# Microtik: Клиент Site-to-site OpenVPN с Ubuntu

Задача: Сервер OpenVPN на Ubuntu, Microtik в качестве клиента, подключение Site-to-site

Что понадобится:

- Сервер на Ubuntu с белым адресом (или проброшенным портом)
- Mikrotik
- 15 минут времени

## Сервер OpenVPN на Ubuntu

Примечание: Ubuntu 20.04 LTS

### Установка

Ставим пакеты:

```
sudo apt install openvpn easy-rsa -y
```

### Настройка конфигурации OpenVPN

Можно создать отдельного пользователя для генерации ключей, можно использовать существующего, можно все делать под root - выбирайте как больше нравится. Я создаю отдельного пользователя openvpn.

Создание пользователя:

```
useradd -m -s /bin/false openvpn
```

Далее все делаем от этого пользователя.

Создаем папки, даем на них права:

```
mkdir ~/easy-rsa
ln -s /usr/share/easy-rsa/* ~/easy-rsa/

chown openvpn ~/easy-rsa
chmod 700 ~/easy-rsa
```

Настраиваем конфигурацию, которая будет использоваться при генерации ключей:

```
cd ~/easy-rsa
```

```
cp /usr/share/easy-rsa/vars.example vars
```

Далее в файле vars изменяем/добавляем описанные ниже строки.

Настройки для ключей, чтобы не запрашивало каждый раз:

```
set_var EASYRSA_REQ_COUNTRY "RU"
set_var EASYRSA_REQ_PROVINCE "Moscow Region"
set_var EASYRSA_REQ_CITY "Moscow"
set_var EASYRSA_REQ_ORG "YOU_ORGANIZATION"
set_var EASYRSA_REQ_EMAIL "you@email.com"
set_var EASYRSA_REQ_OU "OU_UNIT"
```

Длина ключа:

```
set_var EASYRSA_KEY_SIZE 2048
```

Вид ключа. Можно старый добрый RSA, можно новый и более короткий Elliptic curve. Для работы с Микротиками я оставляю RSA 2048

```
# The default crypto mode is rsa; ec can enable elliptic curve support.
# Note that not all software supports ECC, so use care when enabling it.
# Choices for crypto alg are: (each in lower-case)
#  * rsa
#  * ec

set_var EASYRSA_ALGO rsa

# Define the named curve, used in ec mode only:

#set_var EASYRSA_CURVE secp384r1
```

Срок действия корневого CA сертификата:

```
set_var EASYRSA_CA_EXPIRE 7300
```

Срок действия выдаваемых для пользователей сертификатов:

```
set_var EASYRSA_CERT_EXPIRE 3650
set_var EASYRSA_KEY_EXPIRE 3650
```

Если не планируете заморачиваться со списками отзыва ключей (например, делаем ключи только для редко сменяемого оборудования):

```
set_var EASYRSA_CRL_DAYS 3650
```

## Создание CA и прочих сертификатов:

Инициализируем:

```
./easyrsa init-pki
```

Создаем CA. В процессе создания будет запрошен пароль - запишите его в надежном месте, он понадобится при любой операции по выдаче/удаление сертификатов!

```
./easyrsa build-ca
```

Создаем DH:

```
./easyrsa gen-dh
```

Создаем TA (TLS, но использовать с Микротиком не будем, а вот пользователей сможем дополнительно проверять):

```
openvpn --genkey --secret ~/easy-rsa/pki/ta.key
```

Создаем сертификат сервера (ключ nopass чтобы не требовался пароль):

```
./easyrsa build-server-full VPN-Server nopass
```

Создаем CRL (список отозванных сертификатов):

```
./easyrsa gen-crl
```

Копируем все сертификаты и ключи в папку настроек OpenVPN:

```
sudo cp -rp ~/easy-rsa/pki/{ca.crt,dh.pem,ta.key,crl.pem}
/etc/openvpn/server/
sudo cp -rp ~/easy-rsa/pki/issued/VPN-Server.crt /etc/openvpn/server/
sudo cp -rp ~/easy-rsa/pki/private/VPN-Server.key /etc/openvpn/server/
```

## Создание пользователей

Создание выполняется следующей командой (my-user - имя пользователя, ключ nopass чтобы не требовался пароль):

```
./easyrsa build-client-full my-user nopass
```

## Настройка конфигурационного файла OpenVPN

Немного условий:

- Для VPN у нас будет использоваться подсеть 172.16.1.0 255.255.255.0
- Сеть за сервером Ubuntu: 10.0.0.0 255.255.255.0
- Сеть за Mikrotik: 192.168.1.0 255.255.255.0

Создаем файл /etc/openvpn/mikrotik.conf со следующим содержимым:

```
ca /etc/openvpn/server/ca.crt
cert /etc/openvpn/server/VPN-Server.crt
key /etc/openvpn/server/VPN-Server.key
dh /etc/openvpn/server/dh.pem
crl-verify /etc/openvpn/server/crl.pem

proto tcp
dev tun
topology subnet
server 172.16.1.0 255.255.255.0
client-config-dir /etc/openvpn/ccd
keepalive 10 60
persist-key
persist-tun
auth SHA1
# for OpenVPN < v.2.6.0
#cipher AES-256-CBC
# for OpenVPN >= v.2.6.0
cipher AES-256-CBC
data-ciphers 'AES-256-CBC'
data-ciphers-fallback 'AES-256-CBC'

user nobody
group nogroup

push "route 10.0.0.0 255.255.255.0"
route 192.168.1.0 255.255.255.0

log-append /var/log/openvpn/openvpn-mikrotik.log
status /var/log/openvpn/openvpn-mikrotik-status.log
ifconfig-pool-persist /var/log/openvpn/ipp-mikrotik.txt
verb 3
mute 20
mssfix 0
```

Создаем папку /etc/openvpn/ccd - там будут храниться индивидуальные конфиги для клиентов.

Создаем файл /etc/openvpn/ccd/my-user со следующим содержимым:

```
ifconfig-push 172.16.1.2 255.255.255.0
iroute 192.168.1.0 255.255.255.0
```

Смысл этих настроек следующий:

- В файле mikrotik.conf указали отправлять на сторону клиента маршрут до сети за сервером OpenVPN на Ubuntu (push «route 10.0.0.0 255.255.255.0»)
- Там же прописали маршрут до сети клиента (route 192.168.1.0 255.255.255.0)
- В файле my-user явно указали какой IP назначить данному клиенту (ifconfig-push 172.16.1.2 255.255.255.0)
- Там же явно указали какая сеть у данного клиента (iroute 192.168.1.0 255.255.255.0)

# Mikrotik - настройка клиента

## Импортируем сертификаты

Копируем с сервера Ubuntu следующие файлы (на Windows это можно сделать при помощи WinSCP):

- ~/easy-rsa/pki/ca.crt
- ~/easy-rsa/pki/issued/my-user.crt
- ~/easy-rsa/pki/private/my-user.key

Заливаем сертификаты на Mikrotik: Files → Upload

Импортируем сертификаты через System → Certificates → Import в указанном порядке:

- ca.crt
- my-user.crt
- my-user.key

В итоге должно быть 2 записи:

- ca.crt со статусом «T»
- my-user.crt со статусом «KT»

## Настраиваем подключение

Настраиваем профиль OVPN: PPP → Profiles → Добавить

- Вкладка General
  - Name: ovpn-profile
- Change TCP MSS: yes
- Вкладка Protocols
  - Use IPv6: no
- Use MPLS: no
- Use compression: no
- Use Encryption: yes

Добавляем интерфейс OVPN Client:

- Закладка General
  - Name: OVPN-VPN-Server
- Закладка Dial Out
  - Connect to: указываем адрес нашего сервера Ubuntu OpenVPN
- Port: 1194
- Mode: ip
- User: my-user (пользователь, которому мы сделали сертификат)
- Profile: ovpn-profile
- Certificate: выбираем наш импортированный сертификат пользователя my-user
- Verify Server Certificate: включить

- Auth: Sha1
- Cipher: aes 256
- Use Peer DNS: отключить
- Add Default Route: отключить, но если надумаете весь трафик гнать через данный VPN то следует включить

## Настраиваем разрешения для нового интерфейса

Чтобы трафик ходил через наш интерфейс добавляем 3 правила: IP → Firewall → Filter Rules

Внимание: данные правила разрешают любой трафик через VPN!

### Правило Input

Закладка General

- Chain: input
- In. Interface: OVPN-VPN-Server

Закладка Action:

- Action: accept

### Правило Ouput

Закладка General

- Chain: output
- Out. Interface: OVPN-VPN-Server

Закладка Action:

- Action: accept

### Правило Forward

Закладка General

- Chain: forward
- Out. Interface: OVPN-VPN-Server

Закладка Action:

- Action: accept

Проверяем работу VPN.

# Что дальше

Далее можно ограничить разрешения на трафик через VPN и принять прочие меры безопасности.