**Лекция 13**

**Создание и настройка VPN**

На сегодняшний день технология VPN приобретает все большую популярность. VPN используется обычными пользователями для выхода в Интернет. Использование этого сервиса позволяет обходить региональные блокриовки ресурсов и обезопасить себя от возможного отслеживания извне. При подключении к VPN серверу между компьютером пользователя и сервером создается защищенный туннель, недоступный извне, а точкой выхода в Интернет становится сам VPN сервер. В сети можно найти много как платных так и бесплатных сервисов, предоставляющих услуги VPN, но если по какой-то причине сторонние сервисы вас не устраивают, вы можете настроить VPN сервер самостоятельно.

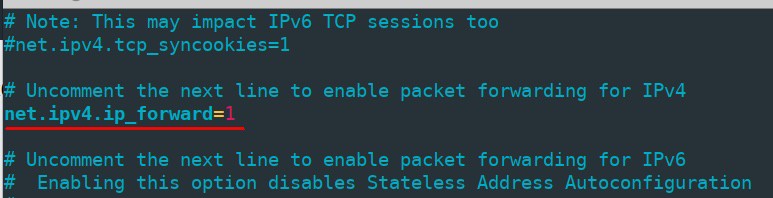
Чтобы создать собственный VPN, необходимо [арендовать подходящий виртуальный сервер](https://profitserver.ru/vps). Для создания VPN соединения существует различное программное обеспечение, которое отличается поддерживаемыми операционными системами и используемыми алгоритмами. В статье рассматривается два независимых друг от друга способа реализации VPN сервера. Первый основан на протоколе PPTP, который на сегодняшний день считается устаревшим и небезопасным, но при этом очень прост в настройке. Второй использует современное и безопасное ПО OpenVPN, но требует установки стороннего клиентского приложения и выполнения более сложных настроек.

В тестовой среде в качестве сервера используется виртуальный сервер под управлением операционной системы Ubuntu Server 18.04. Брандмауэр на сервере отключен, так как его настройка не рассматривается в данной статье. Настройка клиентской части описана на примере Windows 10.

**Подготовительные операции**

Независимо от того, какой из вариантов VPN сервера вы предпочтете, доступ клиентов в Интернет будет реализован штатными средствами операционной системы. Для того, чтобы из внутренней сети открыть доступ в Интернет через внешний интерфейс сервера необходимо разрешить пересылку пакетов между интерфейсами (форвардинг пакетов), и настроить трансляцию адресов.

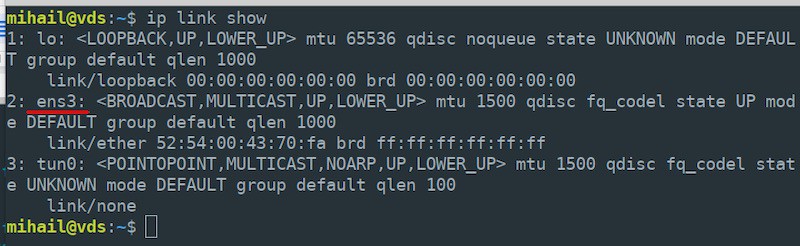
Для включения форвардинга пакетов откроем файл **“/etc/sysctl.conf”** и изменим значение параметра **“net.ipv4.ip\_forward”** на **1**.



Чтобы изменения применились без перезагрузки сервера, выполним команду

sudo sysctl -p /etc/sysctl.conf

Трансляция адресов настраивается средствами **iptables**. Предварительно уточним имя внешнего сетевого интерфейса, выполнив команду **“ip link show”**, оно понадобится на следующем шаге. В нашем случае имя интерфейса **“ens3”**.



Включаем трансляцию адресов на внешнем интерфейсе для всех узлов локальной сети.

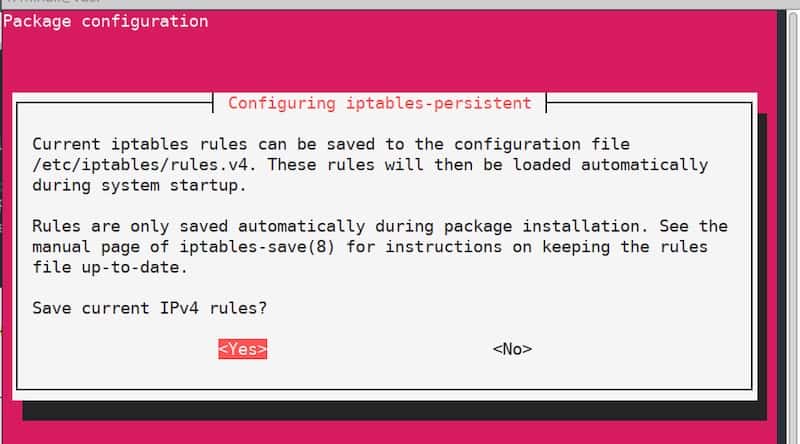
sudo iptables -t nat -A POSTROUTING -o ens3 -j MASQUERADE

Обратите внимание, что в команде необходимо указать реальное имя сетевого интерфейса. На вашем сервере оно может отличаться.

По умолчанию все созданные правила **iptables** сбрасываются после перезагрузки сервера, для того, чтобы этого избежать, воспользуемся утилитой **“iptables-persistent”**  Устанавливаем пакет.

sudo apt install iptables-persistent

В процессе установки откроется окно конфигурации, в котором система предложит сохранить текущие правила iptables. Так как правила уже настроены, соглашаемся и дважды нажимаем **“Yes”**. Теперь, после перезагрузки сервера правила будут восстанавливаться автоматически.



**1. PPTP сервер**

**Настройка сервера**

Устанавливаем пакет

sudo apt install pptpd  
  
После завершения установки открываем в любом текстовом редакторе файл **“/etc/pptpd.conf”** и приводим его к следующему виду.

option /etc/ppp/pptpd-options #путь к файлу с настройками  
logwtmp #механизм логирования клиентских подключений  
connections 100 #количество одновременных подключений  
localip 172.16.0.1 #адрес, который будет шлюзом для клиентов  
remoteip 172.16.0.2-200 #диапазон адресов для клиентов

Далее редактируем файл **“/etc/ppp/pptpd-options”**, большинство параметров уже установлены по умолчанию.

#имя сервиса, потребуется при создании учетных записей для клиентов  
name pptpd

#запрещаем устаревшие методы аутентификации  
refuse-pap  
refuse-chap  
refuse-mschap

#разрешаем более надежный метод аутентификации  
require-mschap-v2

#включаем шифрование  
require-mppe-128

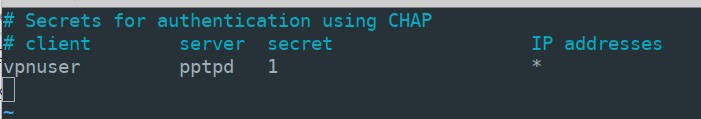
#указываем dns сервера для клиентов, можно указать любые доступные  
ms-dns 8.8.8.8  
ms-dns 8.8.4.4

proxyarp  
nodefaultroute  
lock  
nobsdcomp  
novj  
novjccomp  
nologfd

На следующем этапе необходимо создать учетную запись для подключения клиентов. Предположим, мы хотим добавить пользователя “vpnuser”, с паролем “1” и разрешить для него динамическую адресацию. Открываем файл **“/etc/ppp/chap-secrets”** и добавляем в конец строку с параметрами пользователя.

vpnuser pptpd 1 \*

Значение **“pptpd”** это имя сервиса, которое мы указали в файле “pptpd-options”. Вместо символа “\*” для каждого клиента можно указать фиксированный ip-адрес. В результате содержимое файла “chap-secrets” будет таким.



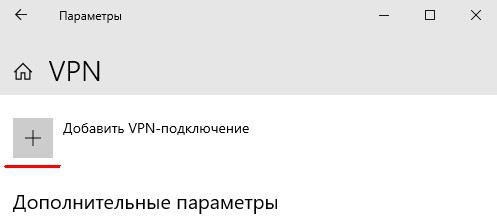
Для применения настроек перезагружаем службу **pptpd** и добавляем её в автозагрузку.

sudo systemctl restart pptpd  
sudo systemctl enable pptpd

Настройка сервера завершена.

**Настройка клиента**

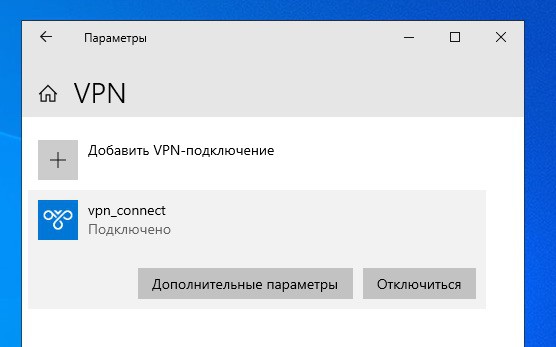
Открываем **“Пуск” - “Параметры” - “Сеть и интернет” - “VPN”** и нажимаем **“Добавить VPN-подключение”**



В открывшемся окне вводим параметры подключения и нажимаем **“Сохранить”**

* Поставщик услуг VPN: “Windows (встроенные)”
* Имя подключения: “vpn\_connect” (можно ввести любое)
* Имя или адрес сервера: (указываем внешний ip адрес сервера)
* Тип VPN: “Автоматически”
* Тип данных для входа: “Имя пользователя и пароль”
* Имя пользователя: **vpnuser** (имя, которое указано в файле “chap-secrets” на сервере)
* Пароль: **1** (так же из файла “chap-secrets”)

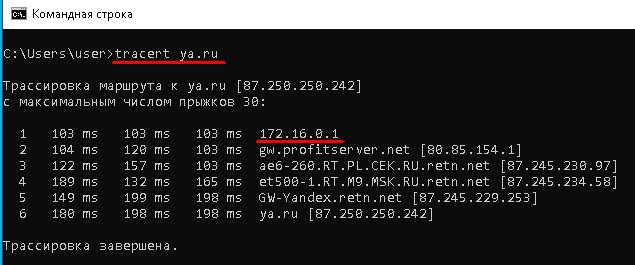
После сохранения параметров, в окне VPN появится новое подключение. Щелкаем по нему левой кнопкой мыши и нажимаем **“Подключиться”**. При успешном соединении с сервером, на значке подключения появится надпись **“Подключено”**.



В свойствах подключения отображаются внутренние адреса клиента и сервера. В поле **“Адрес назначения”** указан внешний адрес сервера.



При установленном соединении внутренний ip-адрес сервера, в нашем случае **172.16.0.1**, становится шлюзом по умолчанию для всех исходящих пакетов.



Воспользовавшись любым онлайн-сервисом вы можете убедиться, что внешний IP адрес компьютера теперь совпадает с IP адресом вашего VPN сервера.

**2. OpenVPN сервер**

**Настройка сервера**

Выполним повышение прав текущего пользователя, так как для всех дальнейших действий требуется root доступ.

sudo -s

Устанавливаем необходимые пакеты. Пакет **“Easy-RSA”** нужен для управления ключами шифрования.

apt install openvpn easy-rsa

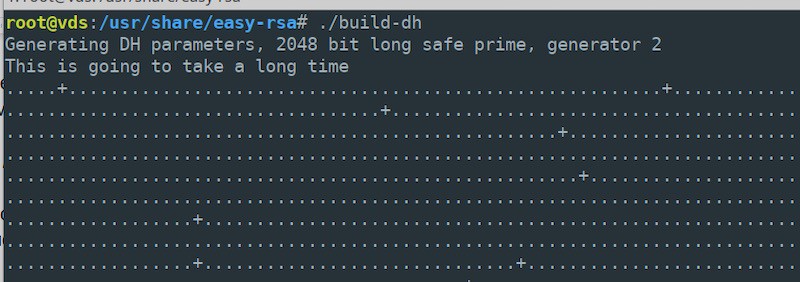
Создаем символическую ссылку на конфигурационный файл OpenSSL, в противном случае система выдаст ошибку при загрузке переменных.

ln -s /usr/share/easy-rsa/openssl-1.0.0.cnf /usr/share/easy-rsa/openssl.cnf

Переходим в рабочий каталог утилиты easy-rsa, загружаем переменные и очищаем старые конфигурации.

cd /usr/share/easy-rsa/  
source ./vars  
./clean-all  
  
Приступаем к созданию ключей. Генерируем ключ Диффи-Хеллмана, процесс может занять некоторое время.

./build-dh



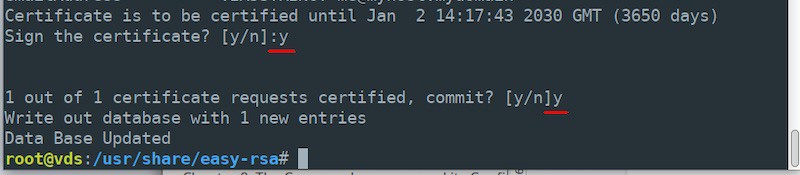
Генерируем центр сертификации.  
  
./build-ca

В процессе необходимо ответить на вопросы и ввести информацию о владельце ключа. Вы можете оставить значения по умолчанию, которые указаны в квадратных скобках. Для завершения ввода нажимаем **“Enter”**.

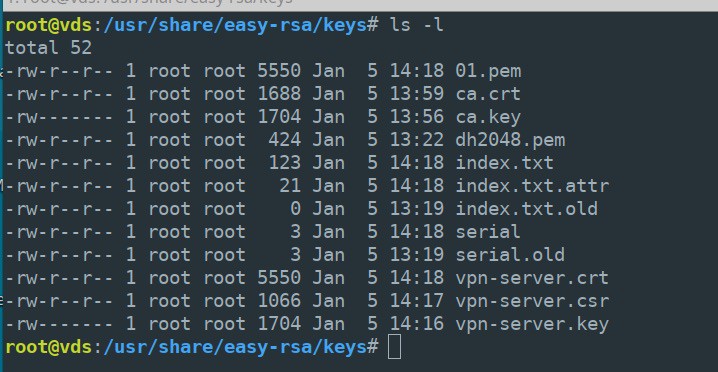
Генерируем ключи для сервера, в качестве аргумента указываем произвольное название, в нашем случае это **“vpn-server”**

./build-key-server vpn-server

Как и на предыдущем шаге отвечаем на вопросы или оставляем значения по умолчанию. На завершающем этапе дважды нажимаем **“y”**.



Генерация ключей сервера завершена, все файлы находятся в папке **“/usr/share/easy-rsa/keys”**.



Создадим в рабочем каталоге OpenVPN папку **“keys”** для хранения ключей и скопируем туда необходимые файлы.

mkdir /etc/openvpn/keys  
cp ca.crt dh2048.pem vpn-server.key vpn-server.crt /etc/openvpn/keys/

Копируем и распаковываем в каталог “/etc/openvpn/” шаблон конфигурационного файла.  
  
cp /usr/share/doc/openvpn/examples/sample-config-files/server.conf.gz /etc/openvpn/  
gzip -d /etc/openvpn/server.conf.gz

Открываем на редактирование файл **“/etc/openvpn/server.conf”** и убеждаемся в наличии следующих строк, при необходимости корректируем.

#Порт, протокол и интерфейс  
port 1194  
proto udp  
dev tun

#Путь к ключам шифрования  
ca /etc/openvpn/keys/ca.crt  
cert /etc/openvpn/keys/vpn-server.crt  
key /etc/openvpn/keys/vpn-server.key  
dh /etc/openvpn/keys/dh2048.pem

#Сетевые параметры  
topology subnet  
server 10.8.0.0 255.255.255.0  
ifconfig-pool-persist /var/log/openvpn/ipp.txt  
push "redirect-gateway def1 bypass-dhcp"  
push “dhcp-option DNS 8.8.8.8”  
push “dhcp-option DNS 8.8.4.4”

#Отключаем дополнительное шифрование  
#tls-auth ta.key 0

#Включаем компрессию  
compress lz4-v2  
push "compress lz4-v2"

#Понижаем привилегии службы OpenVPN после запуска  
user nobody  
group nogroup

#Включаем сохранение параметров после перезапуска  
persist-key  
persist-tun

#Перенаправляем логи  
log /var/log/openvpn/openvpn.log

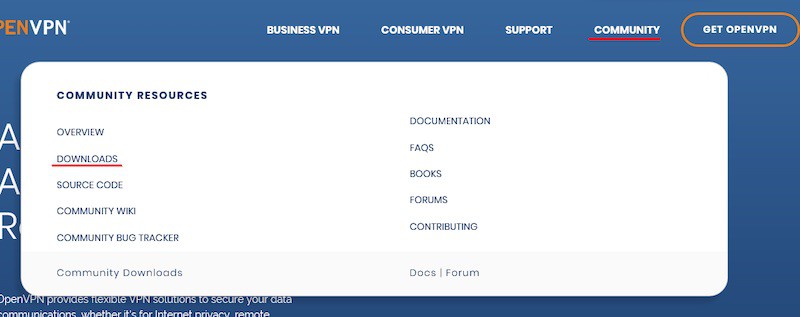
Остальные параметры оставляем без изменений.  
Перезапускаем службу OpenVPN для применения конфигурации.

systemctl restart openvpn

Настройка сервера завершена!

**Настройка клиента**

Заходим на официальный сайт проекта “<https://openvpn.net>”, переходим в раздел **“COMUNITY”** - **“DOWNLOADS”**



и скачиваем инсталлятор для своей версии операционной системы. В нашем случае это Windows 10.



Устанавливаем приложение, оставляя все параметры по умолчанию.

На следующем этапе необходимо подготовить на сервере и передать на компьютер клиента следующие файлы:

* публичный и приватный ключи;
* копия ключа центра сертификации;
* шаблон конфигурационного файла.

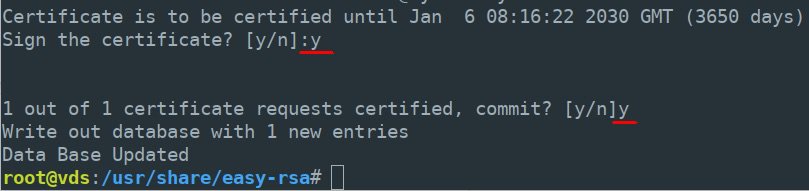
Подключаемся к серверу, повышаем права,переходим в рабочий каталог утилиты **“easy-rsa”** и загружаем переменные.

sudo -s  
cd /usr/share/easy-rsa/  
source ./vars

Генерируем ключевую пару для клиента, в качестве аргумента указываем произвольное имя, в нашем случае **“client1”**.

./build-key client1

Отвечая на вопросы вводим свои данные или просто нажимаем **“ENTER”**, оставляя значения по умолчанию. После этого дважды нажимаем **“y”**



Сгенерированные ключи клиента также находятся в папке **“/usr/share/easy-rsa/keys/”** Для удобства создадим в домашнем каталоге папку **“client1”** и скопируем в нее все файлы предназначенные для переноса на клиентский компьютер.

cd /usr/share/easy-rsa/keys/  
mkdir ~/client1  
cp client1.crt client1.key ca.crt ~/client1/

Скопируем в эту же папку шаблон клиентского конфигурационного файла. При копировании меняем расширение файла на **“ovpn”**.

cp /usr/share/doc/openvpn/examples/sample-config-files/client.conf ~/client1/client.ovpn

Изменим владельца каталога **“~/client1/”** и всех файлов находящихся в нем, для того, чтобы получить возможность перенести их на клиентский компьютер. В нашем случае сделаем владельцем пользователя **“mihail”**

chown -R mihail:mihail ~/client1

Переходим на клиентский компьютер и копируем с сервера содержимое папки **“~/client1/”** любым доступным способом, например с помощью утилиты **“PSCP”**, которая входит в состав клиента Putty.

PSCP -r mihail@[IP\_сервера]:/home/mihail/client1 c:\client1

Файлы ключей **“ca.crt”, “client1.crt”, “client1.key”** можно хранить в любом месте, в нашем случае это папка **“c:\Program Files\OpenVPN\keys”**, а конфигурационный файл **“client.ovpn”** переносим в директорию **“c:\Program Files\OpenVPN\config”**.

Приступаем к конфигурированию клиента. Открываем в блокноте файл **“c:\Program Files\OpenVPN\config\client.ovpn”** и отредактируем следующие строки

#Сообщаем, что мы являемся клиентом  
client

#Интерфейс и протокол так же как на сервере  
dev tun  
proto udp

#IP адрес сервера и порт  
remote ip\_адрес\_сервера 1194

#сохранение параметров при перезапусках  
persist-key  
persist-tun

#Путь к ключам  
ca “c:\\Program Files\\OpenVPN\\keys\\ca.cert”  
cert “c:\\Program Files\\OpenVPN\\keys\\client1.crt”  
key “c:\\Program Files\\OpenVPN\\keys\\client1.key”

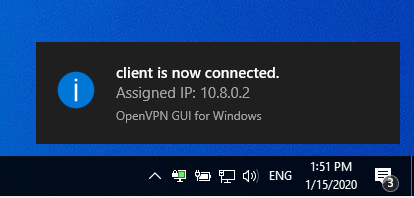
#Включаем проверку подлинности сервера  
remote-cert-tls server

#Отключаем дополнительное шифрование  
#tls-auth ta.key 1

cipher AES-256-CBC  
comp-lzo  
auth-nocache  
verb 3

Остальные параметры оставляем без изменений.

Сохраняем файл и запускаем клиентское приложение **“OpenVPN GUI”**.

  
Для подключения к серверу щелкаем правой кнопкой мыши на иконке в трее и выбираем **“Подключиться”**. При успешном соединении иконка станет зеленого цвета.