

# **Отчёт по лабораторной работе №1 по дисциплине Компьютерный практикум по статистическому анализу данных**

**Введение в Mininet**

Шаповалова Диана Дмитриевна, НПИбд-02-21, 1032211220

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
2.1	Настройка образа VirtualBox . . . . .	6
2.2	Подключение к виртуальной машине . . . . .	8
2.3	Настройка tc . . . . .	10
2.4	Настройка параметров XTerm . . . . .	11
2.5	Настройка соединения X11 для суперпользователя . . . . .	12
2.6	Работа с Mininet из-под Windows . . . . .	13
2.7	Основы работы в Mininet . . . . .	15
2.8	Построение и эмуляция сети в Mininet с использованием графического интерфейса . . . . .	18
<b>3</b>	<b>Выводы</b>	<b>24</b>
<b>4</b>	<b>Список литературы</b>	<b>25</b>

# Список иллюстраций

2.1	Настройка виртуальной машины . . . . .	7
2.2	Настройка виртуальной машины . . . . .	8
2.3	Логинимся в виртуальной машине . . . . .	9
2.4	Настраиваем подключение по ключу, снова подключаемся . . . .	10
2.5	Устанавливаем ms . . . . .	11
2.6	Редактируем файл /etc/X11/app-defaults/XTerm . . . . .	12
2.7	Настраиваем соединение X11 . . . . .	13
2.8	Устанавливаем putty . . . . .	14
2.9	Устанавливаем VcXsrv . . . . .	14
2.10	Выполняем команды . . . . .	16
2.11	Выполняем команды . . . . .	17
2.12	Выполняем команды . . . . .	18
2.13	Открываем Miniedit, строим топологию . . . . .	19
2.14	Настраиваем IP-адрес h1, по аналогии и у h2 . . . . .	20
2.15	Проверяем соединение между хостами . . . . .	21
2.16	Настраиваем автоматическое назначение IP-адресов . . . . .	22
2.17	Проверяем IP-адрес у h1 . . . . .	22
2.18	Создаем каталог, сохраняем файл . . . . .	23

## **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 и через графический интерфейс.

## 2 Выполнение лабораторной работы

### 2.1 Настройка образа VirtualBox

- Перейдите в репозиторий Mininet. Скачайте актуальный релиз ovf-образа виртуальной машины.
- При необходимости переместите скачанный образ в каталог для работы, затем распакуйте его.
- Запустите систему виртуализации и импортируйте файл .ovf.
- Перейдите в настройки системы виртуализации и уточните параметры настройки виртуальной машины. В частности, для VirtualBox выберите импортированную виртуальную машину и перейдите в меню Машина-Настроить. (рис. 2.1 - рис. 2.2).

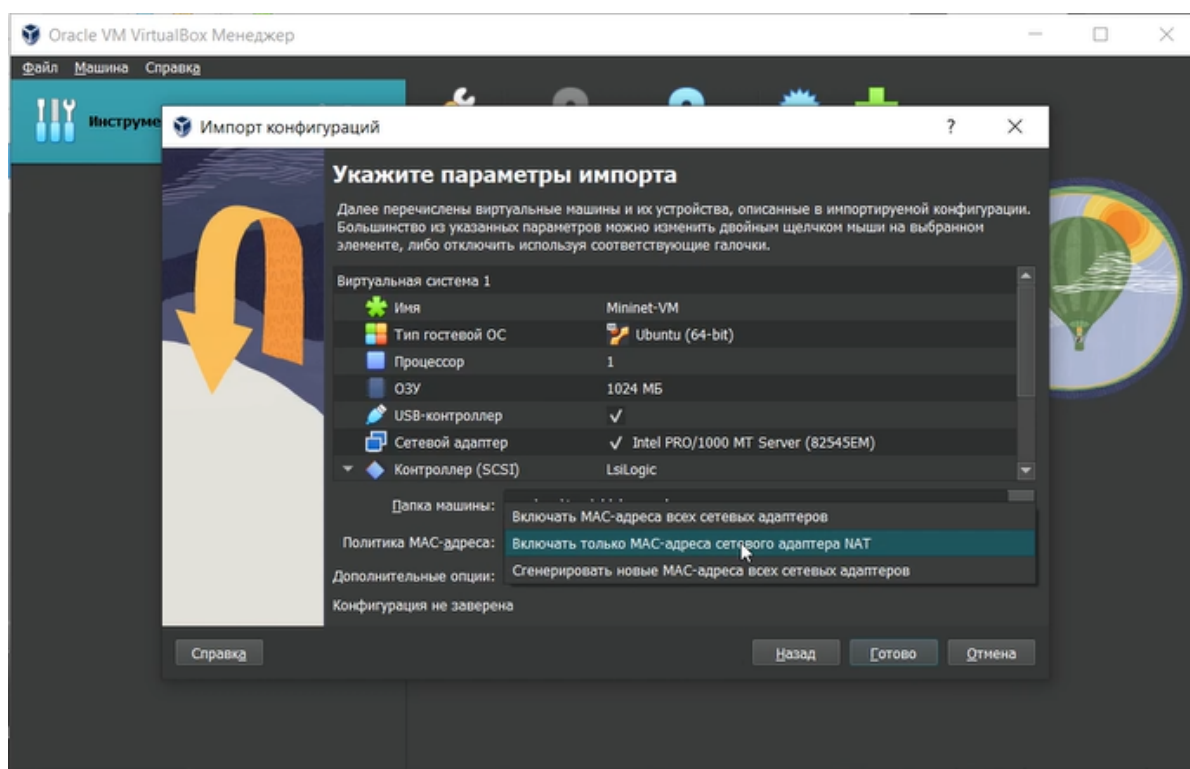


Рис. 2.1: Настройка виртуальной машины

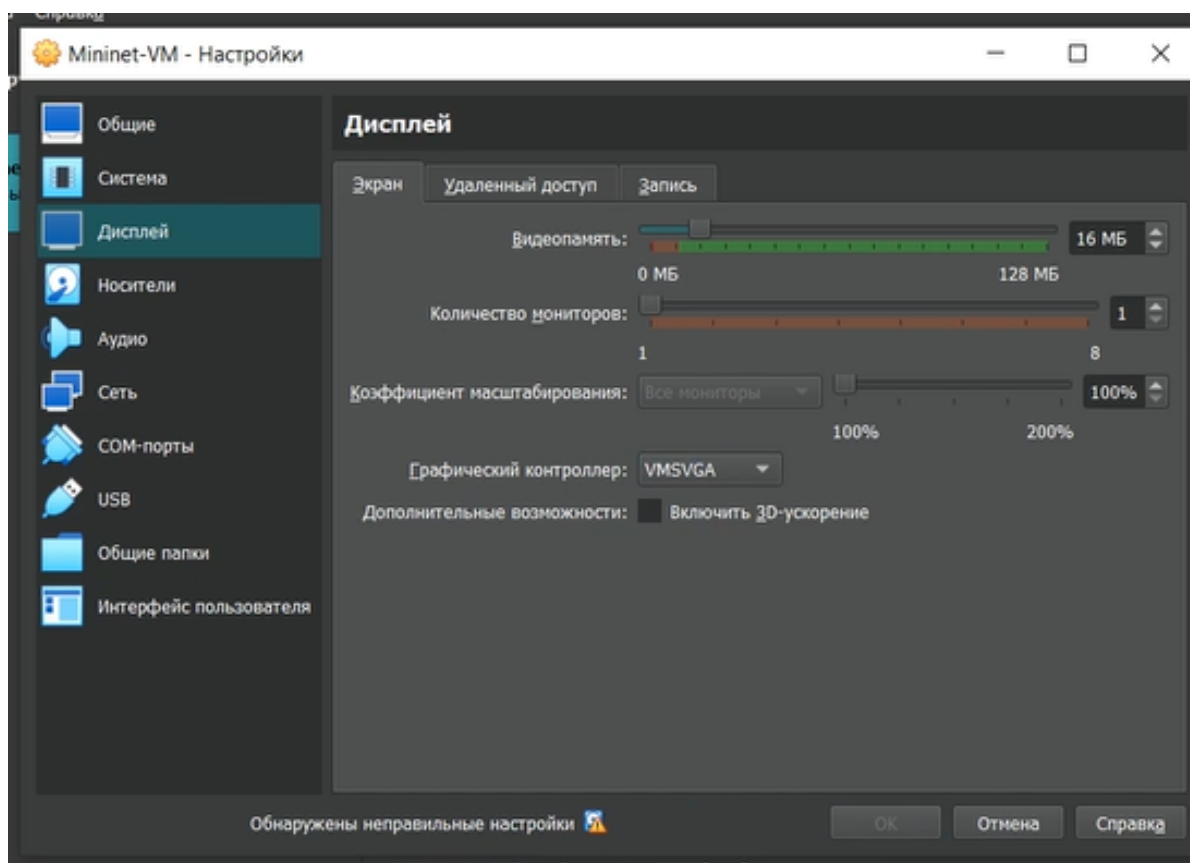


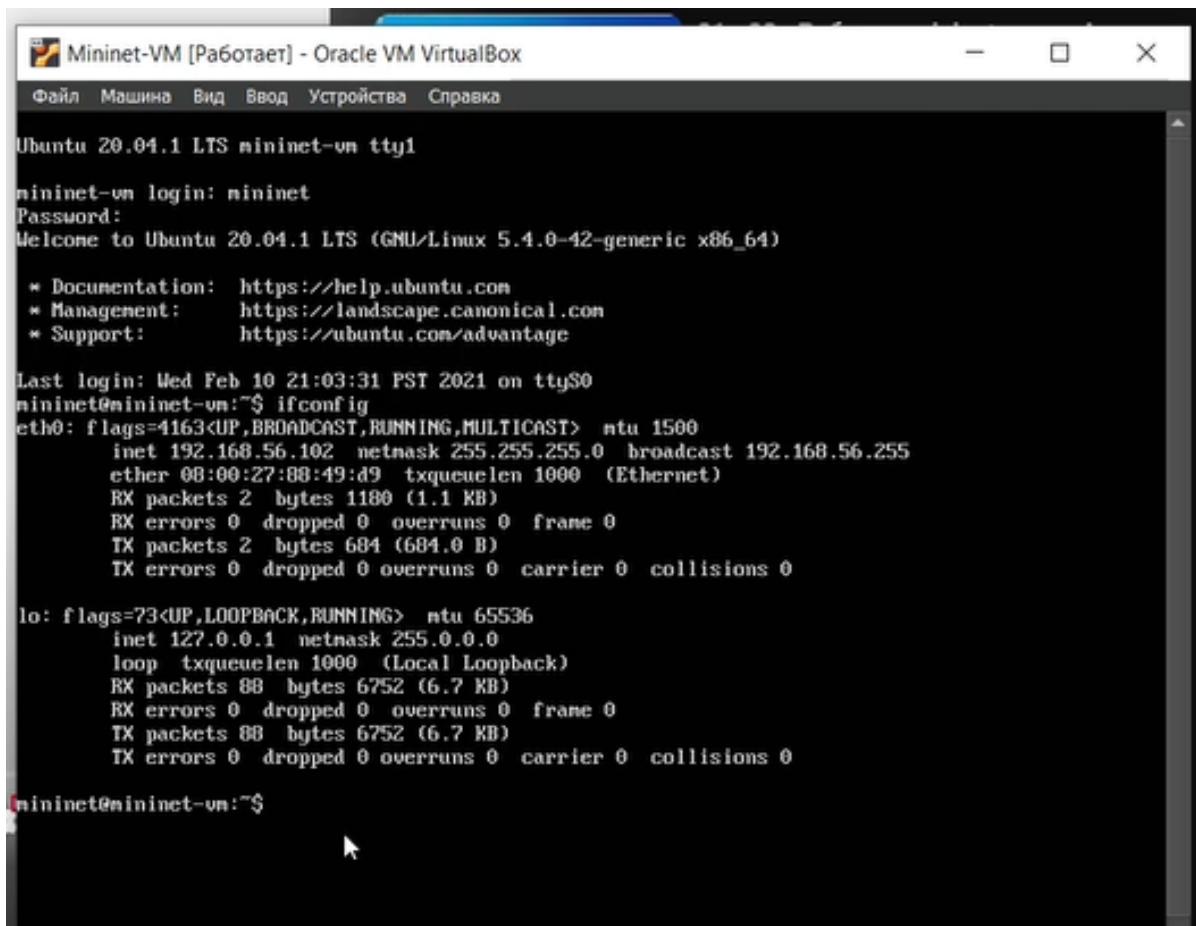
Рис. 2.2: Настройка виртуальной машины

- Запустите виртуальную машину с Mininet.

## 2.2 Подключение к виртуальной машине

- Залогиньтесь в виртуальной машине:
- login: mininet
- password: mininet
- Посмотрите адрес машины: ifconfig





```
Mininet-VM [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка

Ubuntu 20.04.1 LTS mininet-vm tty1

mininet-vm login: mininet
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

Last login: Wed Feb 10 21:03:31 PST 2021 on ttyS0
mininet@mininet-vm:~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.56.102  netmask 255.255.255.0  broadcast 192.168.56.255
    ether 08:00:27:88:49:d9  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

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 88  bytes 6752 (6.7 KB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 88  bytes 6752 (6.7 KB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

mininet@mininet-vm:~$
```

Рис. 2.3: Логинимся в виртуальной машине

- Подключитесь к виртуальной машине (из терминала хостовой машины): `ssh -Y mininet@192.168.x.y`
- Для отключения ssh-соединения с виртуальной машиной нажмите `Ctrl + d`.
- Настройте ssh-подсоединение по ключу к виртуальной машине, для чего в терминале основной Linux-машины перейдите в каталог `.ssh` своего домашнего каталога и введите (вместо `192.168.x.y` укажите внутренний адрес виртуальной машины Mininet): `ssh-copy-id mininet@192.168.x.y`
- Вновь подключитесь к виртуальной машине и убедитесь, что подсоединение происходит успешно и без ввода пароля.

```

mininet@192.168.56.102's password:
Permission denied, please try again.
mininet@192.168.56.102's password:
Permission denied, please try again.
mininet@192.168.56.102's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'mininet@192.168.56.102'"
and check to make sure that only the key(s) you wanted were added.

Di@LAPTOP-F7P8KGFO MINGW64 ~/.ssh
$ ssh -Y mininet@192.168.56.102
Warning: No xauth data; using fake authentication data for X11 forwarding.
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
Internet connection or proxy settings

Last login: Thu Nov 14 11:39:59 2024 from 192.168.56.1
mininet@mininet-vm:~$ logout
Connection to 192.168.56.102 closed.

Di@LAPTOP-F7P8KGFO MINGW64 ~/.ssh
$ ssh -Y mininet@192.168.56.102
Warning: No xauth data; using fake authentication data for X11 forwarding.
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
Internet connection or proxy settings

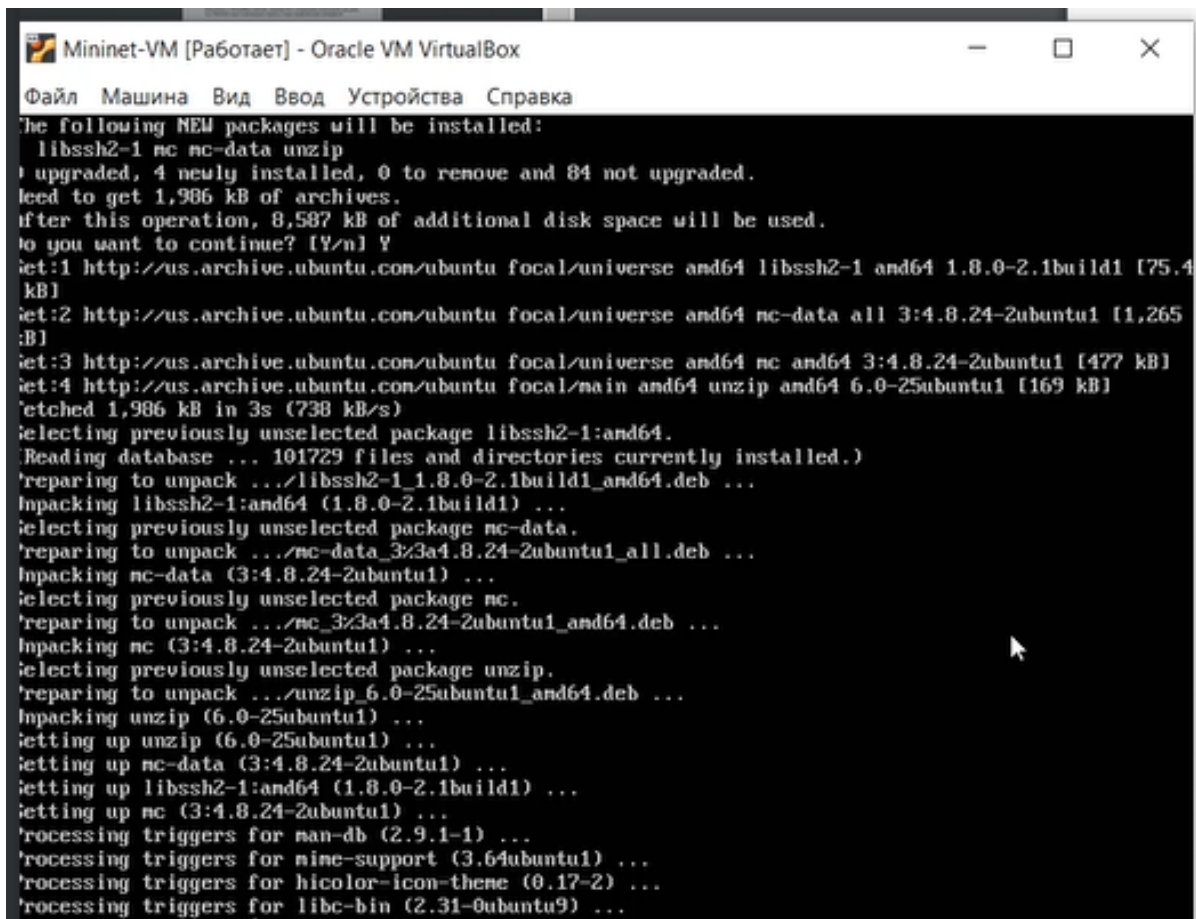
Last login: Thu Nov 14 11:43:27 2024 from 192.168.56.1
mininet@mininet-vm:~$ |

```

Рис. 2.4: Настраиваем подключение по ключу, снова подключаемся

## 2.3 Настройка mc

– Для удобства дальнейшей работы установите mc: `sudo apt install mc`



```
Mininet-VM [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
The following NEW packages will be installed:
  libssh2-1 nc nc-data unzip
0 upgraded, 4 newly installed, 0 to remove and 84 not upgraded.
Need to get 1,986 kB of archives.
After this operation, 8,587 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
get:1 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 libssh2-1 amd64 1.8.0-2.1build1 [75.4
kB]
get:2 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 nc-data all 3:4.8.24-2ubuntu1 [1,265
kB]
get:3 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 nc amd64 3:4.8.24-2ubuntu1 [477 kB]
get:4 http://us.archive.ubuntu.com/ubuntu focal/main amd64 unzip amd64 6.0-25ubuntu1 [169 kB]
Fetched 1,986 kB in 3s (738 kB/s)
Selecting previously unselected package libssh2-1:amd64.
Reading database ... 101729 files and directories currently installed.)
Preparing to unpack .../libssh2-1_1.8.0-2.1build1_amd64.deb ...
Unpacking libssh2-1:amd64 (1.8.0-2.1build1) ...
Selecting previously unselected package nc-data.
Preparing to unpack .../nc-data_3:4.8.24-2ubuntu1_all.deb ...
Unpacking nc-data (3:4.8.24-2ubuntu1) ...
Selecting previously unselected package nc.
Preparing to unpack .../nc_3:4.8.24-2ubuntu1_amd64.deb ...
Unpacking nc (3:4.8.24-2ubuntu1) ...
Selecting previously unselected package unzip.
Preparing to unpack .../unzip_6.0-25ubuntu1_amd64.deb ...
Unpacking unzip (6.0-25ubuntu1) ...
Setting up unzip (6.0-25ubuntu1) ...
Setting up nc-data (3:4.8.24-2ubuntu1) ...
Setting up libssh2-1:amd64 (1.8.0-2.1build1) ...
Setting up nc (3:4.8.24-2ubuntu1) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for mime-support (3.64ubuntu1) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for libc-bin (2.31-0ubuntu9) ...
```

Рис. 2.5: Устанавливаем nc

## 2.4 Настройка параметров XTerm

По умолчанию XTerm использует растровые шрифты малого кегля. Для увеличения размера шрифта и применения векторных шрифтов вместо растровых необходимо внести изменения в файл `/etc/X11/app-defaults/XTerm`. Для этого можно воспользоваться следующей командой: `sudo mcedit /etc/X11/app-defaults/XTerm`

и затем в конце файла добавить строки:

```
xterm*faceName: Monospace
```

```
xterm*faceSize: 12
```

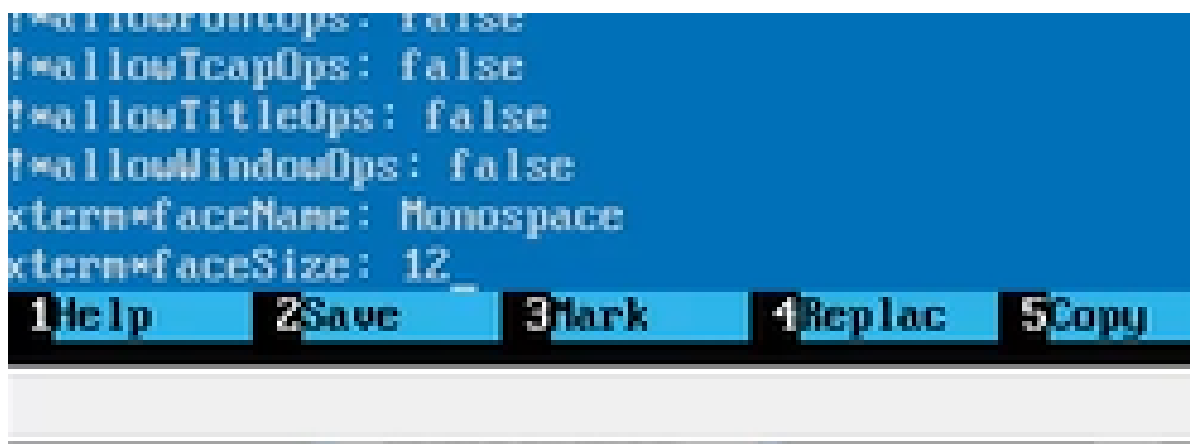


Рис. 2.6: Редактируем файл `/etc/X11/app-defaults/XTerm`

## 2.5 Настройка соединения X11 для суперпользователя

При попытке запуска приложения из-под суперпользователя возникает ошибка:

X11 connection rejected because of wrong authentication.

Ошибка возникает из-за того, что X-соединение выполняется от имени пользователя `mininet`, а приложение запускается от имени пользователя `root` с использованием `sudo`. Для исправления этой ситуации необходимо заполнить файл полномочий `/root/.Xauthority`, используя утилиту `xauth`. Скопируйте значение куки (MIT magic cookie)<sup>1</sup> пользователя `mininet` в файл для пользователя `root`

```

mininet@mininet-vm:~$ xauth list $DISPLAY
mininet-vm/unix:11 MIT-MAGIC-COOKIE-1 efec696b82b464b26d4631f8b6c173f4
mininet@mininet-vm:~$ ^C
mininet@mininet-vm:~$ mininet-vm/unix:11 MIT-MAGIC-COOKIE-1 efec696b82b464b26d4631f8b6c173f4
-bash: mininet-vm/unix:11: No such file or directory
mininet@mininet-vm:~$ xauth add mininet-vm/unix:11 MIT-MAGIC-COOKIE-1 efec696b82b464b26d4631f8b6c173f4
mininet@mininet-vm:~$ sudo -i
root@mininet-vm:~# xterm
PuTTY X11 proxy: No authorisation provided
Warning: This program is an suid-root program or is being run by the root user.
The full text of the error or warning message cannot be safely formatted
in this environment. You may get a more descriptive message by running the
program as a non-root user or by removing the suid bit on the executable.
xterm: Xt error: Can't open display: %s
root@mininet-vm:~# xauth add mininet-vm/unix:11 MIT-MAGIC-COOKIE-1 efec696b82b464b26d4631f8b6c173f4
root@mininet-vm:~# █

```

Рис. 2.7: Настраиваем соединение X11

## 2.6 Работа с Mininet из-под Windows

- Установите putty: choco install putty
- Установите VcXsrv Windows X Server: choco install vcxsrv



```

PS C:\Windows\system32> choco install putty -y
Chocolatey v2.3.0
Installing the following packages:
putty
By installing, you accept licenses for the packages.
Downloading package from source 'https://community.chocolatey.org/api/v2/'
Progress: Downloading putty.portable 0.81.0... 100%

putty.portable v0.81.0 [Approved]
putty.portable package files install completed. Performing other installation steps.
Extracting 64-bit C:\ProgramData\chocolatey\lib\putty.portable\tools\putty_x64.zip to C:\ProgramData\chocolatey\lib\putt
y.portable\tools...
C:\ProgramData\chocolatey\lib\putty.portable\tools
ShimGen has successfully created a gui shim for PAGEANT.EXE
ShimGen has successfully created a shim for PLINK.EXE
ShimGen has successfully created a shim for PSCP.EXE
ShimGen has successfully created a shim for PSFTP.EXE
ShimGen has successfully created a gui shim for PUTTY.EXE
ShimGen has successfully created a gui shim for PUTTYGEN.EXE
The install of putty.portable was successful.
Deployed to 'C:\ProgramData\chocolatey\lib\putty.portable\tools'
Downloading package from source 'https://community.chocolatey.org/api/v2/'
Progress: Downloading putty 0.81.0... 100%

putty v0.81.0 [Approved]
putty package files install completed. Performing other installation steps.
The install of putty was successful.
Deployed to 'C:\ProgramData\chocolatey\lib\putty'

Chocolatey installed 2/2 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).
PS C:\Windows\system32>

```

Рис. 2.8: Устанавливаем putty

```

PS C:\Windows\system32> choco install vcxsrv -y
Chocolatey v2.3.0
Installing the following packages:
vcxsrv
By installing, you accept licenses for the packages.
Downloading package from source 'https://community.chocolatey.org/api/v2/'
Progress: Downloading vcxsrv 21.1.10... 100%

vcxsrv v21.1.10 [Approved]
vcxsrv package files install completed. Performing other installation steps.
Installing 64-bit vcxsrv...
vcxsrv has been installed.
vcxsrv may be able to be automatically uninstalled.
The install of vcxsrv was successful.
Software installed as 'EXE', install location is likely default.

Chocolatey installed 1/1 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).
PS C:\Windows\system32>

```

Рис. 2.9: Устанавливаем VcXsrv

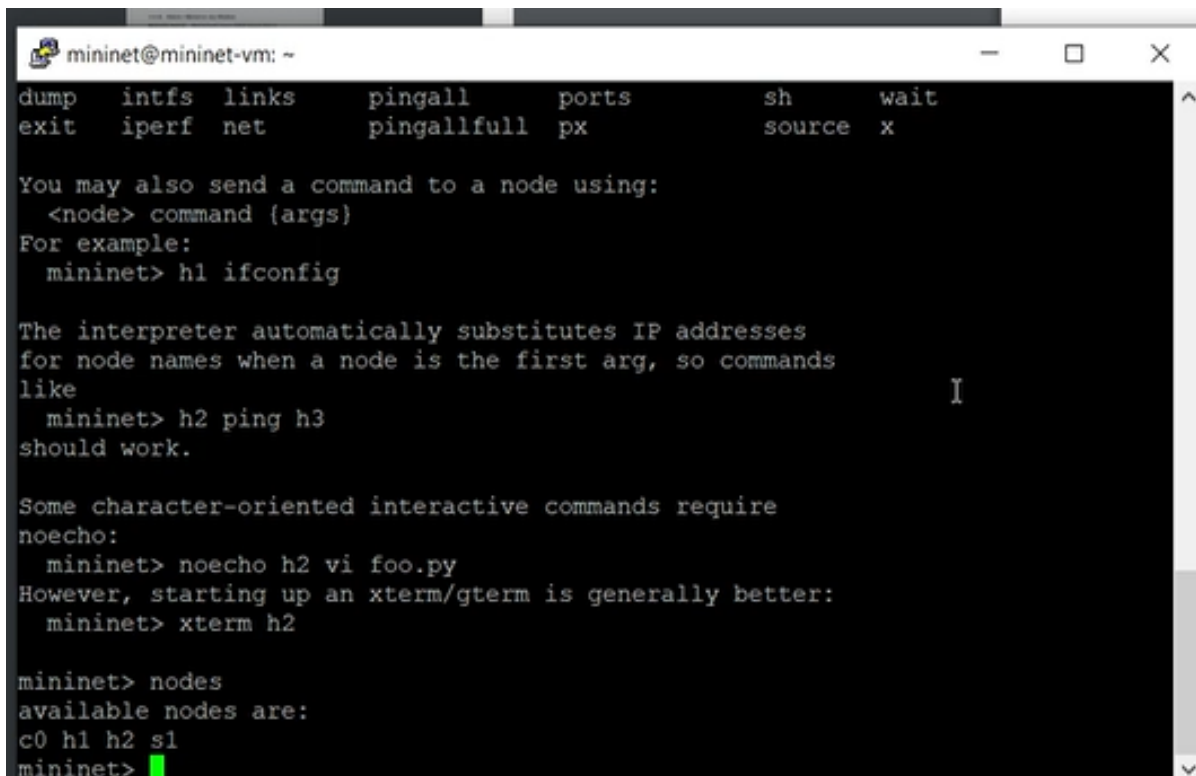
Запуск Xserver.

– Запустите XLaunch.

- Выберите опции:
- Multiple windows;
- Display number: -1;
- Start no client.
- Можно сохранить параметры, тогда при следующем запуске не нужно будет отмечать эти опции.

## 2.7 Основы работы в Mininet

- Для запуска минимальной топологии введите в командной строке: `sudo mn`  
Эта команда запускает Mininet с минимальной топологией, состоящей из коммутатора, подключённого к двум хостам.
- Для отображения списка команд интерфейса командной строки Mininet и примеров их использования введите команду в интерфейсе командной строки Mininet: `help`



```
mininet@mininet-vm: ~
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:
mininet> xterm h2

mininet> nodes
available nodes are:
c0 h1 h2 s1
mininet>
```

Рис. 2.10: Выполняем команды

– Для отображения доступных узлов введите: `nodes`

Вывод этой команды показывает, что есть два хоста (хост `h1` и хост `h2`) и коммутатор (`s1`).

– Иногда бывает полезно отобразить связи между устройствами в Mininet, чтобы понять топологию. Введите команду `net` в интерфейсе командной строки Mininet, чтобы просмотреть доступные линки: `net`

Вывод этой команды показывает:

– Хост `h1` подключён через свой сетевой интерфейс `h1-eth0` к коммутатору на интерфейсе `s1-eth1`.

– Хост `h2` подключён через свой сетевой интерфейс `h2-eth0` к коммутатору на интерфейсе `s1-eth2`.

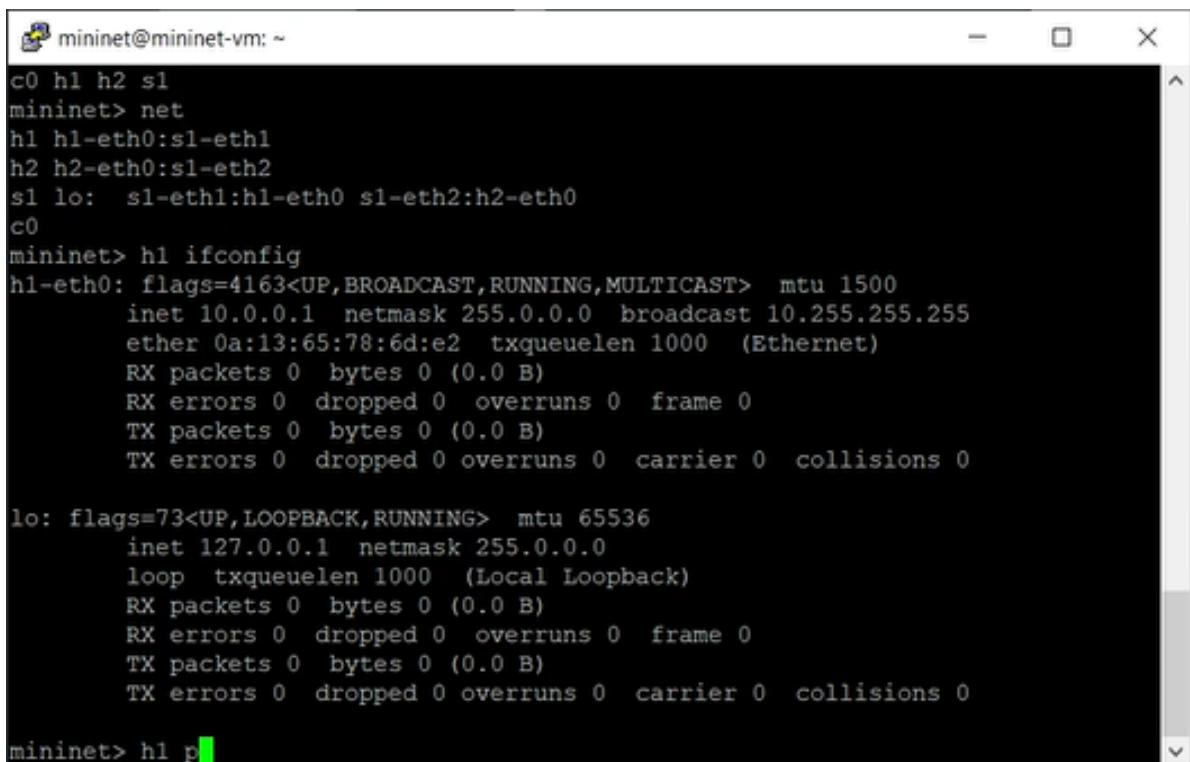
– Коммутатор `s1`:

– имеет петлевой интерфейс `lo`.

– подключается к `h1-eth0` через интерфейс `s1-eth1`.



– подключается к h2-eth0 через интерфейс s1-eth2.



```
mininet@mininet-vm: ~
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> 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 0a:13:65:78:6d:e2 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

mininet> h1 p
```

Рис. 2.11: Выполняем команды

– Mininet позволяет выполнять команды на конкретном устройстве. Чтобы выполнить команду для определенного узла, необходимо сначала указать устройство, а затем команду, например: h1 ifconfig

Эта запись выполняет команду ifconfig на хосте h1 и показывает 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

Проверка связности.

По умолчанию узлам h1 и h2 назначаются IP-адреса 10.0.0.1/8 и 10.0.0.2/8 соответственно. Чтобы проверить связь между ними, вы можете использовать команду ping. Команда ping работает, отправляя сообщения эхо-запроса протокола управляющих сообщений Интернета (ICMP) на удалённый компьютер и ожидая ответа.

Например, команда h1 ping 10.0.0.2 проверяет соединение между хостами h1 и

h2. Для остановки теста нажмите Ctrl + c.

Остановка эмуляции. Для завершения работы режима эмуляции mininet используйте команду exit

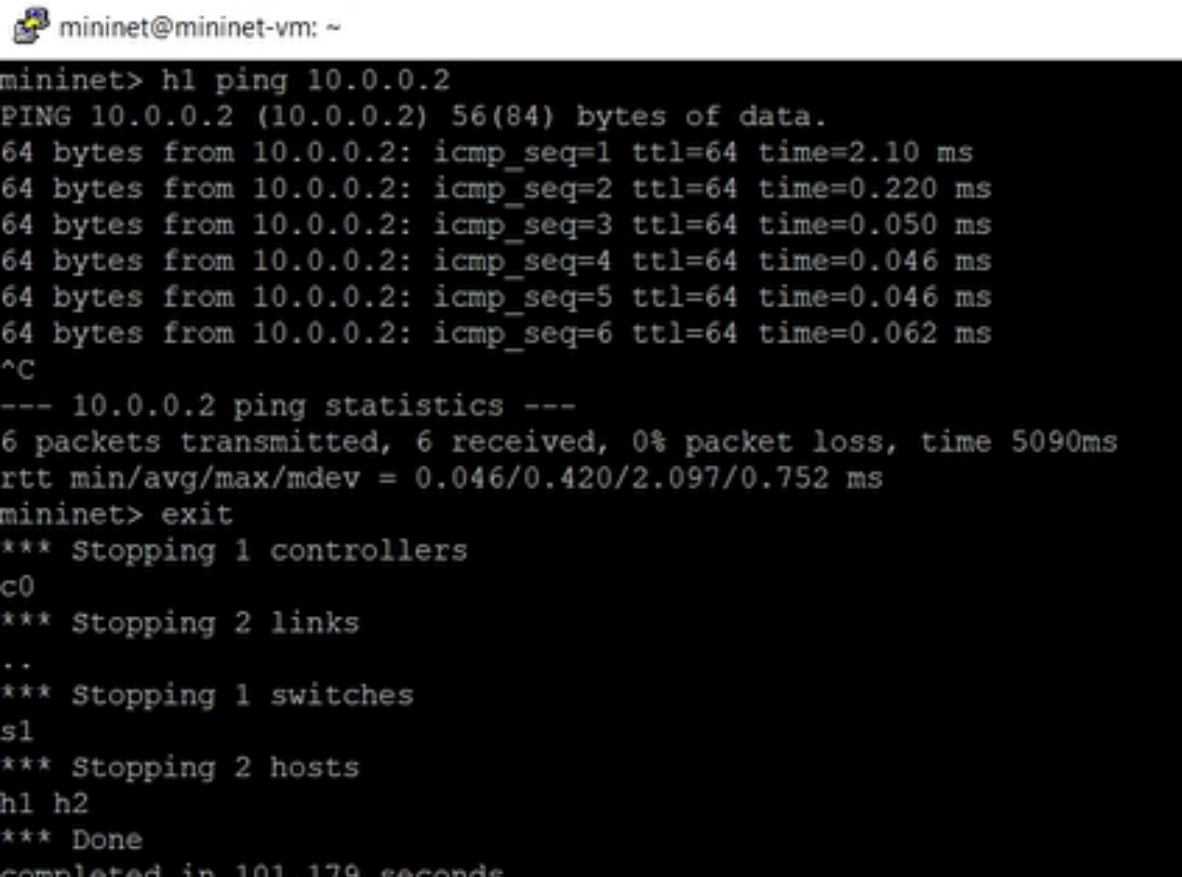
A screenshot of a terminal window with a black background and white text. The prompt is 'mininet@mininet-vm: ~'. The user enters 'mininet> h1 ping 10.0.0.2'. The output shows a series of ping results for 10.0.0.2, each 64 bytes, with varying times. After six pings, the user presses Ctrl+C (^C). The terminal shows '--- 10.0.0.2 ping statistics ---' followed by statistics: '6 packets transmitted, 6 received, 0% packet loss, time 5090ms' and 'rtt min/avg/max/mdev = 0.046/0.420/2.097/0.752 ms'. Then the user enters 'mininet> exit'. The terminal shows a series of status messages: '\*\*\* Stopping 1 controllers', 'c0', '\*\*\* Stopping 2 links', '..', '\*\*\* Stopping 1 switches', 's1', '\*\*\* Stopping 2 hosts', 'h1 h2', '\*\*\* Done', and finally 'completed in 101.179 seconds'.

Рис. 2.12: Выполняем команды

## 2.8 Построение и эмуляция сети в Mininet с использованием графического интерфейса

- В терминале виртуальной машины mininet запустите MiniEdit: `sudo ~/mininet/mininet/examples/miniedit.py`
- Добавьте два хоста и один коммутатор, соедините хосты с коммутатором.

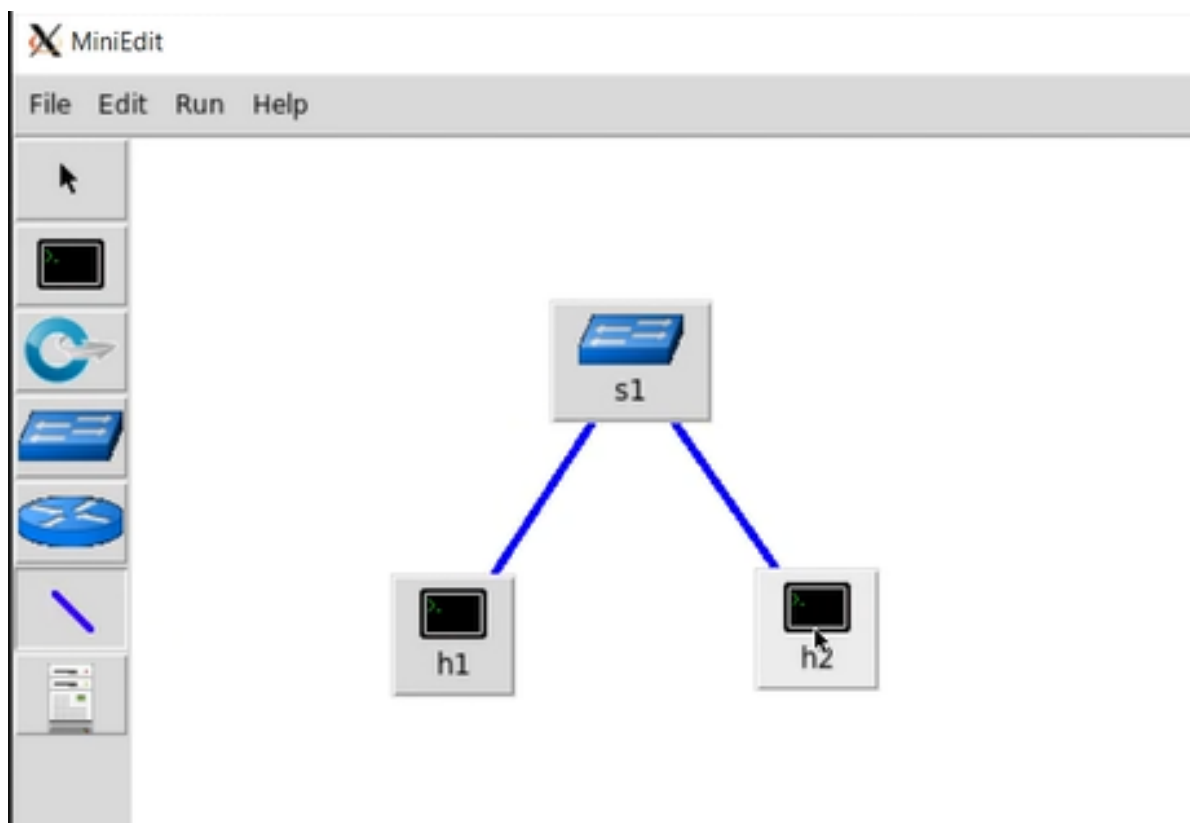


Рис. 2.13: Открываем Miniedit, строим топологию

– Настройте IP-адреса на хостах h1 и h2. Для этого удерживая правую кнопку мыши на устройстве выберите свойства. Для хоста h1 укажите IP-адрес 10.0.0.1/8, а для хоста h2 – 10.0.0.2/8.

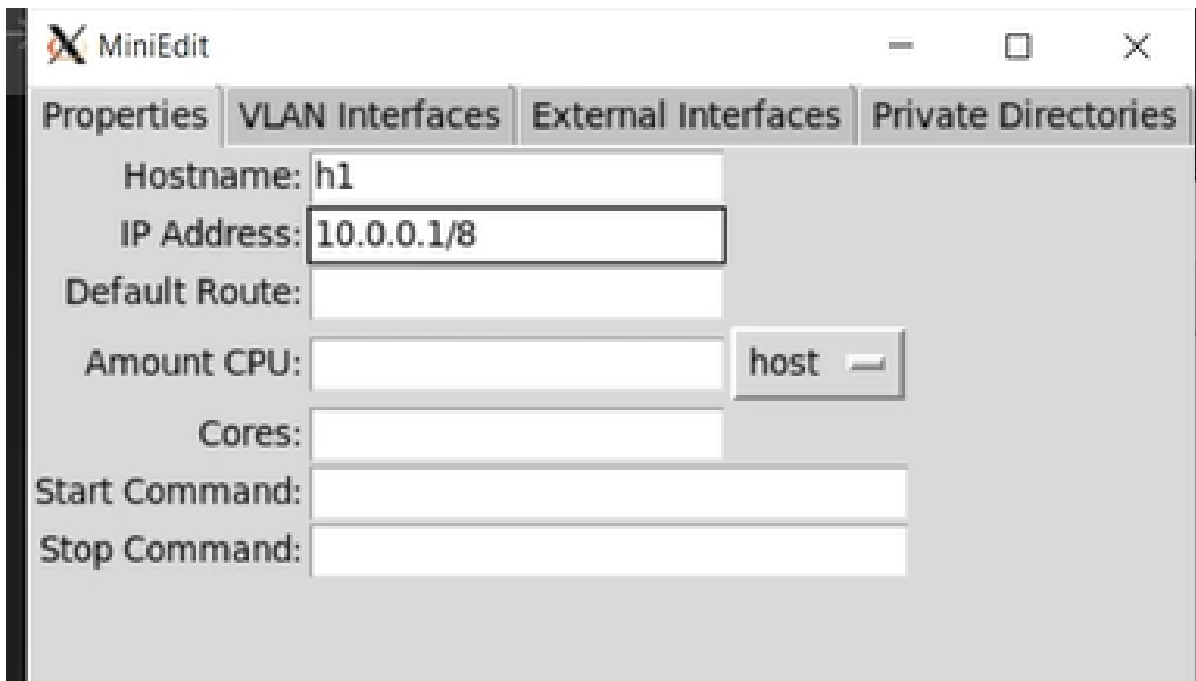


Рис. 2.14: Настраиваем IP-адрес h1, по аналогии и у h2

Проверка связности.

– Перед проверкой соединения между хостом h1 и хостом h2 необходимо запустить эмуляцию. Для запуска эмуляции нажмите кнопку Run. После начала эмуляции кнопки панели MiniEdit станут серыми, указывая на то, что в настоящее время они отключены.

– Откройте терминал на хосте h1, удерживая правую кнопку мыши на хосте h1 и выбрав Terminal. Это действие позволит выполнять команды на хосте h1.

– Откройте терминал на хосте h2.

– На терминале хоста h1 введите команду `ifconfig`, чтобы отобразить назначенные ему IP-адреса. Интерфейс h1-eth0 на хосте h1 должен быть настроен с IP-адресом 10.0.0.1 и маской подсети 255.0.0.0.

– Повторите эти действия на хосте h2. Его интерфейс h2-eth0 должен быть настроен с IP-адресом 10.0.0.2 и маской подсети 255.0.0.0.

– Проверьте соединение между хостами, введя в терминале хоста h1 команду `ping 10.0.0.2`. Для остановки теста нажмите `Ctrl + c`.

– Остановите эмуляцию, нажав кнопку Stop.

```
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=0.288 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.045 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.061 ms
^C
--- 10.0.0.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2029ms
rtt min/avg/max/mdev = 0.045/0.131/0.288/0.110 ms
root@mininet-vm:/home/mininet#
```

Рис. 2.15: Проверяем соединение между хостами

Автоматическое назначение IP-адресов.

Ранее IP-адреса узлам h1 и h2 были назначены вручную. В качестве альтернативы можно полагаться на Mininet для автоматического назначения IP-адресов.

- Удалите назначенный вручную IP-адрес с хостов h1 и h2.
- В MiniEdit нажмите Edit Preferences . По умолчанию в поле базовые значения IP-адресов (IP Base) установлено 10.0.0.0/8. Измените это значение на 15.0.0.0/8.
- Запустите эмуляцию, нажав кнопку Run.
- Откройте терминал на хосте h1, удерживая правую кнопку мыши на хосте h1 и выбрав Terminal.
- Чтобы отобразить IP-адреса, назначенные хосту h1, введите команду ifconfig

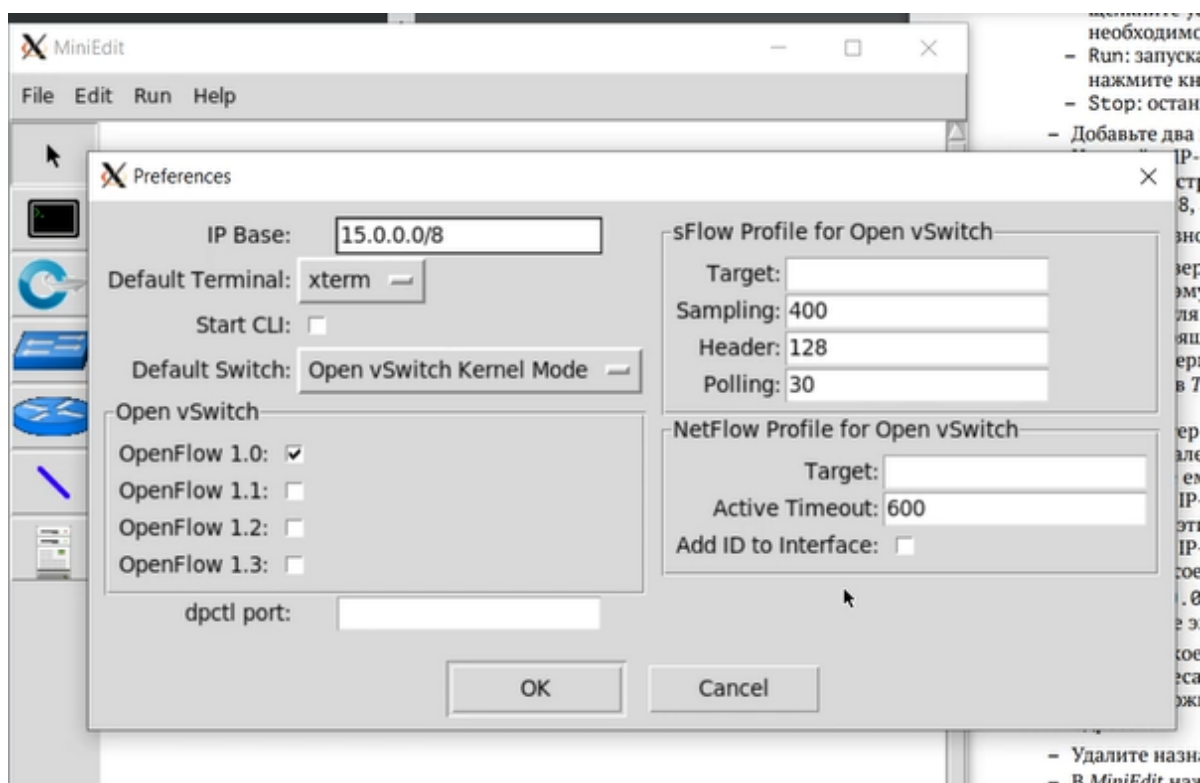


Рис. 2.16: Настраиваем автоматическое назначение IP-адресов

```

Host: h1@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 92:37:ef:b4:e6:c0 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 1093 bytes 298716 (298.7 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1093 bytes 298716 (298.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@mininet-vm:/home/mininet#

```

Рис. 2.17: Проверяем IP-адрес у h1

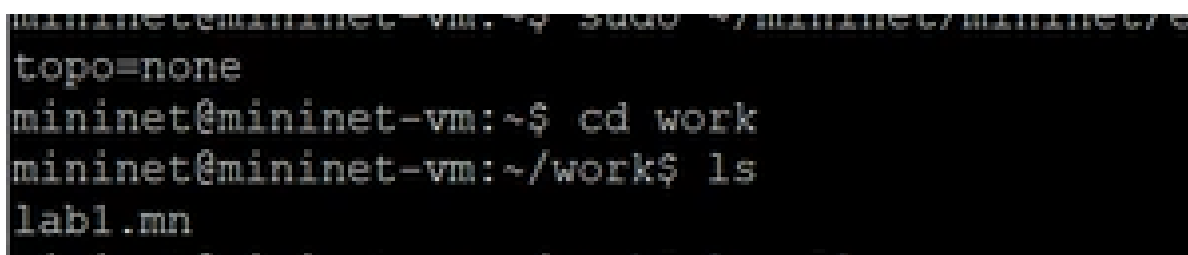
Интерфейс h1-eth0 на узле h1 теперь имеет IP-адрес 15.0.0.1 и маску подсети 255.0.0.0.

Сохранение и загрузка топологии Mininet.

– В домашнем каталоге виртуальной машины mininet создайте каталог для работы с проектами mininet: `mkdir ~/work`

Для сохранения топологии сети в файл нажмите в MiniEdit File-Save. Укажите имя для топологии и сохраните на своём компьютере.

– После сохранения проекта поменяйте права доступа к файлам в каталоге проекта: `sudo chown -R mininet:mininet ~/work`



```
mininet@mininet-vm:~$ sudo mkdir ~/mininet/mininet/c
topo=None
mininet@mininet-vm:~$ cd work
mininet@mininet-vm:~/work$ ls
lab1.mn
```

Рис. 2.18: Создаем каталог, сохраняем файл

## 3 Выводы

Мы развернули в системе виртуализации mininet, познакомились с основными командами для работы с Mininet через командную строку и через графический интерфейс.



## 4 Список литературы

[1] Mininet: <https://mininet.org/>