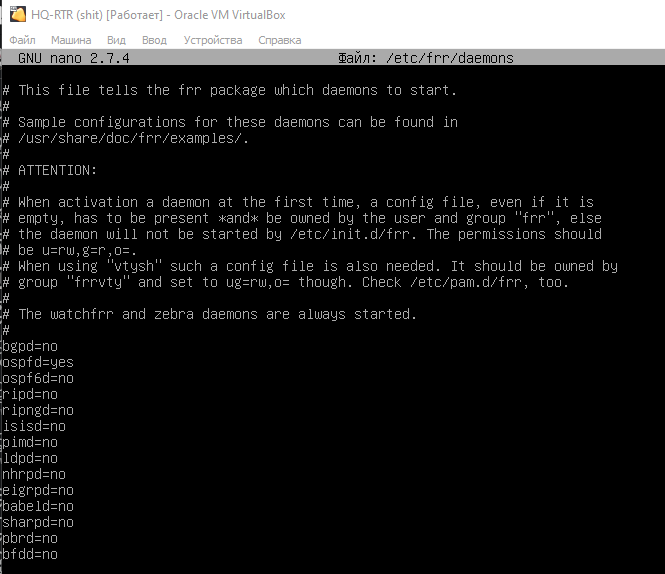
У нас он уже установлен, поэтому не обращаем внимание, скриншот нужен для того, чтобы вы поняли.

Затем нам нужно включить настройку ospf через конфигурационный файл /etc/frr/daemons:

**mcedit /etc/frr/daemons**

Находим в нём следующую строку и приводим её к такому виду:

**ospfd=yes**



А затем перезагрузим службу командой:

**systemctl restart frr**

А затем начнём настройку:

**vtysh** (зайти в режим настройки)

**conf t** (режим конфигурации, ВСПОМИНАЕМ ЦИСКО, РЕБЯТКИ!)

**router ospf**

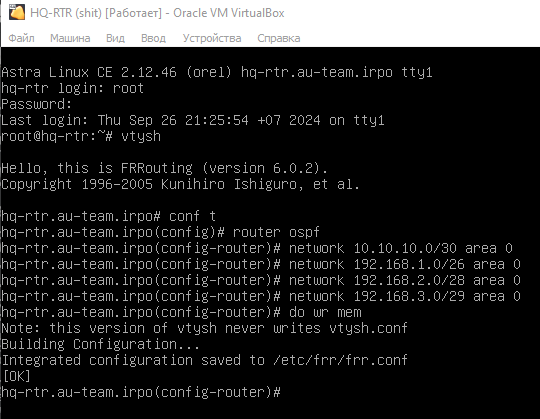
**network 10.10.10.0/30 area 0**

**network 192.168.1.0/26 area 0**

**network 192.168.2.0/28 area 0**

**network 192.168.3.0/29 area 0**

**do wr mem**



Теперь настроим парольную защиту на нашем GRE туннеле через frr:

**vtysh**

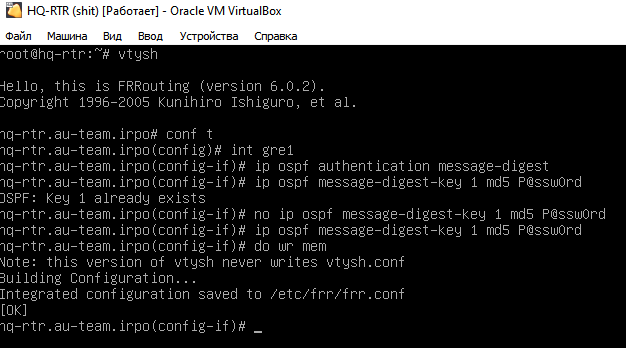
**conf t**

**int gre1**

**ip ospf authentication message-digest**

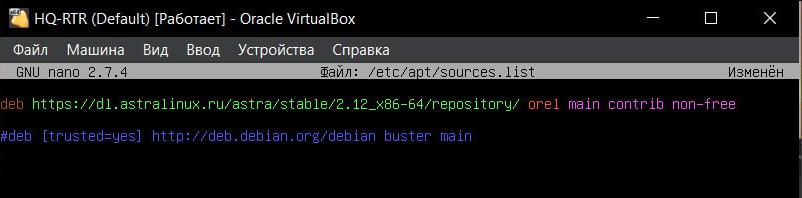
**ip ospf message-digest-key 1 md5 P@ssw0rd**

**do wr mem**



OSPF на HQ-RTR настроен.

ПОСЛЕ ПРОДЕЛАННОЙ РАБОТЫ, РАСКОММЕНТИРУЙТЕ РЕПОЗИТОРИЙ АСТРЫ И ЗАКОММЕНТИРУЙТЕ РЕПОЗИТОРИЙ DEBIAN!!! ВОТ ТАК:



**BR-RTR:**

Проделываем тоже самое с репозиториями.

Для начала зайдём туда следующей комадной:

**mcedit /etc/apt/sources.list**

Комментируем первую строку знаком #:

**#deb** [**https://dl.astralinux.ru/astra/stables/2.12\_x86-64/repository/**](https://dl.astralinux.ru/astra/stables/2.12_x86-64/repository/) **orel main contrib non-free**

Ниже пишем следующую строку:

**deb [trusted=yes]** [**http://deb.debian.org/debian**](http://deb.debian.org/debian) **buster main**

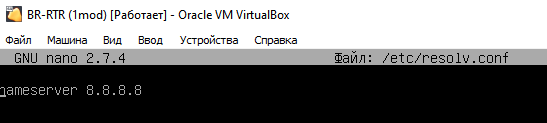


Теперь нам нужно добавить в /etc/resolv.conf сервер Google:

**mcedit /etc/resolv.conf**

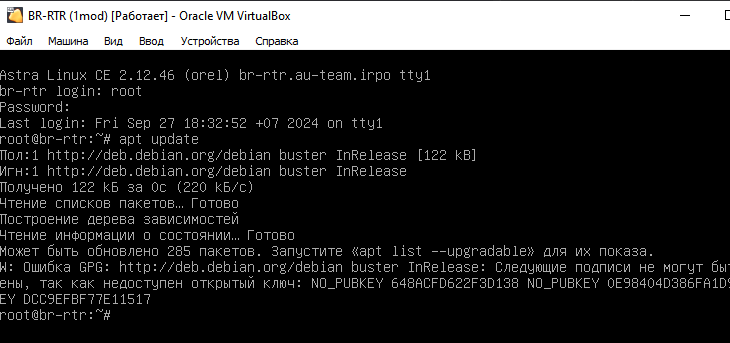
И добавляем следующую строку в него:

**nameserver 8.8.8.8**



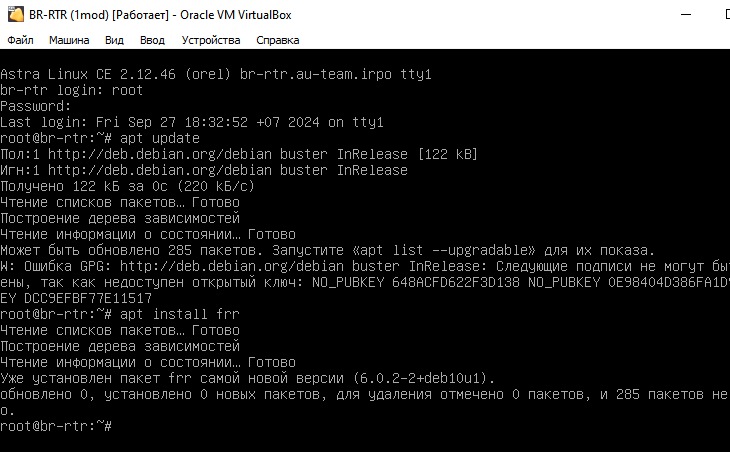
Сохраняем и идём теперь обновлять список пакетов:

**apt update**



Теперь качаем сам пакет frr:

**apt install frr**

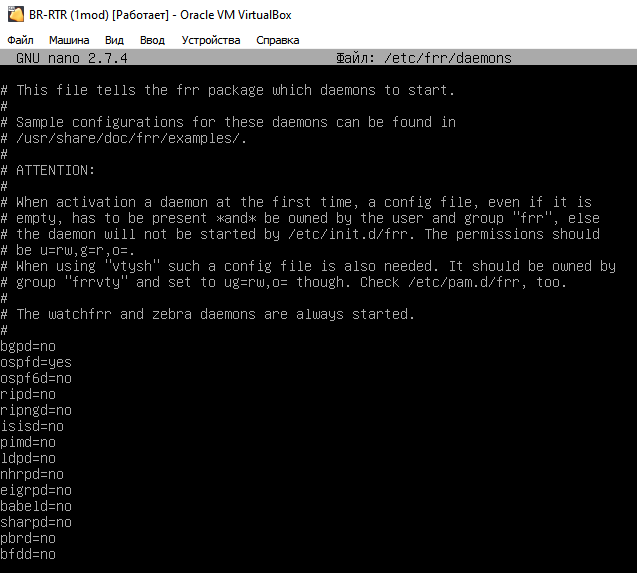


Затем нам нужно включить настройку ospf через конфигурационный файл **/etc/frr/daemons**:

**mcedit /etc/frr/daemons**

Находим в нём следующую строку и приводим её к такому виду:

**ospfd=yes**



А затем перезагрузим службу командой:

**systemctl restart frr**

А затем начнём настройку:

**vtysh**

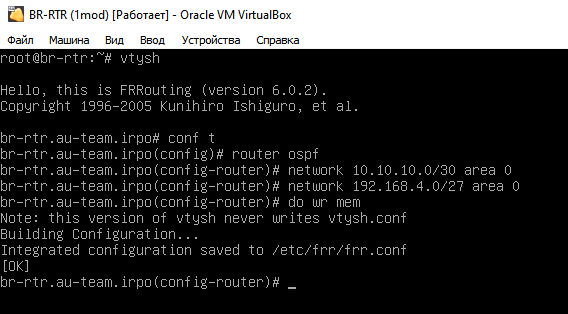
**conf t**

**router ospf**

**network 10.10.10.0/30 area 0**

**network 192.168.4.0/27 area 0**

**do wr mem**



Теперь настроим парольную защиту на нашем GRE туннеле через **frr** на второй стороне тоже:

**vtysh**

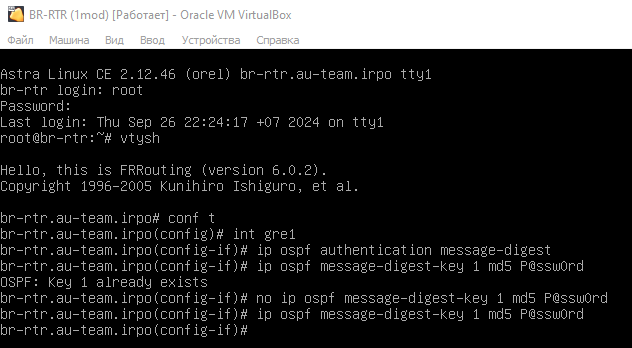
**conf t**

**int gre1**

**ip ospf authentication message-digest**

**ip ospf message-digest-key 1 md5 P@ssw0rd**

**do wr mem**



**OSPF** на **BR-RTR** настроен.

Также нужно вернуть репозиторий астры обратно, смотрите выше, как мы это делали, но в обратном порядке выполняя шаги.

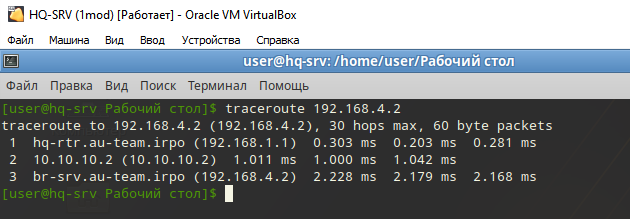
**OSPF** полностью настроен, теперь пинг должен идти везде и по туннелям, проверим это.

**ИНОГДА НУЖНО ЧУТЬ ПОДОЖДАТЬ, ПОКА ПОЯВИТСЯ СОСЕД, ПОЭТОМУ ПИНГ И ТРАССИРОВКА МОГУТ СРАЗУ НЕ ПОЙТИ, ПРОВЕРЯЙТЕ СОСЕДЕЙ ЧЕРЕЗ VTYSH С ПОМОЩЬЮ КОМАНДЫ:**

**do show ip ospf neighbor**

Сделаем трассировку от сервера **HQ-SRV** до **BR-SRV**:

**traceroute 192.168.4.2**



Всё отлично проходит **через** наш **туннель**, поздравляю!

(Доменные имена будут показываться после настройки **DNS**, просто мы это проделали для себя ранее, а вы следуйте пунктам дальше!)

1. **Настройка протокола динамической конфигурации хостов (DHCP):**

Настройка будет производиться на **HQ-RTR**!

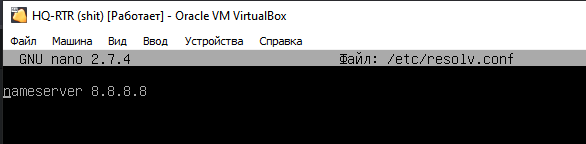
Использовать в качестве **DHCP** мы будем **dnsmasq**, служба, которой по умолчанию нет в наших ОС Российского производства.

Ещё нам нужно добавить в **resolv.conf** сервер **Google**, иначе мы не сможем обновить репозитории, поэтому идём его редактировать следующей командой:

**mcedit /etc/resolv.conf**

И добавляем следующую строку в него:

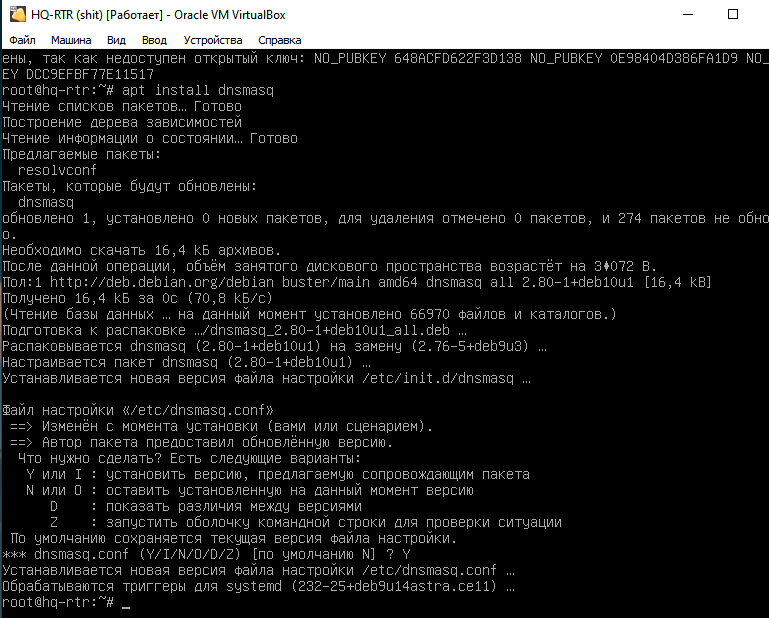
**nameserver 8.8.8.8**



Обновим пакеты и установим её командами:

**apt update**

**apt install dnsmasq**



Затем зайдем в настройки конфигурационного файла командой:

**mcedit /etc/dnsmasq.conf**

И внесем в него следующие строки (можно прямо в начало файла):

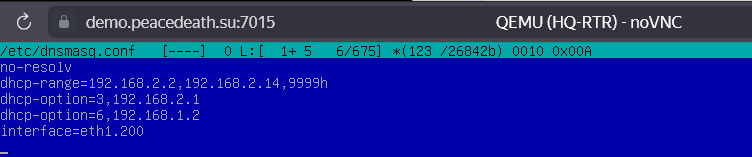
**no-resolv**

**dhcp-range=192.168.2.2,192.168.2.14,9999h**

**dhcp-option=3,192.168.2.1**

**dhcp-option=6,192.168.1.2**

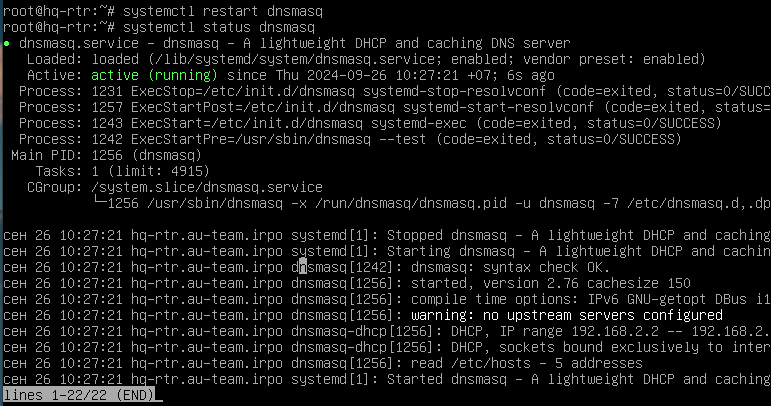
**interface=eth1.200**



Затем перезапускаем службу и посмотрим её статус:

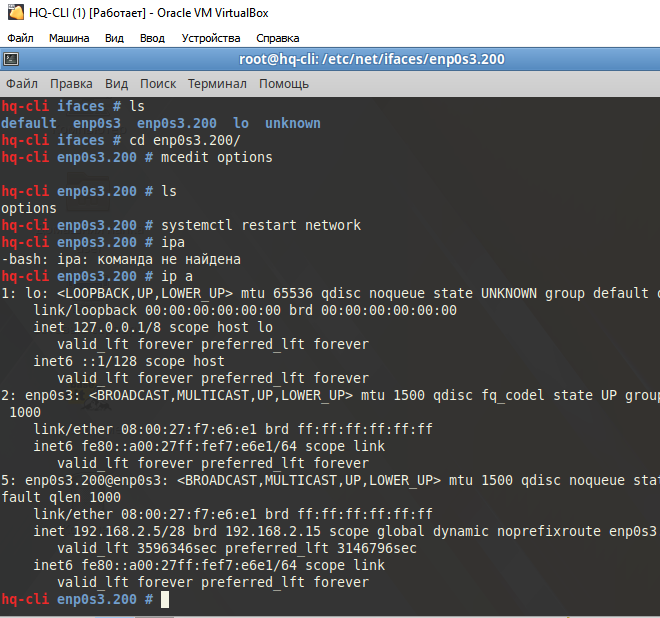
**systemctl restart dnsmasq**

**systemctl status dnsmasq**



Проверим работу службы на **HQ-CLI**, перезапускаем службу **network** на нём и посмотрим, выдался ли нам адрес:  
**systemctl restart network**

**ip a**



**enp0s3.200** на **HQ-CLI** успешно получил адрес из диапазона.

1. **Настройка DNS для офисов HQ и BR:**

Для начала необходимо отключить несовместимую службу bind если она есть, командой

**systemctl disable --now bind**

Для работы **DNS** есть служба **dnsmasq** (она же и для **DHCP**)

Установим её на наш сервер **HQ-SRV** (если есть, как у нас, то переходите к следующему шагу).

Ещё нам нужно добавить в **resolv.conf** сервер **Google**, иначе мы не сможем обновить репозитории, поэтому идём его редактировать следующей командой:

**mcedit /etc/resolv.conf**

И добавляем следующую строку в него:

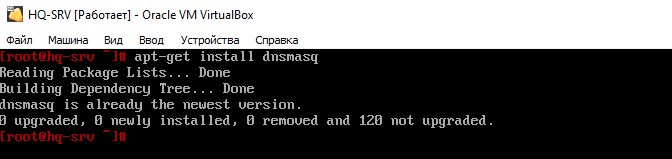
**nameserver 8.8.8.8**

Обновим пакеты и установим её командами:

**apt-get update**

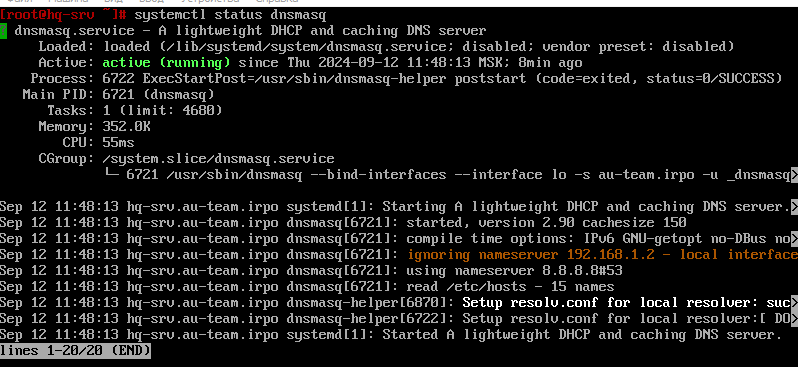
**apt-get install dnsmasq** (Установка пакета dnsmasq)

**systemctl enable --now dnsmasq** (Добавление службы в автозапуск)



Проверим её состояние перед работой:

**systemctl status dnsmasq**



Затем откроем файл для редактирования конфигурации нашего DNS-сервера:

**mcedit /etc/dnsmasq.conf**

И добавляем в неё строки (для удобства прям с первой строки файла):

**no-resolv** (не будет использовать /etc/resolv.conf)

**domain=au-team.irpo**

**server=8.8.8.8** (адрес общедоступного DNS-сервера)

**interface=\*** (на каком интерфейсе будет работать служба)

**address=/hq-rtr.au-team.irpo/192.168.1.1**

**ptr-record=1.1.168.192.in-addr.arpa,hq-rtr.au-team.irpo**

**cname=moodle.au-team.irpo,hq-rtr.au-team.irpo**

**cname=wiki.au-team.irpo,hq-rtr.au-team.irpo**

**address=/br-rtr.au-team.irpo/192.168.4.1**

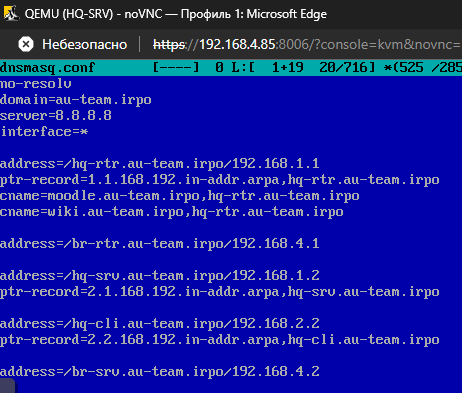
**address=/hq-srv.au-team.irpo/192.168.1.2**

**ptr-record=2.1.168.192.in-addr.arpa,hq-srv.au-team.irpo**

**address=/hq-cli.au-team.irpo/192.168.2.2 (**Смотрите адрес на **HQ-CLI**, т.к он выдаётся по DHCP**)**

**ptr-record=2.2.168.192.in-addr.arpa,hq-cli.au-team.irpo**

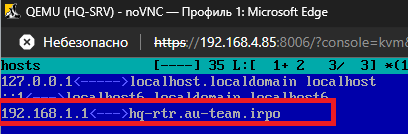
**address=/br-srv.au-team.irpo/192.168.4.2**

****

Сохраняем файл нажатием кнопки **F2**, а затем выход с помощью **F10.**

Теперь необходимо добавить строку **192.168.1.1 hq-rtr.au-team.irpo** в файл /etc/hosts:

**mcedit /etc/hosts**



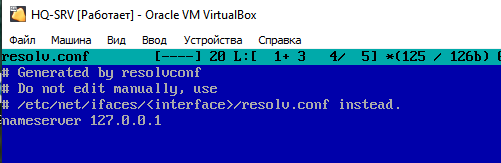
Сохраняем файл, выходим из редактора.

И также нужно изменить файл **resolv.conf**:

**mcedit /etc/resolv.conf**

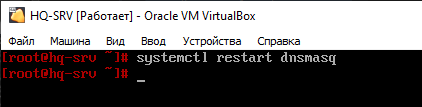
Теперь там должен находиться следующий адрес:

**127.0.0.1**

****

Перезапускаем службу командой:

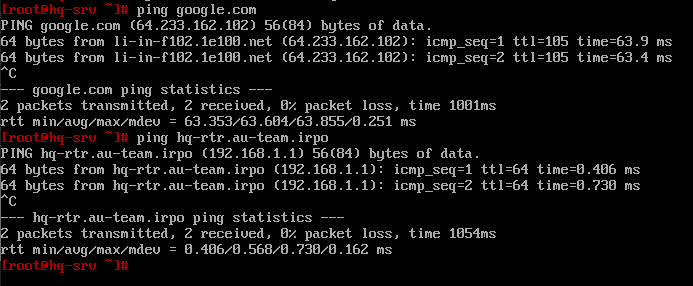
**systemctl restart dnsmasq**

****

Проверим пинг сначала с HQ-SRV на google.com и hq-rtr.au-team.irpo:

**ping google.com**

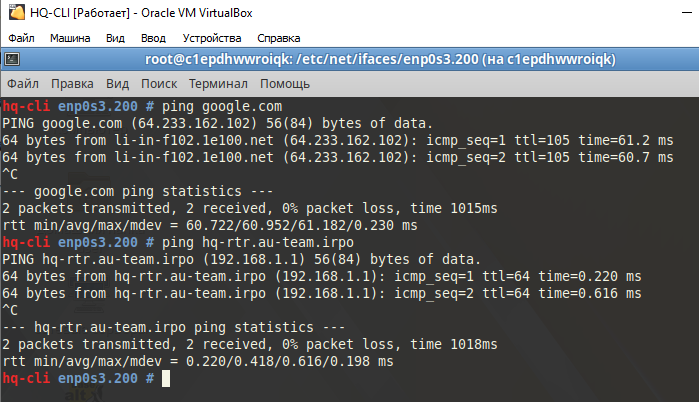
**ping hq-rtr.au-team.irpo**

****

Теперь проверим пинг с HQ-CLI:

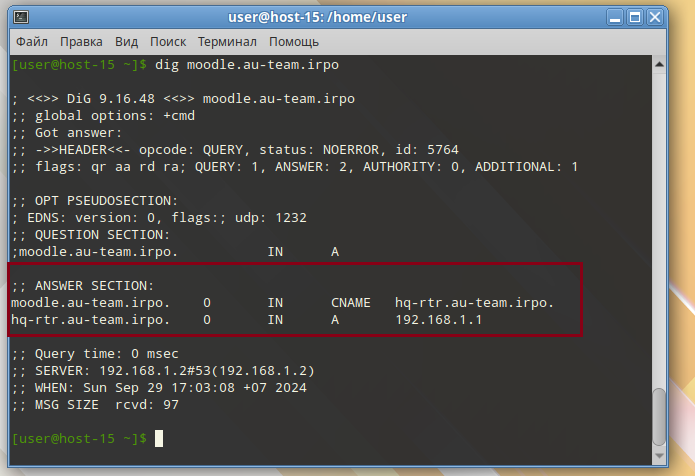
**ping google.com**

**ping hq-rtr.au-team.irpo**

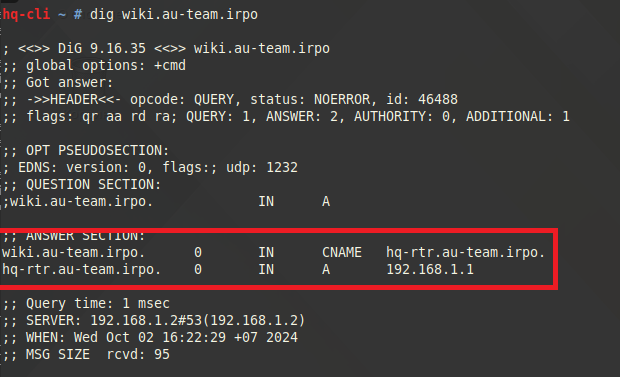


И проверим работу CNAME записей с HQ-CLI:

**dig moodle.au-team.irpo**



**dig wiki.au-team.irpo**



Наш DNS-сервер настроен.

1. **Создание локальных учетных записей:**

Создание на **HQ-SRV:**

Для создания пользователя с определённым идентификатором на машине под управлением ОС Alt Linux нужно использовать команду:

**useradd sshuser -u 1010**



Для проверки можно использовать команду:

**id sshuser**

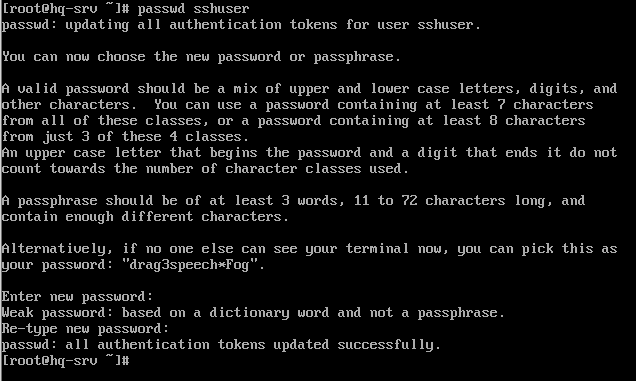


Чтобы задать пользователю новый пароль нужно использовать команду:

**passwd sshuser**

После чего ввести и подтвердить новый пароль:

**P@ssw0rd**

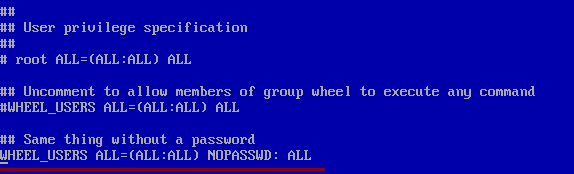


Чтобы **sshuser** мог запускать **sudo без дополнительной аутентификации**, необходимо убрать комментарий с двух строчек в файле **/etc/sudoers**, откроем его командой:

**mcedit /etc/sudoers**

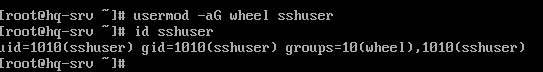
И уберём комментарий на следующей строке:

**WHEEL\_USERS ALL=(ALL:ALL) NOPASSWD: ALL**



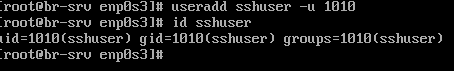
После чего добавить пользователя sshuser в группу wheel:

**usermod -aG wheel sshuser**

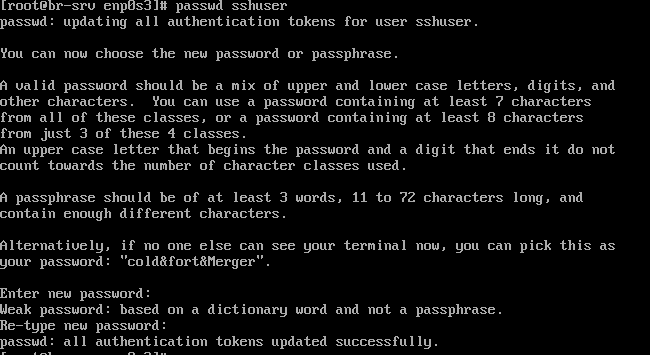


Создание на **BR-SRV:**

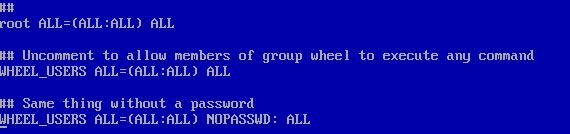
Создание пользователя:

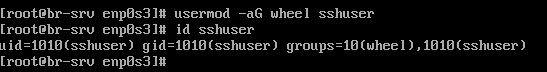


Редактирование пароля:



Повышение прав:

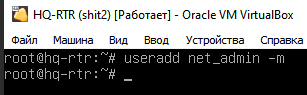




**Создание на HQ-RTR:**

Для создания пользователя на машине под управлением ОС Astra Linux нужно использовать команду:

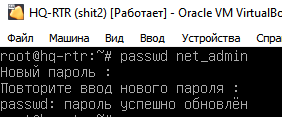
**useradd** **net\_admin -m**



Изменим пароль:

**passwd net\_admin**

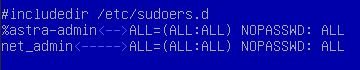
**P@$$word**

****

Чтобы **net\_admin** мог запускать **sudo** без дополнительной аутентификации необходимо добавить следующую строчку в файл **/etc/sudoers**, в самый конец:

**mcedit /etc/sudoers**

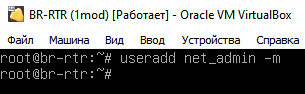
**net\_admin ALL=(ALL:ALL) NOPASSWD: ALL**



Создание на **BR-RTR:**

Создание пользователя:

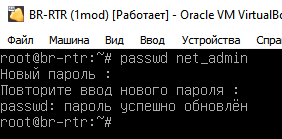
**useradd net\_admin -m**



Изменение пароля:

**passwd net\_admin**

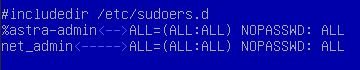
**P@$$word**



Повышение прав:

**mcedit /etc/sudoers**

**net\_admin ALL=(ALL:ALL) NOPASSWD: ALL**

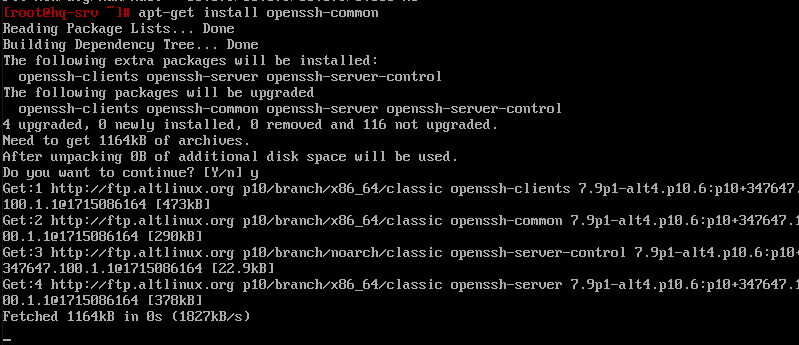


1. **Настройка безопасного удаленного доступа на серверах HQ-SRV и BR-SRV (SSH):**

**Настройка на HQ-SRV** **:**

Для работы SSH нам понадобится служба **openssh-common**, которой изначально нет, поэтому установим её:

**apt-get install openssh-common**



Затем зайдём в файл конфигурации для внесения изменений:

**mcedit /etc/openssh/sshd\_config**

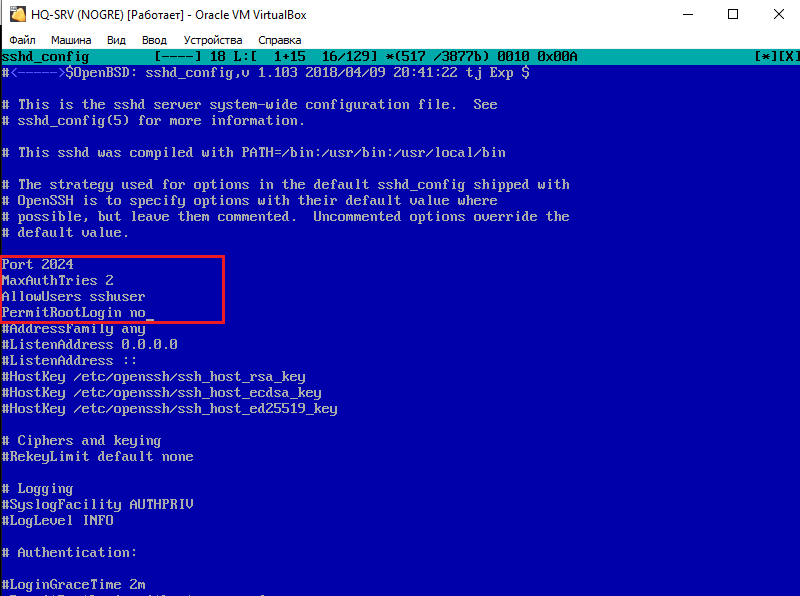
И внесём туда следующие строки:

**Port 2024**

**MaxAuthTries 2**

**AllowUsers sshuser**

**PermitRootLogin no**

****

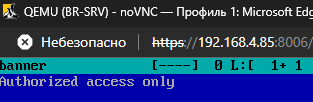
Далее нам нужен баннер.

Создаём его, вносим предложение, которое требуется по заданию через команду:

**mcedit /root/banner**

Пишем туда следующую строку (ОБЯЗАТЕЛЬНО ПОСЛЕ НЕЁ НАЖАТЬ ENTER, чтобы под ней была пустая строка):

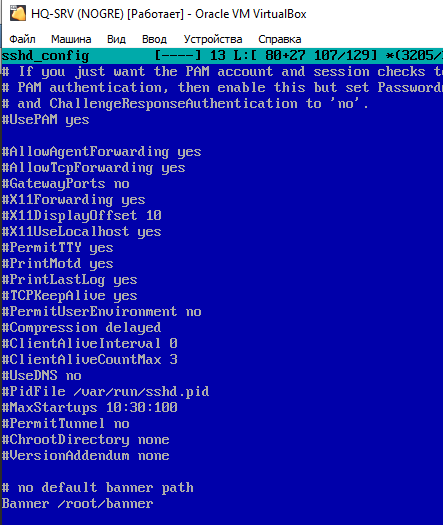
**Authorized access only**

****

Затем сохраняем и возвращаемся в **/etc/openssh/sshd\_config**.

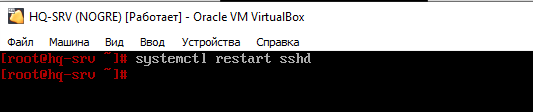
Добавляем/Редактируем следующую строку:

**Banner /root/banner**



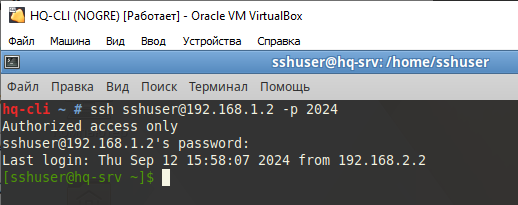
После внесения изменений, сохраняем и выходим. И делаем перезапуск службы:  
**systemctl enable --now sshd**

**systemctl restart sshd**



Затем попробуем подключиться по SSH через HQ-CLI:

**ssh sshuser@192.168.1.2 -p 2024**



**sshuser** – пользователь, под которым вы подключаетесь

**192.168.1.2** – адрес сервера, к которому мы подключаемся (**HQ-SRV**)

**-p 2024** – порт, по которому мы подключаемся (мы заменили со стандартного 22 на 2024)

**Проделываем все тоже самое и на сервере** **BR-SRV**.

Сервис безопасного удаленного доступа настроен.