

Отчёт по лабораторной работе №4

Эмуляция и измерение задержек в глобальных сетях

Лушин Артём Андреевич

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Вывод	21

Список иллюстраций

2.1	Права запуска	6
2.2	Простейшая топология	6
2.3	Адреса хостов	7
2.4	Проверка подключений	7
2.5	Задержка на хосте 1	7
2.6	Соединение от хоста 1	7
2.7	Задержка на хосте 2	8
2.8	Проверка соединений между хостами	8
2.9	Понижение задержка на хостах	8
2.10	Соединение между хостами	8
2.11	Восстановление конфигурации на компах	9
2.12	Проверка без задержке	9
2.13	Задержка с отклонением	9
2.14	Проверка соединений с отклонением	9
2.15	Восстановление конфигурации	10
2.16	Добавление корреляции	10
2.17	Отправка пакетов с корреляцией	10
2.18	Нормальное распределение на хосте	10
2.19	Пингование с нормальным распределением	11
2.20	Обновление репозитория	11
2.21	Установка geeqie	11
2.22	Создание каталога	12
2.23	Скрипт lab_netem_i	12
2.24	Скрипт для визуализации	12
2.25	Права доступа	13
2.26	Makefile	13
2.27	Запуск эксперимента	13
2.28	График после первого эксперимента	14
2.29	Удаление данных из файла	14
2.30	Построение нового графика	15
2.31	Собственный скрипт	15
2.32	Изменение Makefile	16
2.33	Демонстрация результата	16
2.34	Изменение задержек	16
2.35	График к новому скрипту	17
2.36	Результаты скрипта	17
2.37	Добавление отклонения	17

2.38 Построение графика	18
2.39 Результаты с отклонением	18
2.40 Эксперимент с корреляцией	18
2.41 График с корреляцией	19
2.42 Результаты работы с корреляцией	19
2.43 Эксперимент с нормальным распределением	19
2.44 График с нормальным распределением	20
2.45 Результат работы с нормальным распределением	20

1 Цель работы

Основной целью работы является знакомство с NETEM - инструментов для тестирования производительности приложений в визуальной сети, а также получение навыков проведения интерактивного и воспроизводимого экспериментов по измерению задержки и её дрожания в моделируемой сети в среде Mininet.

2 Выполнение лабораторной работы

- 1) Я запустил виртуальную среду с mininet и подключился к основной ос. В виртуальной машине исправил права запуска. Скопировал значение куки своего пользователя для пользователя рут.

```
mininet@mininet-vm:~$ xauth list $DISPLAY
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 6dd50f8007da69cd71606cae34fee113
mininet@mininet-vm:~$ sudo -i
sudo-i: command not found
mininet@mininet-vm:~$ sudo -i
root@mininet-vm:~# xauth add mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 6dd50f8007da69cd71606cae34fee113
root@mininet-vm:~# logout
mininet@mininet-vm:~$
```

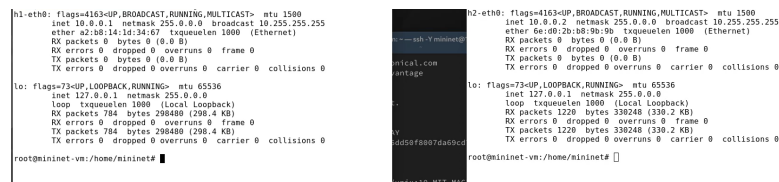
Рис. 2.1: Права запуска

- 2) Задал простейшую топологию, состоящую из двух хостов, коммутатора и контроллера.

```
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:10.0
*** Starting controller
c0
*** Starting 1 switches
s1 ...
```

Рис. 2.2: Простейшая топология

3) На хостах ввёл команду, что определения IP.



The image shows two terminal windows side-by-side. The left window shows the configuration for host h1, and the right window shows the configuration for host h2. Both hosts are configured with IP 10.0.0.1, netmask 255.0.0.0, and broadcast 10.255.255.255. The left window also shows the configuration for the loopback interface lo (127.0.0.1).

```
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 a2:8b:14:1b:34:67 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 784 bytes 208400 (208.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 784 bytes 208400 (208.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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

```
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 6a:d0:2b:b8:9b:9b 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 1220 bytes 330240 (330.2 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1220 bytes 330240 (330.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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

Рис. 2.3: Адреса хостов

4) Проверил подключение между хостами с помощью команды пинг. Минимальная задержка - 0,034. Среднее - 2,11. Максимальная 7,72.



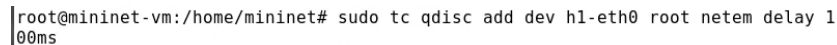
The image shows two terminal windows side-by-side. The left window shows the output of a ping command from host h1 to host h2 (10.0.0.2). The right window shows the output of a ping command from host h2 to host h1 (10.0.0.1). Both show 6 ping requests with varying times.

```
root@mininet-vn:/home/mininet# ping -c 6 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=2.28 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.039 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.060 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.139 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.002 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.059 ms
```

```
root@mininet-vn:/home/mininet# ping -c 6 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=7.72 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.166 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.179 ms
64 bytes from 10.0.0.1: icmp_seq=4 ttl=64 time=0.036 ms
64 bytes from 10.0.0.1: icmp_seq=5 ttl=64 time=0.034 ms
64 bytes from 10.0.0.1: icmp_seq=6 ttl=64 time=0.045 ms
```

Рис. 2.4: Проверка подключений

5) На хосте 1 добавил задержку 100мс.

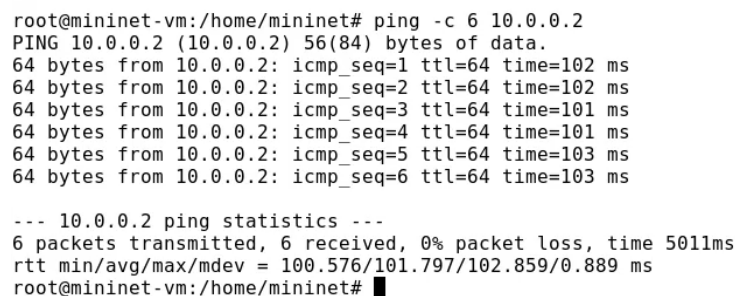


The image shows a terminal window where the command 'sudo tc qdisc add dev h1-eth0 root netem delay 100ms' is entered to add a 100ms delay to the network interface h1-eth0.

```
root@mininet-vn:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 100ms
```

Рис. 2.5: Задержка на хосте 1

6) Проверил соединение от хоста 1 к хосту 2. Минимальная задержка - 101. Средняя - 102. Максимальная - 103.



The image shows a terminal window where the command 'ping -c 6 10.0.0.2' is entered. The output shows 6 ping requests with times ranging from 101ms to 103ms. Below the ping results, the 'ping statistics' are displayed, showing 6 packets transmitted, 6 received, 0% packet loss, and a total time of 5011ms.

```
root@mininet-vn:/home/mininet# ping -c 6 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=102 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=102 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=101 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=103 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5011ms
rtt min/avg/max/mdev = 100.576/101.797/102.859/0.889 ms
root@mininet-vn:/home/mininet#
```

Рис. 2.6: Соединение от хоста 1

7) На хосте 2 добавил такую же задержку 100 мс.

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

Рис. 2.7: Задержка на хосте 2

8) Проверил соединение. Теперь минимальная задержка - 201. Средняя - 202,4. Максимальная - 204.

```
root@mininet-vm:/home/mininet# ping -c 6 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=204 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=201 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=203 ms
64 bytes from 10.0.0.1: icmp_seq=4 ttl=64 time=202 ms
64 bytes from 10.0.0.1: icmp_seq=5 ttl=64 time=201 ms
64 bytes from 10.0.0.1: icmp_seq=6 ttl=64 time=201 ms

--- 10.0.0.1 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5008ms
rtt min/avg/max/mdev = 200.727/202.006/204.003/1.079 ms
root@mininet-vm:/home/mininet#
```

Рис. 2.8: Проверка соединений между хостами

9) Изменил задержку на двух хостах до 50 мс.

```
root@mininet-vm:/home/mininet# sudo tc qdisc change dev h1-eth0 root netem delay 50ms
root@mininet-vm:/home/mininet#
root@mininet-vm:/home/mininet# sudo tc qdisc change dev h2-eth0 root netem delay 50ms
root@mininet-vm:/home/mininet#
```

Рис. 2.9: Понижение задержка на хостах

10) Проверил соединение от первого хоста, ко второму. Теперь минимальная задержка составила 101. Средняя 102.3. Максимальная - 104.

```
root@mininet-vm:/home/mininet# ping -c 6 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=104 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=101 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=102 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=101 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5009ms
rtt min/ava/max/mdev = 100.741/102.251/104.230/1.164 ms
```

Рис. 2.10: Соединение между хостами

11) Восстановил конфигурацию, удалив все правила.

```
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet# []
```

```
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h2-eth0 root netem
root@mininet-vm:/home/mininet#
```

Рис. 2.11: Восстановление конфигурации на компах

12) Проверил соединение между хостами. Искусственная задержка теперь отсутствует.

```
root@mininet-vm:/home/mininet# ping -c 6 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=2.36 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.490 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.336 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.041 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.053 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.175 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5087ms
rtt min/avg/max/mdev = 0.041/0.575/2.360/0.813 ms
```

Рис. 2.12: Проверка без задержке

13) Добавил задержку на первый хост 100мс с отклонением в 10.

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 1
00ms 10ms
root@mininet-vm:/home/mininet#
```

Рис. 2.13: Задержка с отклонением

14) Проверил соединение между хостами. Теперь минимальная задержка - 100. Максимальная - 110.

```
root@mininet-vm:/home/mininet# ping -c 6 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=109 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=101 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=110 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=109 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=100 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=106 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5007ms
```

Рис. 2.14: Проверка соединений с отклонением

15) Восстановил конфигурацию по умолчанию.

```
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet#
```

Рис. 2.15: Восстановление конфигурации

16) Добавил к конфигурации задержку в 100мс с отклонением в 10мс. А так же значение корреляции 25%.

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 1
00ms 10ms 25%
```

Рис. 2.16: Добавление корреляции

17) Убедился, что пакеты имеют задержку в 100 мс с отклонением в 10 и корреляцией.

```
root@mininet-vm:/home/mininet# ping -c 20 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=107 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=110 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=108 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=105 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=98.8 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=102 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=104 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=109 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=107 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=96.2 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=99.3 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=96.8 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=103 ms
64 bytes from 10.0.0.2: icmp_seq=15 ttl=64 time=105 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=97.7 ms
```

Рис. 2.17: Отправка пакетов с корреляцией

18) Указал нормальное распределение задержки на хосте 1.

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 1
00ms 20ms distribution normal
root@mininet-vm:/home/mininet#
```

Рис. 2.18: Нормальное распределение на хосте

19) Проверил подключение с нормальным распределением задержки.

```
root@mininet-vm:/home/mininet# ping -c 10 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=118 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=122 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=113 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=72.2 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=98.6 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=111 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=90.5 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=132 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=67.1 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=93.6 ms

--- 10.0.0.2 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9015ms
```

Рис. 2.19: Пингование с нормальным распределением

20) Обновил репозиторий программного обеспечения на виртуальной машине.

```
mininet@mininet-vm:~$ sudo apt-get update
Hit:1 http://security.ubuntu.com/ubuntu focal-security InRelease
Hit:2 http://us.archive.ubuntu.com/ubuntu focal InRelease
Get:3 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease [128 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease [128 kB]
Fetched 256 kB in 2s (141 kB/s)
Reading package lists... Done
```

Рис. 2.20: Обновление репозитория

21) Установил пакеты geeqie.

```
libxfont2 libxklavier16 libxslt1.1 libxvidcore4 libxvnc1 libyelp0 libzmq5 libzvb1-common
libzvb10 mesa-vulkan-drivers mesa-vdpau-drivers mobile-broadband-provider-info modemmanager
mousesteaks mutter mutter-common network-manager network-manager-gnome
network-manager-pptp ocl-icd-libopencl1 p11-kit p11-kit-modules packagekit
packagekit-tools pinentry-gnome3 polycykit-1 ppp pptp-linux pulseaudio
pulseaudio-module-bluetooth pulseaudio-utils python3-apport python3-aptdaemon
python3-aptdaemon.gtk3widgets python3-blinker python3-cairo python3-cups
python3-cupsdhelpers python3-defer python3-entrpoints python3-httplib2 python3-ibus-1.0
python3-jwt python3-keyring python3-lanchapadlib python3-lazr.restfulclient
python3-lazr.uri python3-ldb python3-macharoonbakery python3-oauthlib
python3-problem-report python3-protobuf python3-requests-unixsocket python3-rfc3339
python3-secretstorage python3-simplejson python3-systemd python3-talloc python3-tz
python3-wadllib rtkit rygel samba-libs sane-utlis session-migration sgml-base sgml-data
switcheroo-control system-config-printer system-config-printer-common
system-config-printer-udev ubuntu-docs ubuntu-session-ubuntu-wallpapers
ubuntu-wallpapers-focal update-inetd upower usb-modeswitch usb-modeswitch-data usbmuxd
va-driver-all vdpau-driver-all whoopsie-preferences wpasupplicant x11-xkb-utils
x11-xserver-utils xdg-dbus-proxy xfce4-base xfce4-encodings xfce4-utils xml-core
xserver-common xserver-xephyr xserver-xorg xserver-xorg-core xserver-xorg-input-all
xserver-xorg-input-libinput xserver-xorg-input-wacom xserver-xorg-legacy
xserver-xorg-video-all xserver-xorg-video-amdgpu xserver-xorg-video-ati
xserver-xorg-video-fbdev xserver-xorg-video-intel xserver-xorg-video-nouveau
xserver-xorg-video-qxl xserver-xorg-video-radeon xserver-xorg-video-vesa
xserver-xorg-video-vmware xwayland yaru-theme-gnome-shell yelp yelp-xsl zenity
zenity-common
```

The following packages will be upgraded:

```
dbus language-selector-common libcupsc2 libdbus-1-3 libdrm-amdgpu1 libegl1 libegl1
```

Рис. 2.21: Установка geeeie

22) Создал каталог где буду выполнять лабораторную.

```
mininet@mininet-vm:~$ mkdir -p ~/work/lab_netem_i/simple_delay
mininet@mininet-vm:~$ cd ~/work/lab_netem_i/simple_delay/
```

Рис. 2.22: Создание каталога

23) Создал файл lab_netem_i.py. Вписал туда скрипт для эксперимента.

```
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 setupNet():
    """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 \'/time/g' -e \'/icmp_seq/g' > ping.dat')
    info('*** Stopping network\n')
    net.stop()

if __name__ == '__main__':
    setupNet()
```

Рис. 2.23: Скрипт lab_netem_i

24) Создал скрипт для визуализации ping_plot. Этот скрипт будет визуализировать результаты эксперимента.

```
#!/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. 2.24: Скрипт для визуализации

25) Задал права доступа к файлам скрипта.

```
mininet@mininet-vm:~/work/lab_netem_1/simple_delay$ chmod +x ping_plot
```

Рис. 2.25: Права доступа

26) Создал Makefile для управления процессом проведения эксперимента.

```
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. 2.26: Makefile

27) Запустил эксперимент.

```
mininet@mininet-vm:~/work/lab_netem_1/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',)
```

Рис. 2.27: Запуск эксперимента

28) Посмотрел какой график построил наш скрипт.

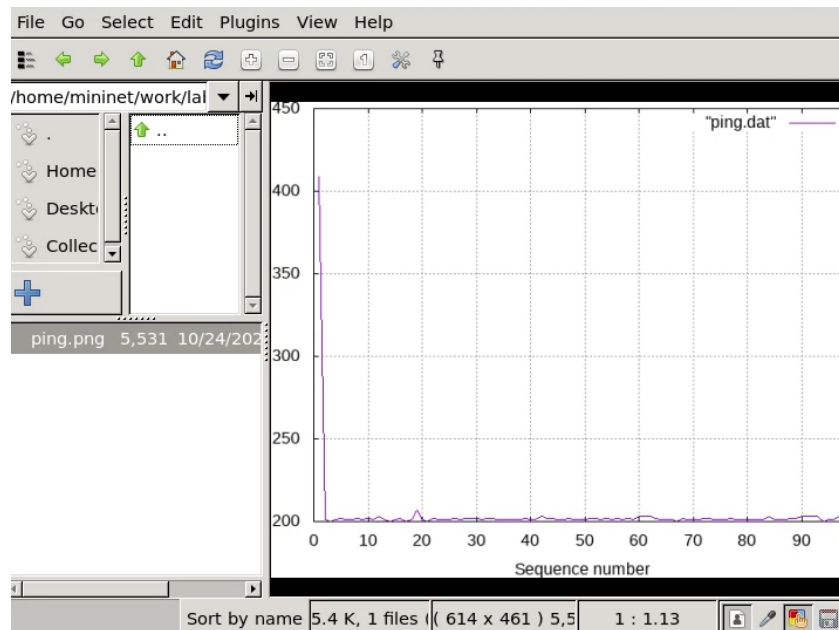


Рис. 2.28: График после первого эксперимента

29) Из файла ping.dat удалил первую строку и заново построил график. График изменился.

```
GNU nano 4.8 /home/mininet/work/lab_netem_i/simple-delay/ping.dat
2 201
3 202
4 200
5 201
6 202
7 201
8 200
9 201
10 201
11 200
12 202
13 201
14 200
15 201
16 201
17 201
18 201
19 201
```

Рис. 2.29: Удаление данных из файла

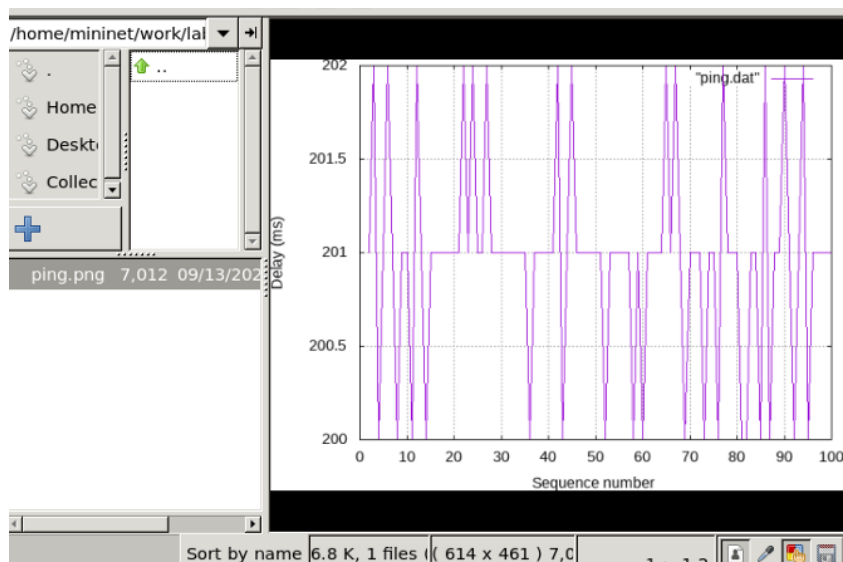


Рис. 2.30: Построение нового графика

30) Разработал собственный скрипт для вычисления. Добавил правило в файл Makefile. Запустил скрипт и затем очистил выполненную работу.

```
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]
    #print(s)
    std=(sum([(i-(sum(s)/len(s)))**2 for i in s])/(len(s)-1))**0.5
    print(f"min: {min(s)} \n max: {max(s)} \n avg: {sum(s)/len(s)} \n std: {std}")
```

Рис. 2.31: Собственный скрипт

```

ping.png: ping.dat
        ./ping_plot
clean:
        -rm -f *.dat *.png
ty:
        sudo python ty.py

```

Рис. 2.32: Изменение Makefile

```

mininet@mininet-vm:~/work/lab_netem_i/simple_delay$ make ty
sudo python ty.py
min: 200
max: 418
avg: 203.82
std: 21.66987388849866
mininet@mininet-vm:~/work/lab_netem_i/simple_delay$

```

Рис. 2.33: Демонстрация результата

31) Реализовал воспроизводимые эксперименты по изменению задержек. Построил график для каждого нового эксперимента.

```

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 50ms' )
h2.cmdPrint( 'tc qdisc add dev h2-eth0 root netem delay 50ms' )
time.sleep(10) # Wait 10 seconds
info( '*** Ping\n' )
h1.cmdPrint( 'ping -c 100', h2.IP(), '| grep "time=" | awk \'{print $5, $7}\'} | sed -e \'/s/time=//g\' -e \'/s/icmp_seq=//g\' > ping.dat' )
info( '*** Stopping network' )

```

Рис. 2.34: Изменение задержек

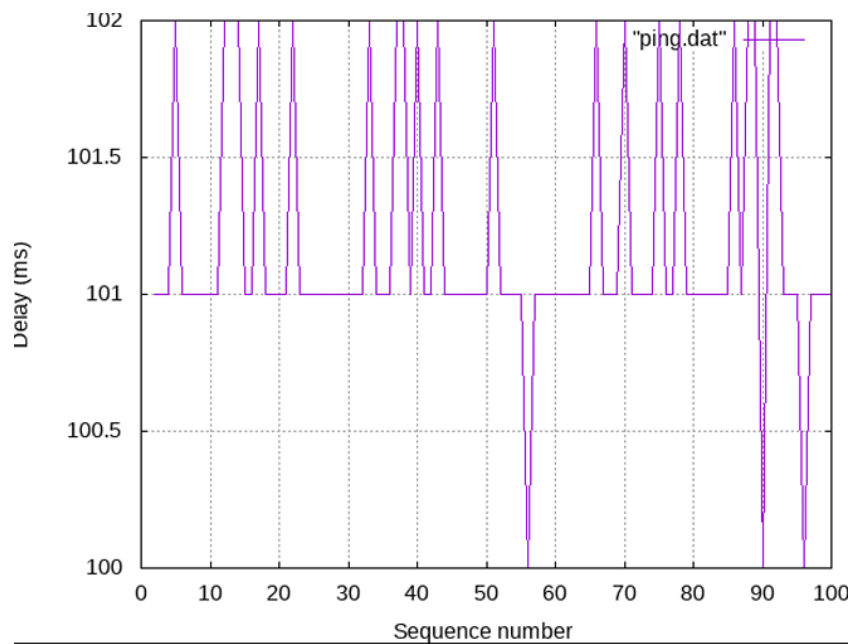


Рис. 2.35: График к новому скрипту

```
sudo python rtt.py
min: 100
max: 102
avg: 101.181818181819
std: 0.45989431713313733
```

Рис. 2.36: Результаты скрипта

32) Изменил задержку и добавил отклонение.

```
info( '*** Set delay\n')
h1.cmdPrint( 'tc qdisc add dev h1-eth0 root netem delay 50ms 10ms' )
h2.cmdPrint( 'tc qdisc add dev h2-eth0 root netem delay 50ms 10ms' )
```

Рис. 2.37: Добавление отклонения

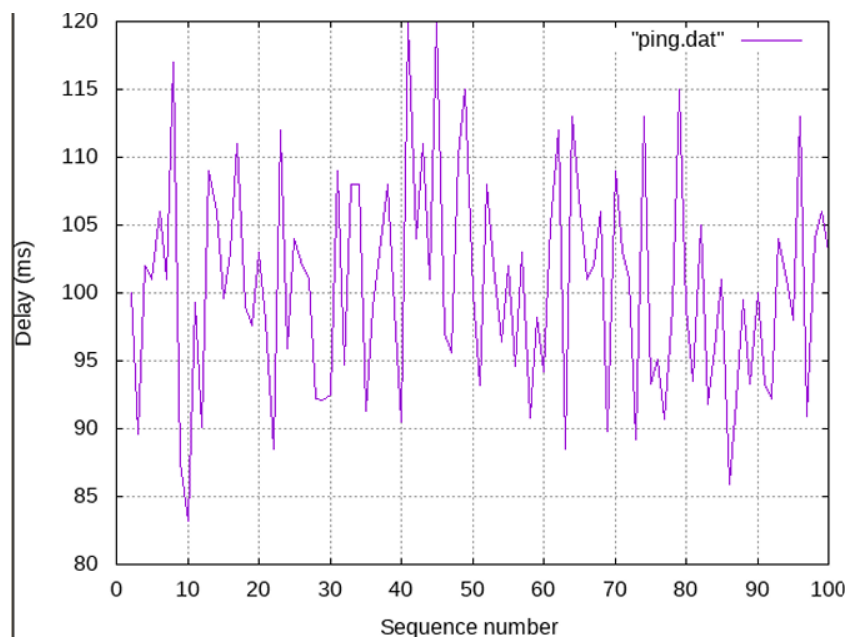


Рис. 2.38: Построение графика

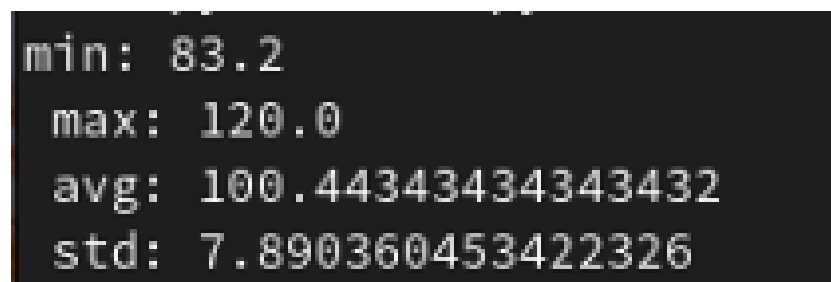


Рис. 2.39: Результаты с отклонением

33) Добавил корреляцию и провёл новый эксперимент.

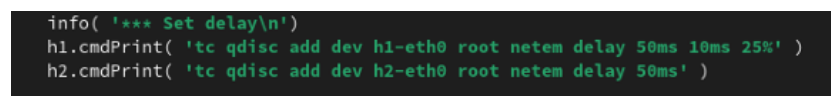


Рис. 2.40: Эксперимент с корреляцией

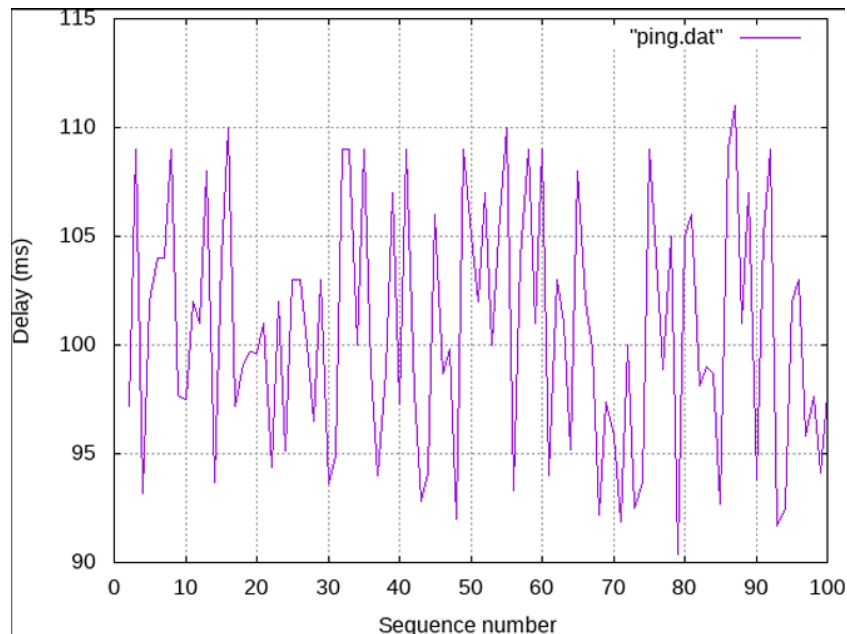


Рис. 2.41: График с корреляцией

```
min: 90.4
max: 111.0
avg: 100.67070707070708
std: 5.595392647768751
```

Рис. 2.42: Результаты работы с корреляцией

34) Изменил эксперимент, чтобы было нормальное распределение. Провёл те же эксперименты и посмотрел на график.

```
info( '*** Set delay\n')
h1.cmdPrint( 'tc qdisc add dev h1-eth0 root netem delay 50ms 10ms distribution normal' )
h2.cmdPrint( 'tc qdisc add dev h2-eth0 root netem delay 50ms' )
time.sleep(10) # Wait 10 seconds
```

Рис. 2.43: Эксперимент с нормальным распределением

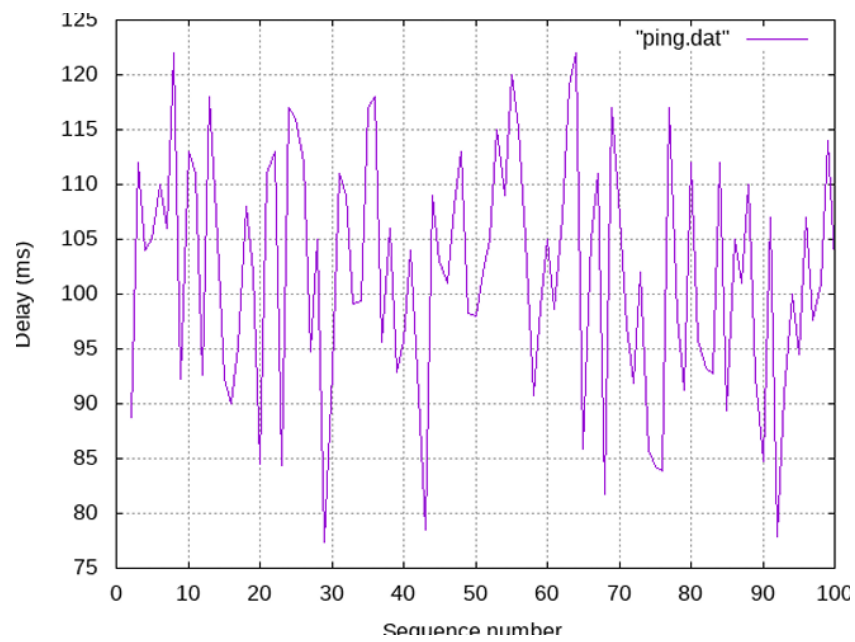


Рис. 2.44: График с нормальным распределением

```
min: 77.4  
max: 122.0  
avg: 101.76060606060605  
std: 10.857676849479262
```

Рис. 2.45: Результат работы с нормальным распределением

3 Вывод

Я познакомился с NETEM - инструментом для тестирования производительности приложений в визуальной сети, а также получил навыки проведения интерактивных и воспроизводимых экспериментов по измерению задержки и её дрожанию в моделируемой сети.