

Практическая работа №5

Цели работы: получение практических навыков настройки безопасного удаленного доступа к сети.

Ход работы

1. Выбрать и установить серверную ОС российского производства.

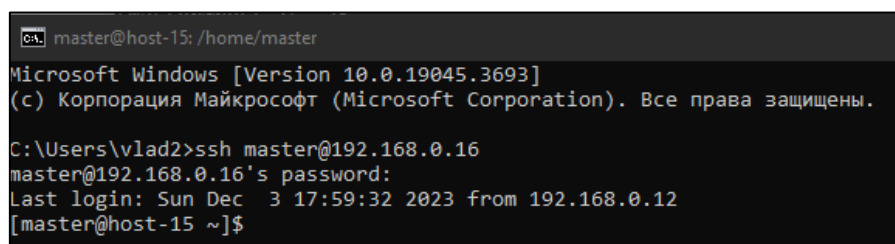
Мой выбор пал на ОС **AltServer**. Для установки ОС на ВМ, я скачал .iso-образ с официального сайта altdesktop и воспользовался ПО VB.

2. Настраиваете доступ с использованием SSH.

По стандарту каждый из разработчиков ОС внедряет ssh-сервер в свою ОС.

Для соединения можно воспользоваться популярными утилитами подключения по SSH (mobaXTerm или Pytty), а также можно воспользоваться CMD Windows или PoweShell (как вариант, можно также установить WSL на Windows и подключаться через оболочку linux-ubuntu).

Я выбрал стандартный метод – это подключение через cmd.



```
master@host-15: /home/master
Microsoft Windows [Version 10.0.19045.3693]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\vlad2>ssh master@192.168.0.16
master@192.168.0.16's password:
Last login: Sun Dec  3 17:59:32 2023 from 192.168.0.12
[master@host-15 ~]$
```

Рисунок 1. Подключение к ВМ по SSH

3. Настроить авторизацию SSH по сертификату/ключу.

Для начала нужно создать пару ключей (открытый/закрытый). После создания, открытый ключ мы передаем на сервер. (ключи создаются на клиенте).

```

C:\Users\vlad2>ssh-keygen -t rsa -b 2048 -f D:\ssh\id_rsa
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in D:\ssh\id_rsa.
Your public key has been saved in D:\ssh\id_rsa.pub.
The key fingerprint is:
SHA256:Y9Bx5l1T1bcbK1E3sZ9ervJKdMQkpzw11opz7tNgEN0 vlad2@DESKTOP-G4SCL79
The key's randomart image is:
+---[RSA 2048]-----+
|  . *.0oB. |
|  . * B @.o |
|  . o oo*o+ |
|  . .+.+*E.. |
|  S .+=+ ..o |
|  . ..O..... |
|  . o. . . . |
|  . =. . . . |
|  . o*o      |
+---[SHA256]-----+

```

Рисунок 2. Создание ключей

Далее я отправил публичный ключ на сервер и попробовал подключиться без пароля по SSH.

```

master@host-15: /home/master
master@DESKTOP-G4SCL79 ~$ ssh -i ~/.ssh/id_rsa master@192.168.0.16
Last login: Sun Dec 3 18:31:33 2023 from 192.168.0.12
[master@host-15 ~]$

```

Рисунок 3. Вход по SSH при помощи ключ (передача ключа и подключение через Linux (WSL))

```

C:\Users\vlad2>scp "D:\ssh_keys\id_rsa.pub" master@192.168.0.16:~/.ssh/authorized_keys
master@192.168.0.16's password:
id_rsa.pub 100% 404 201.6KB/s 00:00
C:\Users\vlad2>ssh -i D:\ssh_keys\id_rsa master@192.168.0.16
Last login: Sun Dec 3 18:46:46 2023 from 192.168.0.12
[master@host-15 ~]$

```

Рисунок 4. Вход по SSH при помощи ключа (передача ключа и подключение через CMD Windows)

4. Переопределяете сетевой порт службы SSH с 22 на 7021.

Для переопределения сетевого порта я отредактировал файл конфигурации сервиса sshd.

```

root@host-15: /etc/openssh
GNU nano 7.2 sshd_config
$OpenBSD: sshd_config,v 1.103 2018/04/09 20:41:22 tj Exp $

# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/bin:/usr/bin:/usr/local/bin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

Port 7021
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/openssh/ssh_host_rsa_key
#HostKey /etc/openssh/ssh_host_ecdsa_key
#HostKey /etc/openssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTHPRIV

Справка Записать Поиск Вырезать Выполнить Позиция Отмена Установить
Выход ЧитФайл Замена Вставить Выровнять К строке Повтор Копировать

```

Рисунок 5. Редактирование конфигурации

```
D:\ssh_keys>ssh -i id_rsa -p 7021 master@192.168.0.16
Last login: Sun Dec  3 19:01:22 2023 from 192.168.0.12
[master@host-15 ~]$
```

Рисунок 6. Вход по порту 7021

5. Настраиваете VPN сервер таким образом, чтобы клиенты попадали во внутреннюю сеть сервера и могли выходить в интернет.

В качестве VPN сервера, я выбрал широко известный – OpenVPN.

OpenVPN предоставляет возможность создания зашифрованных туннелей для безопасного соединения между удаленными клиентами и сервером. Однако, помимо обеспечения безопасной передачи данных, многим пользователям также требуется возможность доступа к ресурсам внутренней сети, а также выхода в интернет через сервер VPN.

Для настройки и конфигурации VPN сервера, я воспользовался web-оболочкой «ЦУС».

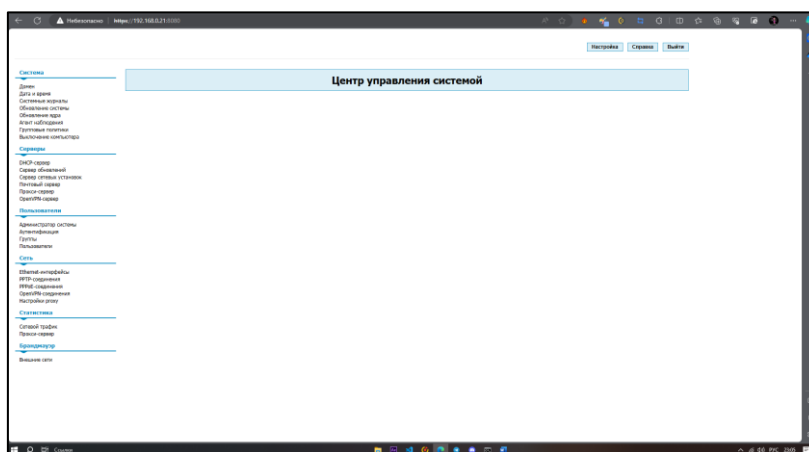


Рисунок 7. Центр управления системой

Перед работой требуется включить экспертный режим.

☐ Основной режим

☒ Режим эксперта

Применить

Сбросить

Рисунок 8. Переключение режима эксперта

Требуется создать ssl-ключ и подписать его в удостоверяющем центре.

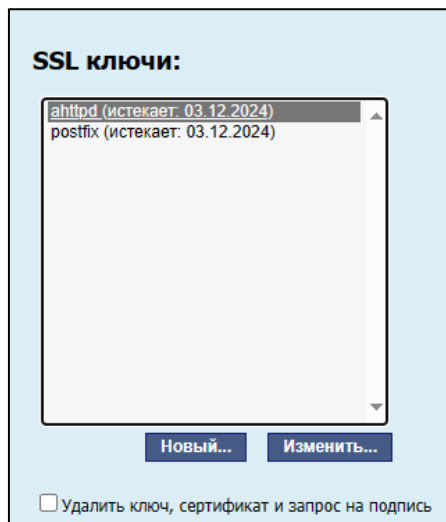


Рисунок 9. Создание нового ssl-ключа

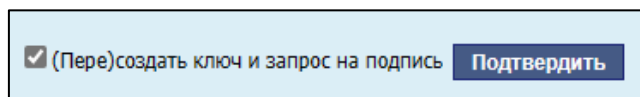


Рисунок 10. Создание ключа и запроса на подпись

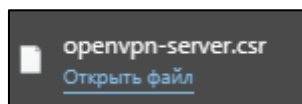


Рисунок 11. Созданный ключ и запрос на подпись

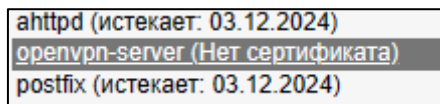


Рисунок 12. Ключ создан, но не подписан

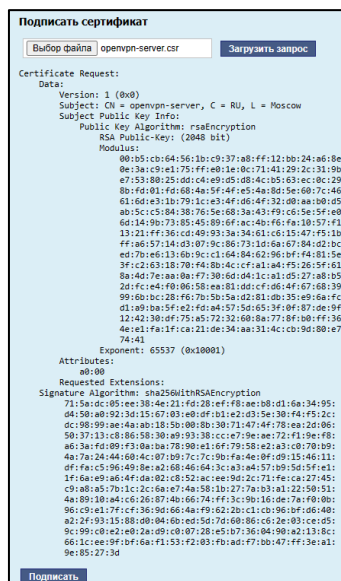


Рисунок 13. Подписание сертификата

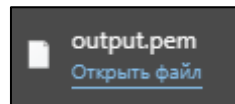


Рисунок 14. Подписанный сертификат

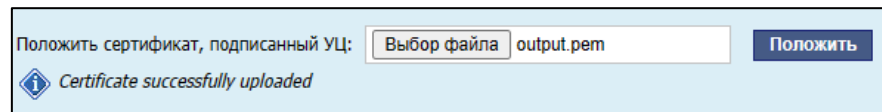


Рисунок 15. Подписание ключа

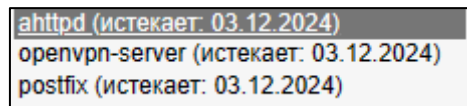


Рисунок 16. Ключ создан и подписан

Далее, я настроил и запустил OpenVPN Server.

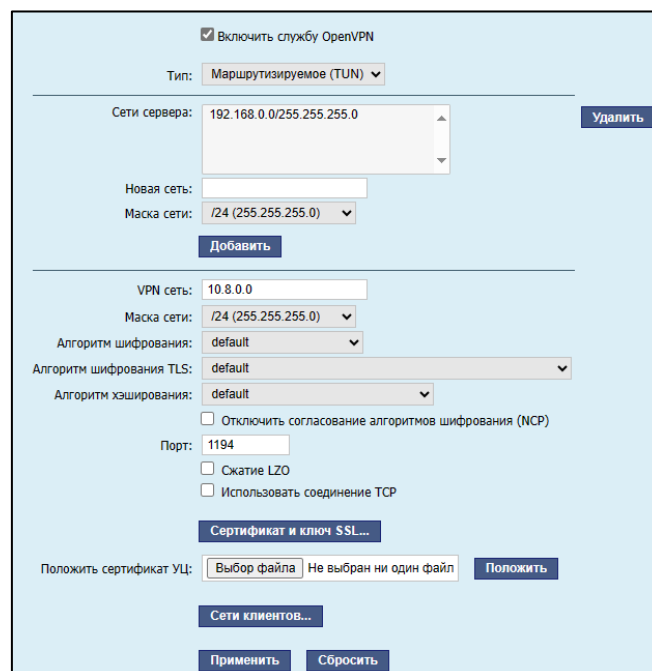


Рисунок 17. Запуск службы OpenVPN

Теперь требуется загрузить сертификат удостоверяющего центра.

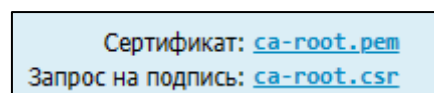


Рисунок 18. Сертификат УЦ

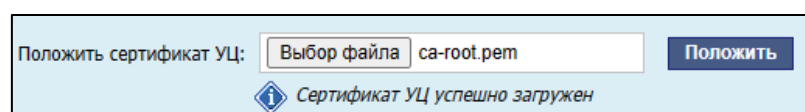


Рисунок 19. Успешная загрузка сертификата УЦ

Данный сертификат понадобится на сервере клиента для подписания сертификатов и ключей клиента.

```
master@host-15: /home/master
master@host-15 ~]$ ip a
lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:2b:7c:0a brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.21/24 brd 192.168.0.255 scope global dynamic noprefixroute enp0s3
        valid_lft 85191sec preferred_lft 85191sec
    inet6 fe80::a00:27ff:fe2b:7c0a/64 scope link
        valid_lft forever preferred_lft forever
tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOWN group default qlen 500
    link/none
    inet 10.8.0.1 peer 10.8.0.2/32 scope global tun0
        valid_lft forever preferred_lft forever
    inet6 fe80::7810:94c5:5fe9:e097/64 scope link stable-privacy
        valid_lft forever preferred_lft forever
master@host-15 ~]$
```

Рисунок 20. Сетевой интерфейс OpenVPN

```
PORT      STATE SERVICE
1194/udp  open  openvpn
MAC Address: 08:00:27:2B:7C:0A (Oracle VirtualBox virtual NIC)
```

Рисунок 21. Работоспособность порта

Для того, чтобы трафик шифровался корректным способом, требуется изменить конфигурацию OpenVPN сервера. Это можно сделать двумя способами: как и раньше через web-оболочку или же через конфигурационный файл `server.conf`.

```
user nobody
group nogroup
persist-key
persist-tun
keepalive 10 120
topology subnet
server 10.8.0.0 255.255.255.0
ifconfig-pool-persist ipp.txt
push "dhcp-option DNS 8.8.8.8"
push "dhcp-option DNS 8.8.4.4"
push "redirect-gateway def1 bypass-dhcp"
dh none
ecdh-curve prime256v1
tls-crypt tls-crypt.key
crl-verify crl.pem
ca ca.crt
cert server_NK1RyQ81LknZJFG2.crt
key server_NK1RyQ81LknZJFG2.key
auth SHA256
cipher AES-128-GCM
ncp-ciphers AES-128-GCM
tls-server
tls-version-min 1.2
tls-cipher TLS-ECDHE-ECDSA-WITH-AES-128-GCM-SHA256
client-config-dir /etc/openvpn/ccd
status /var/log/openvpn/status.log
verb 3
```

Рисунок 22. Файл конфигурации сервера

В моей конфигурации OpenVPN применяется симметричное шифрование AES-256-CBC для защиты данных, а также асимметричное шифрование и обмен ключами для обеспечения безопасности процесса установки соединения.

7. Проверить работоспособность VPN с гостевой машины.

Для этого я создал на сервере сертификат и ключ для клиента, подписал их и создал конфигурационный файл.

Для проверки работоспособности, я воспользовался ПО с открытым исходным кодом – OpenVPN GUI, импортировал конфигурационный файл и подключился к серверу.

```
client
dev tun
proto udp
remote 192.168.0.14 1194
resolv-retry infinite
nobind
persist-key
persist-tun
ca D:\\certs\\ca.crt
cert D:\\certs\\client.crt
key D:\\certs\\client.key
comp-lzo no
cipher AES-256-GCM
data-ciphers AES-256-GCM:AES-128-GCM
remote-cert-tls server
allow-compression no
verb 3
```

Рисунок 23. Конфигурационный файл клиента

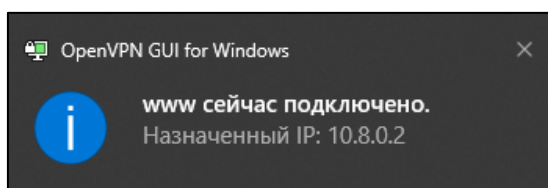


Рисунок 24. Успешное подключение к серверу

На данном рисунке можно заметить, что сервер выдал нам IP во внутренней сети сервера.

```
Назначенный IP: 10.8.0.2
Входящие байты: 6157334 (5.9 MiB) Исходящие байты: 258102 (252.1 KiB) OpenVPN GUI 11.46.0.0/2.6.8
```

Рисунок 25. Информация по входящим/исходящим байтам

После для проверки прогонки траффика, я открыл сайт wink и понаблюдавал за количеством входящего и исходящего траффика.

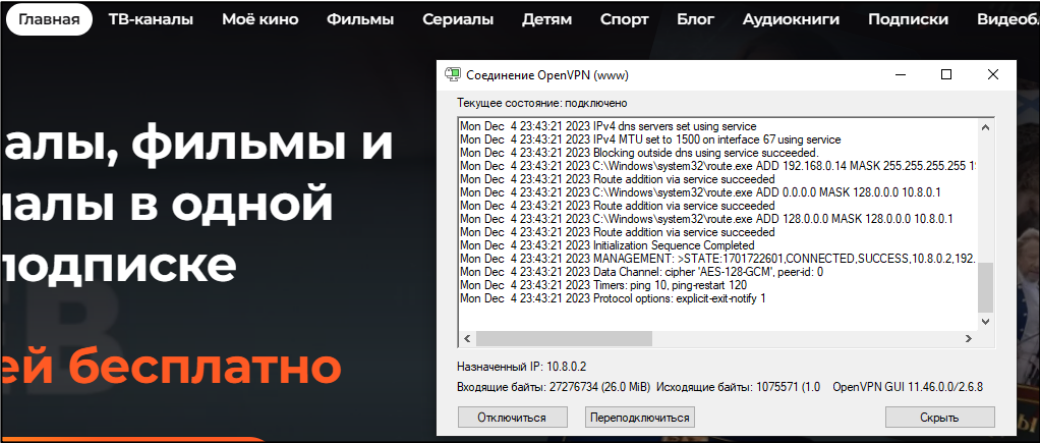


Рисунок 26. Мониторинг трафика

Можно заметить, что трафик увеличился в разы.

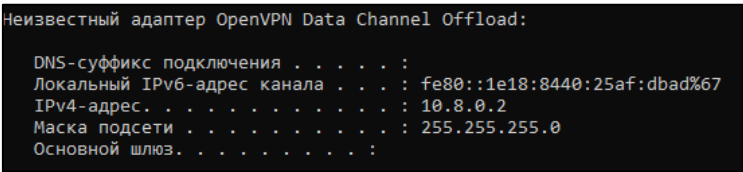


Рисунок 27. Сетевой адаптер OpenVPN

ВЫВОД:

Выполняя данное практическое задание, я приобрел необходимые практические навыки реализации VPN сервера и создания своей частной виртуальной сети.

Выполнил:	Студент группы ИСП-Б-о20 Кузин В.С.
Проверил:	Преподаватель СПО Миргородский А.И.