

Отчет по лабораторной работе №5

Моделирование сетей передачи данных

Оразгелдиев Язгелди

Содержание

1 Цель работы	6
2 Задание	7
3 Выполнение лабораторной работы	8
4 Выводы	21

Список иллюстраций

3.1	<i>Изменение прав запуска X-соединения.</i>	8
3.2	<i>Создание топологии</i>	8
3.3	<i>Команда ifconfig на хосте h1</i>	9
3.4	<i>Команда ifconfig на хосте h2</i>	9
3.5	<i>Пингование</i>	9
3.6	<i>Добавление процента потерь</i>	10
3.7	<i>Пингование</i>	10
3.8	<i>Пингование</i>	10
3.9	<i>Добавление процента потерь</i>	10
3.10	<i>Пингование</i>	11
3.11	<i>Пингование</i>	11
3.12	<i>Восстановление конфигурации</i>	11
3.13	<i>Пингование</i>	12
3.14	<i>Добавление коэффициента потерь с корреляцией</i>	12
3.15	<i>Пингование</i>	13
3.16	<i>Добавление повреждения</i>	13
3.17	<i>Запуск сервера</i>	13
3.18	<i>Запуск клиента</i>	14
3.19	<i>Добавление переупорядочивания пакетов</i>	14
3.20	<i>Пингование</i>	15
3.21	<i>Добавление дублирования</i>	15
3.22	<i>Пингование</i>	16
3.23	<i>Создание каталога</i>	16
3.24	<i>Создание подкаталога</i>	16
3.25	<i>Скрипт lab_netem_ii.py</i>	17
3.26	<i>Изменение в скрипте lab_netem_ii.py</i>	17
3.27	<i>Создание Makefile</i>	17
3.28	<i>Выполнение эксперимента</i>	18
3.29	<i>Добавление коэффициента потерь с корреляцией</i>	18
3.30	<i>Выполнение эксперимента</i>	18
3.31	<i>Добавление повреждения</i>	19
3.32	<i>Выполнение эксперимента</i>	19
3.33	<i>Добавление переупорядочивания пакетов</i>	19
3.34	<i>Выполнение эксперимента</i>	20
3.35	<i>Добавление дублирования</i>	20

3.36 <i>Выполнение эксперимента</i>	20
---	----

Список таблиц

1 Цель работы

Основной целью работы является получение навыков проведения интерактивных экспериментов в среде Mininet по исследованию параметров сети, связанных с потерей, дублированием, изменением порядка и повреждением пакетов при передаче данных. Эти параметры влияют на производительность протоколов и сетей.

2 Задание

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 исследованию параметров сети, связанных с потерей, дублированием, изменением порядка и повреждением пакетов при передаче данных.
3. Реализуйте воспроизводимый эксперимент по добавлению правила отбрасывания пакетов в эмулируемой глобальной сети. На экран выведите сводную информацию о потерянных пакетах.
4. Самостоятельно реализуйте воспроизводимые эксперименты по исследованию параметров сети, связанных с потерей, изменением порядка и повреждением пакетов при передаче данных. На экран выведите сводную информацию о потерянных пакетах.

3 Выполнение лабораторной работы

Запустил виртуальную среду с mininet. Из основной ОС подключился к виртуальной машине и исправил права запуска X-соединения. Скопировал значение куки своего пользователя mininet в файл для пользователя root.

```
yazik@fedora:~$ ssh -Y mininet@192.168.56.104
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

Last login: Sat Oct 25 11:20:14 2025 from 192.168.56.1
mininet@mininet-vm:~$ xauth list $DISPLAY
mininet-vm/unix:10  MIT-MAGIC-COOKIE-1  e6c84167b31fc7dd2fd2b69be88a4dcd
mininet@mininet-vm:~$ sudo -i
root@mininet-vm:~# xauth add mininet-vm/unix:10  MIT-MAGIC-COOKIE-1  e6c84167b31fc7dd2fd2b69be88a4dcd
root@mininet-vm:~# logout
```

Рисунок 3.1: Изменение прав запуска X-соединения.

2. Задал простейшую топологию, состоящую из двух хостов и коммутатора с назначенной по умолчанию mininet сетью 10.0.0.0/8.

```
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 ...
*** Starting CLI:
mininet> █
```

Рисунок 3.2: Создание топологии

3. На хостах h1 и h2 ввел команду ifconfig, чтобы отобразить информацию, относящуюся к их сетевым интерфейсам и назначенным им IP-адресам.

```
"host: h1"
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 e6:c9:b1:82:fc:45 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 902 bytes 308540 (308.5 KB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 902 bytes 308540 (308.5 KB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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

Рисунок 3.3: Команда ifconfig на хосте h1

```
"host: h2"
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 a2:c3:5a:4f:d8:bd 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 984 bytes 309672 (309.6 KB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 984 bytes 309672 (309.6 KB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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

Рисунок 3.4: Команда ifconfig на хосте h2

4. Проверил подключение между хостами h1 и h2 с помощью команды ping с параметром -c 6.

```
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=1.47 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.153 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.040 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.053 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.048 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.057 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5088ms
rtt min/avg/max/mdev = 0.040/0.303/1.468/0.522 ms
```

Рисунок 3.5: Пингование

- На хосте h1 добавил 10% потерь пакетов к интерфейсу h1-eth0.

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem loss 10%
root@mininet-vm:/home/mininet#
```

Рисунок 3.6: Добавление процента потерь

- Проверил, что на соединении от хоста h1 к хосту h2 имеются потери пакетов, используя команду ping с параметром -c 100 с хоста h1.

```
root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 100
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=0.710 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.276 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.147 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.049 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.046 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.053 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=0.053 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.053 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=0.053 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.055 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=0.055 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=0.047 ms
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=0.048 ms
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=20 ttl=64 time=0.053 ms
64 bytes from 10.0.0.2: icmp_seq=22 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=23 ttl=64 time=0.044 ms
64 bytes from 10.0.0.2: icmp_seq=24 ttl=64 time=0.051 ms
```

Рисунок 3.7: Пингование

```
64 bytes from 10.0.0.2: icmp_seq=100 ttl=64 time=0.051 ms
--- 10.0.0.2 ping statistics ---
100 packets transmitted, 86 received, 14% packet loss, time 101382ms
rtt min/avg/max/mdev = 0.043/0.063/0.710/0.074 ms
```

Рисунок 3.8: Пингование

- Для эмуляции глобальной сети с потерей пакетов в обоих направлениях к соответствующему интерфейсу на хосте h2 также добавил 10% потерь пакетов.

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h2-eth0 root netem loss 10%
root@mininet-vm:/home/mininet#
```

Рисунок 3.9: Добавление процента потерь

8. Проверил, что соединение между хостом h1 и хостом h2 имеет больший процент потерянных данных

```
root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 100
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=0.420 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.398 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.146 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.046 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.056 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=0.048 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.047 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=0.058 ms
64 bytes from 10.0.0.2: icmp_seq=15 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=0.058 ms
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=0.046 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=0.059 ms
64 bytes from 10.0.0.2: icmp_seq=20 ttl=64 time=0.057 ms
64 bytes from 10.0.0.2: icmp_seq=22 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=23 ttl=64 time=0.051 ms
```

Рисунок 3.10: Пингование

```
64 bytes from 10.0.0.2: icmp_seq=95 ttl=64 time=0.054 ms
64 bytes from 10.0.0.2: icmp_seq=96 ttl=64 time=0.054 ms
64 bytes from 10.0.0.2: icmp_seq=98 ttl=64 time=0.056 ms
64 bytes from 10.0.0.2: icmp_seq=99 ttl=64 time=0.047 ms

--- 10.0.0.2 ping statistics ---
100 packets transmitted, 89 received, 11% packet loss, time 101363ms
rtt min/avg/max/mdev = 0.045/0.061/0.420/0.053 ms
```

Рисунок 3.11: Пингование

9. Восстановил конфигурацию по умолчанию, удалив все правила, применённые к сетевому планировщику соответствующего интерфейса. Для отправителя h1 и для получателя h2.

```
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h2-eth0 root netem
root@mininet-vm:/home/mininet# 
```

Рисунок 3.12: Восстановление конфигурации

10. Убедился, что соединение от хоста h1 к хосту h2 не имеет явной потери пакетов, запустив команду ping с терминала хоста h1.

```

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=0.448 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.249 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.115 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.045 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.059 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=0.045 ms
^C
--- 10.0.0.2 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8173ms
rtt min/avg/max/mdev = 0.045/0.123/0.448/0.130 ms

```

Рисунок 3.13: Пингование

- Добавил на интерфейсе узла h1 коэффициент потери пакетов 50%, и каждая последующая вероятность зависит на 50% от последней.

```

root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem loss 50% 50%
root@mininet-vm:/home/mininet#

```

Рисунок 3.14: Добавление коэффициента потерь с корреляцией

- Проверила, что на соединении от хоста h1 к хосту h2 имеются потери пакетов, используя команду ping с параметром -c 50 с хоста h1. Процент потерянных пакетов после завершения передачи: 50%. Восстановил для узла h1 конфигурацию по умолчанию, удалив все правила, применённые к сетевому планировщику соответствующего интерфейса.

```

root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 50
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.506 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.285 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.125 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.043 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.048 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=0.056 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.049 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=0.047 ms
64 bytes from 10.0.0.2: icmp_seq=21 ttl=64 time=0.058 ms
64 bytes from 10.0.0.2: icmp_seq=22 ttl=64 time=0.047 ms
64 bytes from 10.0.0.2: icmp_seq=34 ttl=64 time=0.180 ms
64 bytes from 10.0.0.2: icmp_seq=36 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=38 ttl=64 time=0.060 ms
64 bytes from 10.0.0.2: icmp_seq=39 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=40 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=41 ttl=64 time=0.048 ms
th0 root 64 bytes from 10.0.0.2: icmp_seq=44 ttl=64 time=0.046 ms
64 bytes from 10.0.0.2: icmp_seq=45 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=46 ttl=64 time=0.060 ms
64 bytes from 10.0.0.2: icmp_seq=47 ttl=64 time=0.049 ms
64 bytes from 10.0.0.2: icmp_seq=48 ttl=64 time=0.054 ms
64 bytes from 10.0.0.2: icmp_seq=50 ttl=64 time=0.050 ms

--- 10.0.0.2 ping statistics ---
50 packets transmitted, 25 received, 50% packet loss, time 50178ms
rtt min/avg/max/mdev = 0.043/0.086/0.506/0.100 ms

```

Рисунок 3.15: Пингование

- Добавила на интерфейсе узла h1 0,01% повреждения пакетов.

```

root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem corrupt 0.01%
root@mininet-vm:/home/mininet#

```

Рисунок 3.16: Добавление повреждения

- Проверил конфигурацию с помощью инструмента iPerf3 для проверки повторных передач. Для этого запустила iPerf3 в режиме сервера в терминале хоста h2.

```

root@mininet-vm:/home/mininet# iperf3 -s
warning: this system does not seem to support IPv6 - trying IPv4
-----
Server listening on 5201
-----

```

Рисунок 3.17: Запуск сервера

- Запустил iPerf3 в клиентском режиме в терминале хоста h1.

```

root@mininet-vm:/home/mininet# iperf3 -c 10.0.0.2
Connecting to host 10.0.0.2, port 5201
[ 7] local 10.0.0.1 port 54630 connected to 10.0.0.2 port 5201
[ ID] Interval      Transfer     Bitrate      Retr  Cwnd
[ 7]  0.00-1.00  sec  6.15 GBytes  52.8 Gbits/sec   11  2.76 MBytes
[ 7]  1.00-2.00  sec  5.78 GBytes  49.7 Gbits/sec   12  2.07 MBytes
[ 7]  2.00-3.00  sec  5.75 GBytes  49.4 Gbits/sec    8  2.56 MBytes
[ 7]  3.00-4.00  sec  5.82 GBytes  50.0 Gbits/sec   12  1.75 MBytes
[ 7]  4.00-5.00  sec  5.87 GBytes  50.5 Gbits/sec    8  2.26 MBytes
[ 7]  5.00-6.00  sec  5.40 GBytes  46.4 Gbits/sec   12  2.35 MBytes
[ 7]  6.00-7.00  sec  5.52 GBytes  47.4 Gbits/sec    6  1.88 MBytes
[ 7]  7.00-8.00  sec  5.58 GBytes  47.9 Gbits/sec   13  1.92 MBytes
[ 7]  8.00-9.00  sec  5.75 GBytes  49.4 Gbits/sec    8  2.24 MBytes
[ 7]  9.00-10.00 sec  5.67 GBytes  48.8 Gbits/sec   9  2.02 MBytes
[ ID] Interval      Transfer     Bitrate      Retr
[ 7]  0.00-10.00 sec  57.3 GBytes  49.2 Gbits/sec   95
[ 7]  0.00-10.00 sec  57.3 GBytes  49.2 Gbits/sec
sender
receiver
iperf Done.

```

Рисунок 3.18: Запуск клиента

16. Восстановил для узла h1 конфигурацию по умолчанию, удалив все правила, применённые к сетевому планировщику соответствующего интерфейса.
17. Добавил на интерфейсе узла h1 следующее правило. Здесь 25% пакетов (со значением корреляции 50%) будут отправлены немедленно, а остальные 75% будут задержаны на 10 мс.

```

root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 10ms reorder 25% 50%
root@mininet-vm:/home/mininet#

```

Рисунок 3.19: Добавление переупорядочивания пакетов

18. Проверил, что на соединении от хоста h1 к хосту h2 имеются потери пакетов, используя команду ping с параметром -c 20 с хоста h1. Отсутствующих из-за потери пакетов номеров последовательности нет. Процент потерянных пакетов после завершения передачи: 0%. Восстановил конфигурацию интерфейса по умолчанию на узле h1.

```

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=11.4 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.333 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.127 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=10.8 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=10.8 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=10.6 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=10.8 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=10.6 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.049 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.046 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=0.048 ms
64 bytes from 10.0.0.2: icmp_seq=15 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=0.047 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=20 ttl=64 time=0.040 ms

--- 10.0.0.2 ping statistics ---
20 packets transmitted, 20 received, 0% packet loss, time 19162ms
rtt min/avg/max/mdev = 0.040/6.497/11.358/5.232 ms

```

Рисунок 3.20: Пингование

- Для интерфейса узла h1 задала правило с дублированием 50% пакетов (т.е. 50% пакетов должны быть получены дважды).

```

root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem duplicate 50%
root@mininet-vm:/home/mininet#

```

Рисунок 3.21: Добавление дублирования

- Проверил, что на соединении от хоста h1 к хосту h2 имеются дублированные пакеты, используя команду ping с параметром -с 20 с хоста h1. Восстановил конфигурацию интерфейса по умолчанию на узле h1.

```

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=0.465 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.270 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.117 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.047 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.059 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.059 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.044 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.044 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.049 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.049 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=0.079 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.054 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.054 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.053 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.054 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.049 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.049 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=0.046 ms
64 bytes from 10.0.0.2: icmp_seq=15 ttl=64 time=0.046 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=0.056 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=0.057 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=0.048 ms
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=0.048 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=0.051 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=20 ttl=64 time=0.050 ms

--- 10.0.0.2 ping statistics ---
20 packets transmitted, 20 received, +9 duplicates, 0% packet loss, time 19458ms
rtt min/avg/max/mdev = 0.044/0.075/0.465/0.084 ms

```

Рисунок 3.22: Пингование

- Для каждого воспроизводимого эксперимента `expname` создал свой каталог, в котором будут размещаться файлы эксперимента.

```

mininet@mininet-vm:~$ mkdir -p ~/work/lab_netem_ii/expname
mininet@mininet-vm:~$ 

```

Рисунок 3.23: Создание каталога

- В виртуальной среде `mininet` в своём рабочем каталоге с проектами создал каталог `simple-drop` и перешла в него.

```

mininet@mininet-vm:~$ mkdir -p ~/work/lab_netem_ii/expname
mininet@mininet-vm:~$ cd ~/work/lab_netem_ii/expname
mininet@mininet-vm:~/work/lab_netem_ii/expname$ touch lab_netem_ii.py
mininet@mininet-vm:~/work/lab_netem_ii/expname$ ls
lab_netem_ii.py
mininet@mininet-vm:~/work/lab_netem_ii/expname$ 

```

Рисунок 3.24: Создание подкаталога

- Создал скрипт для эксперимента `lab_netem_ii.py`.

```

mininet@mininet-vm:~/work/lab_neterm_ii/exptname -- ssh -Y mininet@192.168.56.104
/home/mininet/work/lab_neterm_ii/exptname/lab_neterm_ii.py [-M--] 8 L:[ 7+41 48/ 48] <(1155/1163b) 101 0x065

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 loss 10%')
    h2.cmdPrint('tc qdisc add dev h2-eth0 root netem loss 10%')

    time.sleep(10) # Wait 10 seconds
    info( '*** Ping\n' )
    h1.cmdPrint('ping -c 100', h2.IP(), '| grep "packet loss" | awk \'{print $5, $7}\' | sed -e \'$a/times//g\' -e \'$a/icmp_seq//g\' > ping.dat')
    info( '*** Stopping network\n' )
    net.stop()

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

```

Рисунок 3.25: Скрипт *lab_neterm_ii.py*

24. Скорректировал скрипт так, чтобы в отдельный файл выводилась информация о потерях пакетов.

```

>info( '*** Ping\n' )
>h1.cmdPrint('ping -c 100', h2.IP(), '| grep "packet loss" | awk \'{print $5, $7}\' | sed -e \'$a/times//g\' -e \'$a/icmp_seq//g\' > ping.dat')
info: *** Stopping network

```

Рисунок 3.26: Изменение в скрипте *lab_neterm_ii.py*

25. Создал Makefile для управления процессом проведения эксперимента.

```

all: ping.dat

ping.dat:
<---->sudo python lab_neterm_ii.py
<---->sudo chown mininet:mininet ping.dat

clean:
<---->-rm -f *.dat

```

Рисунок 3.27: Создание Makefile

26. Выполнил эксперимент.

```

mininet@mininet-vm:~/work/lab_netem_ii/exptname$ make
sudo python lab_netem_ii.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 loss 10%',)
*** h2 : ('tc qdisc add dev h2-eth0 root netem loss 10%',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "packet loss" | awk \'print $5, $7, $8\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat

```

Рисунок 3.28: Выполнение эксперимента

27. Самостоятельно реализовал воспроизводимые эксперименты по исследованию параметров сети, связанных с потерей пакетов и добавление значения корреляции для потери пакетов.

```

-->info( "*** Set delay\n")
-->h1.cmdPrint( 'tc qdisc add dev h1-eth0 root netem loss 50% 50%' )
-->#h2.cmdPrint( 'tc qdisc add dev h2-eth0 root netem loss 10%' )

```

Рисунок 3.29: Добавление коэффициента потерь с корреляцией

```

mininet@mininet-vm:~/work/lab_netem_ii/exptname$ make
sudo python lab_netem_ii.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 loss 50% 50%',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "packet loss" | awk \'print $5, $7, $8\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat

```

Рисунок 3.30: Выполнение эксперимента

28. Самостоятельно реализовал воспроизводимые эксперименты по исследованию параметров сети, связанных добавлением повреждения пакетов.

```
----->info( '*** Set delay\n')
----->h1.cmdPrint( 'tc qdisc add dev h1-eth0 root netem corrupt 0.01%' )
----->#h2.cmdPrint( 'tc qdisc add dev h2-eth0 root netem loss 10%' )
```

Рисунок 3.31: Добавление повреждения

```
mininet@mininet-vm:/work/lab_netem_ii/exptests$ make
sudo python lab_netem_ii.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 corrupt 0.01%',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "packet loss" | awk \'{print $5, $7, $8}\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
```

Рисунок 3.32: Выполнение эксперимента

29. Самостоятельно реализовал воспроизводимые эксперименты по исследованию параметров сети, связанных добавлением переупорядочивания пакетов.

```
----->info( '*** Set delay\n')
----->h1.cmdPrint( 'tc qdisc add dev h1-eth0 root netem delay 10ms reorder 25% 50%' )
----->#h2.cmdPrint( 'tc qdisc add dev h2-eth0 root netem loss 10%' )
```

Рисунок 3.33: Добавление переупорядочивания пакетов

```

mininet@mininet-vm:~/work/lab_netem_ii/exptname$ make
sudo python lab_netem_ii.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 10ms reorder 25% 50%',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "packet loss" | awk \'print $5, $7, $8\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
...
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat

```

Рисунок 3.34: Выполнение эксперимента

30. Самостоятельно реализовал воспроизводимые эксперименты по исследованию параметров сети, связанных добавлением дублирования пакетов.

```

---->info( "*** Set delay\n")
---->h1.cmdPrint( 'tc qdisc add dev h1-eth0 root netem duplicate 50%' )
---->#h2.cmdPrint( 'tc qdisc add dev h2-eth0 root netem loss 10%' )

```

Рисунок 3.35: Добавление дублирования

```

mininet@mininet-vm:~/work/lab_netem_ii/exptname$ make
sudo python lab_netem_ii.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 duplicate 50%',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "packet loss" | awk \'print $5, $7, $8\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
...
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat

```

Рисунок 3.36: Выполнение эксперимента

4 Выводы

Я получил навыки проведения интерактивных экспериментов в среде Mininet по исследованию параметров сети, связанных с потерей, дублированием, изменением порядка и повреждением пакетов при передаче данных.