

Лабораторная работа 4

Эмуляция и измерение задержек в глобальных сетях

Ланцова Я. И.

Российский университет дружбы народов, Москва, Россия

Информация

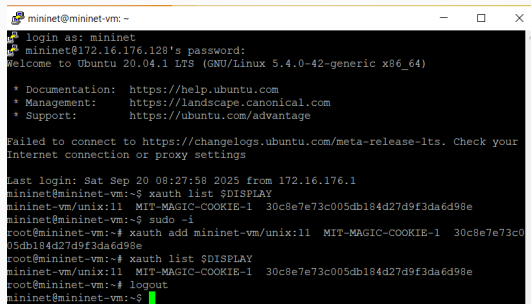
- Ланцова Яна Игоревна
- студентка
- Российский университет дружбы народов

Основной целью работы является знакомство с NETEM — инструментом для тестирования производительности приложений в виртуальной сети, а также получение навыков проведения интерактивного и воспроизводимого экспериментов по измерению задержки и её дрожания (jitter) в моделируемой сети в среде Mininet.

1. Задайте простейшую топологию, состоящую из двух хостов и коммутатора с назначенной по умолчанию mininet сетью 10.0.0.0/8.
2. Проведите интерактивные эксперименты по добавлению/изменению задержки, джиттера, значения корреляции для джиттера и задержки, распределения времени задержки в эмулируемой глобальной сети.

3. Реализуйте воспроизводимый эксперимент по заданию значения задержки в эмулируемой глобальной сети. Постройте график.
4. Самостоятельно реализуйте воспроизводимые эксперименты по изменению задержки, джиттера, значения корреляции для джиттера и задержки, распределения времени задержки в эмулируемой глобальной сети. Постройте графики.

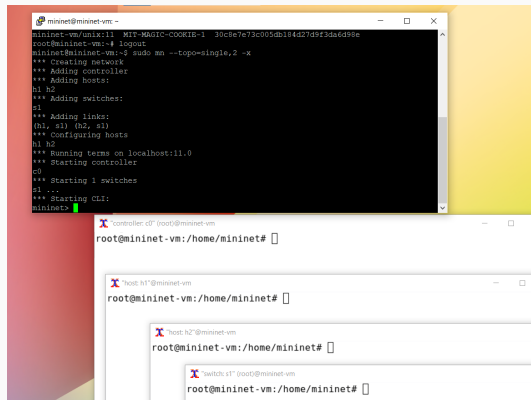
Выполнение лабораторной работы

A terminal window titled 'mininet@mininet-vm: ~' with standard window controls. The terminal shows a login sequence for 'mininet' on '172.16.176.128' using Ubuntu 20.04.1 LTS. It displays system information and links for documentation, management, and support. A message indicates a failed connection to the Ubuntu changelogs. The user then runs 'xauth list \$DISPLAY', showing a single MIT-MAGIC-COOKIE entry. They then run 'sudo -i' to become root. As root, they run 'xauth add mininet-vm/unix:11 MIT-MAGIC-COOKIE-1 30c8e7e73c005db184d27d9f3da6d98e'. Finally, they run 'xauth list \$DISPLAY' again, showing the updated list with two entries. They then run 'logout' and return to the 'mininet' prompt.

```
mininet@mininet-vm: ~  
login as: mininet  
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: Sat Sep 20 08:27:58 2025 from 172.16.176.1  
mininet@mininet-vm:~$ xauth list $DISPLAY  
mininet-vm/unix:11 MIT-MAGIC-COOKIE-1 30c8e7e73c005db184d27d9f3da6d98e  
mininet@mininet-vm:~$ sudo -i  
root@mininet-vm:~# xauth add mininet-vm/unix:11 MIT-MAGIC-COOKIE-1 30c8e7e73c0  
05db184d27d9f3da6d98e  
root@mininet-vm:~# xauth list $DISPLAY  
mininet-vm/unix:11 MIT-MAGIC-COOKIE-1 30c8e7e73c005db184d27d9f3da6d98e  
root@mininet-vm:~# logout  
mininet@mininet-vm:~$
```

Рис. 1: Исправление MIT magic cookie

Выполнение лабораторной работы



```
mininet@mininet-vm: -
mininet-vm/unix:11 MIT-MAGIC-COOKIE-1 30c8e7e73c005db184d27d9f3da6d98e
root@mininet-vm:~# logout
mininet@mininet-vm:~$ sudo mn --topo=single,2 -x
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
*** Configuring hosts
h1 h2
*** Running terms on localhost:11.0
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet>
```

controller: c0 (root@mininet-vm)

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

host: h1 (root@mininet-vm)

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

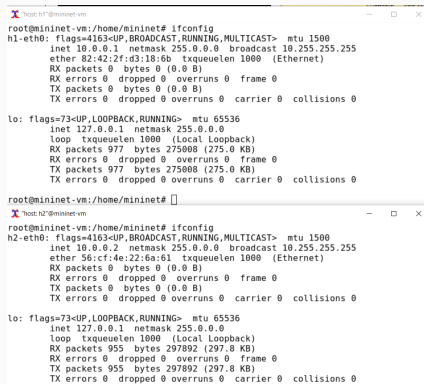
host: h2 (root@mininet-vm)

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

switch: s1 (root@mininet-vm)

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

Рис. 2: Простейшая топология



The image shows two terminal windows side-by-side. The top window is titled 'host: h1@mininet-vm' and shows the output of the 'ifconfig' command for host h1. The bottom window is titled 'host: h2@mininet-vm' and shows the output of the 'ifconfig' command for host h2. Both windows show configuration for the 'eth0' interface and the loopback 'lo' interface.

```
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 82:42:2f:d3:18:6b 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 977 bytes 275008 (275.0 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 977 bytes 275008 (275.0 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@mininet-vm:/home/mininet#

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 56:cf:4e:22:6a:61 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 955 bytes 297892 (297.8 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 955 bytes 297892 (297.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Рис. 3: ifconfig на хостах h1 и h2

```
root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 6
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=23.8 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.904 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.104 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.152 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.211 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.220 ms
|
--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5079ms
rtt min/avg/max/mdev = 0.104/4.234/23.814/8.760 ms
```

Рис. 4: Проверка подключения между хостами

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 100ms
root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 6
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=103 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=103 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=102 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=101 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=101 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=100 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5008ms
rtt min/avg/max/mdev = 100.418/101.560/103.122/1.147 ms
```

Рис. 5: Добавление задержки в 100мс


Выполнение лабораторной работы

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h2-eth0 root netem delay 100ms
root@mininet-vm:/home/mininet# █

root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 6
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=206 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=204 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=202 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=202 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=202 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=202 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5011ms
rtt min/avg/max/mdev = 201.505/202.907/205.568/1.391 ms
root@mininet-vm:/home/mininet# ▢
```

Рис. 6: Двухнаправленная задержка соединения

```
root@mininet-vm:/home/mininet# sudo tc qdisc change dev h2-eth0 root netem delay 50ms
root@mininet-vm:/home/mininet# 
X host: h1" @mininet-vm
lay 50ms
1: command not found
root@mininet-vm:/home/mininet# sudo tc qdisc change dev h1-eth0 root netem dela
y 50ms
Error: Qdisc not found. To create specify NLM_F_CREATE flag.
root@mininet-vm:/home/mininet# sudo tc qdisc change dev h1-eth0 root netem dela
y 50ms
Error: Qdisc not found. To create specify NLM_F_CREATE flag.
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 100ms
root@mininet-vm:/home/mininet# sudo tc qdisc change dev h1-eth0 root netem delay 50ms
root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 6
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=210 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=103 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=102 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=101 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=102 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=103 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5013ms
rtt min/avg/max/mdev = 101.434/120.245/209.959/40.126 ms
```

Рис. 7: Изменение задержки на 50мс

```
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h2-eth0 root netem
X *host: h1*@mininet-vm
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=103 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=102 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=101 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=102 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=103 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5013ms
rtt min/avg/max/mdev = 101.434/120.245/209.959/40.126 ms
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 6
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=3.60 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=2.57 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.352 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.115 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.159 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.151 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5057ms
rtt min/ava/max/mdev = 0.115/1.157/3.600/1.396 ms
```

Рис. 8: Восстановление исходных значений задержки

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 100ms 10ms
root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 6
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=114 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=98.3 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=94.1 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=111 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=99.5 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=109 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5012ms
rtt min/avg/max/mdev = 94.050/104.215/113.511/7.245 ms
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet#
```

Рис. 9: Добавление значения дрожания задержки в интерфейс подключения

Выполнение лабораторной работы

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 100ms 10ms 25%
root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 20
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=95.6 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=109 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=101 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=96.7 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=103 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=105 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=97.4 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=103 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=98.0 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=99.6 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=99.0 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=111 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=109 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=105 ms
64 bytes from 10.0.0.2: icmp_seq=15 ttl=64 time=102 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=108 ms
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=99.0 ms
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=104 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=91.3 ms
64 bytes from 10.0.0.2: icmp_seq=20 ttl=64 time=93.7 ms

--- 10.0.0.2 ping statistics ---
20 packets transmitted, 20 received, 0% packet loss, time 19047ms
rtt min/avg/max/mdev = 91.287/101.498/111.086/5.185 ms
root@mininet-vm:/home/mininet#
```

Рис. 10: Добавление значения корреляции для джиттера и задержки в интерфейс подключения

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 100ms 20ms distribution normal
root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 10
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=79.5 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=116 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=61.3 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=108 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=109 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=98.7 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=115 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=130 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=96.1 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=114 ms

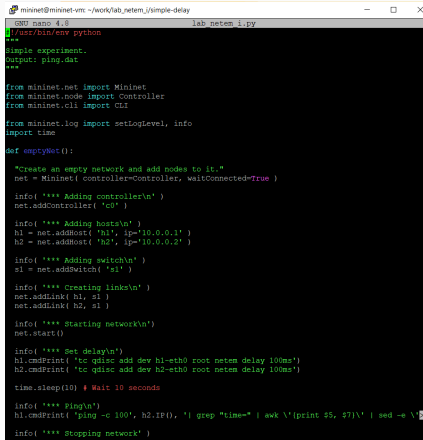
--- 10.0.0.2 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9021ms
rtt min/avg/max/mdev = 61.331/102.699/129.986/18.905 ms
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet#
```

Рис. 11: Распределение задержки в интерфейсе подключения

```
mininet@mininet-vm:~$ mkdir -p ~/work/lab_netem_i/simple-delay
mininet@mininet-vm:~$ sudo apt install geeqie
Reading package lists... Done
Building dependency tree
Reading state information... Done
geeqie is already the newest version (1:1.5.1-8build1).
0 upgraded, 0 newly installed, 0 to remove and 374 not upgraded.
mininet@mininet-vm:~$
```

Рис. 12: Установка пакета и создание каталога

Выполнение лабораторной работы



```
mininet@mininet-vm: ~/work/lab_netem/jsimple-delay
GNU nano 4.6 lab_netem_i.py
#!/usr/bin/env python
'''
Simple experiment.
Output: ping.dat
'''

from mininet.net import Mininet
from mininet.node import Controller
from mininet.cli import CLI

from mininet.log import setLogLevel, info
import time

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' )
    s1 = net.addSwitch( 's1' )

    info( '*** Creating links\n' )
    net.addLink( h1, s1 )
    net.addLink( h2, s1 )

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

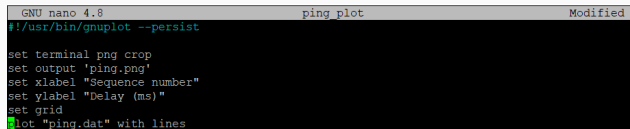
    info( '*** Set delay\n' )
    h1.cmdPrint( 'tc qdisc add dev h1-eth0 root netem delay 100ms' )
    h2.cmdPrint( 'tc qdisc add dev h2-eth0 root netem delay 100ms' )

    time.sleep(10) # Wait 10 seconds

    info( '*** Ping\n' )
    h1.cmdPrint( 'ping -c 100', h2.IP(), '| grep "time=" | awk '{print $5, $7}' | sed -e \'

    info( '*** Stopping network' )
```

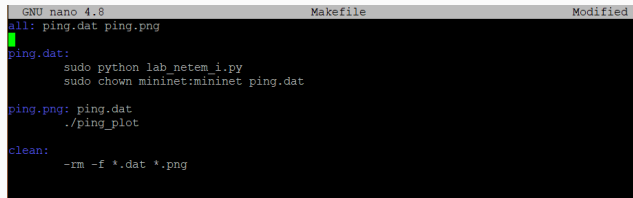
Рис. 13: Скрипт на Python для эксперимента



```
GNU nano 4.8                               ping_plot                               Modified
#!/usr/bin/gnuplot --persist

set terminal png crop
set output 'ping.png'
set xlabel "Sequence number"
set ylabel "Delay (ms)"
set grid
plot "ping.dat" with lines
```

Рис. 14: Скрипт для визуализации ping_plot



```
GNU nano 4.8 Makefile Modified
all: ping.dat ping.png
ping.dat:
    sudo python lab_netem_i.py
    sudo chown mininet:mininet ping.dat
ping.png: ping.dat
    ./ping_plot
clean:
    -rm -f *.dat *.png
```

Рис. 15: Makefile для управления процессом проведения эксперимента

```
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ make
sudo python lab_netem_i.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
*** Starting network
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Waiting for switches to connect
s1
*** Set delay
*** h1 : ('tc qdisc add dev h1-eth0 root netem delay 100ms',)
*** h2 : ('tc qdisc add dev h2-eth0 root netem delay 100ms',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "time=" | awk \'{print $5, $7}\'' | sed -e \'s/time=//g\' -e \'s/icmp_seq=//g\' > ping.dat')
```

Рис. 16: Запуск эксперимента

Выполнение лабораторной работы

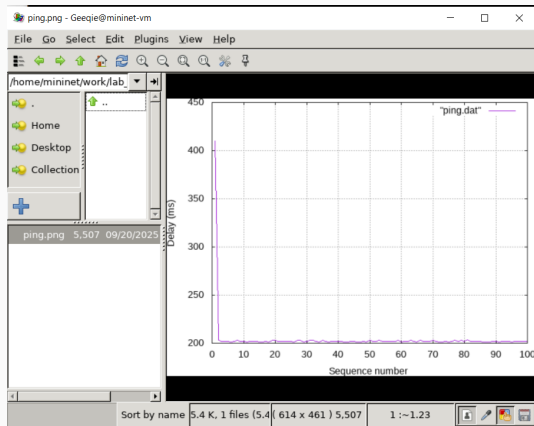
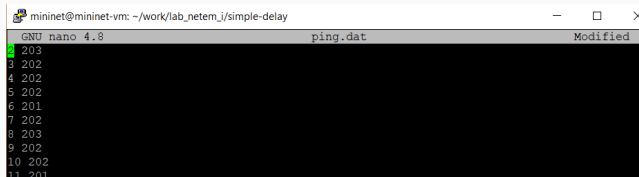


Рис. 17: Визуализация эксперимента



```
mininet@mininet-vm: ~/work/lab_netem_i/simple-delay
GNU nano 4.8 ping.dat Modified
2 203
3 202
4 202
5 202
6 201
7 202
8 203
9 202
10 202
11 201
```

Рис. 18: Удаление строки из файла .dat

Выполнение лабораторной работы

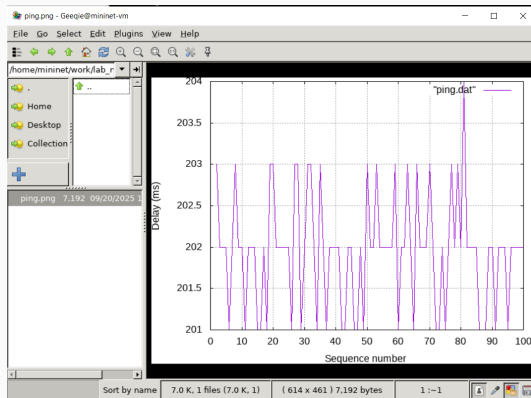
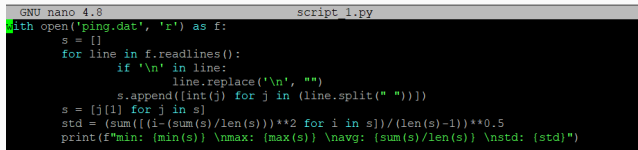


Рис. 19: Визуализация эксперимента

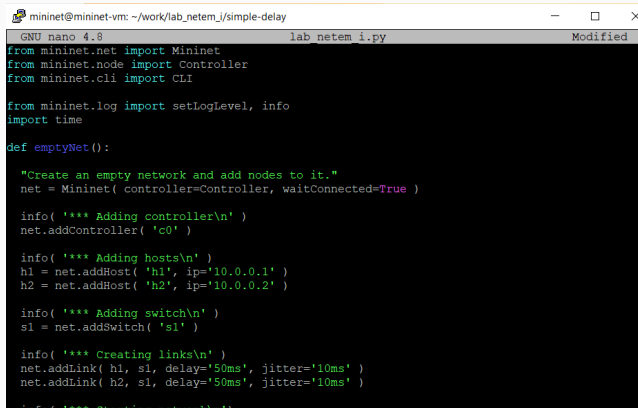


```
GNU nano 4.8 script_1.py
with open('ping.dat', 'r') as f:
    s = []
    for line in f.readlines():
        if '\n' in line:
            line.replace('\n', "")
            s.append([int(j) for j in (line.split(" "))])
    s = [j[1] for j in s]
    std = (sum([(i-(sum(s)/len(s)))**2 for i in s])/(len(s)-1))**0.5
    print(f"min: {min(s)} \nmax: {max(s)} \navg: {sum(s)/len(s)} \nstd: {std}")
```

Рис. 20: Скрипт script_1.py

```
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ sudo python script_1.py  
min: 201  
max: 204  
avg: 201.94949494949495  
std: 0.6757269309257353
```

Рис. 21: Результат работы скрипта script_1.py



```
mininet@mininet-vm: ~/work/lab_netem_i/simple-delay
GNU nano 4.8 lab_netem_i.py Modified
from mininet.net import Mininet
from mininet.node import Controller
from mininet.cli import CLI

from mininet.log import setLogLevel, info
import time

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' )
    s1 = net.addSwitch( 's1' )

    info( '*** Creating links\n' )
    net.addLink( h1, s1, delay='50ms', jitter='10ms' )
    net.addLink( h2, s1, delay='50ms', jitter='10ms' )

    info( '*** Starting network\n' )
```

Рис. 22: Изменение файла lab_netem_i.py

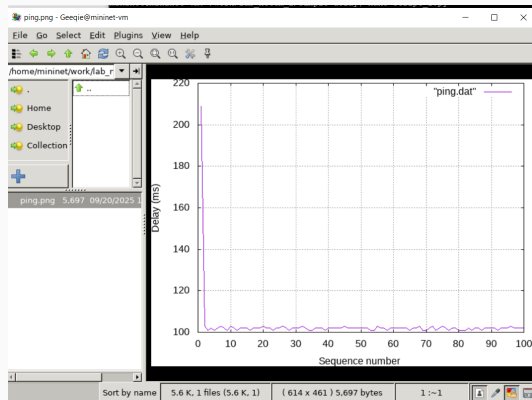


Рис. 23: Визуализация эксперимента

```
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ sudo python script_1.py  
min: 101  
max: 209  
avg: 102.95  
std: 10.72839619181194  
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
```

Рис. 24: Результат работы скрипта script_1.py

Выводы

В результате выполнения данной лабораторной работы я познакомилась с NETEM – инструментом для тестирования производительности приложений в виртуальной сети, а также получила навыки проведения интерактивного и воспроизводимого экспериментов по измерению задержки и её дрожания (jitter) в моделируемой сети в среде Mininet.