

Лабораторная работа №1

Введение в Mininet

Ланцова Яна Игоревна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	24

Список иллюстраций

2.1	Импорт конфигураций	6
2.2	Импорт конфигураций	7
2.3	Настройка сети	8
2.4	Настройка сети	8
2.5	Запуск mininet	9
2.6	Подключение к mininet через SSH	9
2.7	Активируем интерфейс и скачаем новую версию mininet	10
2.8	Настройка параметров XTerm	10
2.9	Настройка соединения X11 для суперпользователя	11
2.10	Установка программного обеспечения	11
2.11	Запуск XServer	12
2.12	Запуск XServer	12
2.13	Запуск XServer	13
2.14	Работа XServer	13
2.15	Запуск Putty	14
2.16	Запуск Putty	15
2.17	Работа с Mininet с помощью командной строки	16
2.18	Работа с Mininet с помощью командной строки	17
2.19	Работа с Mininet с помощью командной строки	17
2.20	Работа с Mininet с помощью командной строки	18
2.21	Работа с Mininet с помощью командной строки	18
2.22	Простейшая сеть	19
2.23	IP-адрес первого хоста	20
2.24	IP-адрес второго хоста	21
2.25	Эмуляция созданной сети	21
2.26	Проверка IP-адресов и соединения	22
2.27	Настройка автоматического назначения IP-адресов	22
2.28	Проверка IP-адресов и соединения	22
2.29	Сохранение топологии	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

Перейдем в репозиторий Mininet, скачаем актуальный релиз ovf-образа виртуальной машины. Запустим систему виртуализации и импортируем файл .ovf и укажем параметры импорта (рис. 2.1;2.2).

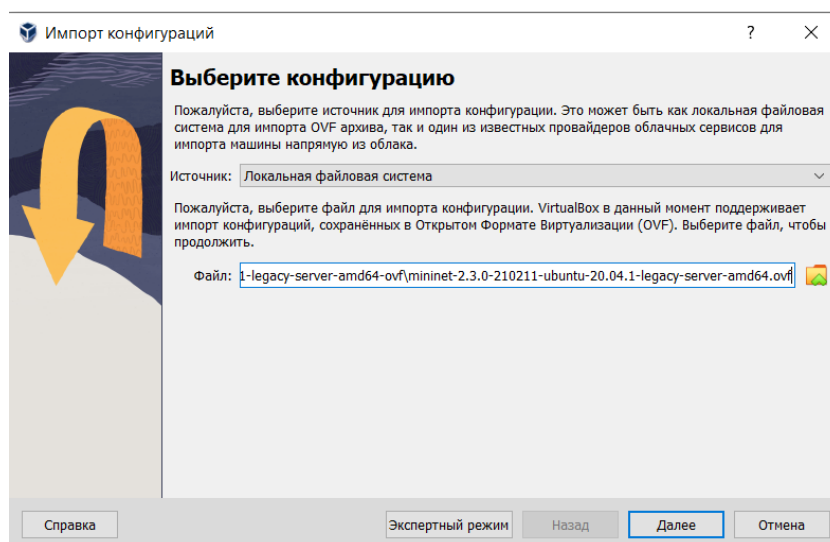


Рис. 2.1: Импорт конфигураций

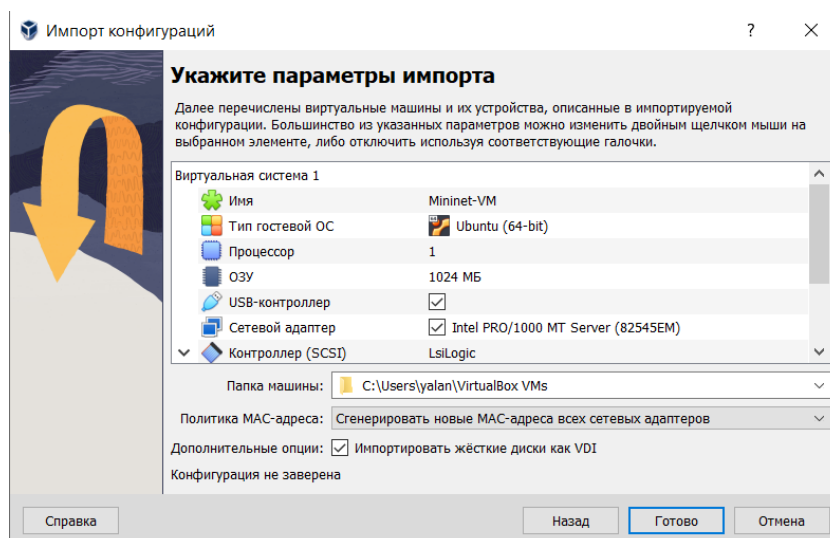


Рис. 2.2: Импорт конфигураций

Перейдем в настройки системы виртуализации и уточним параметры настройки виртуальной машины. Для VirtualBox (однако, я была вынуждена перейти на VMware, поскольку не могла пропинговать адрес своей машины впоследствии) выберем импортированную виртуальную машину и перейдите в меню “Машина -> Настроить”. Перейдем к опции «Система». Если внизу этого окна есть сообщение об обнаружении неправильных настроек, то, следуя рекомендациям, внесем исправления (изменим тип графического контроллера на рекомендуемый). В настройках сети первый адаптер должен иметь подключение типа NAT (рис. 2.3). Для второго адаптера укажите тип подключения host-only network adapter (виртуальный адаптер хоста), который в дальнейшем вы будете использовать для входа в образ виртуальной машины (рис. 2.4).

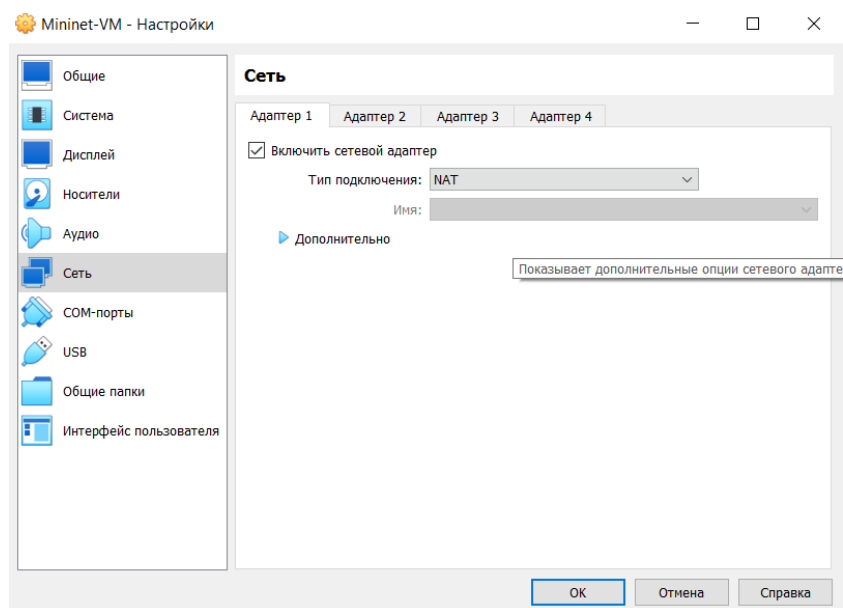


Рис. 2.3: Настройка сети

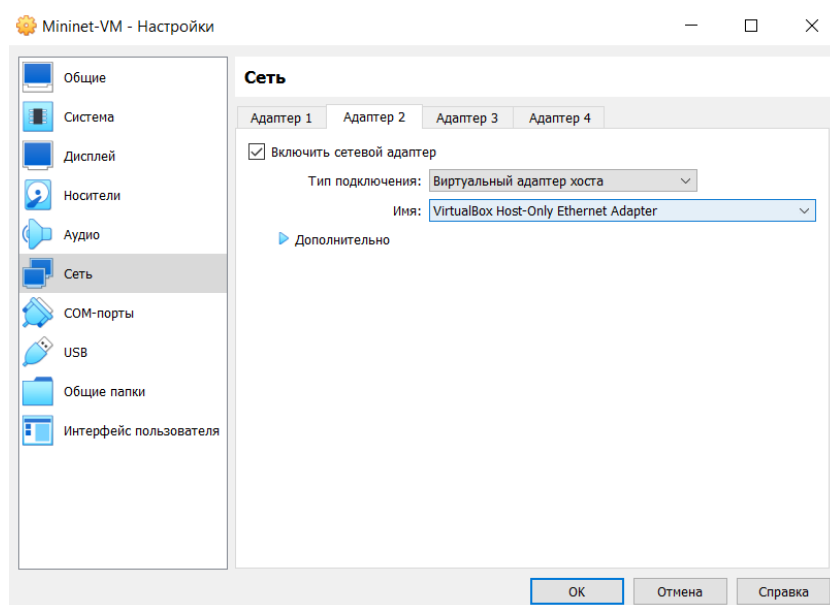


Рис. 2.4: Настройка сети

Посмотрим адрес машины с помощью ifconfig (рис. 2.5).


```

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 172.16.176.128 netmask 255.255.255.0 broadcast 172.16.176.255
    ether 00:0c:29:6d:ce:cb txqueuelen 1000 (Ethernet)
    RX packets 209 bytes 16316 (16.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 208 bytes 17194 (17.1 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 68 bytes 5614 (5.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 68 bytes 5614 (5.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

mininet@mininet-vm:~$ _

```

Рис. 2.5: Запуск mininet

Подключимся к виртуальной машине (из терминала хостовой машины). Настроим ssh-подсоединение по ключу к виртуальной машине. Вновь подключимся к виртуальной машине и убедимся, что подсоединение происходит успешно и без ввода пароля (рис. 2.6).

```

PS C:\Users\yalan\.ssh> ping 172.16.176.128
Обмен пакетами с 172.16.176.128 по 32 байтами данных:
Ответ от 172.16.176.128: число байт=32 время=1мс TTL=64
Ответ от 172.16.176.128: число байт=32 время=1мс TTL=64
Ответ от 172.16.176.128: число байт=32 время=1мс TTL=64
Ответ от 172.16.176.128: число байт=32 время=1мс TTL=64

Статистика Ping для 172.16.176.128:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (0% потерь)

Приблизительное время приема-передачи в мс:
    Минимальное = 0мсек, Максимальное = 1 мсек, Среднее = 0 мсек
PS C:\Users\yalan\.ssh> ssh mininet@172.16.176.128
mininet@172.16.176.128'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: Mon Sep  8 06:19:06 2025 from 172.16.176.1
mininet@mininet-vm:~$ logout
Connection to 172.16.176.128 closed.

```

Рис. 2.6: Подключение к mininet через SSH

Активируем второй интерфейс для доступа к сети интернет (рис. 2.7). Поскольку я выполняла работу на VMware, у меня не получилось активировать интерфейс и подключиться к интернету с виртуальной машины, а следовательно установить тмс и скачать новую версию mininet.

```

mininet@mininet-vm:~$ sudo dhclient eth1
Cannot find device "eth1"
mininet@mininet-vm:~$ mv ~/mininet ~/mininet.orig
mininet@mininet-vm:~$ cd ~
mininet@mininet-vm:~$ git clone https://github.com/mininet/mininet.git
Cloning into 'mininet'...
fatal: unable to access 'https://github.com/mininet/mininet.git': Could not resolve host: github.co
m
mininet@mininet-vm:~$ _

```

Рис. 2.7: Активируем интерфейс и скачаем новую версию mininet

По умолчанию XTerm использует растровые шрифты малого кегля. Для увеличения размера шрифта и применения векторных шрифтов вместо растровых внесем изменения в файл /etc/X11/app-defaults/XTerm:

```

GNU nano 4.8 /etc/X11/app-defaults/XTerm Modified
*SimpleMenu*borderWidth: 2

! xterm can switch at runtime between bitmap (default) and TrueType fonts.
! The "faceSize" resource controls the size of the latter. However, it was
! originally given with a size that makes the two types of fonts different
! sizes. Uncomment this line to use the same size as "fixed".
!*faceSize: 8

! Here is a pattern that is useful for double-clicking on a URL:
*charClass: 33:48,35:48,37-38:48,43-47:48,58:48,61:48,63-64:48,95:48,126:48
!
! Alternatively,
!*on2Clicks: regex [[[:alpha:]]+://([[:alnum:]]!#+,./=?@_~!|C%[[[:xdigit:]]![:xdigit:]]!)]+
!
! 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
!*allowTcpOps: false
!*allowTitledOps: false
!*allowWindowOps: false
xterm*faceName: Monospace
xterm*faceSize: 12

G Get Help  O Write Out  M Where Is  R Cut Text  J Justify  C Cur Pos  U Undo
X Exit      B Read File  N Replace    U Paste Text  T To Spell  G Go To Line  E Redo

```

Рис. 2.8: Настройка параметров XTerm

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. Скопируем значение куки (MIT magic cookie)¹ пользователя mininet в файл для пользователя root(рис. 2.9).

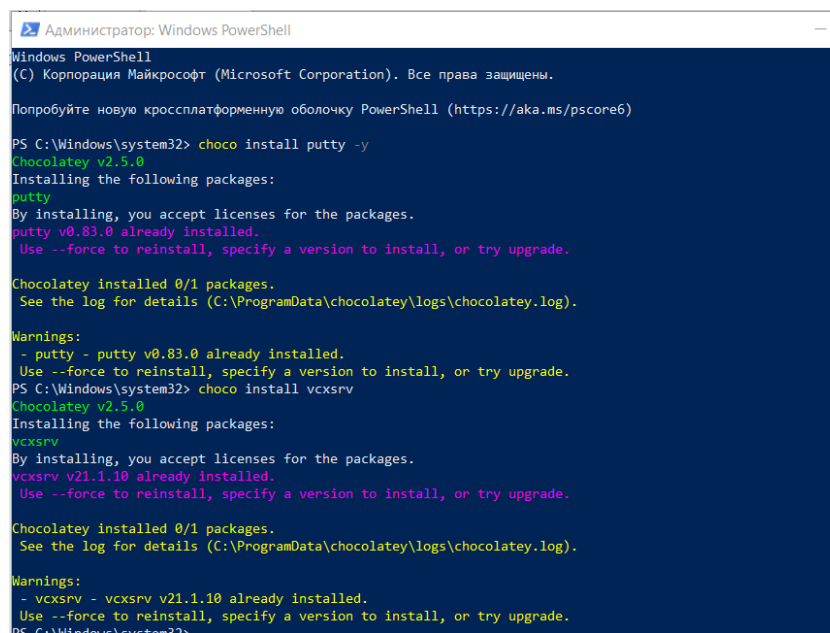
```

mininet@mininet-vm:~$ cd ~
mininet@mininet-vm:~$ xauth list $DISPLAY
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 9cbb9eb5f5a6479154c1352dc848f339
mininet@mininet-vm:~$ 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 9cbb9eb5f5a6479154c1352dc848f339
xauth: file /root/.Xauthority does not exist
root@mininet-vm:~# xauth list $DISPLAY
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 9cbb9eb5f5a6479154c1352dc848f339
root@mininet-vm:~# logout
mininet@mininet-vm:~$ _

```

Рис. 2.9: Настройка соединения X11 для суперпользователя

Для работы с Mininet из-под Windows необходимо установить putty и VcXsrv Windows X Server (рис. 2.10):



```

Администратор: Windows PowerShell
Windows PowerShell
(C) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

Попробуйте новую кроссплатформенную оболочку PowerShell (https://aka.ms/pscore6)

PS C:\Windows\system32> choco install putty -y
Chocolatey v2.5.0
Installing the following packages:
putty
By installing, you accept licenses for the packages.
putty v0.83.0 already installed.
Use --force to reinstall, specify a version to install, or try upgrade.

Chocolatey installed 0/1 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).

Warnings:
- putty - putty v0.83.0 already installed.
Use --force to reinstall, specify a version to install, or try upgrade.
PS C:\Windows\system32> choco install vcxsrv
Chocolatey v2.5.0
Installing the following packages:
vcxsrv
By installing, you accept licenses for the packages.
vcxsrv v21.1.10 already installed.
Use --force to reinstall, specify a version to install, or try upgrade.

Chocolatey installed 0/1 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).

Warnings:
- vcxsrv - vcxsrv v21.1.10 already installed.
Use --force to reinstall, specify a version to install, or try upgrade.
PS C:\Windows\system32>

```

Рис. 2.10: Установка программного обеспечения

Далее запустим Xlaunch и выберем следующие опции(рис. 2.11-2.13).

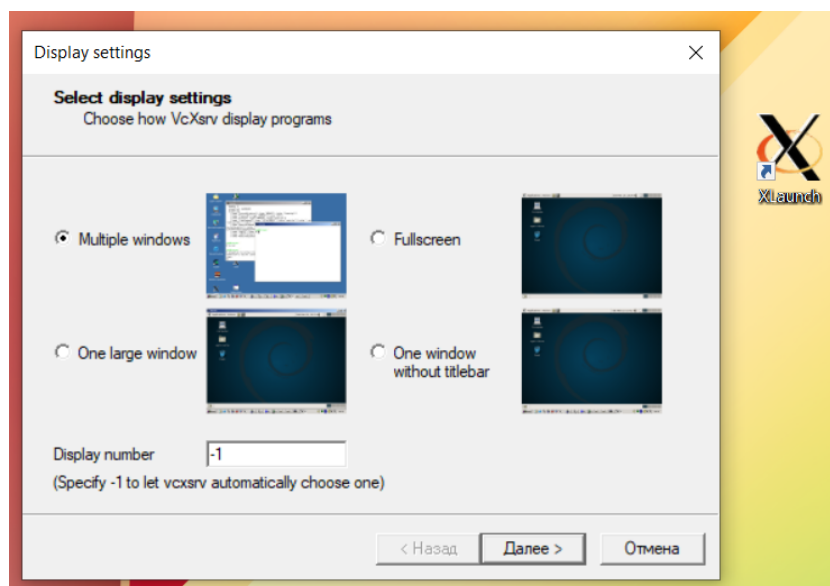


Рис. 2.11: Запуск XServer

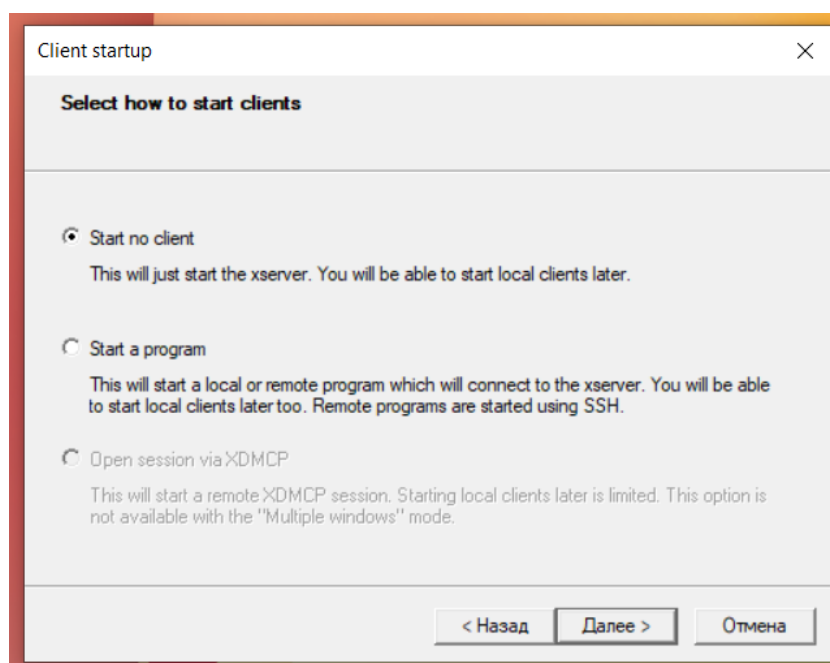


Рис. 2.12: Запуск XServer

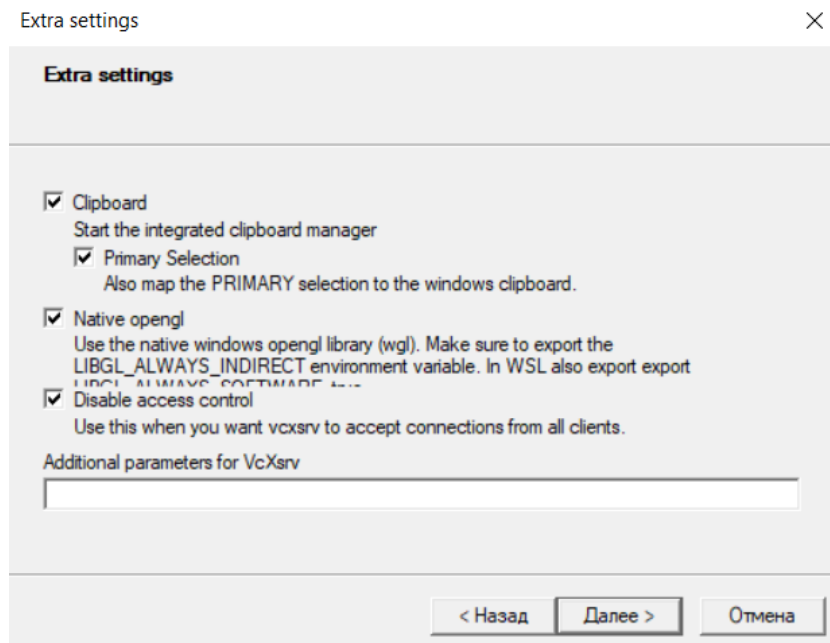


Рис. 2.13: Запуск XServer

Проверим, что XServer запустился через диспетчер задач (рис. 2.14):

Имя	Состояние	36% ЦП	76% Память	1% Диск	0% Сеть	1% GPU	Ядро GPU	Энергопотребл...
McAfee® Personal Security (3)		0%	3.5 MB	0 MB/c	0 MB/c	0%		Очень низкое
Хост Windows Shell Experience ...		0%	3.2 MB	0 MB/c	0 MB/c	0%		Очень низкое
Windows PowerShell (2)		0%	19.5 MB	0 MB/c	0 MB/c	0%		Очень низкое
Набросок на фрагменте экран...		0%	1.4 MB	0 MB/c	0 MB/c	0%		Очень низкое
MyASUS		0%	0 MB	0 MB/c	0 MB/c	0%		Очень низкое
VcXsrv windows xserver		0%	27.7 MB	0 MB/c	0 MB/c	0%		Очень низкое

Рис. 2.14: Работа XServer

Запустим Putty, введем адрес нашей машины и включим нужные опции (рис. 2.15, 2.16):

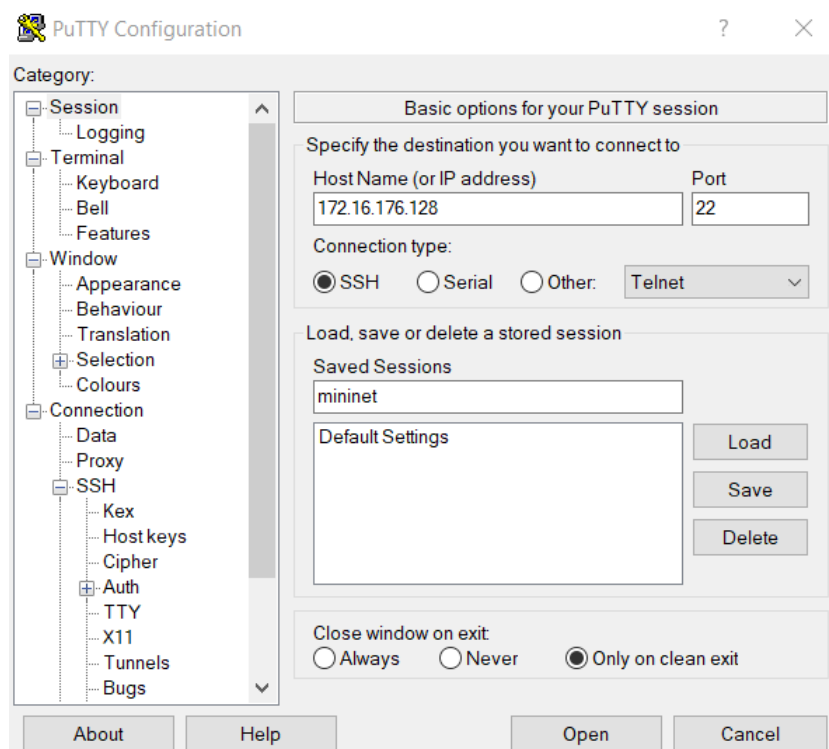


Рис. 2.15: Запуск Putty

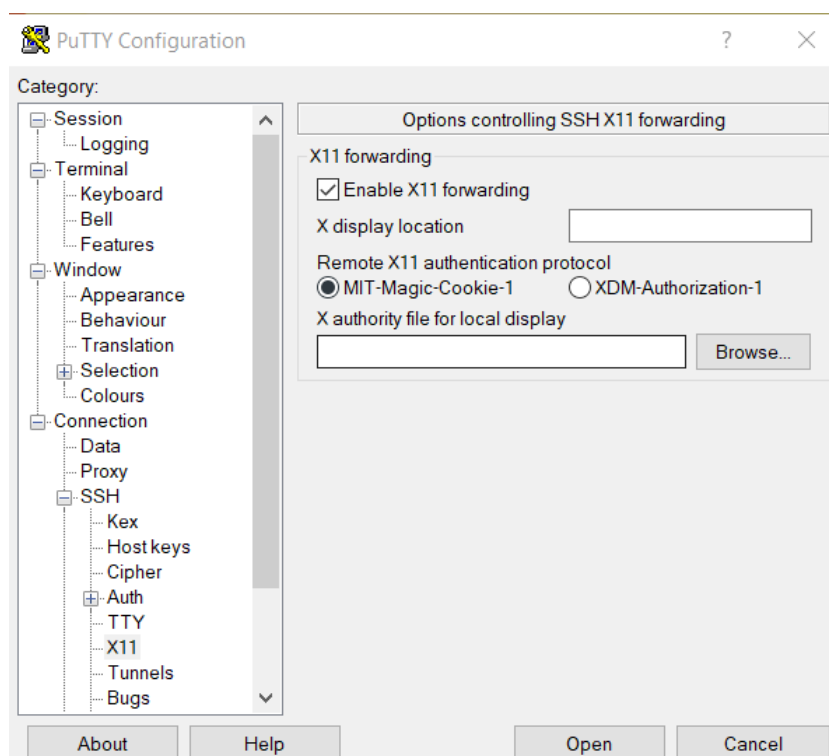


Рис. 2.16: Запуск Putty

Для запуска минимальной топологии введем в командной строке (рис. 2.17): `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 введем команду в интерфейсе командной строки Mininet: `help`

```

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> 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:
mininet> xterm h2

```

Рис. 2.17: Работа с Mininet с помощью командной строки

Для отображения доступных узлов введем: `nodes` (рис. 2.18). 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. Введем 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> 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
```

Рис. 2.18: Работа с Mininet с помощью командной строки

Mininet позволяет выполнять команды на конкретном устройстве. Чтобы выполнить команду для определенного узла, необходимо сначала указать устройство, а затем команду, например: `h1 ifconfig` (рис. 2.19). 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`.

```
mininet> h1 ipconfig
bash: ipconfig: command not found
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 f2:39:f5:0d:84:ea 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. 2.19: Работа с Mininet с помощью командной строки

Посмотрим конфигурацию всех узлов (рис. 2.20).

```

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 82:4b:37:24:15:af 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> s1 ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.16.176.128 netmask 255.255.255.0 broadcast 172.16.176.255
    ether 00:0c:29:6d:ce:cb txqueuelen 1000 (Ethernet)
    RX packets 21073 bytes 2039266 (2.0 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 42382 bytes 32706086 (32.7 MB)
    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 8448 bytes 30863377 (30.8 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8448 bytes 30863377 (30.8 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

s1-eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    ether 9a:bc:ba:a6:b3:71 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 1e:8b:e7:ce:60:e5 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. 2.20: Работа с Mininet с помощью командной строки

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, используем команду ping (рис. 2.21).

```

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=4.55 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.252 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.083 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.091 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.086 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.185 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.117 ms
^C
--- 10.0.0.2 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6112ms
rtt min/avg/max/mdev = 0.083/0.766/4.553/1.546 ms

```

Рис. 2.21: Работа с Mininet с помощью командной строки

В терминале виртуальной машины mininet запустим MiniEdit. Добавим два хоста и один коммутатор, соединим хосты с коммутатором (рис. 2.22):

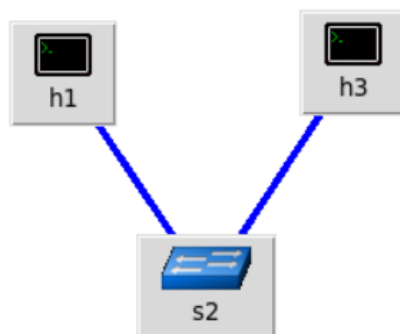


Рис. 2.22: Простейшая сеть

Настроим IP-адрес на хостах h1 и h2. Для первого хоста укажем IP-адрес 10.0.0.1/8, а для хоста второго — 10.0.0.2/8(рис. 2.23, 2.24).

The image shows a window titled "MiniEdit" with standard window controls (minimize, maximize, close). Inside the window, there are four tabs: "Properties", "VLAN Interfaces", "External Interfaces", and "Private Directories". The "Properties" tab is currently selected. It contains several input fields and a dropdown menu:

- Hostname:
- IP Address:
- Default Route:
- Amount CPU:
- Cores:
- Start Command:
- Stop Command:

At the bottom of the dialog, there are two buttons: "OK" and "Cancel".

Рис. 2.23: IP-адрес первого хоста

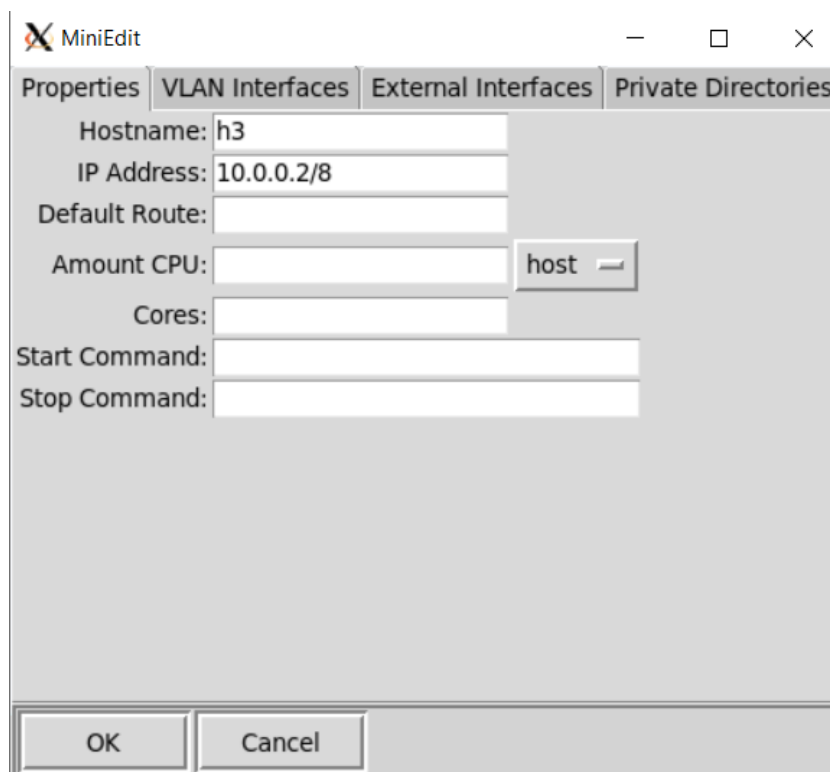


Рис. 2.24: IP-адрес второго хоста

Запустим эмуляцию, нажав на кнопку Run (рис. 2.25):

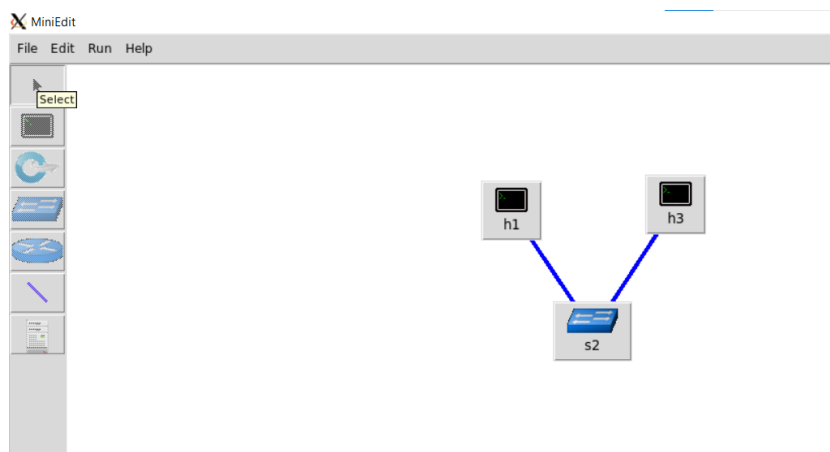


Рис. 2.25: Эмуляция созданной сети

Откроем терминалы на обоих хостах и введем команду `ifconfig`, чтобы отобразить назначенные IP-адреса. Проверим соединение между хостами, введя в терминале хоста h1 команду `ping 10.0.0.2` (рис. 2.26).

```
Host: h1@mininet-vn
root@mininet-vn:/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 8e:7f:18:7f:9b:68 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 936 bytes 235612 (235.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 936 bytes 235612 (235.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@mininet-vn:/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=0.237 ms
 64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.083 ms
 64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.090 ms
 64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.118 ms
^C
--- 10.0.0.2 ping statistics ---
 4 packets transmitted, 4 received, 0% packet loss, time 3055ms
 rtt min/avg/max/mdev = 0.083/0.132/0.237/0.062 ms
root@mininet-vn:/home/mininet#
```

```
Host: h2@mininet-vn
root@mininet-vn:/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 e2:f5:b0:ca:1a:1a txqueuelen 1000 (Ethernet)
    RX packets 6 bytes 476 (476.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 6 bytes 476 (476.0 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 884 bytes 231676 (231.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 884 bytes 231676 (231.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@mininet-vn:/home/mininet#
```

Рис. 2.26: Проверка IP-адресов и соединения

Настроим автоматическое назначение IP-адресов. Для этого в MiniEdit нажмет Edit>Preferences. По умолчанию в поле базовые значения IP-адресов установлено 10.0.0.0/8. Изменим это значение на 15.0.0.0/8 (рис. 2.27).

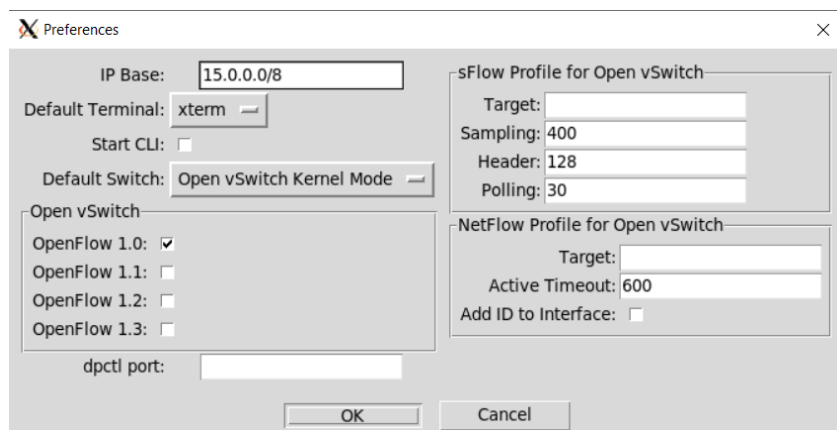


Рис. 2.27: Настройка автоматического назначения IP-адресов

Вновь запустим эмуляцию, откроем терминалы и введем команды ifconfig. Проверим соединение между хостами (рис. 2.28).

```
Host: h1@mininet-vn
root@mininet-vn:/home/mininet# ifconfig
h3-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 15.0.0.3 netmask 255.0.0.0 broadcast 15.255.255.255
    ether 36:95:c0:0d:9f:96 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 786 bytes 263244 (263.2 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 786 bytes 263244 (263.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@mininet-vn:/home/mininet# ping 10.0.0.1
ping: connect: Network is unreachable
root@mininet-vn:/home/mininet# ping 15.0.0.1
PING 15.0.0.1 (15.0.0.1) 56(84) bytes of data:
 64 bytes from 15.0.0.1: icmp_seq=1 ttl=64 time=0.626 ms
 64 bytes from 15.0.0.1: icmp_seq=2 ttl=64 time=0.076 ms
^C
```

```
Host: h2@mininet-vn
root@mininet-vn:/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 56:07:49:52:c8:08 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 862 bytes 229540 (229.5 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 862 bytes 229540 (229.5 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@mininet-vn:/home/mininet#
```

Рис. 2.28: Проверка IP-адресов и соединения

В домашнем каталоге виртуальной машины mininet создайте каталог для работы с проектами mininet: `mkdir ~/work`. Сохраним топологию в созданную папку, нажав на File>Save (рис. 2.29).

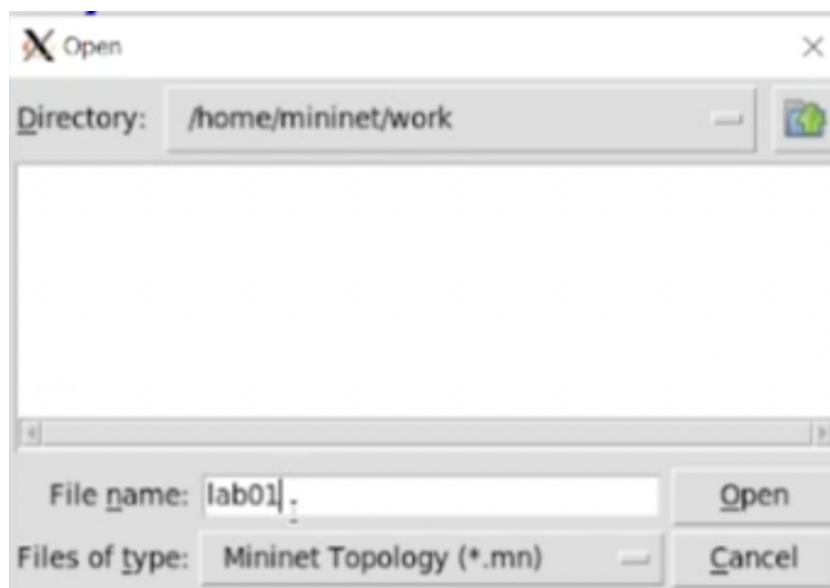


Рис. 2.29: Сохранение топологии

После сохранения проекта поменяем права доступа к файлам в каталоге проекта.

3 Выводы

В результате выполнения данной лабораторной работы я развёрнула mininet в системе виртуализации VirtualBox, а также ознакомилась с основными командами для работы с Mininet через командную строку и через графический интерфейс.