Настройка маршрутов пользователя

1. Нам необходимо определить какой адрес будет у пользователя при подключении и куда ему разрешается ходить.

2. Для начала найдем свободный ip-адрес

```
grep -rn 10.8.0. /etc/openvpn/ccd
```

Команда grep пройдет по всем файлам в директории и выведет в консоль все совпадения с **10.8.0.** – адресация openvpn

3. Выбираем любой адрес, которого нет в списке от 2 до 254

Проверить что вы не просмотрели свободный адрес, можно той же командой grep, вбив адрес целиком. К примеру:

```
grep -rn 10.8.0.38 /etc/openvpn/ccd  - найдет в конфиге такой адрес и выведет его в консоль
grep -rn 10.8.0.119 /etc/openvpn/ccd – такого адреса нет, и в консоль ничего не выведет
```

```
[root@petr easy-rsa]# grep -rn 10.8.0.38 /etc/openvpn/ccd
/etc/openvpn/ccd/gerasimov-d:1:ifconfig-push 10.8.0.38 255.255.255.0
[root@petr easy-rsa]# grep -rn 10.8.0.111 /etc/openvpn/ccd
[root@petr easy-rsa]#
```

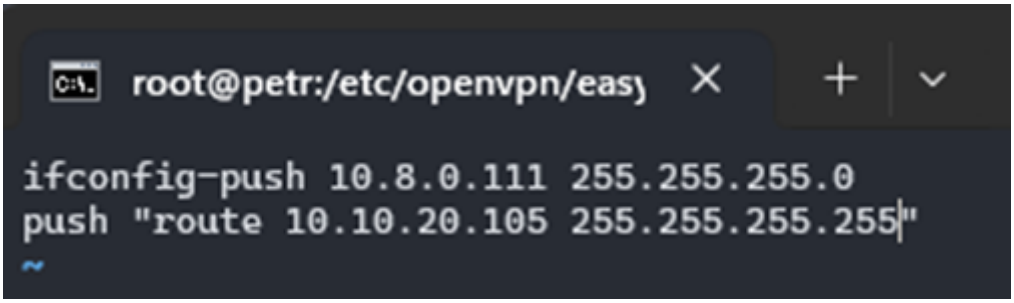
4. Создаем конфигурационный файл пользователя

```
touch /etc/openvpn/ccd/fursov.m
```

5. Прописываем в нем конфигурацию

```
nano /etc/openvpn/ccd/fursov.m
```

> **ifconfig-push 10.8.0.111 255.255.255.0**
>
> **push "route 10.10.20.105 255.255.255.255"**



Ifconfig-push – какой адрес и какую маску присваивать клиенту

push route – какой маршрут ему давать (до какого узла внутри сети можно будет ходить). В данном случае задан один адрес, но при желании можно указать несколько.

push "route 10.10.20.105 255.255.255.255"

push "route 10.10.20.106 255.255.255.255"

push "route 10.10.20.107 255.255.255.255"

и т.д.

6. Сохраняем данный конфиг и оставляем в этой директории.

Настройка фаервола

6.1. На данном шаге у нас уже имеются сертификаты пользователя, его конфиг и настройка. Осталось добавить разрешающие правила в фаервол, чтобы сервер пропустил клиента к его рабочему месту.

6.2. В качестве фаервола на сервере настроен iptables

6.3. Чтобы посмотреть текущие настройки iptables, можно ввести команду

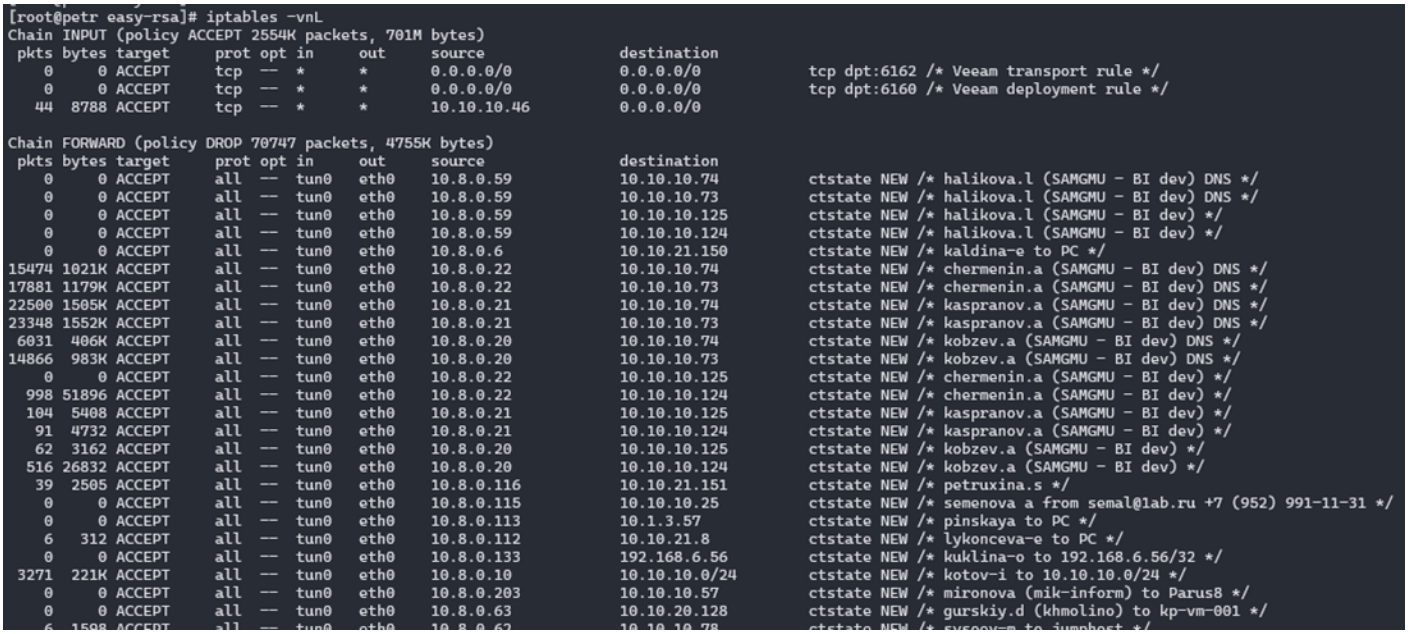
```
iptables -vnL
```

где ключи

>

> **v** – verbose т.е. подробный вывод
>
> **n** – numeric, вывод адресов в цифровом формате. По умолчанию он пытается вместо адресов прописать имена хостов
>
> **L** – list. Собственно, вывести все содержимое

В ответ мы получим достаточно большую табличку с правилами.



```
[root@petr easy-rsa]# iptables -vnL
Chain INPUT (policy ACCEPT 2554K packets, 701M bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           tcp dpt:6162 /* Veeam transport rule */
    0     0 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           tcp dpt:6160 /* Veeam deployment rule */
   44  8788 ACCEPT     tcp  --  *      *       10.10.10.46          0.0.0.0/0

Chain FORWARD (policy DROP 70747 packets, 4755K bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.59            10.10.10.74         ctstate NEW /* halikova.l (SAMGMU - BI dev) DNS */
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.59            10.10.10.73         ctstate NEW /* halikova.l (SAMGMU - BI dev) DNS */
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.59            10.10.10.125        ctstate NEW /* halikova.l (SAMGMU - BI dev) */
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.59            10.10.10.124        ctstate NEW /* halikova.l (SAMGMU - BI dev) */
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.6             10.10.21.150        ctstate NEW /* kaldina-e to PC */
15474 1021K ACCEPT     all  --  tun0   eth0    10.8.0.22            10.10.10.74         ctstate NEW /* chermenin.a (SAMGMU - BI dev) DNS */
17881 1179K ACCEPT     all  --  tun0   eth0    10.8.0.22            10.10.10.73         ctstate NEW /* chermenin.a (SAMGMU - BI dev) DNS */
22500 1505K ACCEPT     all  --  tun0   eth0    10.8.0.21            10.10.10.74         ctstate NEW /* kaspranov.a (SAMGMU - BI dev) DNS */
23348 1552K ACCEPT     all  --  tun0   eth0    10.8.0.21            10.10.10.73         ctstate NEW /* kaspranov.a (SAMGMU - BI dev) DNS */
 6031  406K ACCEPT     all  --  tun0   eth0    10.8.0.20            10.10.10.74         ctstate NEW /* kobzev.a (SAMGMU - BI dev) DNS */
14866  983K ACCEPT     all  --  tun0   eth0    10.8.0.20            10.10.10.73         ctstate NEW /* kobzev.a (SAMGMU - BI dev) DNS */
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.22            10.10.10.125        ctstate NEW /* chermenin.a (SAMGMU - BI dev) */
  998 51896 ACCEPT     all  --  tun0   eth0    10.8.0.22            10.10.10.124        ctstate NEW /* chermenin.a (SAMGMU - BI dev) */
  104  5408 ACCEPT     all  --  tun0   eth0    10.8.0.21            10.10.10.125        ctstate NEW /* kaspranov.a (SAMGMU - BI dev) */
   91  4732 ACCEPT     all  --  tun0   eth0    10.8.0.21            10.10.10.124        ctstate NEW /* kaspranov.a (SAMGMU - BI dev) */
   62  3162 ACCEPT     all  --  tun0   eth0    10.8.0.20            10.10.10.125        ctstate NEW /* kobzev.a (SAMGMU - BI dev) */
  516 26832 ACCEPT     all  --  tun0   eth0    10.8.0.20            10.10.10.124        ctstate NEW /* kobzev.a (SAMGMU - BI dev) */
   39  2505 ACCEPT     all  --  tun0   eth0    10.8.0.116           10.10.21.151        ctstate NEW /* petruxina.s */
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.115           10.10.10.25         ctstate NEW /* semenova a from semal@1ab.ru +7 (952) 991-11-31 */
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.113           10.1.3.57           ctstate NEW /* pinskaya to PC */
    6   312 ACCEPT     all  --  tun0   eth0    10.8.0.112           10.10.21.8          ctstate NEW /* lykonceva-e to PC */
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.133           192.168.6.56        ctstate NEW /* kuklina-o to 192.168.6.56/32 */
 3271  221K ACCEPT     all  --  tun0   eth0    10.8.0.10            10.10.10.0/24       ctstate NEW /* kotov-i to 10.10.10.0/24 */
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.203           10.10.10.57         ctstate NEW /* mironova (mik-inform) to Parus8 */
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.63            10.10.20.128        ctstate NEW /* gurskiy.d (khmolino) to kp-vm-001 */
    6  1598 ACCEPT     all  --  tun0   eth0    10.8.0.62            10.10.10.78         ctstate NEW /* fursov.m to jumphost */
```

Основными полями являются **source** (откуда) **destination** (куда) ну и комментарий, чтобы каждый раз не искать по конфигам, чей это адрес.

4. Для того чтобы добавить новое правило, пишем

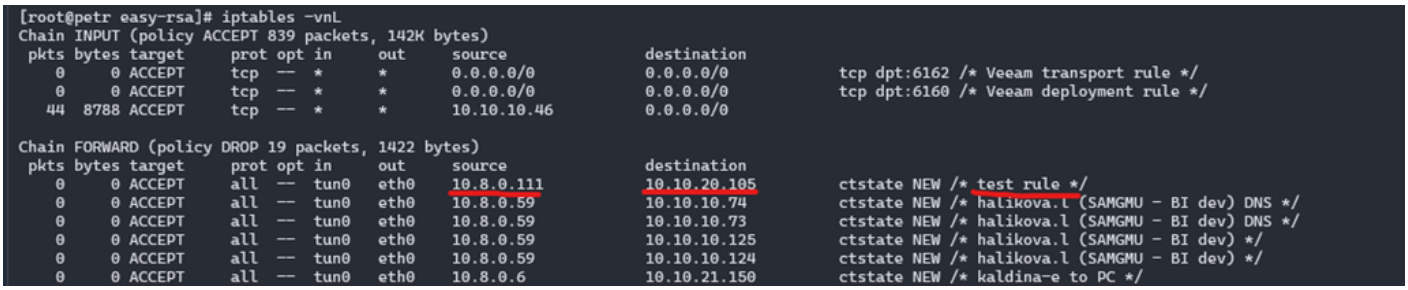
```
iptables -I FORWARD -i tun0 -o eth0 -s 10.8.0.122/32 -d 10.10.10.96/32 -m conntrack --ctstate NEW -j ACCEPT -m comment --comment "fursov.m– 1C"
```

Соответственно **-s** источник отправления (наш адрес клиента), **-d** адрес назначения, куда мы ему разрешаем ходить. Ну и комментарий.

5. Если необходимо прописать еще и порт

```
iptables -I FORWARD -i tun0 -o eth0 -s 10.8.0.119/32 -d 10.10.21.151/32 -p tcp --destination-port 3389 -m conntrack --ctstate NEW -j ACCEPT -m comment --comment "fursov.m – 1C"
```

6. Еще раз выведем iptables и удостоверимся, что правило появилось



```
[root@petr easy-rsa]# iptables -vnL
Chain INPUT (policy ACCEPT 839 packets, 142K bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           tcp dpt:6162 /* Veeam transport rule */
    0     0 ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           tcp dpt:6160 /* Veeam deployment rule */
   44  8788 ACCEPT     tcp  --  *      *       10.10.10.46          0.0.0.0/0

Chain FORWARD (policy DROP 19 packets, 1422 bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.111           10.10.20.105        ctstate NEW /* test rule */
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.59            10.10.10.74         ctstate NEW /* halikova.l (SAMGMU - BI dev) DNS */
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.59            10.10.10.73         ctstate NEW /* halikova.l (SAMGMU - BI dev) DNS */
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.59            10.10.10.125        ctstate NEW /* halikova.l (SAMGMU - BI dev) */
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.59            10.10.10.124        ctstate NEW /* halikova.l (SAMGMU - BI dev) */
    0     0 ACCEPT     all  --  tun0   eth0    10.8.0.6             10.10.21.150        ctstate NEW /* kaldina-e to PC */
```

7. (Опционально) Перезапуск openvpn. На случай если при подключении от пользователя, по какой-то причине не применились правила, или что-то работает не корректно. При данном действии, у других пользователей могут порваться сессии, и он отвалятся.

```
systemctl restart openvpn@server
```

8. ОБЯЗАТЕЛЬНО ЗАРЕЗЕРВИРОВАТЬ IP КОМПЬЮТЕРА, ИНАЧЕ ПОСЛЕ СМЕНЫ IP НЕОБХОДИМО БУДЕТ МЕНЯТЬ ПРАВИЛО

Revision #3
Created 25 March 2025 11:30:43 by admin_vy
Updated 28 May 2025 08:54:38 by admin_vy