

# Лабораторная работа №5 Эмуляция и измерение потерь пакетов в глобальных сетях

- Выполнил:
- Студент группы: НПИбд-02-21
- Студенческий билет: № 1032217060
- ФИО студента: Королев Адам Маратович

# Цели 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.

# Выполнение работы:

## 5.4.1. Запуск лабораторной топологии

- 5.4.1. Запуск лабораторной топологии 1. Запустите виртуальную среду с mininet. 2. Из основной ОС подключитесь к виртуальной машине: 1 ssh -Y [mininet@192.168.x.y](#) 3. В виртуальной машине mininet при необходимости исправьте права запуска X-соединения. Скопируйте значение куки (MIT magic cookie)1 своего пользователя mininet в файл для пользователя root: 1 mininet@mininet-vm:~\$ xauth list \$DISPLAY 2 mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 ↪ 295acad8e35d17636924c5ab80e8462d 3 4 mininet@mininet-vm:~\$ sudo -i 5 root@mininet-vm:~# xauth add mininet-vm/unix:10 ↪ MIT-MAGIC-COOKIE-1 295acad8e35d17636924c5ab80e8462d 6 root@mininet-vm:~# logout После выполнения этих действий графические приложения должны запускаться под пользователем mininet.
- 4. Задайте простейшую топологию, состоящую из двух хостов и коммутатора с назначенной по умолчанию mininet сетью 10.0.0.0/8: 1 sudo mn --topo=single,2 -x После введения этой команды запустятся терминалы двух хостов, коммутатора и контроллера. Терминалы коммутатора и контроллера можно закрыть.

"controller: c0" (root)@mininet-vm

root@mininet-vm:~#

root@mininet-vm: ~

```
mininet@mininet-vm:~$ xauth list $DISPLAY
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 c382fb527fee766ff386d2035c06e6c0
mininet@mininet-vm:~$ sudo -i
root@mininet-vm:~# xauth add mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 c382fb527fee7
66ff386d2035c06e6c0
root@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>
```

5. На хостах h1 и h2 введите команду ifconfig, чтобы отобразить информацию, относящуюся к их сетевым интерфейсам и назначенным им IP-адресам. В дальнейшем при работе с NETEM и командой tc будут использоваться интерфейсы h1-eth0 и h2-eth0.

```
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 c2:25:10:c7:a6:90 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 1118 bytes 278668 (278.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1118 bytes 278668 (278.6 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 7a:c1:70:58:cd:a3 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 1322 bytes 353352 (353.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1322 bytes 353352 (353.3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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

## 6. Проверьте подключение между хостами h1 и h2 с помощью команды ping с параметром -c 6.

```
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=1.44 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.193 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.043 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.069 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.044 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.033 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5083ms
rtt min/avg/max/mdev = 0.033/0.303/1.439/0.510 ms
root@mininet-vm:/home/mininet#
```

```
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=0.622 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.051 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.046 ms
64 bytes from 10.0.0.1: icmp_seq=4 ttl=64 time=0.050 ms
64 bytes from 10.0.0.1: icmp_seq=5 ttl=64 time=0.056 ms
64 bytes from 10.0.0.1: icmp_seq=6 ttl=64 time=0.050 ms

--- 10.0.0.1 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5098ms
rtt min/avg/max/mdev = 0.046/0.145/0.622/0.212 ms
root@mininet-vm:/home/mininet#
```



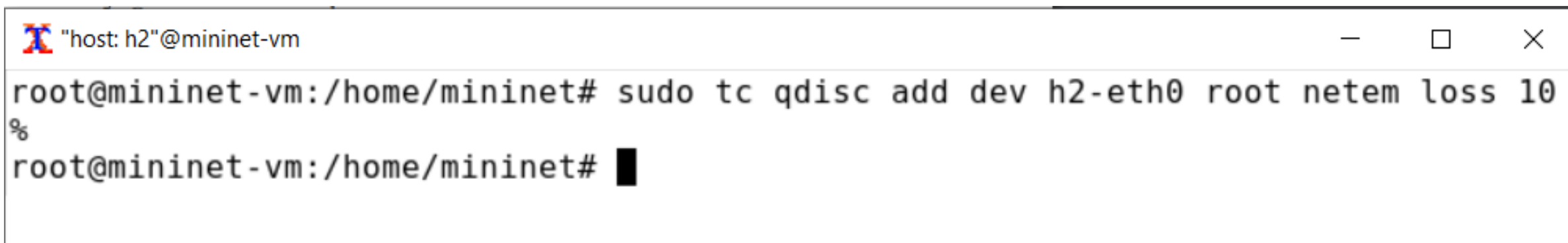
## 5.4.2. Интерактивные эксперименты

5.4.2.1. Добавление потери пакетов на интерфейс, подключённый к эмулируемой глобальной сети. Пакеты могут быть потеряны в процессе передачи из-за таких факторов, как битовые ошибки и перегрузка сети. Скорость потери данных часто измеряется как процентная доля потерянных пакетов по отношению к количеству отправленных пакетов. 1. На хосте h1 добавьте 10% потерь пакетов к интерфейсу h1-eth0: 1 sudo tc qdisc add dev h1-eth0 root netem loss 10%

2. Проверьте, что на соединении от хоста h1 к хосту h2 имеются потери пакетов, используя команду ping с параметром -c 100 с хоста h1. Параметр -c указывает общее количество пакетов для отправки.

```
root@mininet-vm:/home/mininet# ping -c 6 10.0.0.2
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem loss 10
%
root@mininet-vm:/home/mininet# ping -c 100 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=0.749 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.368 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=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.054 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.058 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=0.044 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=0.039 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=1.17 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.040 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=0.065 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=0.043 ms
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=0.054 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=0.101 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=21 ttl=64 time=0.049 ms
64 bytes from 10.0.0.2: icmp_seq=22 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=23 ttl=64 time=0.054 ms
64 bytes from 10.0.0.2: icmp_seq=25 ttl=64 time=0.054 ms
64 bytes from 10.0.0.2: icmp_seq=26 ttl=64 time=0.055 ms
64 bytes from 10.0.0.2: icmp_seq=27 ttl=64 time=0.057 ms
64 bytes from 10.0.0.2: icmp_seq=28 ttl=64 time=0.055 ms
64 bytes from 10.0.0.2: icmp_seq=29 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=31 ttl=64 time=0.055 ms
64 bytes from 10.0.0.2: icmp_seq=32 ttl=64 time=0.054 ms
^C
--- 10.0.0.2 ping statistics ---
32 packets transmitted, 28 received, 12.5% packet loss, time 31714ms
rtt min/avg/max/mdev = 0.039/0.132/1.169/0.243 ms
root@mininet-vm:/home/mininet#
```

3. Для эмуляции глобальной сети с потерей пакетов в обоих направлениях необходимо к соответствующему интерфейсу на хосте h2 также добавить 10% потерь пакетов: `1 sudo tc qdisc add dev h2-eth0 root netem loss 10%`



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



4. Проверьте, что соединение между хостом h1 и хостом h2 имеет большой процент потерянных данных (10% от хоста h1 к хосту h2 и 10% от хоста h2 к хосту h1), повторив команду ping с параметром -c 100 на терминале хоста h1.

```
64 bytes from 10.0.0.2: icmp_seq=36 ttl=64 time=0.060 ms
64 bytes from 10.0.0.2: icmp_seq=38 ttl=64 time=0.059 ms
64 bytes from 10.0.0.2: icmp_seq=40 ttl=64 time=0.055 ms
64 bytes from 10.0.0.2: icmp_seq=41 ttl=64 time=0.054 ms
64 bytes from 10.0.0.2: icmp_seq=42 ttl=64 time=0.054 ms
64 bytes from 10.0.0.2: icmp_seq=43 ttl=64 time=0.053 ms
64 bytes from 10.0.0.2: icmp_seq=44 ttl=64 time=0.055 ms
64 bytes from 10.0.0.2: icmp_seq=45 ttl=64 time=0.060 ms
64 bytes from 10.0.0.2: icmp_seq=46 ttl=64 time=0.054 ms
64 bytes from 10.0.0.2: icmp_seq=47 ttl=64 time=0.057 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=49 ttl=64 time=0.057 ms
64 bytes from 10.0.0.2: icmp_seq=50 ttl=64 time=0.057 ms
64 bytes from 10.0.0.2: icmp_seq=51 ttl=64 time=0.069 ms
64 bytes from 10.0.0.2: icmp_seq=53 ttl=64 time=0.054 ms
^C
--- 10.0.0.2 ping statistics ---
53 packets transmitted, 47 received, 11.3208% packet loss, time 53237ms
rtt min/avg/max/mdev = 0.048/0.071/0.639/0.087 ms
root@mininet-vm:/home/mininet#
```

5. Восстановите конфигурацию по умолчанию, удалив все правила, применённые к сетевому планировщику соответствующего интерфейса. Для отправителя h1: `1 sudo tc qdisc del dev h1-eth0 root netem`  
Для получателя h2:  
`1 sudo tc qdisc del dev h2-eth0 root netem`
6. Убедитесь, что соединение от хоста h1 к хосту h2 не имеет явной потери пакетов, запустив команду `ping` с терминала хоста h1 и затем нажав `Ctrl + c`, чтобы остановить тест.

```
"host: h1"@mininet-vm
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet# ping -c 10.0.0.2
ping: invalid argument: '10.0.0.2'
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=0.518 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.344 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.135 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.052 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=0.047 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.050 ms
^C
--- 10.0.0.2 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9
rtt min/avg/max/mdev = 0.047/0.134/0.518/0.155 ms
root@mininet-vm:/home/mininet#
```

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

5.4.2.2. Добавление значения корреляции для потери пакетов в эмулируемой глобальной сети 1. Добавьте на интерфейсе узла h1 коэффициент потери пакетов 50% (такой высокий уровень потери пакетов маловероятен), и каждая последующая вероятность зависит на 50% от последней: 1 sudo tc qdisc add dev h1-eth0 root netem loss 50% 50%

2. Проверьте, что на соединении от хоста h1 к хосту h2 имеются потери пакетов, используя команду ping с параметром -c 50 с хоста h1.

3. Восстановите для узла h1 конфигурацию по умолчанию, удалив все правила, применённые к сетевому планировщику соответствующего интерфейса: 1 sudo tc qdisc del dev h1-eth0 root netem

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem loss 50
% 50%
root@mininet-vm:/home/mininet# ping -c 50 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=3 ttl=64 time=0.517 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.435 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.243 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=1025 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.506 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=0.063 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=0.055 ms
64 bytes from 10.0.0.2: icmp_seq=21 ttl=64 time=0.053 ms
64 bytes from 10.0.0.2: icmp_seq=23 ttl=64 time=0.053 ms
64 bytes from 10.0.0.2: icmp_seq=24 ttl=64 time=0.055 ms
64 bytes from 10.0.0.2: icmp_seq=25 ttl=64 time=0.074 ms
^C
--- 10.0.0.2 ping statistics ---
25 packets transmitted, 11 received, 56% packet loss, time 24581ms
rtt min/avg/max/mdev = 0.053/93.350/1024.796/294.549 ms, pipe 2
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet#
```

5.4.2.3. Добавление повреждения пакетов в эмулируемой глобальной сети 1. При необходимости восстановите конфигурацию интерфейсов по умолчанию на узлах h1 и h2. 2. Добавьте на интерфейсе узла h1 0,01% повреждения пакетов: `1 sudo tc qdisc add dev h1 -eth0 root netem corrupt 0.01%`

3. Проверьте конфигурацию с помощью инструмента iPerf3 для проверки повторных передач. Для этого: – запустите iPerf3 в режиме сервера в терминале хоста h2: `1 iperf3 -s` – запустите iPerf3 в клиентском режиме в терминале хоста h1: `1 iperf3 -c 10.0.0.2`

– В отчёте отразите значения повторной передачи на каждом временном интервале и общее количество повторно переданных пакетов. – Для остановки сервера iPerf3 нажмите Ctrl + c в терминале хоста h2. 4. Восстановите для узла h1 конфигурацию по умолчанию, удалив все правила, применённые к сетевому планировщику соответствующего интерфейса.

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem corrupt 0.01%
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 35586 connected to 10.0.0.2 port 5201
[ ID] Interval      Transfer     Bitrate      Retr  Cwnd
[ 7] 0.00-1.00 sec  4.57 GBytes  39.2 Gbits/sec  65   1.10 MBytes
[ 7] 1.00-2.00 sec  3.78 GBytes  32.4 Gbits/sec   6   1.42 MBytes
[ 7] 2.00-3.00 sec  4.02 GBytes  34.6 Gbits/sec   9   567 KBytes
[ 7] 3.00-4.00 sec  4.14 GBytes  35.5 Gbits/sec   5   1.89 MBytes
[ 7] 4.00-5.00 sec  4.87 GBytes  41.9 Gbits/sec  10   1.12 MBytes
[ 7] 5.00-6.00 sec  4.48 GBytes  38.5 Gbits/sec  19   635 KBytes
^C[ 7] 6.00-6.97 sec  4.41 GBytes  39.3 Gbits/sec   8   529 KBytes
-----
[ ID] Interval      Transfer     Bitrate      Retr  sender receiver
[ 7] 0.00-6.97 sec  30.3 GBytes  37.3 Gbits/sec  122
[ 7] 0.00-6.97 sec  0.00 Bytes  0.00 bits/sec
iperf3: interrupt - the client has terminated
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 7a:c1:70:58:cd:a3 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 frame 0
root@mininet-vm:/home/mininet# iperf3 -s
warning: this system does not seem to support IPv6 - trying IPv4
Server listening on 5201
-----
Accepted connection from 10.0.0.1, port 35584
[ 7] local 10.0.0.2 port 5201 connected to 10.0.0.1 port 35586
[ ID] Interval      Transfer     Bitrate
[ 7] 0.00-1.00 sec  4.57 GBytes  39.2 Gbits/sec
[ 7] 1.00-2.00 sec  3.76 GBytes  32.3 Gbits/sec
[ 7] 2.00-3.00 sec  4.01 GBytes  34.4 Gbits/sec
[ 7] 3.00-4.00 sec  4.15 GBytes  35.7 Gbits/sec
[ 7] 4.00-5.00 sec  4.87 GBytes  41.8 Gbits/sec
[ 7] 5.00-6.00 sec  4.46 GBytes  38.4 Gbits/sec
[ 7] 5.00-6.00 sec  4.46 GBytes  38.4 Gbits/sec
-----
[ ID] Interval      Transfer     Bitrate
[ 7] 0.00-6.00 sec  30.2 GBytes  43.3 Gbits/sec
iperf3: the client has terminated
Server listening on 5201
```


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



5.4.2.4. Добавление переупорядочивания пакетов в интерфейс подключения к эмулируемой глобальной сети 1. При необходимости восстановите конфигурацию интерфейсов по умолчанию на узлах h1 и h2. 2. Добавьте на интерфейсе узла h1 следующее правило: 1 sudo tc qdisc add dev h1-eth0 root netem delay 10ms ↪ reorder 25% 50% Здесь 25% пакетов (со значением корреляции 50%) будут отправлены немедленно, а остальные 75% будут задержаны на 10 мс. 3. Проверьте, что на соединении от хоста h1 к хосту h2 имеются потери пакетов, используя команду ping с параметром -c 20 с хоста h1. Убедитесь, что часть пакетов не будут иметь задержки (один из четырех, или 25%), а последующие несколько пакетов будут иметь задержку около 10 миллисекунд (три из четырех, или 75%). 4. Восстановите конфигурацию интерфейса по умолчанию на узле h1.

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 10ms reorder 25% 50%
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=11.3 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=11.1 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=10.9 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=10.3 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=10.8 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=10.3 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=10.2 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=10.9 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=10.3 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=0.045 ms
^C
--- 10.0.0.2 ping statistics ---
16 packets transmitted, 16 received, 0% packet loss, time 15030ms
rtt min/avg/max/mdev = 0.045/10.014/11.265/2.590 ms
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet#
```

5.4.2.5. Добавление дублирования пакетов в интерфейс подключения к эмулируемой глобальной сети 1. При необходимости восстановите конфигурацию интерфейсов по умолчанию на узлах h1 и h2. 2. Для интерфейса узла h1 задайте правило с дублированием 50% пакетов (т.е. 50% пакетов должны быть получены дважды): 1 sudo tc qdisc add dev h1 -eth0 root netem duplicate 50% 3. Проверьте, что на соединении от хоста h1 к хосту h2 имеются дублированные пакеты, используя команду ping с параметром -c 20 с хоста h1. Дубликаты пакетов помечаются как DUP!. Измеренная скорость дублирования пакетов будет приближаться к настроенной скорости по мере выполнения большего количества попыток. 4. Восстановите конфигурацию интерфейса по умолчанию на узле h1.

 "host: h1"@mininet-vm

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem duplicate 50%
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=0.592 ms
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.610 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.313 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.166 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.662 ms (DUP!)
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=4 ttl=64 time=0.053 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.054 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.055 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.058 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=0.034 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=0.039 ms
^C
--- 10.0.0.2 ping statistics ---
9 packets transmitted, 9 received, +4 duplicates, 0% packet loss, time 8194ms
rtt min/avg/max/mdev = 0.034/0.210/0.662/0.236 ms
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet#
```




mininet@mininet-vm: ~

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

5.4.3. Воспроизведение экспериментов 5.4.3.1. Предварительная подготовка 1. Для каждого воспроизводимого эксперимента expname создайте свой каталог, в котором будут размещаться файлы эксперимента: 1 mkdir -p ~/work/lab\_netem\_ii/expname Здесь expname может принимать значения simple-drop, correlationdrop и т.п. 2. Для каждого случая создайте скрипт для проведения эксперимента lab\_netem\_ii.py.



 mininet@mininet-vm: ~/work/lab\_netem\_i/simple-drop

```
mininet@mininet-vm:~$ mkdir -p ~/work/lab_netem_i/simple-drop
mininet@mininet-vm:~$ cd ~/work/lab_netem_i/simple-drop/
mininet@mininet-vm:~/work/lab_netem_i/simple-drop$ █
```

5.4.3.2. Добавление потери пакетов на интерфейс, подключённый к эмулируемой глобальной сети С помощью API Mininet воспроизведите эксперимент по добавлению потери пакетов для интерфейса хоста, подключающегося к эмулируемой глобальной сети. 1. В виртуальной среде mininet в своём рабочем каталоге с проектами создайте каталог simple-drop и перейдите в него: 1 mkdir -p ~/work/lab\_netem\_i/simple-drop 2 cd ~/work/lab\_netem\_i/simple-drop

## 2. Создаёте скрипт для эксперимента lab\_netem\_ii.py:

```
mininet@mininet-vm: ~/work/lab_netem_ii/simple-drop

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(f'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\n')
net.stop()

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

:wq

### 3. В отчёте поясните содержание скрипта lab\_netem\_ii.py. В каких строках скрипта задается значение потери пакетов для интерфейса хоста?

- Создание сети:
- В строке 18 создаётся объект Mininet, представляющий виртуальную сеть.
- Контроллер добавляется в строке 21.
- Узлы (хосты) h1 и h2 добавляются в строках 24 и 25, соответственно.
- Коммутатор s1 добавляется в строке 28.
- В строках 31 и 32 создаются связи между хостами и коммутатором.
- Настройка потерь пакетов:
- Значения потерь пакетов задаются в строках 38 и 39 с помощью команды `tc qdisc add`. Потери пакетов настраиваются для интерфейсов хостов h1 и h2 с параметром `loss 10%`.
- Проведение эксперимента:
- В строке 41 вводится задержка на 10 секунд для эмуляции работы сети.
- В строке 44 запускается команда `ping`, результаты которой фильтруются с помощью `grep`, `awk` и `sed` и сохраняются в файл `ping.dat`.
- Остановка сети:
- В строке 47 вызывается метод `net.stop()`, завершающий работу сети.

4. Скорректируйте скрипт так, чтобы на экран или в отдельный файл выводилась информация о потерях пакетов.

```
mininet@mininet-vm: ~/work/lab_netem_ii/simple-drop

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(f'ping -c 100 {h2.IP()} | tee ping_output.log | grep "time=" | a
wk \'{{print $5, $7}}\' | sed -e \'s/time=//g\' -e \'s/icmp_seq=//g\' > ping.dat
')

info('*** Analyzing packet loss\n')
h1.cmdPrint(f'grep "packet loss" ping_output.log | awk \'{{print $6}}\' > pa
cket_loss.dat')

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

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

mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$
```

## 5. Создайте Makefile для управления процессом проведения эксперимента:

[illegible]

## 6. Выполните эксперимент:

```
mininet@mininet-vm: ~/work/lab_netem_ii/simple-drop
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 | tee ping_output.log | grep "time=" | awk '{p
int $5, $7}\'' | sed -e 's/time=//g' -e 's/icmp_seq=//g' > ping.dat',)
*** Analyzing packet loss
*** h1 : ('grep "packet loss" ping_output.log | awk '{print $6}\'' > packet_lo
s.dat',)
*** Stopping network
*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$
```

```
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ make clean  
rm -f *.dat  
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$
```

7. Очистите каталог от результатов проведения экспериментов:



# Вывод:

- В процессе выполнения работы я приобрел навыки проведения интерактивных экспериментов в среде Mininet по исследованию параметров сети, связанных с потерей, дублированием, изменением порядка и повреждением пакетов при передаче данных.