Презентация по лабораторной работе №3

Моделирование сетей передачи данных

Еюбоглу Тимур

7 октября 2025 г.

Российский университет дружбы народов, Москва, Россия

Докладчик

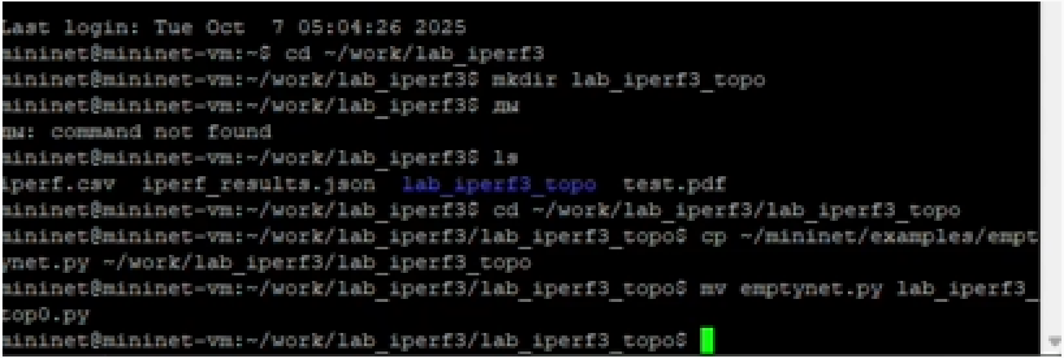
- Еюбоглу Тимур
- Студент группы НПИбд-01-22
- Студ. билет 1032224357
- Российский университет дружбы народов имени Патриса Лумумбы

Цель лабораторной работы

- Познакомиться с инструментом для измерения пропускной способности сети в режиме реального времени — iPerf3, а также получить навыки проведения воспроизводимого эксперимента по измерению пропускной способности моделируемой сети в среде Mininet.

Выполнение лабораторной работы

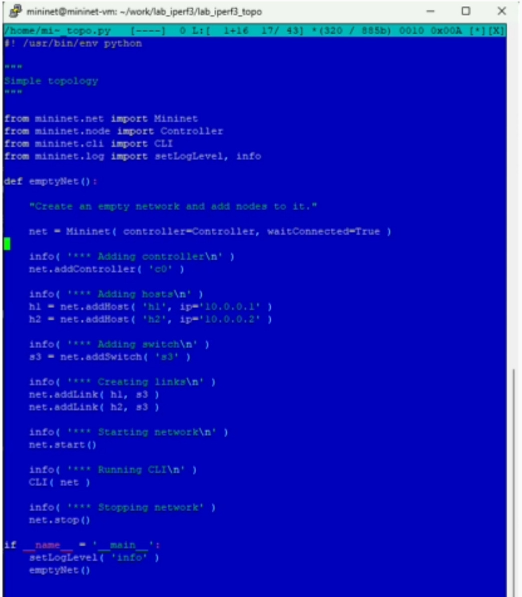
Создание простейшей топологии сети



```
Last login: Tue Oct  7 05:04:26 2025
mininet@mininet-vm:~$ cd ~/work/lab_iperf3
mininet@mininet-vm:~/work/lab_iperf3$ mkdir lab_iperf3_topo
mininet@mininet-vm:~/work/lab_iperf3$ mw
mw: command not found
mininet@mininet-vm:~/work/lab_iperf3$ ls
iperf.csv  iperf_results.json  lab_iperf3_topo  test.pdf
mininet@mininet-vm:~/work/lab_iperf3$ cd ~/work/lab_iperf3/lab_iperf3_topo
mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ cp ~/mininet/examples/empt
ynet.py ~/work/lab_iperf3/lab_iperf3_topo
mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ mv emptynet.py lab_iperf3_
topo0.py
mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ 
```

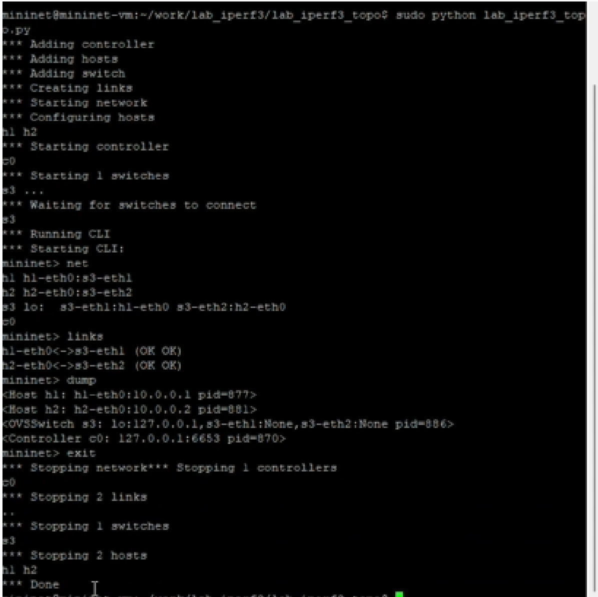
Рис. 1: Создание подкаталога, копирование файла с примером скрипта (описывающего стандартную простую топологию сети mininet)

Создание простейшей топологии сети



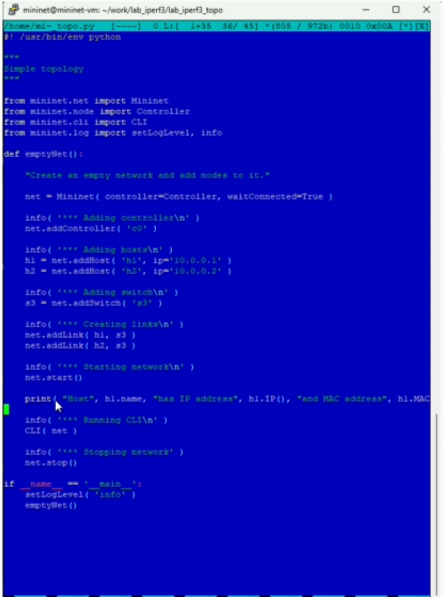
```
/home/mi~_topo.py   [----]  0 L:[  1+16  17/ 43] *(320 / 885b) 0010 0x00A [*][X]
#! /usr/bin/env python

"""
Simple topology
"""

from mininet.net import Mininet
from mininet.node import Controller
from mininet.cli import CLI
from mininet.log import setLogLevel, info

def emptyNet():

    "Create an empty network and add nodes to it."

    net = Mininet( controller=Controller, waitConnected=True )

    info( '*** Adding controller\n' )
    net.addController( 'c0' )

    info( '*** Adding hosts\n' )
    h1 = net.addHost( 'h1', ip='10.0.0.1' )
    h2 = net.addHost( 'h2', ip='10.0.0.2' )

    info( '*** Adding switch\n' )
    s3 = net.addSwitch( 's3' )

    info( '*** Creating links\n' )
    net.addLink( h1, s3 )
    net.addLink( h2, s3 )

    info( '*** Starting network\n' )
    net.start()

    info( '*** Running CLI\n' )
    CLI( net )

    info( '*** Stopping network' )
    net.stop()

if __name__ == '__main__':
    setLogLevel( 'info' )
    emptyNet()
```

Создание простейшей топологии сети



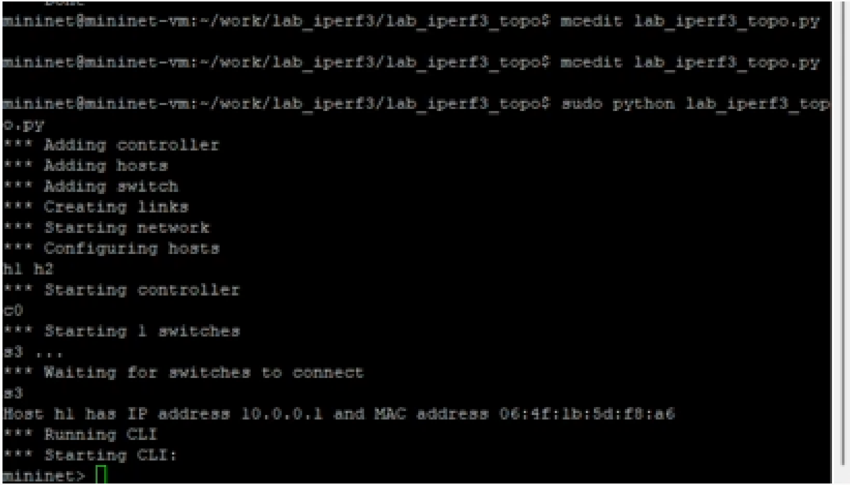
```
mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ sudo python lab_iperf3_topo.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
*** Starting network
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s3 ...
*** Waiting for switches to connect
s3
*** Running CLI
*** Starting CLI:
mininet> net
h1 h1-eth0:s3-eth1
h2 h2-eth0:s3-eth2
s3 lo:  s3-eth1:h1-eth0 s3-eth2:h2-eth0
c0
mininet> links
h1-eth0<->s3-eth1 (OK OK)
h2-eth0<->s3-eth2 (OK OK)
mininet> dump
<Host h1: h1-eth0:10.0.0.1 pid=877>
<Host h2: h2-eth0:10.0.0.2 pid=881>
<OVSSwitch s3: lo:127.0.0.1,s3-eth1:None,s3-eth2:None pid=886>
<Controller c0: 127.0.0.1:6653 pid=870>
mininet> exit
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s3
*** Stopping 2 hosts
h1 h2
*** Done
mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$
```

Внесение изменений в скрипт



```
/home/mi~_topo.py   [----]  0 L:[  1+35  36/ 45] *(805 / 972b) 0010 0x00A [*][X]
#! /usr/bin/env python

"""
Simple topology
"""

from mininet.net import Mininet
from mininet.node import Controller
from mininet.cli import CLI
from mininet.log import setLogLevel, info

def emptyNet():

    "Create an empty network and add nodes to it."

    net = Mininet( controller=Controller, waitConnected=True )

    info( '*** Adding controller\n' )
    net.addController( 'c0' )

    info( '*** Adding hosts\n' )
    h1 = net.addHost( 'h1', ip='10.0.0.1' )
    h2 = net.addHost( 'h2', ip='10.0.0.2' )

    info( '*** Adding switch\n' )
    s3 = net.addSwitch( 's3' )

    info( '*** Creating links\n' )
    net.addLink( h1, s3 )
    net.addLink( h2, s3 )

    info( '*** Starting network\n' )
    net.start()

    print( "Host", h1.name, "has IP address", h1.IP(), "and MAC address", h1.MAC


    info( '*** Running CLI\n' )
    CLI( net )

    info( '*** Stopping network' )
    net.stop()

if __name__ == '__main__':
    setLogLevel( 'info' )
    emptyNet()
```

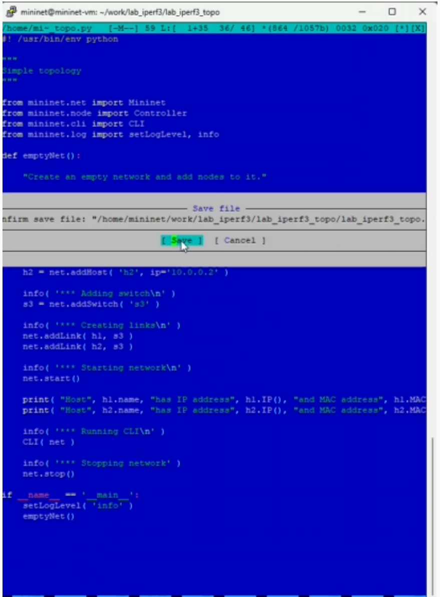
Проверка



```
mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ mcedit lab_iperf3_topo.py

mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ mcedit lab_iperf3_topo.py

mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ sudo python lab_iperf3_top
o.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
*** Starting network
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s3 ...
*** Waiting for switches to connect
s3
Host h1 has IP address 10.0.0.1 and MAC address 06:4f:1b:5d:f8:a6
*** Running CLI
*** Starting CLI:
mininet> 
```

Рис. 5: Проверка корректности отработки скрипта

Внесение изменений в скрипт

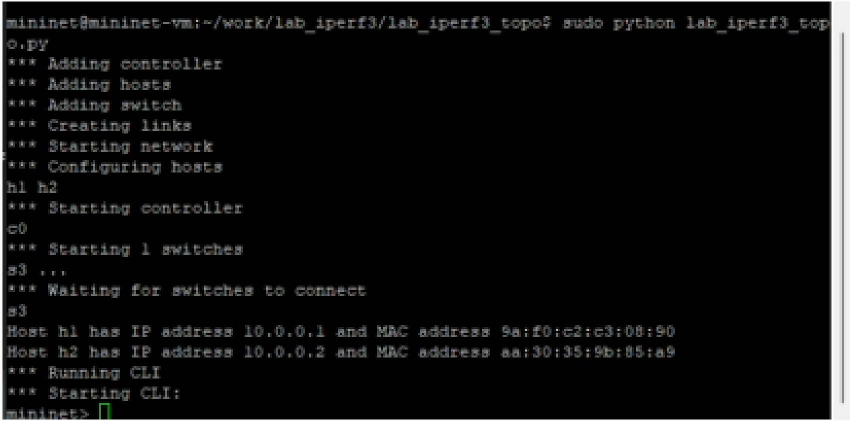


```
/home/mi~_topo.py   [-M--] 59 L:[  1+35  36/ 46] *(864 /1057b) 0032 0x020 [*][X]
#! /usr/bin/env python

"""
Simple topology
"""

from mininet.net import Mininet
from mininet.node import Controller
from mininet.cli import CLI
from mininet.log import setLogLevel, info

def emptyNet():

    "Create an empty network and add nodes to it."
```

```
Save file
nfirm save file: "/home/mininet/work/lab_iperf3/lab_iperf3_topo/lab_iperf3_topo.
```

```
    h2 = net.addHost( 'h2', ip='10.0.0.2' )

    info( '*** Adding switch\n' )
    s3 = net.addSwitch( 's3' )

    info( '*** Creating links\n' )
    net.addLink( h1, s3 )
    net.addLink( h2, s3 )

    info( '*** Starting network\n' )
    net.start()

    print( "Host", h1.name, "has IP address", h1.IP(), "and MAC address", h1.MAC
    print( "Host", h2.name, "has IP address", h2.IP(), "and MAC address", h2.MAC

    info( '*** Running CLI\n' )
    CLI( net )

    info( '*** Stopping network' )
    net.stop()

if __name__ == '__main__':
    setLogLevel( 'info' )
    emptyNet()
```

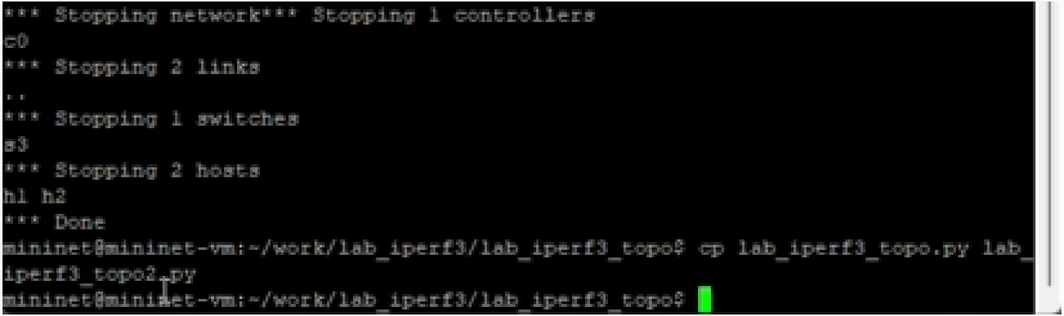
Проверка



```
mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ sudo python lab_iperf3_top
o.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
*** Starting network
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s3 ...
*** Waiting for switches to connect
s3
Host h1 has IP address 10.0.0.1 and MAC address 9a:f0:c2:c3:08:90
Host h2 has IP address 10.0.0.2 and MAC address aa:30:35:9b:85:a9
*** Running CLI
*** Starting CLI:
mininet> 
```

Рис. 7: Проверка корректности отработки скрипта

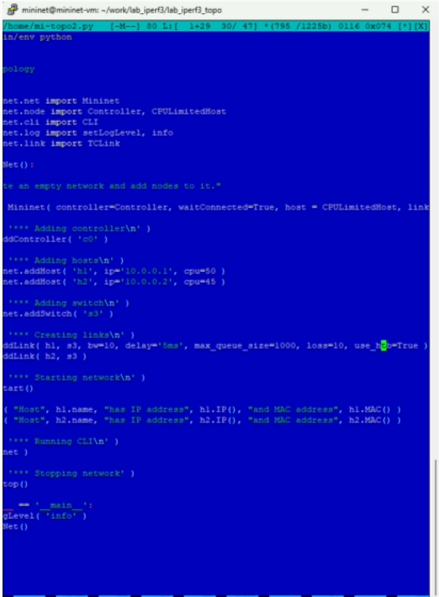
Добавление в скрипт настроек параметров производительности



```
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s3
*** Stopping 2 hosts
h1 h2
*** Done
mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ cp lab_iperf3_topo.py lab_
iperf3_topo2.py
mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ 
```

Рис. 8: Создание копии скрипта lab_iperf3_topo.py

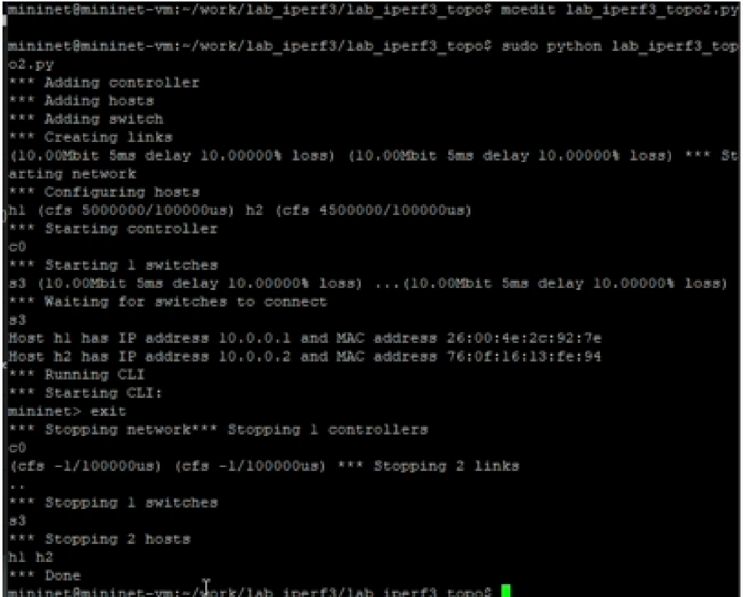
Добавление в скрипт настроек параметров производительности



mininet@mininet-vm: ~/work/lab_iperf3/lab_iperf3_topo

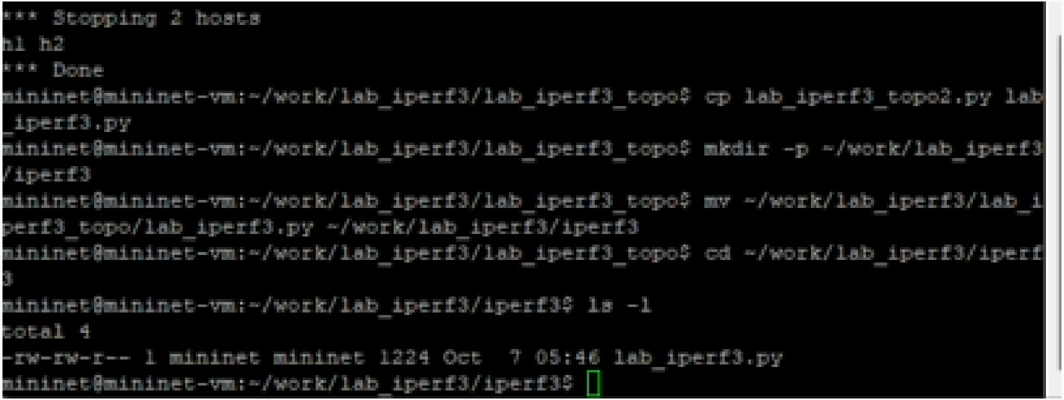
```
/home/mi-topo2.py   [-M--] 80 L:[  1+29  30/ 47] *(795 /1225b) 0116 0x074 [*][X]
in/env python


pology


net.net import Mininet
net.node import Controller, CPULimitedHost
net.cli import CLI
net.log import setLogLevel, info
net.link import TCLink

Net():

te an empty network and add nodes to it."

 Mininet( controller=Controller, waitConnected=True, host = CPULimitedHost, link

 '*** Adding controller\n' )
ddController( 'c0' )

 '*** Adding hosts\n' )
net.addHost( 'h1', ip='10.0.0.1', cpu=50 )
net.addHost( 'h2', ip='10.0.0.2', cpu=45 )

 '*** Adding switch\n' )
net.addSwitch( 's3' )

 '*** Creating links\n' )
ddLink( h1, s3, bw=10, delay='5ms', max_queue_size=1000, loss=10, use_htb=True )
ddLink( h2, s3 )

 '*** Starting network\n' )
tart()

( "Host", h1.name, "has IP address", h1.IP(), "and MAC address", h1.MAC() )
( "Host", h2.name, "has IP address", h2.IP(), "and MAC address", h2.MAC() )

 '*** Running CLI\n' )
net )

 '*** Stopping network' )
top()

__ == '__main__':
gLevel( 'info' )
Net()
```

Добавление в скрипт настроек параметров производительности



```
mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ mcedit lab_iperf3_topo2.py

mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ sudo python lab_iperf3_top
o2.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
(10.00Mbit 5ms delay 10.00000% loss) (10.00Mbit 5ms delay 10.00000% loss) *** St
arting network
*** Configuring hosts
h1 (cfs 5000000/100000us) h2 (cfs 4500000/100000us)
*** Starting controller
c0
*** Starting 1 switches
s3 (10.00Mbit 5ms delay 10.00000% loss) ...(10.00Mbit 5ms delay 10.00000% loss)
*** Waiting for switches to connect
s3
Host h1 has IP address 10.0.0.1 and MAC address 26:00:4e:2c:92:7e
Host h2 has IP address 10.0.0.2 and MAC address 76:0f:16:13:fe:94
*** Running CLI
*** Starting CLI:
mininet> exit
*** Stopping network*** Stopping 1 controllers
c0
(cfs -1/100000us) (cfs -1/100000us) *** Stopping 2 links
..
*** Stopping 1 switches
s3
*** Stopping 2 hosts
h1 h2
*** Done
mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$
```

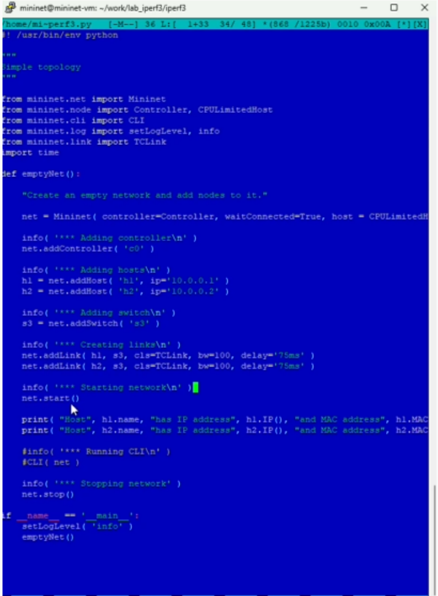
Добавление в скрипт настроек параметров производительности



```
*** Stopping 2 hosts
h1 h2
*** Done
mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ cp lab_iperf3_topo2.py lab
_iperf3.py
mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ mkdir -p ~/work/lab_iperf3
/iperf3
mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ mv ~/work/lab_iperf3/lab_i
perf3_topo/lab_iperf3.py ~/work/lab_iperf3/iperf3
mininet@mininet-vm:~/work/lab_iperf3/lab_iperf3_topo$ cd ~/work/lab_iperf3/iperf
3
mininet@mininet-vm:~/work/lab_iperf3/iperf3$ ls -l
total 4
-rw-rw-r-- 1 mininet mininet 1224 Oct  7 05:46 lab_iperf3.py
mininet@mininet-vm:~/work/lab_iperf3/iperf3$ 
```

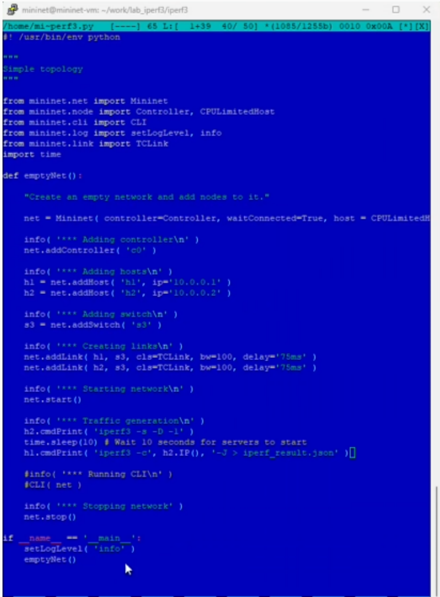
Рис. 11: Создание копии скрипта lab_iperf3_topo2.py и его дальнейшее помещение в подкаталог iperf

Добавление в скрипт настроек параметров производительности



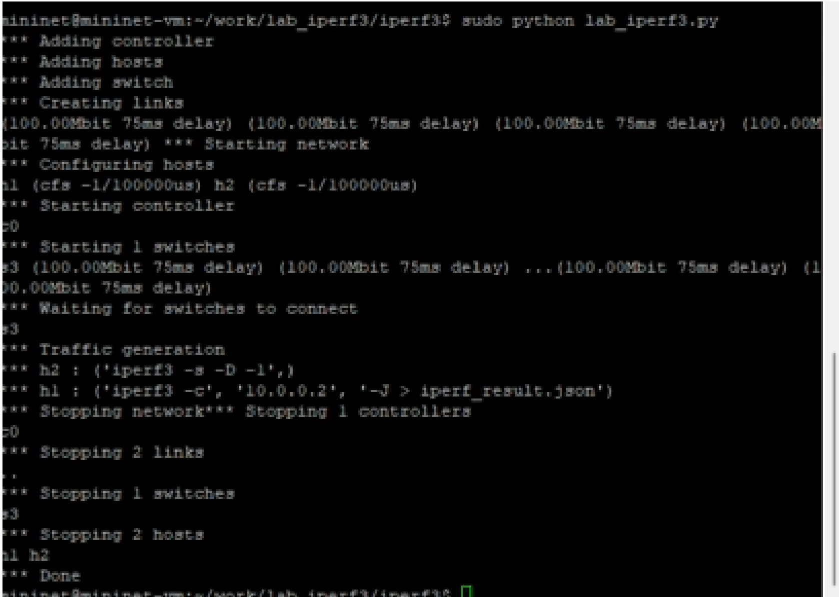
```
/home/mi-perf3.py   [-M--] 36 L:[  1+33  34/ 48] *(868 /1225b) 0010 0x00A [*][X]
#! /usr/bin/env python

"""
Simple topology
"""

from mininet.net import Mininet
from mininet.node import Controller, CPULimitedHost
from mininet.cli import CLI
from mininet.log import setLogLevel, info
from mininet.link import TCLink
import time

def emptyNet():

    "Create an empty network and add nodes to it."

    net = Mininet( controller=Controller, waitConnected=True, host = CPULimitedH

    info( '*** Adding controller\n' )
    net.addController( 'c0' )

    info( '*** Adding hosts\n' )
    h1 = net.addHost( 'h1', ip='10.0.0.1' )
    h2 = net.addHost( 'h2', ip='10.0.0.2' )

    info( '*** Adding switch\n' )
    s3 = net.addSwitch( 's3' )

    info( '*** Creating links\n' )
    net.addLink( h1, s3, cls=TCLink, bw=100, delay='75ms' )
    net.addLink( h2, s3, cls=TCLink, bw=100, delay='75ms' )

    info( '*** Starting network\n' )
    net.start()

    print( "Host", h1.name, "has IP address", h1.IP(), "and MAC address", h1.MAC
    print( "Host", h2.name, "has IP address", h2.IP(), "and MAC address", h2.MAC

    #info( '*** Running CLI\n' )
    #CLI( net )

    info( '*** Stopping network' )
    net.stop()

if __name__ == '__main__':
    setLogLevel( 'info' )
    emptyNet()
```

Добавление в скрипт настроек параметров производительности



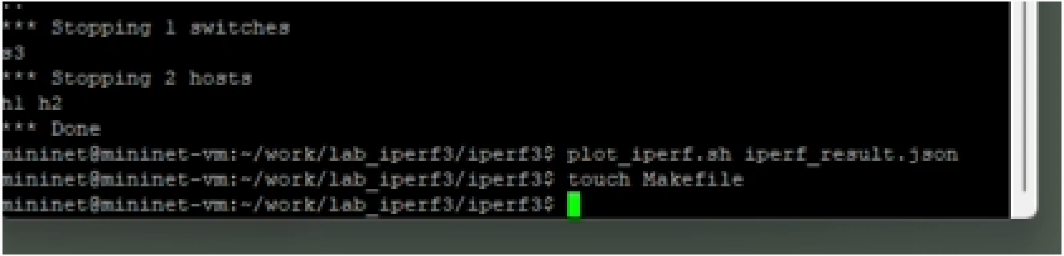
```
/home/mi-perf3.py   [----] 65 L:[  1+39  40/ 50] *(1085/1255b) 0010 0x00A [*][X]
#! /usr/bin/env python

"""
Simple topology
"""

from mininet.net import Mininet
from mininet.node import Controller, CPULimitedHost
from mininet.cli import CLI
from mininet.log import setLogLevel, info
from mininet.link import TCLink
import time

def emptyNet():

    "Create an empty network and add nodes to it."

    net = Mininet( controller=Controller, waitConnected=True, host = CPULimitedH

    info( '*** Adding controller\n' )
    net.addController( 'c0' )

    info( '*** Adding hosts\n' )
    h1 = net.addHost( 'h1', ip='10.0.0.1' )
    h2 = net.addHost( 'h2', ip='10.0.0.2' )

    info( '*** Adding switch\n' )
    s3 = net.addSwitch( 's3' )

    info( '*** Creating links\n' )
    net.addLink( h1, s3, cls=TCLink, bw=100, delay='75ms' )
    net.addLink( h2, s3, cls=TCLink, bw=100, delay='75ms' )

    info( '*** Starting network\n' )
    net.start()

    info( '*** Traffic generation\n' )
    h2.cmdPrint( 'iperf3 -s -D -1' )
    time.sleep(10) # Wait 10 seconds for servers to start
    h1.cmdPrint( 'iperf3 -c', h2.IP(), '-J > iperf_result.json' )

    #info( '*** Running CLI\n' )
    #CLI( net )

    info( '*** Stopping network' )
    net.stop()

if __name__ == '__main__':
    setLogLevel( 'info' )
    emptyNet()
```

Добавление в скрипт настроек параметров производительности



```
mininet@mininet-vm:~/work/lab_iperf3/iperf3$ sudo python lab_iperf3.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
(100.00Mbit 75ms delay) (100.00Mbit 75ms delay) (100.00Mbit 75ms delay) (100.00M
bit 75ms delay) *** Starting network
*** Configuring hosts
h1 (cfs -1/100000us) h2 (cfs -1/100000us)
*** Starting controller
c0
*** Starting 1 switches
s3 (100.00Mbit 75ms delay) (100.00Mbit 75ms delay) ...(100.00Mbit 75ms delay) (1
00.00Mbit 75ms delay)
*** Waiting for switches to connect
s3
*** Traffic generation
*** h2 : ('iperf3 -s -D -1',)
*** h1 : ('iperf3 -c', '10.0.0.2', '-J > iperf_result.json')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s3
*** Stopping 2 hosts
h1 h2
*** Done
mininet@mininet-vm:~/work/lab_iperf3/iperf3$ 
```

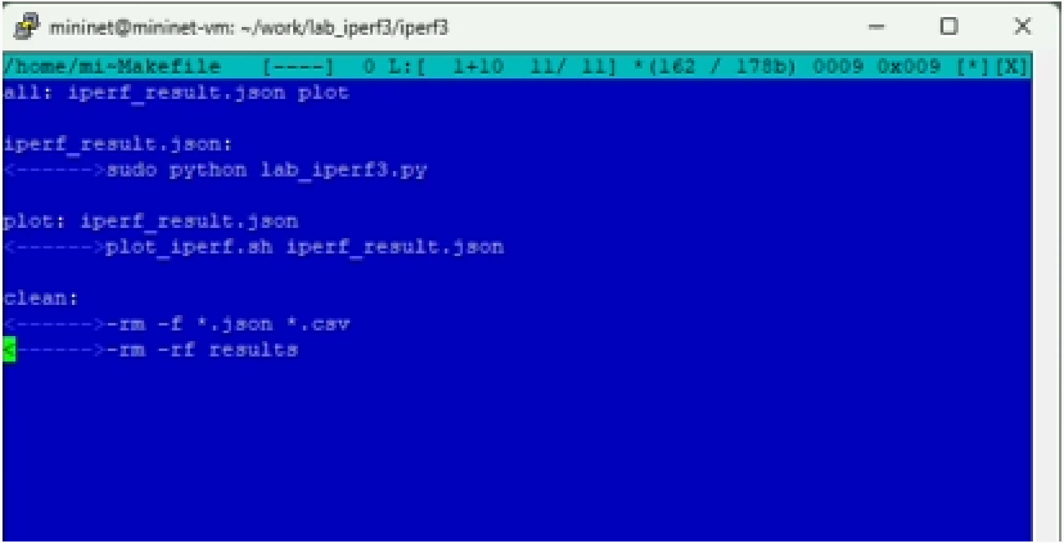
Построение графиков по проводимому эксперименту



```
*** Stopping 1 switches
s3
*** Stopping 2 hosts
h1 h2
*** Done
mininet@mininet-vm:~/work/lab_iperf3/iperf3$ plot_iperf.sh iperf_result.json
mininet@mininet-vm:~/work/lab_iperf3/iperf3$ touch Makefile
mininet@mininet-vm:~/work/lab_iperf3/iperf3$ 
```

Рис. 15: Построение графиков и создание Makefile для проведения всего эксперимента

Построение графиков по проводимому эксперименту

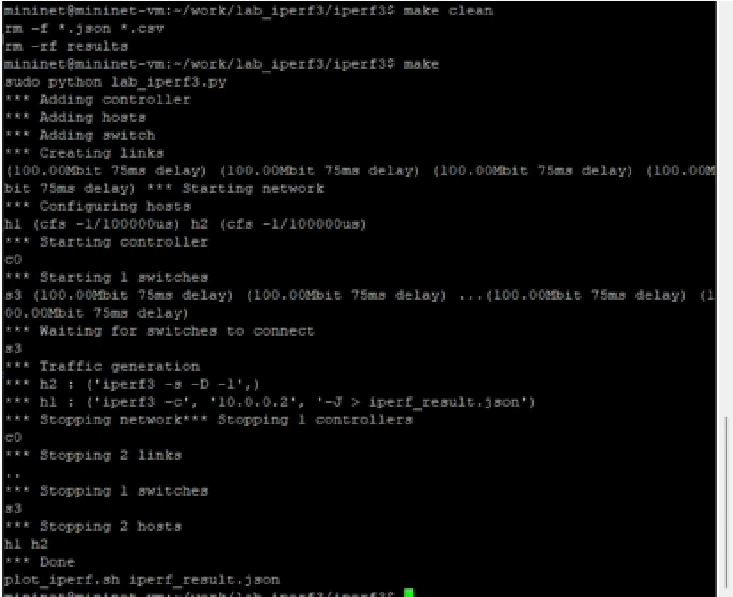


mininet@mininet-vm: ~/work/lab_iperf3/iperf3

```
/home/mi~Makefile    [----]  0 L:[  1+10  11/ 11] *(162 / 178b) 0009 0x009 [*][X]
all: iperf_result.json plot

iperf_result.json:
<------->sudo python lab_iperf3.py

plot: iperf_result.json
<------->plot_iperf.sh iperf_result.json

clean:
<------->-rm -f *.json *.csv
<------->-rm -rf results
```

Рис. 16: Добавление скрипта в Makefile

Построение графиков по проводимому эксперименту



```
mininet@mininet-vm:~/work/lab_iperf3/iperf3$ make clean
rm -f *.json *.csv
rm -rf results
mininet@mininet-vm:~/work/lab_iperf3/iperf3$ make
sudo python lab_iperf3.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
(100.00Mbit 75ms delay) (100.00Mbit 75ms delay) (100.00Mbit 75ms delay) (100.00M
bit 75ms delay) *** Starting network
*** Configuring hosts
h1 (cfs -1/100000us) h2 (cfs -1/100000us)
*** Starting controller
c0
*** Starting 1 switches
s3 (100.00Mbit 75ms delay) (100.00Mbit 75ms delay) ...(100.00Mbit 75ms delay) (1
00.00Mbit 75ms delay)
*** Waiting for switches to connect
s3
*** Traffic generation
*** h2 : ('iperf3 -s -D -1',)
*** h1 : ('iperf3 -c', '10.0.0.2', '-J > iperf_result.json')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s3
*** Stopping 2 hosts
h1 h2
*** Done
plot_iperf.sh iperf_result.json
mininet@mininet-vm:~/work/lab_iperf3/iperf3$
```

Вывод

Вывод

- В ходе выполнения лабораторной работы познакомились с инструментом для измерения пропускной способности сети в режиме реального времени — iPerf3, а также получили навыки проведения воспроизводимого эксперимента по измерению пропускной способности моделируемой сети в среде Mininet

Список литературы. Библиография

Список литературы. Библиография

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

Вывод

Вывод

- В ходе выполнения лабораторной работы познакомились с инструментом для измерения пропускной способности сети в режиме реального времени — iPerf3, а также получили навыки проведения воспроизводимого эксперимента по измерению пропускной способности моделируемой сети в среде Mininet

Список литературы. Библиография

Список литературы. Библиография

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