

Бонусное задание

Пользовательская установка wireguard

Зайти в режим привилегированного пользователя

```
sudo su
```

установить нужные пакеты:

```
apt update  
apt install -y wireguard qrencode
```

Настройка системы

Разрешить перенаправление сетевых пакетов на уровне ядра. Для этого откройте файл `/etc/sysctl.conf` и добавьте в конец такие строки:

```
vi /etc/sysctl.conf
```

```
net.ipv4.ip_forward = 1  
net.ipv6.conf.default.forwarding = 1  
net.ipv6.conf.all.forwarding = 1  
net.ipv4.conf.all.rp_filter = 1  
net.ipv4.conf.default.proxy_arp = 0  
net.ipv4.conf.default.send_redirects = 1  
net.ipv4.conf.all.send_redirects = 0
```

Затем необходимо выполнить команду `sysctl -p` чтобы система перечитала конфигурацию:

```
sysctl -p
```

Генерация ключей сервера

Для сервера надо создать приватный и публичный ключ. Эти ключи, потом надо будет записать в конфигурационный файл сервера и клиента, сами файлы ключей вам не нужны, поэтому можете создавать их где хотите, например, в домашней папке. Так же полученный

ключ можно записать в переменную окружения:

```
wg genkey | sudo tee server_private.key | wg pubkey | sudo tee server_public.key
```

Ключи созданы, утилита tee запишет их в файл, а также выведет на экран, что очень удобно для сохранения значения в переменную

Генерация ключей клиента

Аналогичным образом создаём ключи для клиента. Команда та же:

```
wg genkey | sudo tee client_private.key | wg pubkey | sudo tee client_public.key
```

Конфигурационный файл сервера

Наш конфигурационный файл сервера будет находится по пути `/etc/wireguard/wg0.conf` и будет выглядеть следующим образом:

```
vi /etc/wireguard/wg0.conf
```

```
[Interface]
Address = 10.10.10.1/24
ListenPort = 62666
PrivateKey = "приватный ключ сервера из файла server_private.key"
PostUp = iptables -A FORWARD -i wg0 -j ACCEPT; iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE;
ip6tables -A FORWARD -i wg0 -j ACCEPT; ip6tables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
PostDown = iptables -D FORWARD -i wg0 -j ACCEPT; iptables -t nat -D POSTROUTING -o eth0 -j MASQUERADE;
ip6tables -D FORWARD -i wg0 -j ACCEPT; ip6tables -t nat -D POSTROUTING -o eth0 -j MASQUERADE
[Peer]
PublicKey = "публичный файл клиента из файла client_public.key"
AllowedIPs = 10.10.10.2/32
```

Файл разделен на две секции:

- `Interface` - настройка сервера;
- `peer` - настройка клиентов, которые могут подключаться к серверу, секций `Peer` может быть несколько.

В данном случае мы настраиваем сервер WireGuard для работы с IPv4 вот, что значат основные параметры:

- `Address` - адрес сервера в сети VPN;
- `ListenPort` - порт, на котором будет ожидать подключения WireGuard;

- PrivateKey - приватный ключ сервера, сгенерированный ранее;
- PostUp - команда, которая выполняется после запуска сервера. В данном случае включается поддержка MASQUERADE для интерфейса enp0s8, а также разрешается прием пакетов на интерфейсе wg0. Сетевые интерфейсы вам придется заменить на свои.
- PostDown - выполняется после завершения работы WireGuard, в данном случае удаляет все правила, добавленные в PostUp.

Секции Peer содержат настройки клиентов, которые могут подключиться к серверу:

- PublicKey - публичный ключ клиента, сгенерированный ранее;
- AllowedIPs - IP адрес, который может взять клиент. Обратите внимание, маска для IPv4 должна быть 32.

Теперь можно переходить к созданию конфигурационного файла непосредственно для клиента.

Конфигурационный файл клиента Конфигурационный файл клиента будет выглядеть примерно так:

```
vi client.conf
```

```
[Interface]
PrivateKey = "содержимое файла client_private.key"
Address = 10.10.10.2DNS = 172.17.1.10
[Peer]
PublicKey = "публичный ключ сервера server_public.key"
Endpoint = "ip адрес вашего инстанса":62666
AllowedIPs = 0.0.0.0/0
```

Обратите внимание, что все ключи мы генерируем на сервере, а затем уже скидываем конфигурационный файл клиента на компьютер, который надо подключить к сети.

Рассмотрим подробнее что за что отвечает:

- PrivateKey - приватный ключ клиента, сгенерированный ранее;
- Address - IP адрес интерфейса wg0 клиента;
- DNS - серверы DNS, которые будут использоваться для разрешения доменных имён;
- PublicKey - публичный ключ сервера, к которому надо подключиться.
- Endpoint - здесь надо указать IP адрес сервера, на котором установлен WireGuard и порт;
- AllowedIPs - IP адреса, трафик с которых будет перенаправляться в сеть VPN, в данном примере выбраны все адреса.

Запуск сервера

Для запуска сервера используйте такую команду:

```
sudo systemctl start wg-quick@wg0
```

С помощью `systemd` можно настроить автозагрузку интерфейса:

```
sudo systemctl enable wg-quick@wg0
```

Подключение клиента

Вывести в консоль qr код, для подключения к vpn. Подключаться к VPN с использованием клиента `wireguard` с мобильного телефона. Для этого в консоли сгенерировать qr код:

```
qrencode -t ansiutf8 < client.conf
```

Далее необходимо проверить, что установленный сервер работает.

Автоматизация установки wireguard

Для автоматизации установки необходимо создать скрипт, автоматически устанавливающий и запускающий все нужные сервисы. Скрипт – последовательное выполнение действий оболочкой командной строки, без участия оператора. Например, все команды, которые оператор ввёл в командной строке, могут быть записаны в один файл, и при вызове этого файла, они все подряд будут исполняться.

Для того, чтобы создать скрипт автоматизации, необходимо:

1. Создать файл скрипта

```
vi script.sh
```

2. Записать в заголовке (самой верхней строке) файла используемый интерпретатор:

```
#!/bin/bash
```

3. Записывать все введённые в терминале команды подряд (из части 1)
4. В случае, если необходимо изменить содержимое файла напрямую из командной строки, можно использовать конструкцию

```
cat <<EOF >>file.txt  
text  
EOF
```

Такая конструкция допишет вывод "text" в конец файла file.txt (так как используется >>)
При необходимости заменить содержимое файла, необходимо использовать >

```
cat <<EOF >file.txt
text
EOF
```

Такая конструкция перезапишет содержимое файла file.txt, и добавит в него только строку text.

Данную схему удобно использовать, при необходимости изменения файлов, например при необходимости дописать содержимое /etc/sysctl.conf

```
cat << EOF >>/etc/sysctl.conf
net.ipv4.ip_forward = 1
net.ipv6.conf.default.forwarding = 1
net.ipv6.conf.all.forwarding = 1
net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.default.proxy_arp = 0
net.ipv4.conf.default.send_redirects = 1
net.ipv4.conf.all.send_redirects = 0
EOF
```

Так же, при необходимости добавить в файл содержимое какой либо переменной, это можно сделать с помощью указания этой переменной в выводе EOF:

```
cat <<EOF > file.txt
$HOSTNAME
EOF
```

Запишет в файл file.txt имя вашего хоста. Чтобы создать переменную, можно в консоли просто указать необходимое ей значение

```
myhost=$HOSTNAME
```

создаст переменную myhost и присвоит ей значение переменной \$HOSTNAME

Для получения текущего сетевого адреса можно использовать следующую команду с различными фильтрами:

```
ip a | grep 172.17 | awk '{print $2}' | awk -F "/" '{print $1}'
```

Для того, чтобы записать этот адрес в переменную (например, `myaddr`), можно использовать следующую команду:

```
myaddr=$(ip a | grep 172.17 | awk '{print $2}' | awk -F "/" '{print $1}')
```

И проверить результат её исполнения

```
echo $myaddr
```

Таким образом в файл можно записывать как текст, так и значения переменных, что необходимо для автоматизации различных процессов 5. Для того, чтобы сделать скрипт исполняемым, необходимо выдать ему права на исполнение командой `chmod`. 6. На запущенной виртуальной машине необходимо запустить этот скрипт и проверить его исполнение. Запустить скрипт можно указав в командной строке команду на запуск в следующем виде:

```
# ./«имя скрипта»
```

После того, как скрипт отработал, он должен вывести QR код для подключения к VPN серверу. Проверить правильность работы можно подключившись к VPN серверу.

Установка wireguard из готового контейнера

Контейнер сам по себе является операционной системой минимального размера, с установленным внутри необходимым программным обеспечением. Контейнеры могут быть преднастроенными, и всё, что необходимо с ними сделать, это установить, передав нужные аргументы. Для работы с контейнерами чаще всего используются docker контейнеры, настроенные на необходимый режим работы путём передачи в них переменных окружения.

Перед установкой Docker нужно выполнить все необходимые настройки системы.

Установка Docker

Docker является набором утилит, для работы с контейнерами. Установку лучше всего выполнять из репозитория самого Docker. Для начала необходимо установить набор утилит, помогающих работать со сторонними репозиториями.

```
sudo apt-get update
sudo apt-get install -y apt-transport-https ca-certificates curl gnupg lsb-release
```

Далее необходимо скачать ключи доступа к репозиториям (одной командой):

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg \  
| sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
```

После этого необходимо добавить нужные репозитории (одной командой):

```
echo "deb [arch=$(dpkg --print-architecture) \  
signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] \  
https://download.docker.com/linux/ubuntu \  
$(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

Добавив репозитории, можно установить все необходимые пакеты

```
apt-get update  
apt-get install -y docker-ce docker-ce-cli containerd.io nftables
```

Запуск контейнера

Как только все необходимые приготовления сделаны, необходимо запустить контейнер с нужными параметрами (одной командой):

```
sudo docker run -ti -d --restart=always --network host \  
--entrypoint "/wireguard-ui" -v /tmp/wireguard-ui:/data \  
--privileged embarkstudios/wireguard-ui:latest \  
--data-dir=/data --wg-listen-port=51820 \  
--wg-endpoint="«ip адрес вашего виртуального сервера»:51820" \  
--wg-allowed-ips=0.0.0.0/0 --wg-dns="172.17.1.10" \  
--wg-device-name="wg0" --listen-address=":80" \  
--nat --nat-device="eth0" --client-ip-range="10.0.8.1/24"
```

Настройка VPN тоннелей

Как только контейнер запущен, необходимо подключиться к web интерфейсу VPN сервера для того, чтобы добавить клиентов для подключения. Для этого необходимо открыть в браузере адрес: <http://<ip адрес вашего виртуального сервера>> В открывшемся окне нажать + в правом нижнем углу


WireGuard VPN Logged in as anonymous

My VPN Clients

(anonymous)

Instructions


1. [Install WireGuard](#)
2. Download your WireGuard config
3. Connect to the VPN server



Powered by [WG UI](#).
Copyright © 2021 [Embark Studios](#).

В появившемся окне ввести имя клиента и нажать create

WireGuard VPN Logged in as anonymous




Create New Device Configuration

Client Name

Label

Generate a Pre-shared Key

CREATE 

Клиент для подключения создан, так же создан конфигурационный файл для клиента и QR код, по которому можно скачать содержимое этого конфигурационного файла

WireGuard VPN Logged in as anonymous

My VPN Clients

Instructions

После создания клиента VPN сервер готов к подключениям внешних клиентов, и можно переходить к проверке.

Версия #2

Артем Швидкий создал 15 июня 2022 13:38:19

Тарабанов Илья Федорович обновил 15 июня 2023 09:57:29