

# Лабораторная работа №1

Введение в Mininet

---

Хрусталеv В.Н.

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Хрусталев Влад Николаевич
- студент
- Российский университет дружбы народов
- 1132222011@pfur.ru

## Цель работы

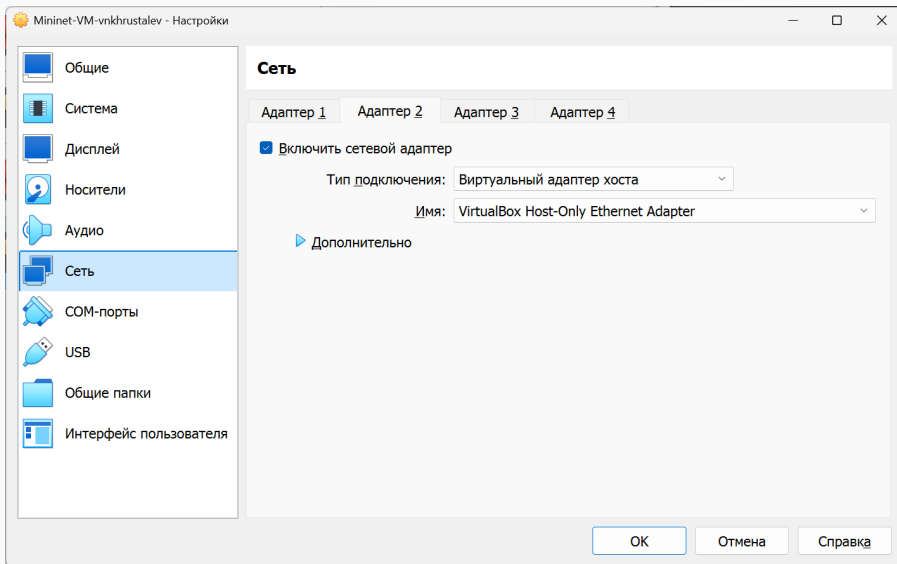
---

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

## Выполнение лабораторной работы

---

# Установка и настройка виртуальной машины



## Вход и просмотр адреса виртуальной машины

```
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

New release '22.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Sat Nov 29 09:42:35 PST 2025 from 192.168.56.1 on pts/3
mininet@mininet-vm:~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.105 netmask 255.255.255.0 broadcast 192.168.56.255
    ether 08:00:27:e8:34:b9 txqueuelen 1000 (Ethernet)
    RX packets 5 bytes 1379 (1.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 16 bytes 2084 (2.0 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

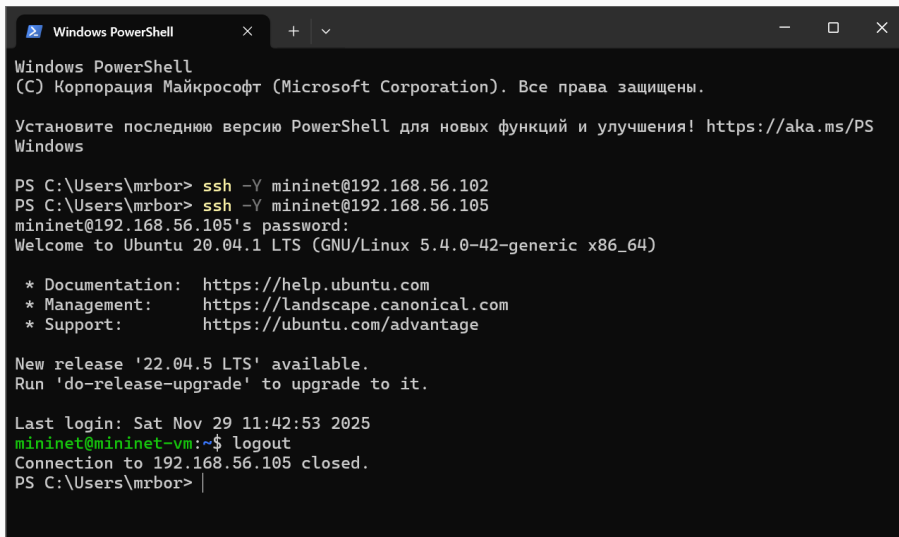
eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    ether 08:00:27:d0:e5:b3 txqueuelen 1000 (Ethernet)
    RX packets 207 bytes 22603 (22.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 229 bytes 21087 (21.0 KB)
    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 52 bytes 4518 (4.5 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 52 bytes 4518 (4.5 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

mininet@mininet-vm:~$
```



## Подключение к виртуальной машине из терминала хостовой машины



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

Установите последнюю версию PowerShell для новых функций и улучшения! https://aka.ms/PSWindows

PS C:\Users\mrbor> ssh -Y mininet@192.168.56.102
PS C:\Users\mrbor> ssh -Y mininet@192.168.56.105
mininet@192.168.56.105'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

New release '22.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Sat Nov 29 11:42:53 2025
mininet@mininet-vm:~$ logout
Connection to 192.168.56.105 closed.
PS C:\Users\mrbor> |
```

Рис. 3: Подключение к виртуальной машине из терминала хостовой машины

## Установка putty

```
Windows PowerShell
PS C:\Users\mrbor> choco install putty -y
Chocolatey v2.2.2
Chocolatey detected you are not running from an elevated command shell
(cmd/powershell).

You may experience errors - many functions/packages
require admin rights. Only advanced users should run choco w/out an
elevated shell. When you open the command shell, you should ensure
that you do so with "Run as Administrator" selected. If you are
attempting to use Chocolatey in a non-administrator setting, you
must select a different location other than the default install
location. See
https://docs.chocolatey.org/en-us/choco/setup#non-administrative-install
for details.

For the question below, you have 20 seconds to make a selection.

Do you want to continue?([Y]es/[N]o): Y

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:\Users\mrbor>
```

# Установка VcXsrv Windows X Server

```
PS C:\Users\mrbor> choco install vcxsrv -y
Chocolatey v2.2.2
Chocolatey detected you are not running from an elevated command shell
(cmd/powershell).

You may experience errors - many functions/packages
require admin rights. Only advanced users should run choco w/out an
elevated shell. When you open the command shell, you should ensure
that you do so with "Run as Administrator" selected. If you are
attempting to use Chocolatey in a non-administrator setting, you
must select a different location other than the default install
location. See
https://docs.chocolatey.org/en-us/choco/setup#non-administrative-install
for details.

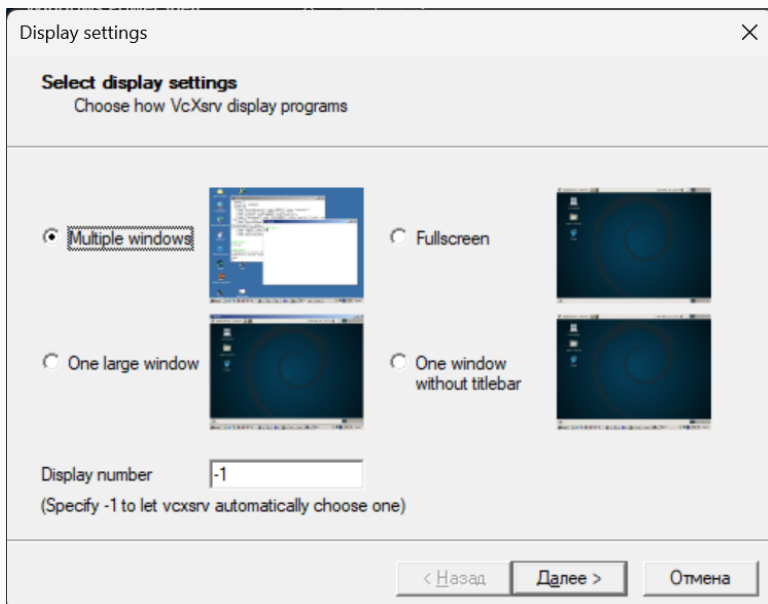
For the question below, you have 20 seconds to make a selection.

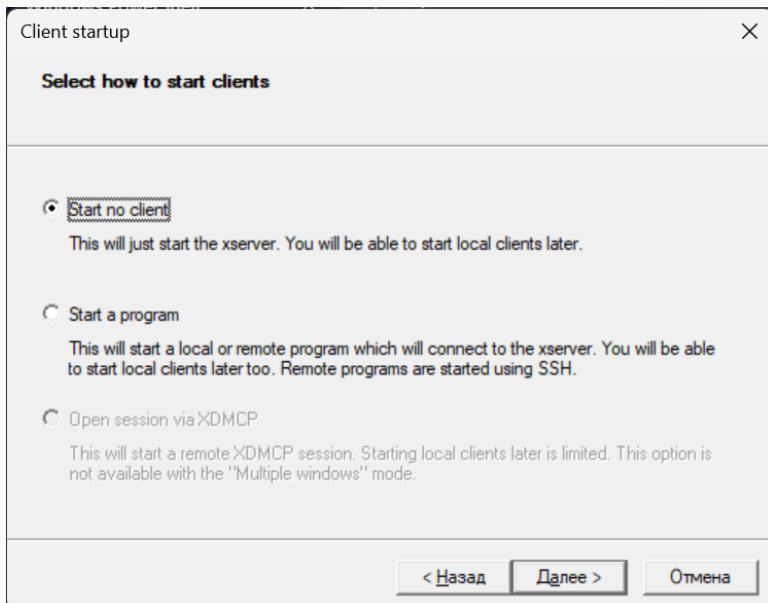
Do you want to continue?([Y]es/[N]o): Y

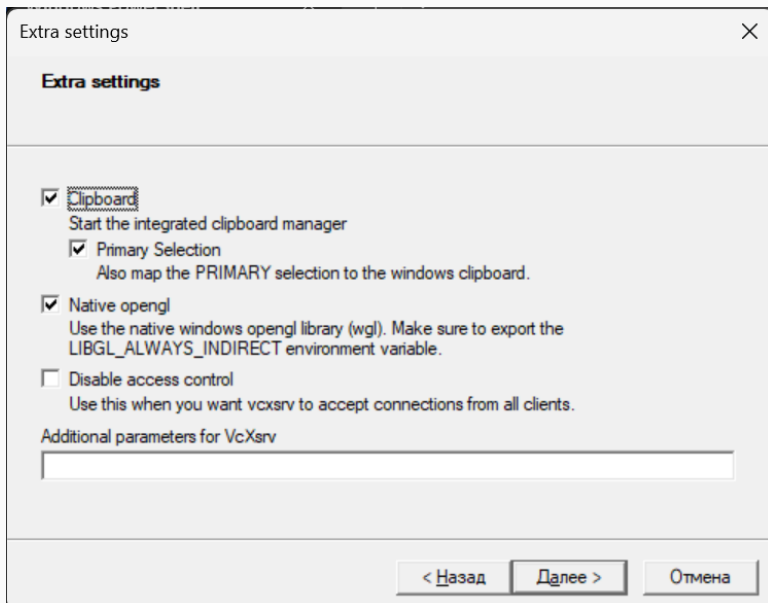
Installing the following packages:
vcxsrv
By installing, you accept licenses for the packages.
vcxsrv v21.1.10 already installed.
Use --force to reinstall, specify a version to install, or try upgrade.

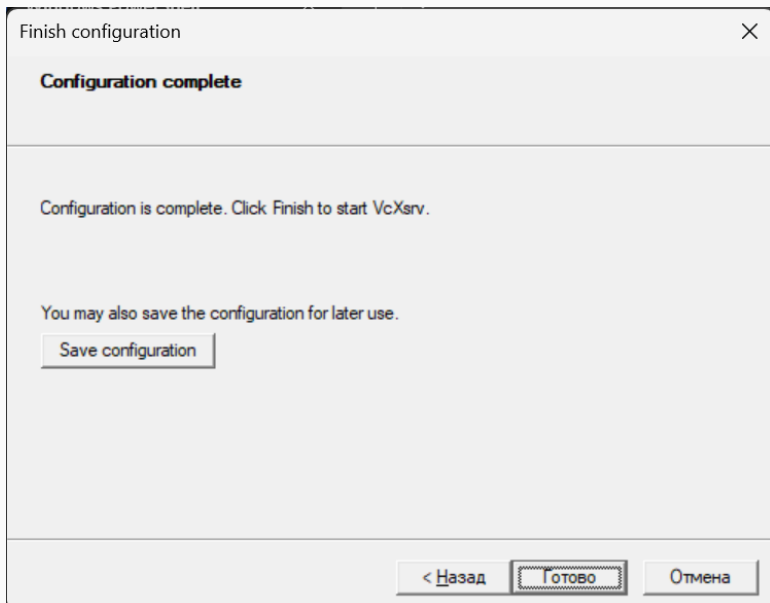
Chocolatey installed 0/1 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).

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

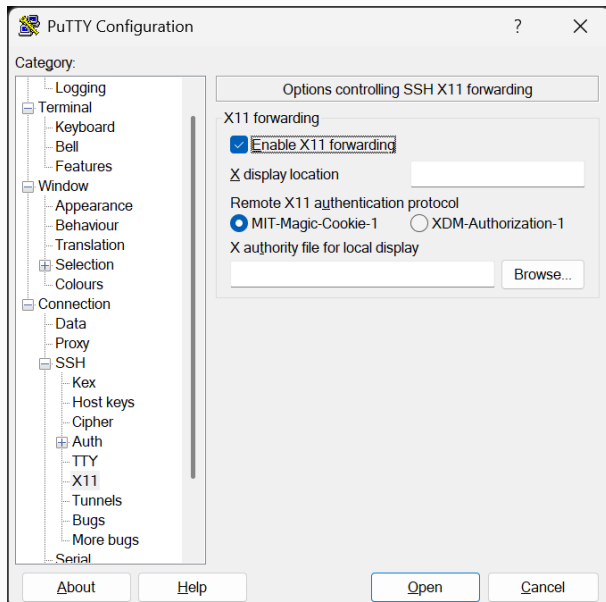








## Опция перенаправления X11





```
mininet@mininet-vm:~$ cat /etc/netplan/01-netcfg.yaml
# 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
    eth1:
      dhcp4: yesmininet@mininet-vm:~$ _
```

Рис. 11: Файл /etc/netplan/01-netcfg.yaml

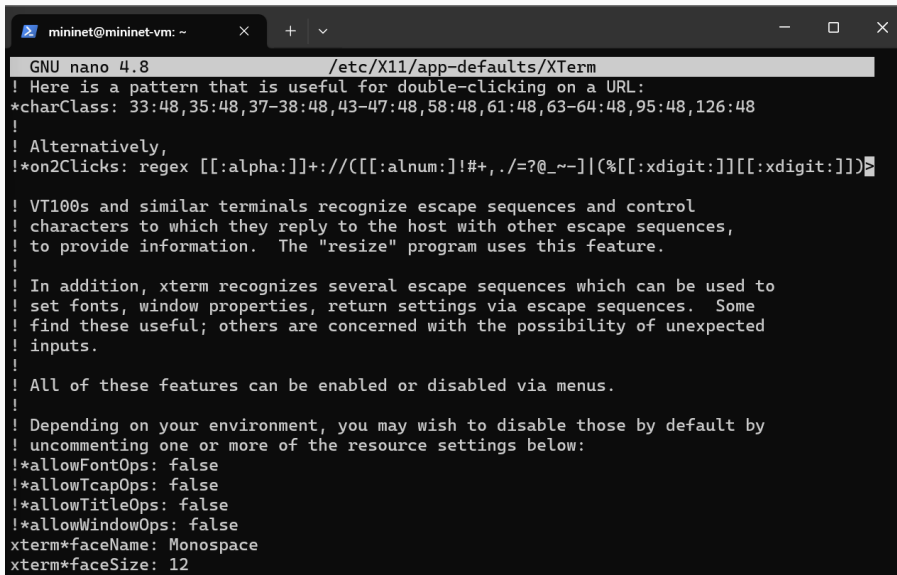
```
mininet@mininet-vm:~$ rm -rf mininet.orig/
mininet@mininet-vm:~$ mv -R ~/mininet ~/mininet.orig
mv: invalid option -- 'R'
Try 'mv --help' for more information.
mininet@mininet-vm:~$ mv ~/mininet ~/mininet.orig
mininet@mininet-vm:~$ cd ~
mininet@mininet-vm:~$ git clone https://github.com/mininet/mininet.git
Cloning into 'mininet'...
remote: Enumerating objects: 10388, done.
remote: Counting objects: 100% (131/131), done.
remote: Compressing objects: 100% (60/60), done.
remote: Total 10388 (delta 104), reused 71 (delta 71), pack-reused 10257 (from 3)
Receiving objects: 100% (10388/10388), 3.36 MiB | 3.18 MiB/s, done.
Resolving deltas: 100% (6906/6906), done.
mininet@mininet-vm:~$ cd ~/mininet
mininet@mininet-vm:~/mininet$ sudo make install
```

Рис. 12: Обновление Mininet

```
Successfully installed mininet-2.3.1b4  
mininet@mininet-vm:~/mininet$ mn --version  
2.3.1b4  
mininet@mininet-vm:~/mininet$ _
```

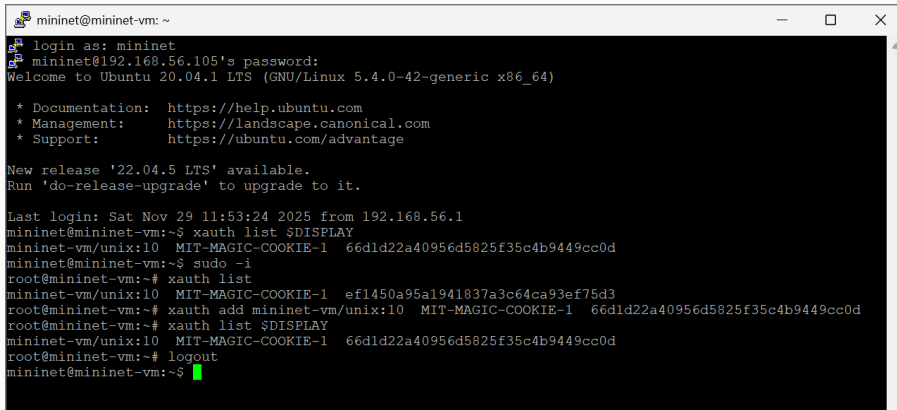
Рис. 13: Обновление Mininet/ВЕРСИЯ

# Настройка шрифтов XTerm



```
mininet@mininet-vm: ~
GNU nano 4.8 /etc/X11/app-defaults/XTerm
! 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:]]!#+,./=?@_~-]|(%[[: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
!*allowTcapOps: false
!*allowTitleOps: false
!*allowWindowOps: false
xterm*faceName: Monospace
xterm*faceSize: 12
```

## Настройка соединения X11 для суперпользователя



```
mininet@mininet-vm: ~  
login as: mininet  
mininet@192.168.56.105'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  
  
New release '22.04.5 LTS' available.  
Run 'do-release-upgrade' to upgrade to it.  
  
Last login: Sat Nov 29 11:53:24 2025 from 192.168.56.1  
mininet@mininet-vm:~$ xauth list $DISPLAY  
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 66d1d22a40956d5825f35c4b9449cc0d  
mininet@mininet-vm:~$ sudo -i  
root@mininet-vm:~# xauth list  
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 ef1450a95a1941837a3c64ca93ef75d3  
root@mininet-vm:~# xauth add mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 66d1d22a40956d5825f35c4b9449cc0d  
root@mininet-vm:~# xauth list $DISPLAY  
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 66d1d22a40956d5825f35c4b9449cc0d  
root@mininet-vm:~# logout  
mininet@mininet-vm:~$
```

Рис. 15: Настройка соединения X11 для суперпользователя

## Mininet с использованием топологии по умолчанию

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

## Отображение результата help команды

```
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

mininet> █
```

Рис. 17: Отображение результата help команды

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

Рис. 18: Отображение доступных узлов



```
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. 19: Просмотр доступных линков

## Команда h1 ifconfig

```
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 16:92:40:70:3e:7c 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> █
```

Рис. 20: Команда h1 ifconfig

## Команда h1 ping 10.0.0.2

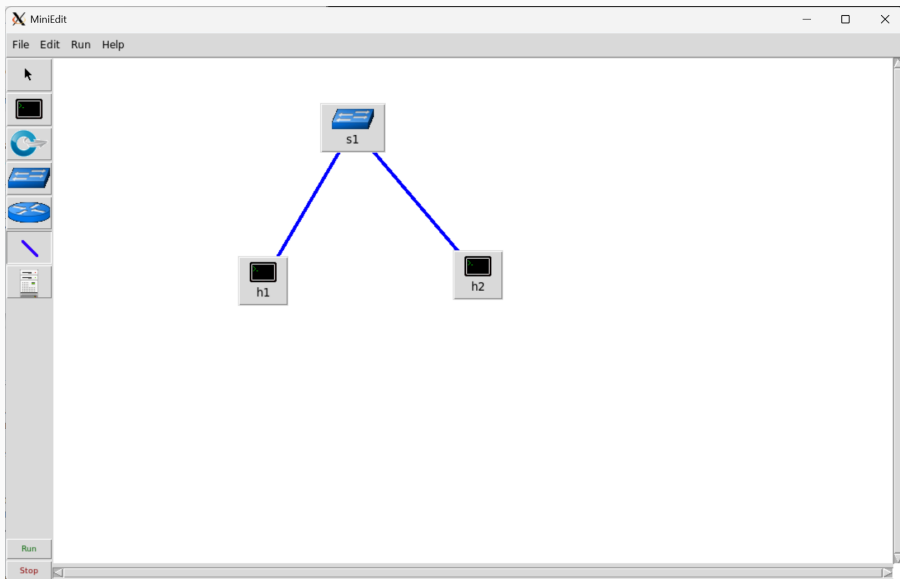
```
mininet> h1 ping 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=3.65 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.394 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.092 ms
64 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=0.088 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.089 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.087 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=0.135 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.081 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=0.069 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.080 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.105 ms
^C
--- 10.0.0.2 ping statistics ---
13 packets transmitted, 13 received, 0% packet loss, time 12274ms
rtt min/avg/max/mdev = 0.052/0.384/3.645/0.944 ms
mininet> █
```

Рис. 21: Команда h1 ping 10.0.0.2

## Очистка предыдущего экземпляра Mininet

```
mininet> exit
*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
completed in 123.883 seconds
mininet@mininet-vm:~$ sudo mn -c
*** Removing excess controllers/ofprotocols/ofdatapaths/pings/noxes
killall controller ofprotocol ofdatapath ping nox_core lt-nox_core ovs-openflowd ovs-controller ovs-test
controller udpbwtest mnexec ivs ryu-manager 2> /dev/null
killall -9 controller ofprotocol ofdatapath ping nox_core lt-nox_core ovs-openflowd ovs-controller ovs-t
estcontroller udpbwtest mnexec ivs ryu-manager 2> /dev/null
pkill -9 -f "sudo mnexec"
*** Removing junk from /tmp
rm -f /tmp/vconn* /tmp/vlogs* /tmp/*.out /tmp/*.log
*** Removing old X11 tunnels
*** Removing excess kernel datapaths
ps ax | egrep -o 'dp[0-9]+' | sed 's/dp/nl:/'
*** Removing OVS datapaths
ovs-vsctl --timeout=1 list-br
ovs-vsctl --timeout=1 list-br
*** Removing all links of the pattern foo-ethX
ip link show | egrep -o '([_-[:alnum:]]+-eth[[:digit:]]+)'
ip link show
*** Killing stale mininet node processes
pkill -9 -f mininet:
*** Shutting down stale tunnels
pkill -9 -f Tunnel=Ethernet
pkill -9 -f .ssh/mn
rm -f ~/.ssh/mn/*
*** Cleanup complete.
mininet@mininet-vm:~$
```

## Добавление двух хостов и одного коммутатора



## Настройка IP-адреса на хосте h1 и h2

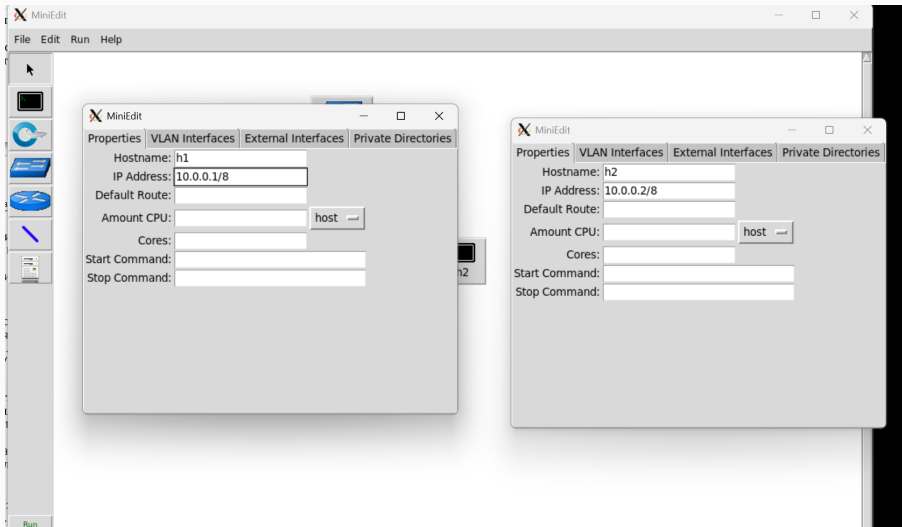


Рис. 24: Настройка IP-адреса на хосте h1 и h2

## Проверка IP-адресов. Пинг

```
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 f6:99:d2:05:30:c0 txqueuelen 1000 (Ethernet)
    RX packets 283 bytes 27118 (27.1 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 283 bytes 27118 (27.1 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 5732 bytes 1206400 (1.2 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 5732 bytes 1206400 (1.2 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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=0.046 ms
■
```

```
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 2a:e0:1a:c8:6e:49 txqueuelen 1000 (Ethernet)
    RX packets 283 bytes 27118 (27.1 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 283 bytes 27118 (27.1 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 2897 bytes 621460 (621.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2897 bytes 621460 (621.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@mininet-vm:/home/mininet# ping 10.0.0.1
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=0.363 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.062 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.089 ms
■
```

Рис. 25: Проверка IP-адресов. Пинг

## Смена базового IP-адреса

Preferences

IP Base:

Default Terminal:

Start CLI: ☐

Default Switch:

Open vSwitch

OpenFlow 1.0: ☒

OpenFlow 1.1: ☐

OpenFlow 1.2: ☐

OpenFlow 1.3: ☐

dpctl port:

sFlow Profile for Open vSwitch

Target:

Sampling:

Header:

Polling:

NetFlow Profile for Open vSwitch

Target:

Active Timeout:

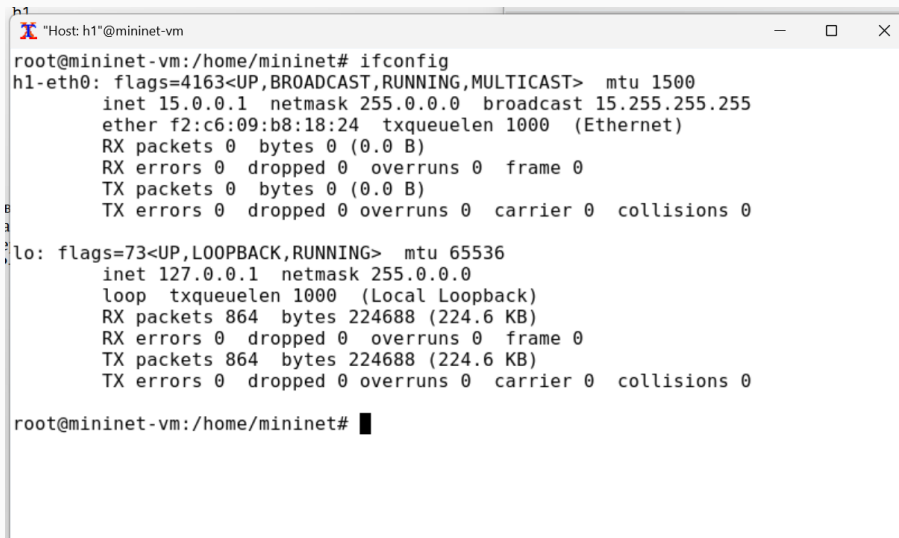
Add ID to Interface: ☐

OK Cancel

Рис. 26: Смена базового IP-адреса



## Просмотр IP-адреса на h1

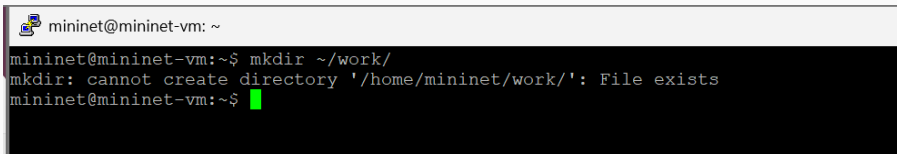
A terminal window titled "Host: h1" @mininet-vm. The window shows the output of the 'ifconfig' command. It displays details for the 'h1-eth0' interface, including its flags, MTU, IP address (15.0.0.1), netmask (255.0.0.0), broadcast address (15.255.255.255), MAC address (f2:c6:09:b8:18:24), and statistics for RX and TX packets, errors, and drops. It also shows details for the 'lo' (loopback) interface, including its flags, MTU, IP address (127.0.0.1), netmask (255.0.0.0), and statistics. The prompt 'root@mininet-vm:/home/mininet#' is visible at the bottom.

```
h1
"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 f2:c6:09:b8:18:24 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 864 bytes 224688 (224.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 864 bytes 224688 (224.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

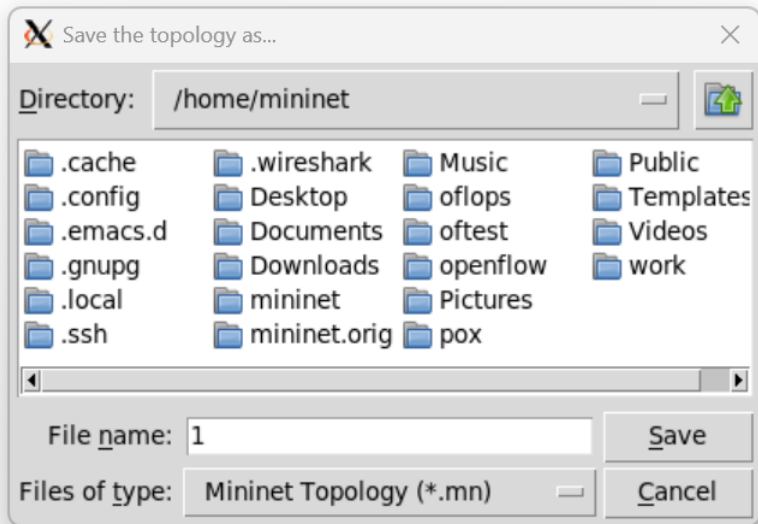
root@mininet-vm:/home/mininet#
```

Рис. 27: Просмотр IP-адреса на h1

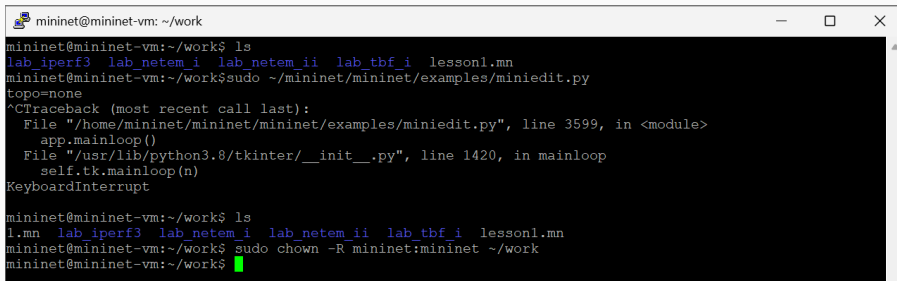


```
mininet@mininet-vm: ~  
mininet@mininet-vm:~$ mkdir ~/work/  
mkdir: cannot create directory '/home/mininet/work/': File exists  
mininet@mininet-vm:~$
```

Рис. 28: Создание нового каталога



## Изменение прав доступа к файлам в каталоге проекта

A terminal window titled 'mininet@mininet-vm: ~/work' with standard window controls. The terminal shows a sequence of commands and their outputs. First, 'ls' lists files: 'lab\_iperf3', 'lab\_netem\_i', 'lab\_netem\_ii', 'lab\_tbf\_i', and 'lesson1.mn'. Then, 'sudo ~/mininet/mininet/examples/miniedit.py' is run, followed by 'topo=None' and a Ctrl-C interrupt, which results in a traceback from 'mininet/examples/miniedit.py' line 3599 to 'tkinter/\_\_init\_\_.py' line 1420, ending with 'KeyboardInterrupt'. Finally, 'sudo chown -R mininet:mininet ~/work' is executed, and the prompt returns. A green cursor is visible at the end of the last command line.

```
mininet@mininet-vm: ~/work
mininet@mininet-vm:~/work$ ls
lab_iperf3 lab_netem_i lab_netem_ii lab_tbf_i lesson1.mn
mininet@mininet-vm:~/work$ sudo ~/mininet/mininet/examples/miniedit.py
topo=None
^CTraceback (most recent call last):
  File "/home/mininet/mininet/mininet/examples/miniedit.py", line 3599, in <module>
    app.mainloop()
  File "/usr/lib/python3.8/tkinter/__init__.py", line 1420, in mainloop
    self.tk.mainloop(n)
KeyboardInterrupt
mininet@mininet-vm:~/work$ ls
1.mn lab_iperf3 lab_netem_i lab_netem_ii lab_tbf_i lesson1.mn
mininet@mininet-vm:~/work$ sudo chown -R mininet:mininet ~/work
mininet@mininet-vm:~/work$
```

Рис. 30: Изменение прав доступа к файлам в каталоге проекта

## Выводы

---

В результате выполнения работы я развёрнул mininet в системе виртуализации VirtualBox и ознакомилась с основными командами для работы с Mininet через командную строку и через графический интерфейс.

## Список литературы

---

1. Mininet [Электронный ресурс]. Mininet Project Contributors. URL: <http://mini-net.org/> (дата обращения: 06.10.2025).



---