

Отчет по лабораторной работе №1

Дисциплина: Моделирование сетей передачи данных

Лобанова Полина Иннокентьевна

Содержание

1	Цель работы	6
2	Задание	7
3	Выполнение лабораторной работы	8
4	Выводы	26
	Список литературы	27

Список иллюстраций

3.1	Импортирование файла	8
3.2	изменение графического контроллера	9
3.3	Изменение первого адаптера	9
3.4	Изменение второго адаптера	9
3.5	Запуск виртуальной машины	10
3.6	Вход в виртуальную машину	10
3.7	Адрес машины	11
3.8	Подключение к виртуальной машине	11
3.9	Настройка ssh-подсоединения по ключу	12
3.10	IP-адреса машины	12
3.11	Активация второго интерфейса	13
3.12	Установка tc	13
3.13	Открытие файла	13
3.14	Изменения в файле /etc/netplan/01-netcfg.yaml	14
3.15	Переименовывание предыдущей установки Mininet	14
3.16	Скачивание новой версии Mininet	14
3.17	Обновление исполняемых файлов	15
3.18	Версия Mininet	15
3.19	Открытие файла	15
3.20	Изменения в файле /etc/X11/app-defaults/XTerm	16
3.21	Копирование значения куки пользователя mininet в файл для пользо- вателя root	16
3.22	Запуск минимальной топологии	17
3.23	Команда help	17
3.24	Доступные узлы и линки	18
3.25	Интерфейсы хоста h1	18
3.26	Интерфейсы хоста h2	19
3.27	Интерфейсы хоста s1	19
3.28	Пингование	20
3.29	Запуск MiniEdit	20
3.30	Схема сети	20
3.31	IP-адрес на хосте h1	21
3.32	IP-адрес на хосте h2	22
3.33	IP-адреса на хосте h1	22
3.34	IP-адреса на хосте h2	23
3.35	Пингование	23
3.36	Изменение базового значения IP-адресов	24

3.37 IP-адреса на хосте $h1$	24
3.38 Создание каталога	24
3.39 Сохранение топологии	25
3.40 Изменение прав доступа	25

Список таблиц

1 Цель работы

Основной целью работы является развёртывание в системе виртуализации (например, в VirtualBox) mininet, знакомство с основными командами для работы с Mininet через командную строку через графический интерфейс.

2 Задание

1. Выполнить настройку стенда виртуальной машины Mininet.
2. Выполнить работу с Mininet с помощью командной строки.
3. Выполнить построение и эмуляции сети в Mininet с использованием графического интерфейса.

3 Выполнение лабораторной работы

1. Скачала актуальный релиз ovf-образа виртуальной машины. Переместила скачанный образ в каталог для работы, затем распаковала его. Запустила систему виртуализации и импортировала файл .ovf.

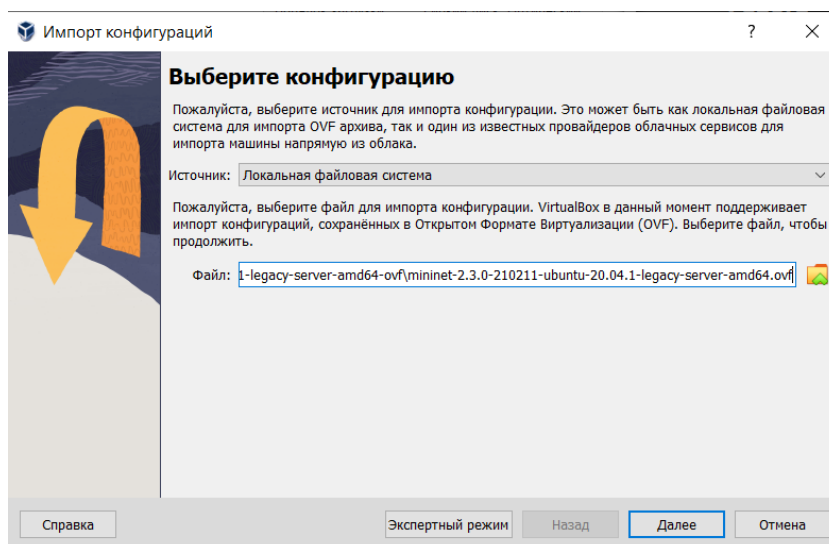


Рис. 3.1: Импорт файла

2. Перешла в настройки системы виртуализации и уточнила параметры настройки виртуальной машины. Изменила тип графического контроллера на рекомендуемый. В настройках сети первый адаптер имеет подключение типа NAT. Для второго адаптера указала тип подключения host-only network adapter (виртуальный адаптер хоста).

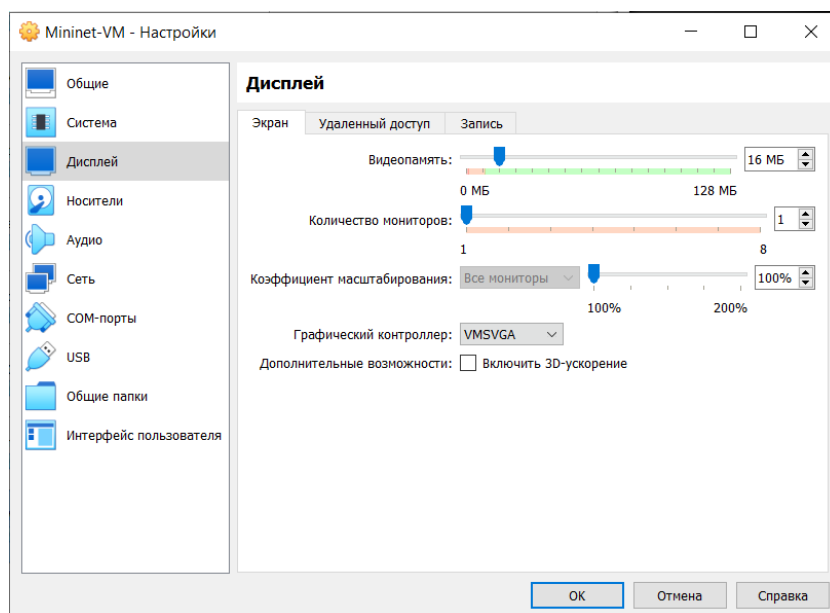


Рис. 3.2: изменение графического контроллера

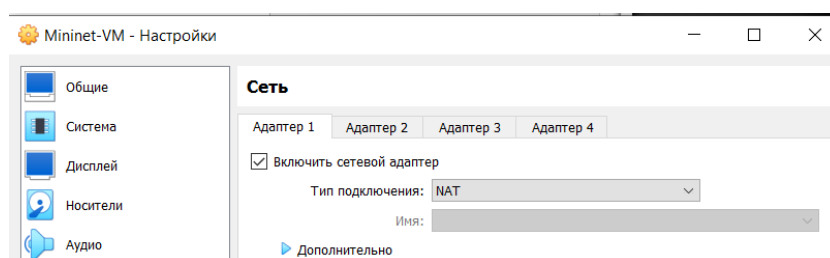


Рис. 3.3: Изменение первого адаптера

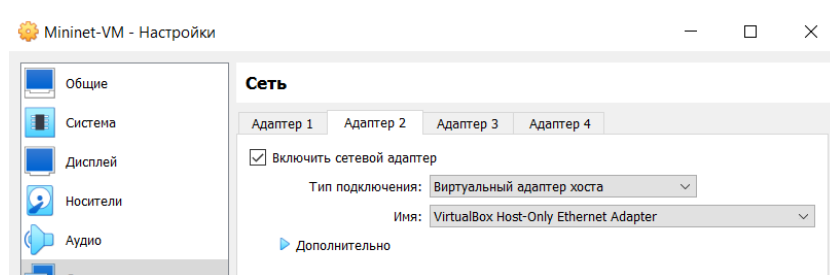


Рис. 3.4: Изменение второго адаптера

3. Запустила виртуальную машину с Mininet.

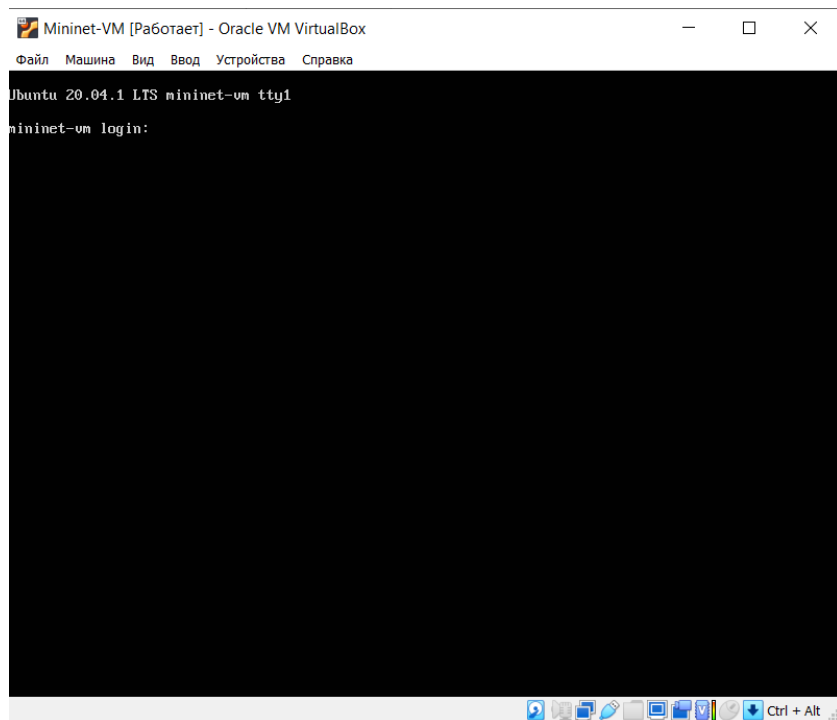


Рис. 3.5: *Запуск виртуальной машины*

4. Залогинилась в виртуальной машине.

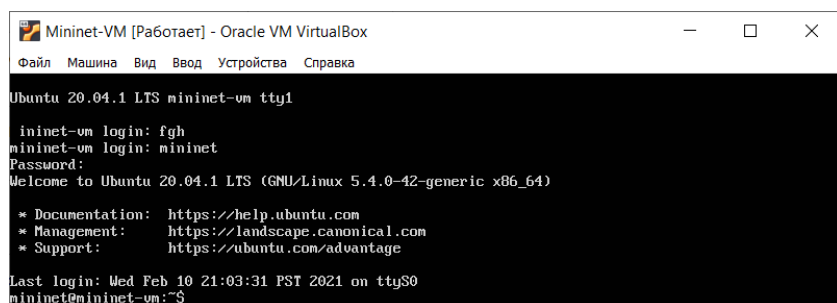


Рис. 3.6: *Вход в виртуальную машину*

5. Посмотрела адрес машины.

```

mininet@mininet-vm:~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.103 netmask 255.255.255.0 broadcast 192.168.56.255
    ether 08:00:27:11:9f:c6 txqueuelen 1000 (Ethernet)
    RX packets 9 bytes 1834 (1.8 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2 bytes 684 (684.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

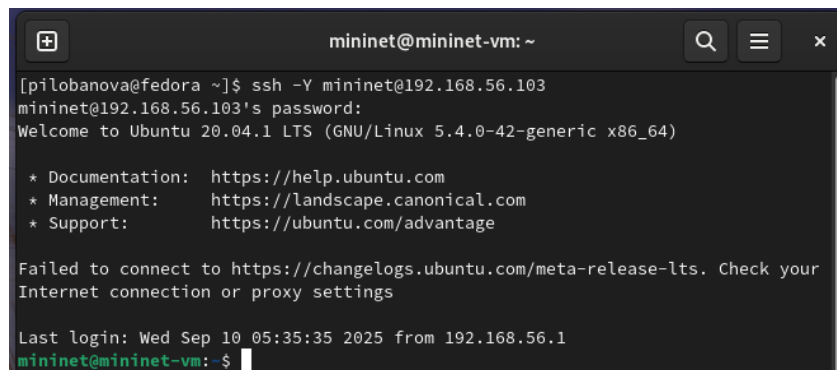
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    loop txqueuelen 1000 (Local Loopback)
    RX packets 56 bytes 4312 (4.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 56 bytes 4312 (4.3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

mininet@mininet-vm:~$ _

```

Рис. 3.7: Адрес машины

6. Подключилась к виртуальной машине.



```

mininet@mininet-vm: ~
[pilobanova@fedora ~]$ ssh -Y mininet@192.168.56.103
mininet@192.168.56.103's password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-42-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your
Internet connection or proxy settings

Last login: Wed Sep 10 05:35:35 2025 from 192.168.56.1
mininet@mininet-vm: $

```

Рис. 3.8: Подключение к виртуальной машине

7. Настроила ssh-подсоединение по ключу к виртуальной машине. Вновь подключилась к виртуальной машине и убедилась, что подсоединение происходит успешно и без ввода пароля.

```
[pilobanova@fedora ~]$ ssh-copy-id mininet@192.168.56.103
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompt
ed now it is to install the new keys
mininet@192.168.56.103's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'mininet@192.168.56.103'"
and check to make sure that only the key(s) you wanted were added.

[pilobanova@fedora ~]$ ssh -Y mininet@192.168.56.103
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-42-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your
Internet connection or proxy settings

Last login: Wed Sep 10 05:36:30 2025 from 192.168.56.1
mininet@mininet-vm:~$
```

Рис. 3.9: *Настройка ssh-подсоединения по ключу*

8. После подключения к виртуальной машине mininet посмотрела IP-адреса машины.

```
mininet@mininet-vm:~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.56.103  netmask 255.255.255.0  broadcast 192.168.56.255
    ether 08:00:27:11:9f:c6  txqueuelen 1000  (Ethernet)
    RX packets 179  bytes 34777 (34.7 KB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 172  bytes 30571 (30.5 KB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
    inet 127.0.0.1  netmask 255.0.0.0
    loop txqueuelen 1000  (Local Loopback)
    RX packets 888  bytes 68032 (68.0 KB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 888  bytes 68032 (68.0 KB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
```

Рис. 3.10: *IP-адреса машины*

9. Для доступа к сети Интернет должен быть активен адрес NAT: 10.0.0.x. У меня был активен только внутренний адрес машины вида 192.168.x.y, поэтому я активировала второй интерфейс.

```
mininet@mininet-vm:~$ sudo dhclient eth1
mininet@mininet-vm:~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.103 netmask 255.255.255.0 broadcast 192.168.56.255
    ether 08:00:27:11:9f:c6 txqueuelen 1000 (Ethernet)
    RX packets 244 bytes 40185 (40.1 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 207 bytes 34821 (34.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    ether 08:00:27:df:ed:48 txqueuelen 1000 (Ethernet)
    RX packets 2 bytes 1180 (1.1 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2 bytes 684 (684.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Рис. 3.11: Активация второго интерфейса

10. Для удобства дальнейшей работы установила `mc`.

```
mininet@mininet-vm:~$ sudo apt install mc
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libssh2-1 mc-data unzip
Suggested packages:
  arj catdvi | texlive-binaries dbview djvulibre-bin epub-utils genisoimage gv
  imagemagick libaspell-dev links | w3m | lynx odt2txt poppler-utils python
  python-boto python-tz xpdf | pdf-viewer zip
The following NEW packages will be installed:
  libssh2-1 mc mc-data unzip
0 upgraded, 4 newly installed, 0 to remove and 84 not upgraded.
Need to get 1.986 kB of archives.
```

Рис. 3.12: Установка `mc`

11. Для удобства дальнейшей работы добавила для `mininet` указание на использование двух адаптеров при запуске. Для этого перешла в режим суперпользователя и внесла изменения в файл `/etc/netplan/01-netcfg.yaml`.

```
mininet@mininet-vm:~$ sudo mcedit /etc/netplan/01-netcfg.yaml
X11 connection rejected because of wrong authentication.
```

Рис. 3.13: Открытие файла

```

/etc/net~cfg.yaml [-M--] 6 L:[ 1+ 9 10/ 10] *(209 / 219b) 100 0x064 [*][X]
# This file describes the network interfaces available on your system
# For more information, see netplan(5).
network:
  version: 2
  renderer: networkd
  ethernets:
    eth0:
      dhcp4: yes
    eht1:
      dhcp4: yes

```

Рис. 3.14: Изменения в файле */etc/netplan/01-netcfg.yaml*

12. В виртуальной машине mininet переименовала предыдущую установку Mininet.

```

mininet@mininet-vm:~$ mv ~/mininet ~/mininet.orig
mininet@mininet-vm:~$

```

Рис. 3.15: Переименовывание предыдущей установки Mininet

13. Скачала новую версию Mininet.

```

mininet@mininet-vm:~$ git clone https://github.com/mininet/mininet.git
Cloning into 'mininet'...
remote: Enumerating objects: 10388, done.
remote: Counting objects: 100% (128/128), done.
remote: Compressing objects: 100% (59/59), done.
remote: Total 10388 (delta 102), reused 69 (delta 69), pack-reused 10260 (from 3)
Receiving objects: 100% (10388/10388), 3.36 MiB | 6.22 MiB/s, done.
Resolving deltas: 100% (6906/6906), done.

```

Рис. 3.16: Скачивание новой версии Mininet

14. Обновила исполняемые файлы.

```

mininet@mininet-vm:~$ cd ~/mininet
mininet@mininet-vm:~/mininet$ sudo make install
cc -Wall -Wextra \
-DVERSION=\"`PYTHONPATH=. python -B bin/mn --version 2>&1`\" mnexec.c -o mnexec
install -D mnexec /usr/bin/mnexec
PYTHONPATH=. help2man -N -n "create a Mininet network." \
--no-discard-stderr "python -B bin/mn" -o mn.1
help2man -N -n "execution utility for Mininet." \
-h "-h" -v "-v" --no-discard-stderr ./mnexec -o mnexec.1
install -D -t /usr/share/man/man1 mn.1 mnexec.1
python -m pip uninstall -y mininet || true
Found existing installation: mininet 2.3.0
Uninstalling mininet-2.3.0:
  Successfully uninstalled mininet-2.3.0
python -m pip install .
Processing /home/mininet/mininet
Requirement already satisfied: setuptools in /usr/lib/python3/dist-packages (from
mininet==2.3.1b4) (45.2.0)
Building wheels for collected packages: mininet

```

Рис. 3.17: Обновление исполняемых файлов

15. Проверила номер установленной версии mininet.

```

mininet@mininet-vm:~/mininet$ mn --version
2.3.1b4

```

Рис. 3.18: Версия Mininet

16. Для увеличения размера шрифта и применения векторных шрифтов вместо растровых внесла изменения в файл /etc/X11/app-defaults/XTerm.

```

mininet@mininet-vm:~/mininet$ sudo mcedit /etc/X11/app-defaults/XTerm

```

Рис. 3.19: Открытие файла

```
/etc/X11~s/XTerm [-M--] 18 L:[246+21 267/267] *(10377/10377b) <EOF> [*][X]
!*on2Clicks: regex [[[:alpha:]]+://([[:alnum:]]!#+,./=?@_~)](%%[[:xdigit:]][:xdig

! VT100s and similar terminals recognize escape sequences and control
! characters to which they reply to the host with other escape sequences,
! to provide information. The "resize" program uses this feature.
!
! In addition, xterm recognizes several escape sequences which can be used to
! set fonts, window properties, return settings via escape sequences. Some
! find these useful; others are concerned with the possibility of unexpected
! inputs.
!
! All of these features can be enabled or disabled via menus.
!
! Depending on your environment, you may wish to disable those by default by
! uncommenting one or more of the resource settings below:
!*allowFontOps: false
!*allowTcapOps: false
!*allowTitleOps: false
!*allowWindowOps: false

xterm*faceName: Monospace
xterm*faceSize: 12
1Help 2Save 3Mark 4Replac 5Copy 6Move 7Search 8Delete 9PullDn10Quit
```

Рис. 3.20: Изменения в файле `/etc/X11/app-defaults/XTerm`

17. Скопировала значение куки пользователя `mininet` в файл для пользователя `root`. После выполнения этих действий графические приложения должны запускаться под пользователем `mininet`.

```
mininet@mininet-vm:~/mininet$ cd
mininet@mininet-vm:~$ xauth list $DISPLAY
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 1ed5b7073481df888a415b284e6973a6
mininet@mininet-vm:~$ sudo -i
root@mininet-vm:~# xauth list
xauth: file /root/.Xauthority does not exist
root@mininet-vm:~# xauth add mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 1ed5b7073481
df888a415b284e6973a6
xauth: file /root/.Xauthority does not exist
root@mininet-vm:~# xauth list $DISPLAY
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 1ed5b7073481df888a415b284e6973a6
root@mininet-vm:~#
```

Рис. 3.21: Копирование значения куки пользователя `mininet` в файл для пользователя `root`

18. Запустила минимальную топологию.


```

mininet@mininet-vm:~$ sudo mn
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet>

```

Рис. 3.22: Запуск минимальной топологии

19. Отобразила список команд интерфейса командной строки Mininet и примеров их использования.

```

mininet> help

Documented commands (type help <topic>):
=====
EOF      gterm  iperfudp  nodes      pingpair    py      switch  xterm
dpctl    help   link      noecho     pingpairfull  quit    time
dump     intfs  links     pingall    ports       sh      wait
exit     iperf  net       pingallfull  px          source  x

You may also send a command to a node using:
  <node> command {args}
For example:
  mininet> h1 ifconfig

The interpreter automatically substitutes IP addresses
for node names when a node is the first arg, so commands
like
  mininet> h2 ping h3
should work.

Some character-oriented interactive commands require
noecho:
  mininet> noecho h2 vi foo.py
However, starting up an xterm/gterm is generally better:

```

Рис. 3.23: Команда help

20. Отобразила доступные узлы и линки.

```
mininet> nodes
available nodes are:
c0 h1 h2 s1
mininet> net
h1 h1-eth0:s1-eth1
h2 h2-eth0:s1-eth2
s1 lo: s1-eth1:h1-eth0 s1-eth2:h2-eth0
c0
mininet>
```

Рис. 3.24: Доступные узлы и линки

21. Посмотрела интерфейсы хоста h1 — хост h1 имеет интерфейс h1-eth0, настроенный с IP-адресом 10.0.0.1, и другой интерфейс lo, настроенный с IP-адресом 127.0.0.1.

```
mininet> h1 ifconfig
h1-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.1 netmask 255.0.0.0 broadcast 10.255.255.255
    ether 76:de:0e:96:ea:53 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Рис. 3.25: Интерфейсы хоста h1

22. Посмотрела интерфейсы хоста h2 — хост h2 имеет интерфейс h2-eth0, настроенный с IP-адресом 10.0.0.2, и другой интерфейс lo, настроенный с IP-адресом 127.0.0.1.

```

mininet> h2 ifconfig
h2-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.2 netmask 255.0.0.0 broadcast 10.255.255.255
    ether 32:d6:69:7f:1f:eb txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

Рис. 3.26: Интерфейсы хоста h2

23. Посмотрела интерфейсы хоста s1 — хост s1 имеет интерфейс s1-eth0, настроенный с IP-адресом 192.168.56.103, интерфейс s1-eth1, настроенный с IP-адресом 10.0.2.15, и другой интерфейс lo, настроенный с IP-адресом 127.0.0.1.

```

mininet> s1 ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.103 netmask 255.255.255.0 broadcast 192.168.56.255
    ether 08:00:27:11:9f:c6 txqueuelen 1000 (Ethernet)
    RX packets 1816 bytes 164605 (164.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1473 bytes 235827 (235.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    ether 08:00:27:df:ed:48 txqueuelen 1000 (Ethernet)
    RX packets 4362 bytes 5897177 (5.8 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2276 bytes 151156 (151.1 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    loop txqueuelen 1000 (Local Loopback)
    RX packets 1266 bytes 92746 (92.7 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1266 bytes 92746 (92.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

s1-eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    ether e6:8b:4a:ba:b5:9f txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

s1-eth2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    ether d2:02:bf:db:9b:53 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

mininet>

```

Рис. 3.27: Интерфейсы хоста s1

24. Чтобы проверить связь между узлами, использовала команду ping.

```
mininet> h1 ping 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=1.65 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.150 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.035 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.037 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.040 ms
^C
--- 10.0.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4063ms
rtt min/avg/max/mdev = 0.035/0.382/1.652/0.636 ms
mininet>
```

Рис. 3.28: Пингование

25. Остановила эмуляцию.

26. В терминале виртуальной машины mininet запустила MiniEdit.

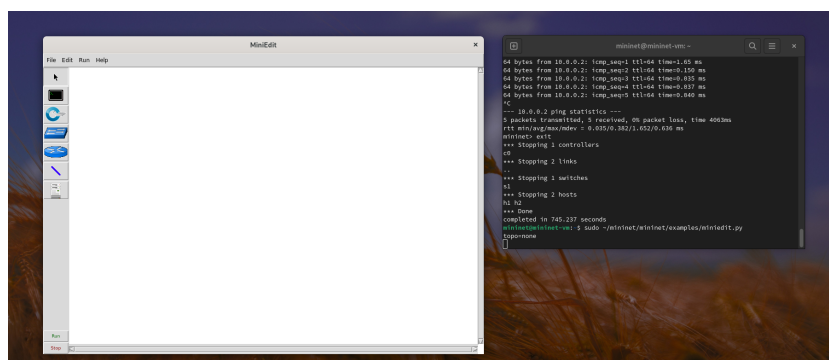


Рис. 3.29: Запуск MiniEdit

27. Добавила два хоста и один коммутатор, соединила хосты с коммутатором.

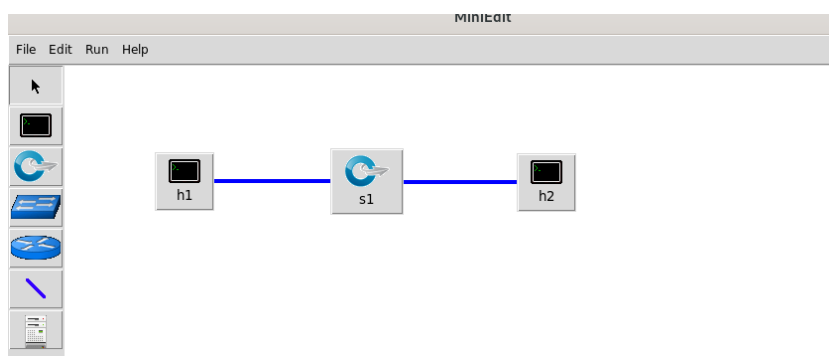


Рис. 3.30: Схема сети

28. Настроила IP-адреса на хостах h1 и h2. Для хоста h1 указала IP-адрес 10.0.0.1/8, а для хоста h2 — 10.0.0.2/8.

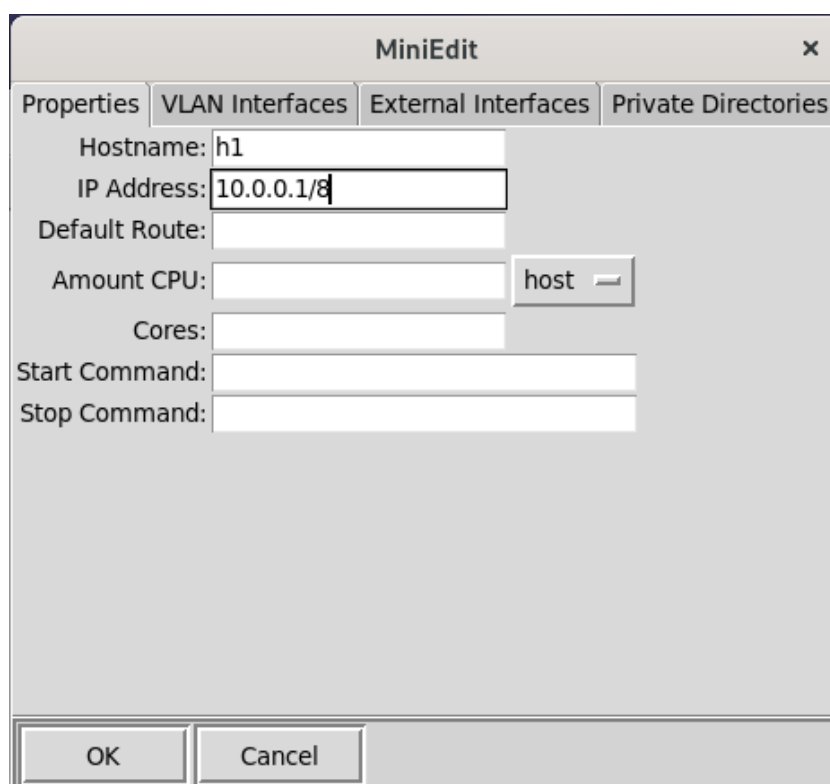


Рис. 3.31: IP-адрес на хосте h1

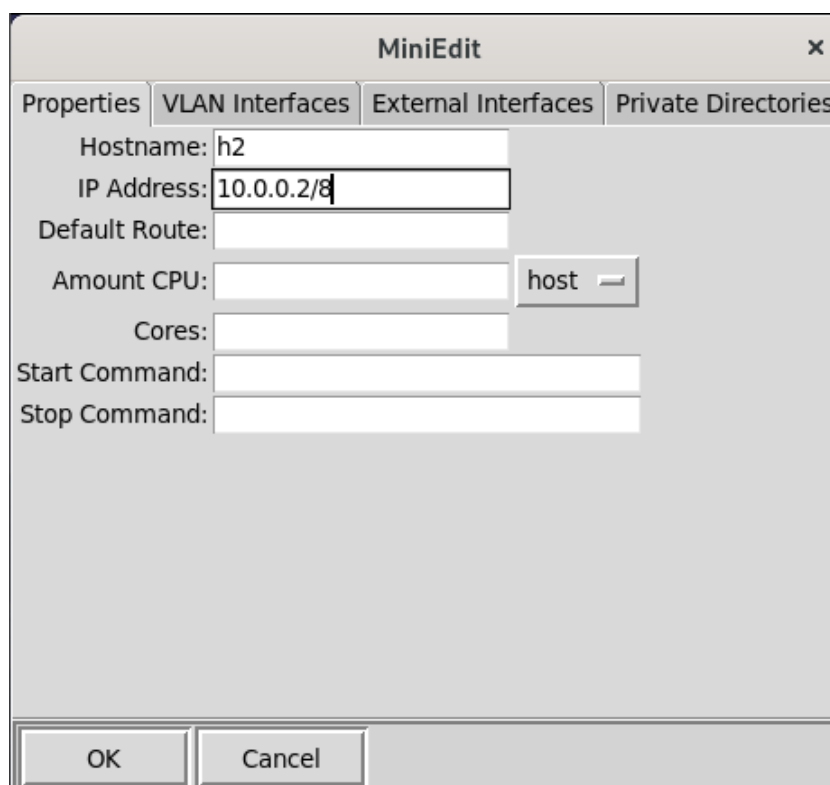


Рис. 3.32: IP-адрес на хосте h2

29. Запустила эмуляцию. Открыла терминал на хосте h1 и ввела команду `ifconfig`, чтобы отобразить назначенные ему IP-адреса.

```

"Host: h1" (на mininet-vm)
root@mininet-vm:/home/mininet# ifconfig
h1-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.1 netmask 255.0.0.0 broadcast 10.255.255.255
    ether 66:fb:de:22:03:a5 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    loop txqueuelen 1000 (Local Loopback)
    RX packets 834 bytes 262256 (262.2 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 834 bytes 262256 (262.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

Рис. 3.33: IP-адреса на хосте h1

30. Повторила эти действия на хосте h2.

```
"Host: h2" (на mininet-vm) x
root@mininet-vm:/home/mininet# ifconfig
h2-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.2 netmask 255.0.0.0 broadcast 10.255.255.255
    ether 9a:07:86:1b:24:2d txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    loop txqueuelen 1000 (Local Loopback)
    RX packets 1068 bytes 273420 (273.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1068 bytes 273420 (273.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Рис. 3.34: IP-адреса на хосте h2

31. Проверила соединение между хостами.

```
"Host: h1" (на mininet-vm) x
root@mininet-vm:/home/mininet# ping 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.349 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.043 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.038 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.054 ms
^C
--- 10.0.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3075ms
rtt min/avg/max/mdev = 0.038/0.121/0.349/0.131 ms
root@mininet-vm:/home/mininet# █
```

Рис. 3.35: Пингование

32. Остановила эмуляцию.

33. Удалила назначенный вручную IP-адрес с хостов h1 и h2.

34. В MiniEdit нажала Edit > Preferences. По умолчанию в поле базовые значения IP-адресов (IP Base) установлено 10.0.0.0/8. Изменила это значение на 15.0.0.0/8.

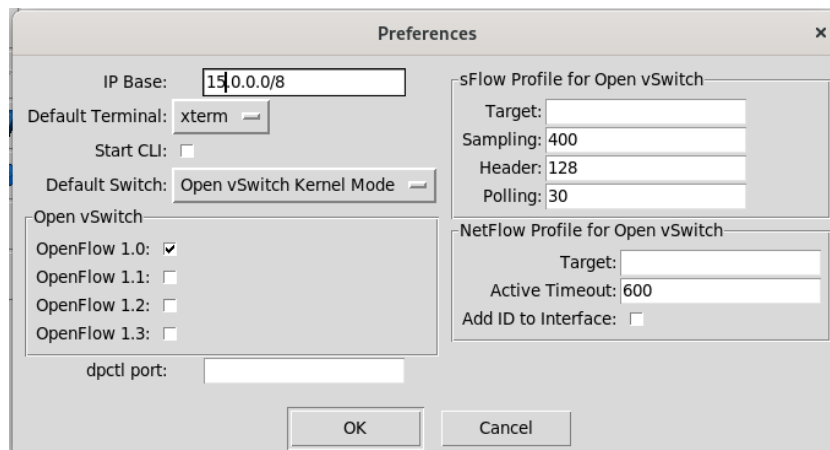


Рис. 3.36: Изменение базового значения IP-адресов

35. Запустила эмуляцию. Открыла терминал на хосте h1 и отобразила IP-адреса, назначенные хосту h1.

```

"Host: h1" (на mininet-vm)
root@mininet-vm:/home/mininet# ifconfig
h1-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 15.0.0.1 netmask 255.0.0.0 broadcast 15.255.255.255
    ether be:3f:d1:1c:cb:94 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    loop txqueuelen 1000 (Local Loopback)
    RX packets 886 bytes 260032 (260.0 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 886 bytes 260032 (260.0 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@mininet-vm:/home/mininet#

```

Рис. 3.37: IP-адреса на хосте h1

36. Остановила эмуляцию. В домашнем каталоге виртуальной машины mininet создала каталог для работы с проектами mininet.

```

mininet@mininet-vm:~$ mkdir ~/work
mininet@mininet-vm:~$

```

Рис. 3.38: Создание каталога

37. Для сохранения топологии сети в файл нажала в MiniEdit File > Save. Указала имя для топологии и сохранила на своём компьютере.

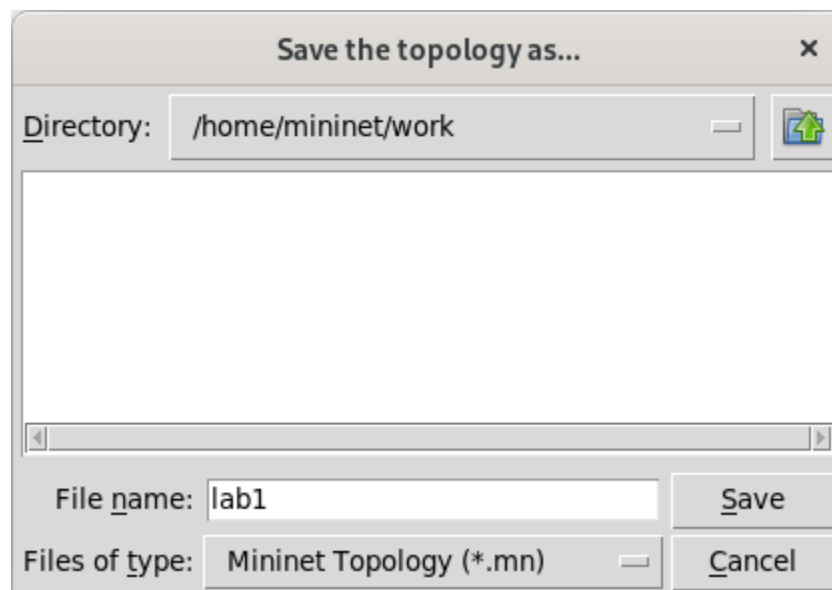


Рис. 3.39: Сохранение топологии

38. После сохранения проекта поменяла права доступа к файлам в каталоге проекта.

```
mininet@mininet-vm:~$ sudo chown -R mininet:mininet ~/work
mininet@mininet-vm:~$
```

Рис. 3.40: Изменение прав доступа

39. Завершила соединение с виртуальной машиной mininet и выключила её.

4 Выводы

Я выполнила развёртывание в системе виртуализации mininet и ознакомилась с основными командами для работы с Mininet через командную строку через графический интерфейс.

Список литературы