Задание:
* 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.

Определяем все доступные сетевые интерфейсы, используя команду ip:

```
sudo ip a
```

Отредактируем файл конфигурации netplan который находится в директории /etc/netplan:

```
sudo vim /etc/netplan/00-installer-config.yaml
```

* 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.

Удаляем существующие правила:

```
iptables -F
```

Разрешаем установленные правила

```
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

Разрешаем циклический трафик

```
iptables -A INPUT -i lo -j ACCEPT
```

Разрешаем SSH (порт 22)

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

Разрешаем HTTP (порт 80)

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

Разрешаем HTTPS (порт 443)

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

Разрешаем подключение к серверу обновлений

```
iptables -A INPUT -p tcp --dport <update server port> -j ACCEPT
```

Запрещаем все остальные подключения

```
iptables -A INPUT -j REJECT
```

* Запретить любой входящий трафик с IP 3.4.5.6.

Удаляем существующие правила:

```
iptables -F
```

Разрешаем установленные правила

```
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

Разрешаем циклический трафик

```
iptables -A INPUT -i lo -j ACCEPT
```

Разрешаем SSH (порт 22)

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

Разрешаем HTTP (порт 80)

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

Разрешаем HTTPS (порт 443)

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

Разрешаем подключение к серверу обновлений

```
iptables -A INPUT -p tcp --dport <update server port> -j ACCEPT
```

Запрещаем любой входящий трафик с указанного IP

```
sudo iptables -t filter -A INPUT -s 3.4.5.6/32 -j DROP
```

(запрещает входящие пакеты без уведомления)
или воспользоваться
sudo iptables -t filter -A INPUT -s 3.4.5.6/32 -j REJECT
(запрещает входящие пакеты от указанного IP с уведомлением о запрете)

* * Запросы на порт 8090 перенаправлять на порт 80 (на этом же сервере).
Очищаем существующие правила
iptables -F
Разрешаем входящий трафик к сервисам HTTP
iptables -A INPUT -p tcp --dport 80 -j ACCEPT

Разрешиаем входящий трафик к сервисам HTTPS
iptables -A INPUT -p tcp --dport 443 -j ACCEPT
Запросы, идущие на порт 8080, перенаправляем на порт 80
iptables -t nat -A PREROUTING -p tcp --dport 8080 -j REDIRECT --to-port 80
Остальной входящий трафик запрещаем
iptables -A INPUT -j DROP

* * Разрешить подключение по SSH только из сети 192.168.0.0/24.
Очищаем существующие правила
iptables -F
Разрешаем входящий трафик к сервисам HTTP
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
Разрешиаем входящий трафик к сервисам HTTPS
iptables -A INPUT -p tcp --dport 443 -j ACCEPT
Запросы, идущие на порт 8080, перенаправляем на порт 80
iptables -t nat -A PREROUTING -p tcp --dport 8080 -j REDIRECT --to-port 80
Делаем входящий трафик по SSH только из указанной сети:
iptables -A INPUT -p tcp --src 95.24.0.0/13 --dport 22 -j ACCEPT
Остальной входящий трафик запрещаем
iptables -A INPUT -j DROP