

Урок 5. Настройка сети в Linux. Работа с IPtables

Домашнее задание:

1. Настроить статическую конфигурацию (без DHCP) в Ubuntu через `ip` и `netplan`. Настроить IP, маршрут по умолчанию и DNS-сервера (1.1.1.1 и 8.8.8.8). Проверить работоспособность сети.

1.1. Восстановление конфигурации сети с помощью команды `ip`

`ip -c a` - адрес хоста

`ip addr add 192.168.193.144/24 broadcast 192.168.193.255 dev enp0s3`

- настроим маршрут по умолчанию

`ip -c r` - адрес роутера (можно спросить у сетевого администратора)

`ip route add default via 192.168.193.40 dev enp0s3`

`ip r` - проверим маршруты (автоматический и вручную настроенный)

- можем удалить один маршрут

`ip route del default via 192.168.193.40 dev enp0s3`

`ip r` - проверим маршруты (остался один автоматический)

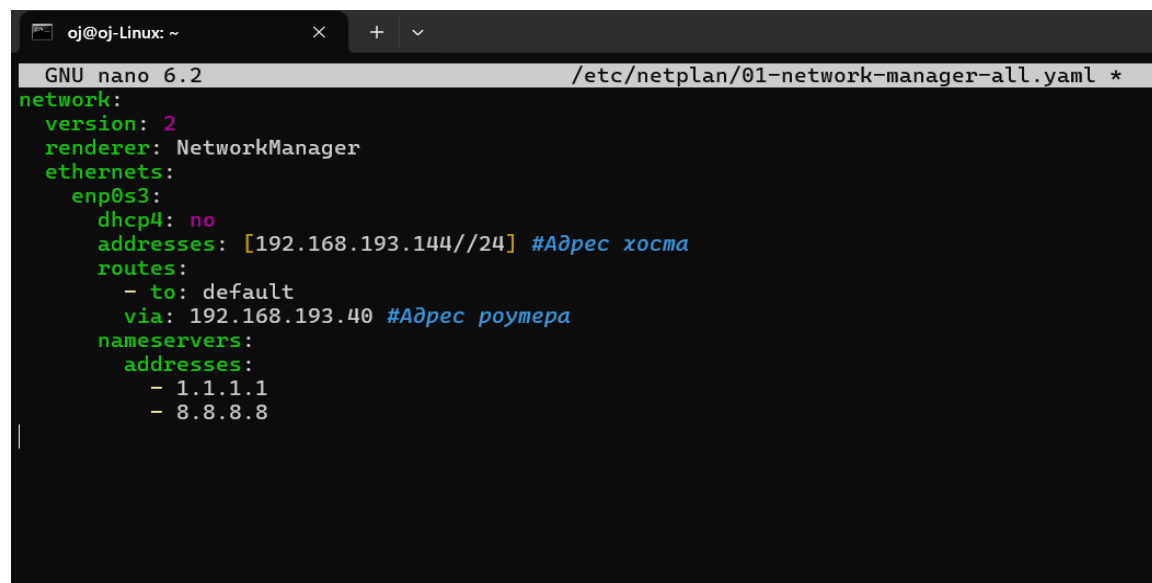
1.2. Внесем настройки через `netplan` (Важны отступы):

`nano /etc/netplan/01-network-manager-all.yaml`

или

`cd /etc/netplan`

`nano 00-installer - config.yaml`



```
oj@oj-Linux: ~
GNU nano 6.2 /etc/netplan/01-network-manager-all.yaml *
network:
  version: 2
  renderer: NetworkManager
  ethernet:
    enp0s3:
      dhcp4: no
      addresses: [192.168.193.144/24] #Адрес хоста
      routes:
        - to: default
          via: 192.168.193.40 #Адрес роутера
      nameservers:
        addresses:
          - 1.1.1.1
          - 8.8.8.8
```

Применение изменений:

netplan apply

Сохраняем этот файл и выходим из netplan

netplan try

ip a

ping ya.ru - проверяем, что сеть доступна, маршрутизатор есть

2. Настроить правила iptables для доступности сервисов на TCP-портах 22, 80 и 443. Также сервер должен иметь возможность устанавливать подключения к серверу обновлений. Остальные подключения запретить.

- Посмотрим, что из правил уже установлено
iptables -nvL

- Настройка правил iptables

iptables -A INPUT -p tcp --dport 22 -j ACCEPT

iptables -A INPUT -p tcp --dport 80 -j ACCEPT

iptables -A INPUT -p tcp --dport 443 -j ACCEPT

iptables -A OUTPUT -p tcp --dport 80 -j ACCEPT

iptables -A OUTPUT -p tcp --dport 443 -j ACCEPT

- Удаляем ненужный порт
сохраним все настройки:

cd

iptables-save > ip.rules

отредактируем в требуемом нам формате

nano ip.rules

удаляем строчку с ненужным портом

меняем перенаправление с порта 8090 на порт 80 (задание 4.)

A PREROUTING -p tcp --dport 8090 -j REDIRECT --to-port 80

сохраняем все изменения и перенаправляем этот файл в netplan

iptables-restore < ip.rules

проверяем список правил

iptables -nvL

3. Запретить любой входящий трафик с IP 3.4.5.6.

```
iptables -I INPUT -s 3.4.5.6 -j DROP ( I-добавление правила в самый верх)
iptables -nvL
```

4. Запросы на порт 8090 перенаправлять на порт 80 (на этом же сервере).

```
iptables -nvL
iptables -t nat -A PREROUTING -p tcp --dport 8090 -j REDIRECT --to-port 80
```

5. Разрешить подключение по SSH только из сети 192.168.0.0/24.

```
iptables -nvL
iptables -I INPUT -p tcp -s 192.168.0.0/24 --dport 22 -j ACCEPT
iptables -nvL
```

*Тренировка

Удаление ненужных правил

```
iptables -nvL --line-numbers
iptables -D INPUT 2 (номер правила)
```

Финальный список правил

```
iptables -nvL --line-numbers
```

Удаляем правило, которое разрешает всем IP-адресам доступ по SSH

```
iptables -D INPUT 2 (номер правила после допустимой подсети)
iptables -nvL --line-numbers
```

Сохраним все изменения:

```
iptables-save > ip.rules
```