

Отчёт по лабораторной работе №5 по дисциплине Моделирование сетей передачи данных

Эмуляция и измерение потерь пакетов в глобальных сетях

Шаповалова Диана Дмитриевна, НПИбд-02-21, 1032211220

Содержание

1	Цель работы	5
2	Выполнение работы	6
2.1	Запуск лабораторной топологии	6
2.2	Интерактивные эксперименты	6
2.3	Добавление потери пакетов на интерфейс, подключённый к эмулируемой глобальной сети	6
2.4	Добавление значения корреляции для потери пакетов в эмулируемой глобальной сети	8
2.5	Добавление повреждения пакетов в эмулируемой глобальной сети	10
2.6	Добавление переупорядочивания пакетов в интерфейс подключения к эмулируемой глобальной сети	11
2.7	Добавление дублирования пакетов в интерфейс подключения к эмулируемой глобальной сети	12
3	Воспроизведение экспериментов	14
3.1	Предварительная подготовка	14
3.2	Задание для самостоятельной работы	16
4	Выводы	21
5	Список литературы	22

Список иллюстраций

2.1	Добавление потери пакетов на интерфейс	8
2.2	Добавление значения корреляции для потери пакетов	9
2.3	Добавление повреждения пакетов	10
2.4	Добавление переупорядочивания пакетов	12
2.5	Добавление дублирования пакетов	13
3.1	Создаем каталоги	14
3.2	Создаем скрипт	15
3.3	Выполняем эксперимент	16
3.4	Выполняем эксперимент	17
3.5	Выполняем эксперимент	18
3.6	Выполняем эксперимент	19
3.7	Выполняем эксперимент	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 Выполнение работы

2.1 Запуск лабораторной топологии

Запустите виртуальную среду с mininet.

Из основной ОС подключитесь к виртуальной машине:

```
ssh -Y mininet@192.168.x.y
```

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

После выполнения этих действий графические приложения должны запускаться под пользователем mininet.

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

2.2 Интерактивные эксперименты

2.3 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:

```
sudo tc qdisc add dev h1-eth0 root netem loss 10%
```

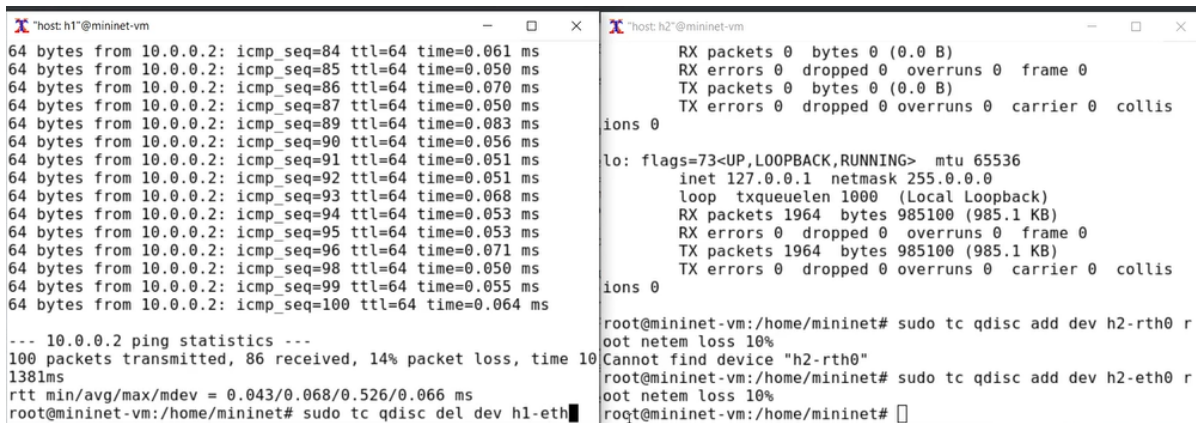
Здесь:

- sudo: выполнить команду с более высокими привилегиями;
- tc: вызвать управление трафиком Linux;
- qdisc: изменить дисциплину очередей сетевого планировщика;
- add: создать новое правило;
- dev h1-eth0: указать интерфейс, на котором будет применяться правило;
- netem: использовать эмулятор сети;
- loss 10%: 10% потерь пакетов

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. Обратите внимание на значения icmp_seq. Некоторые номера последовательности отсутствуют из-за потери пакетов. В сводном отчёте ping сообщает о проценте потерянных пакетов после завершения передачи.
3. Для эмуляции глобальной сети с потерей пакетов в обоих направлениях необходимо к соответствующему интерфейсу на хосте h2 также добавить 10% потерь пакетов:

```
sudo tc qdisc add dev h2-eth0 root netem loss 10%
```

4. Проверьте, что соединение между хостом h1 и хостом h2 имеет больший 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. Укажите в отчёте отсутствующие из-за потери пакетов номера последовательности (значения icmp_seq), процент потерянных пакетов после завершения передачи.

The image shows two terminal windows side-by-side. The left window, titled "host: h1" @mininet-vm, displays the output of a ping command from 10.0.0.2 to 10.0.0.2, showing 100 packets transmitted with 14% packet loss and a time of 101381ms. It also shows the command "sudo tc qdisc del dev h1-eth0 root netem" being executed. The right window, titled "host: h2" @mininet-vm, shows the configuration of the h2-eth0 interface with "tc qdisc add dev h2-eth0 root netem loss 10%" and the command "sudo tc qdisc del dev h2-eth0 root netem" being executed. It also shows the configuration of the h2-rth0 interface with "tc qdisc add dev h2-rth0 root netem loss 10%".

```
host: h1" @mininet-vm
64 bytes from 10.0.0.2: icmp_seq=84 ttl=64 time=0.061 ms
64 bytes from 10.0.0.2: icmp_seq=85 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=86 ttl=64 time=0.070 ms
64 bytes from 10.0.0.2: icmp_seq=87 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=89 ttl=64 time=0.083 ms
64 bytes from 10.0.0.2: icmp_seq=90 ttl=64 time=0.056 ms
64 bytes from 10.0.0.2: icmp_seq=91 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=92 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=93 ttl=64 time=0.068 ms
64 bytes from 10.0.0.2: icmp_seq=94 ttl=64 time=0.053 ms
64 bytes from 10.0.0.2: icmp_seq=95 ttl=64 time=0.053 ms
64 bytes from 10.0.0.2: icmp_seq=96 ttl=64 time=0.071 ms
64 bytes from 10.0.0.2: icmp_seq=98 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=99 ttl=64 time=0.055 ms
64 bytes from 10.0.0.2: icmp_seq=100 ttl=64 time=0.064 ms

--- 10.0.0.2 ping statistics ---
100 packets transmitted, 86 received, 14% packet loss, time 101381ms
rtt min/avg/max/mdev = 0.043/0.068/0.526/0.066 ms
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem

host: h2" @mininet-vm
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 1964 bytes 985100 (985.1 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 1964 bytes 985100 (985.1 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@mininet-vm:/home/mininet# sudo tc qdisc add dev h2-rth0 root netem loss 10%
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h2-eth0 root netem loss 10%
root@mininet-vm:/home/mininet#
```

Рис. 2.1: Добавление потери пакетов на интерфейс

5. Восстановите конфигурацию по умолчанию, удалив все правила, применённые к сетевому планировщику соответствующего интерфейса. Для отправителя h1:

```
sudo tc qdisc del dev h1-eth0 root netem
```

Для получателя h2:

```
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.

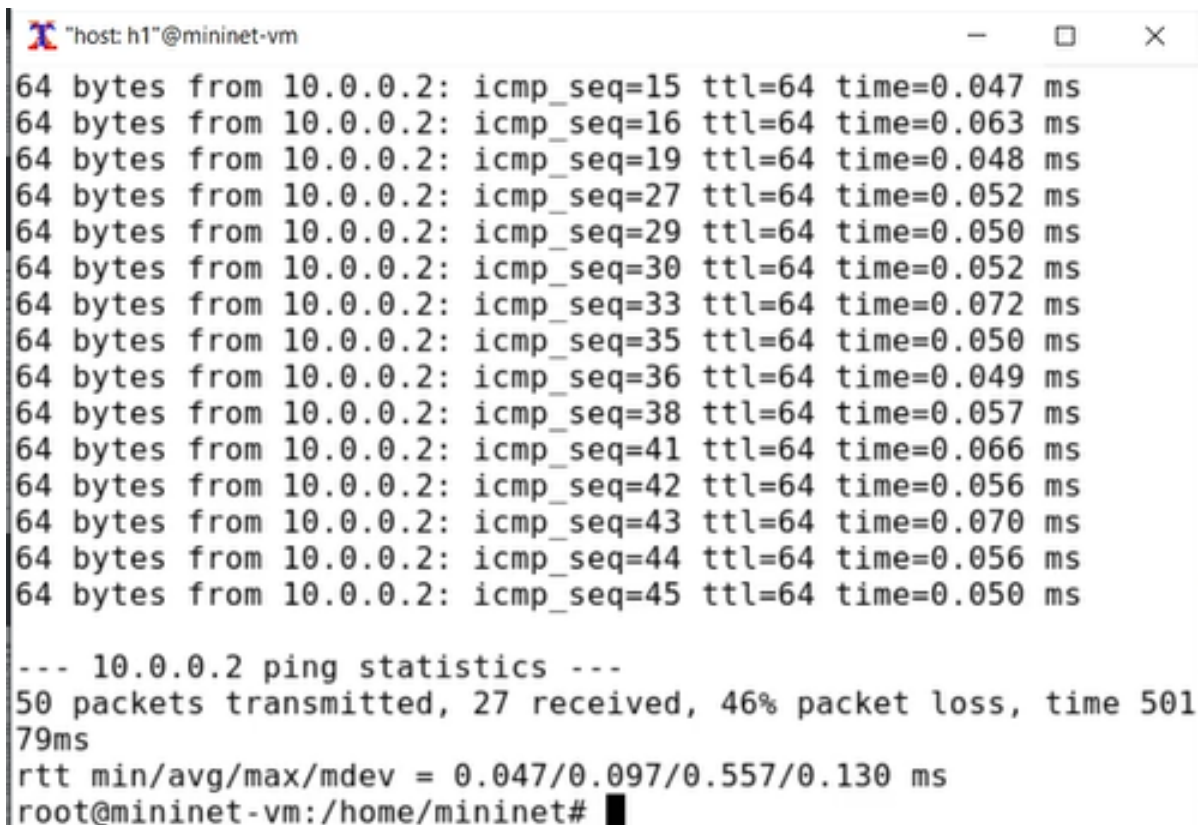
2.4 Добавление значения корреляции для потери пакетов в эмулируемой глобальной сети

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:

```
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. Укажите в отчёте отсутствующие из-за потери пакетов номера последовательности (значения icmp_seq), процент потерянных пакетов после завершения передачи.
3. Восстановите для узла h1 конфигурацию по умолчанию, удалив все правила, применённые к сетевому планировщику соответствующего интерфейса:

```
sudo tc qdisc del dev h1-eth0 root netem
```



```
"host: h1"@mininet-vm
64 bytes from 10.0.0.2: icmp_seq=15 ttl=64 time=0.047 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=0.063 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=0.048 ms
64 bytes from 10.0.0.2: icmp_seq=27 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=29 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=30 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=33 ttl=64 time=0.072 ms
64 bytes from 10.0.0.2: icmp_seq=35 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=36 ttl=64 time=0.049 ms
64 bytes from 10.0.0.2: icmp_seq=38 ttl=64 time=0.057 ms
64 bytes from 10.0.0.2: icmp_seq=41 ttl=64 time=0.066 ms
64 bytes from 10.0.0.2: icmp_seq=42 ttl=64 time=0.056 ms
64 bytes from 10.0.0.2: icmp_seq=43 ttl=64 time=0.070 ms
64 bytes from 10.0.0.2: icmp_seq=44 ttl=64 time=0.056 ms
64 bytes from 10.0.0.2: icmp_seq=45 ttl=64 time=0.050 ms

--- 10.0.0.2 ping statistics ---
50 packets transmitted, 27 received, 46% packet loss, time 501
79ms
rtt min/avg/max/mdev = 0.047/0.097/0.557/0.130 ms
root@mininet-vm:/home/mininet#
```

Рис. 2.2: Добавление значения корреляции для потери пакетов

2.5 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:

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:
iperf3 -s
 - запустите iPerf3 в клиентском режиме в терминале хоста h1:
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.

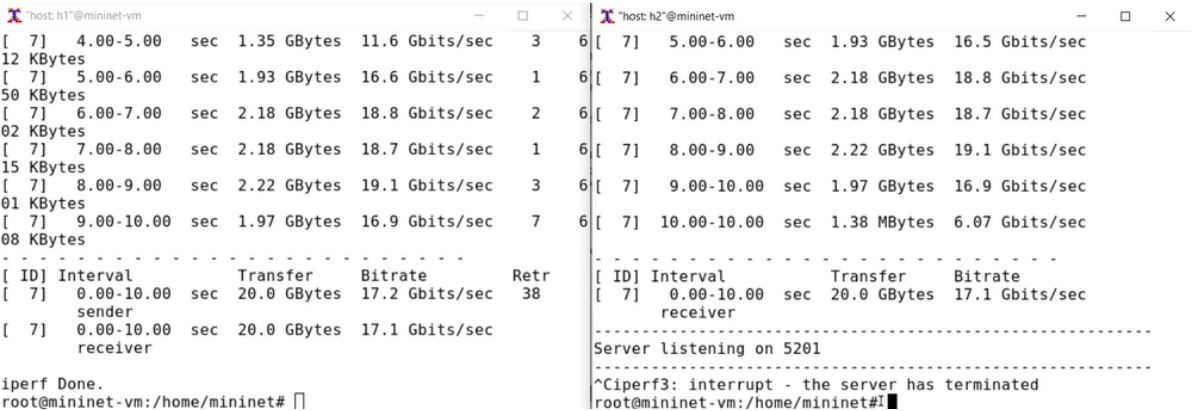


Рис. 2.3: Добавление повреждения пакетов

2.6 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:

```
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%). При необходимости повторите тест. Укажите в отчёте отсутствующие из-за потери пакетов номера последовательности (значения `icmp_seq`), процент потерянных пакетов после завершения передачи.
4. Восстановите конфигурацию интерфейса по умолчанию на узле h1.

```
host h1@mininet-vm
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=10.1 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.049 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=10.7 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=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=10.8 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=10.4 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.075 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=10.8 ms
64 bytes from 10.0.0.2: icmp_seq=15 ttl=64 time=10.8 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=10.3 ms
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=10.8 ms
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=10.8 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=20 ttl=64 time=10.1 ms
--- 10.0.0.2 ping statistics ---
20 packets transmitted, 20 received, 0% packet loss, time 1909
5ms
rtt min/avg/max/mdev = 0.049/9.108/11.720/3.816 ms
root@mininet-vm:/home/mininet#
```

```
host h2@mininet-vm
[ 7] 5.00-6.00 sec 1.93 GBytes 16.5 Gbits/sec
[ 7] 6.00-7.00 sec 2.18 GBytes 18.8 Gbits/sec
[ 7] 7.00-8.00 sec 2.18 GBytes 18.7 Gbits/sec
[ 7] 8.00-9.00 sec 2.22 GBytes 19.1 Gbits/sec
[ 7] 9.00-10.00 sec 1.97 GBytes 16.9 Gbits/sec
[ 7] 10.00-10.00 sec 1.38 MBytes 6.07 Gbits/sec
-----
[ ID] Interval Transfer Bitrate
[ 7] 0.00-10.00 sec 20.0 GBytes 17.1 Gbits/sec
receiver
-----
Server listening on 5201
-----
^Ciperf3: interrupt - the server has terminated
root@mininet-vm:/home/mininet#
```

Рис. 2.4: Добавление переупорядочивания пакетов

2.7 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):

```
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
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.048 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=0.271 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=0.271 ms (DUP!)
)
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=12 ttl=64 time=0.056 ms (DUP!)
)
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.063 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=0.050 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=0.045 ms
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=0.069 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=0.057 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=0.057 ms (DUP!)
)
64 bytes from 10.0.0.2: icmp_seq=20 ttl=64 time=0.054 ms

--- 10.0.0.2 ping statistics ---
20 packets transmitted, 20 received, +8 duplicates, 0% packet
loss, time 19425ms
rtt min/avg/max/mdev = 0.045/0.212/1.278/0.335 ms
root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 20
```

Рис. 2.5: Добавление дублирования пакетов

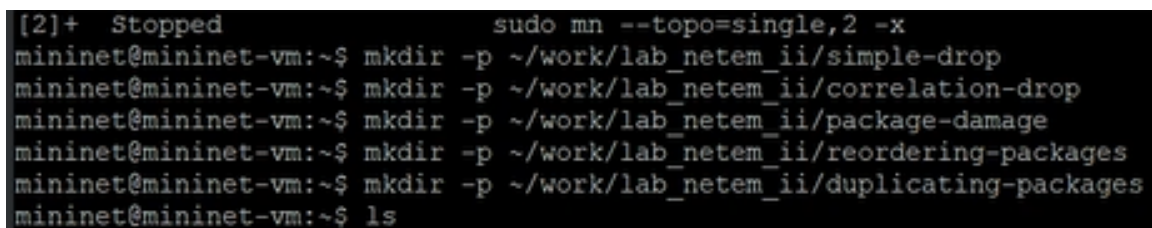
3 Воспроизведение экспериментов

3.1 Предварительная подготовка

1. Для каждого воспроизводимого эксперимента `expname` создайте свой каталог, в котором будут размещаться файлы эксперимента:

```
mkdir -p ~/work/lab_netem_ii/expname
```

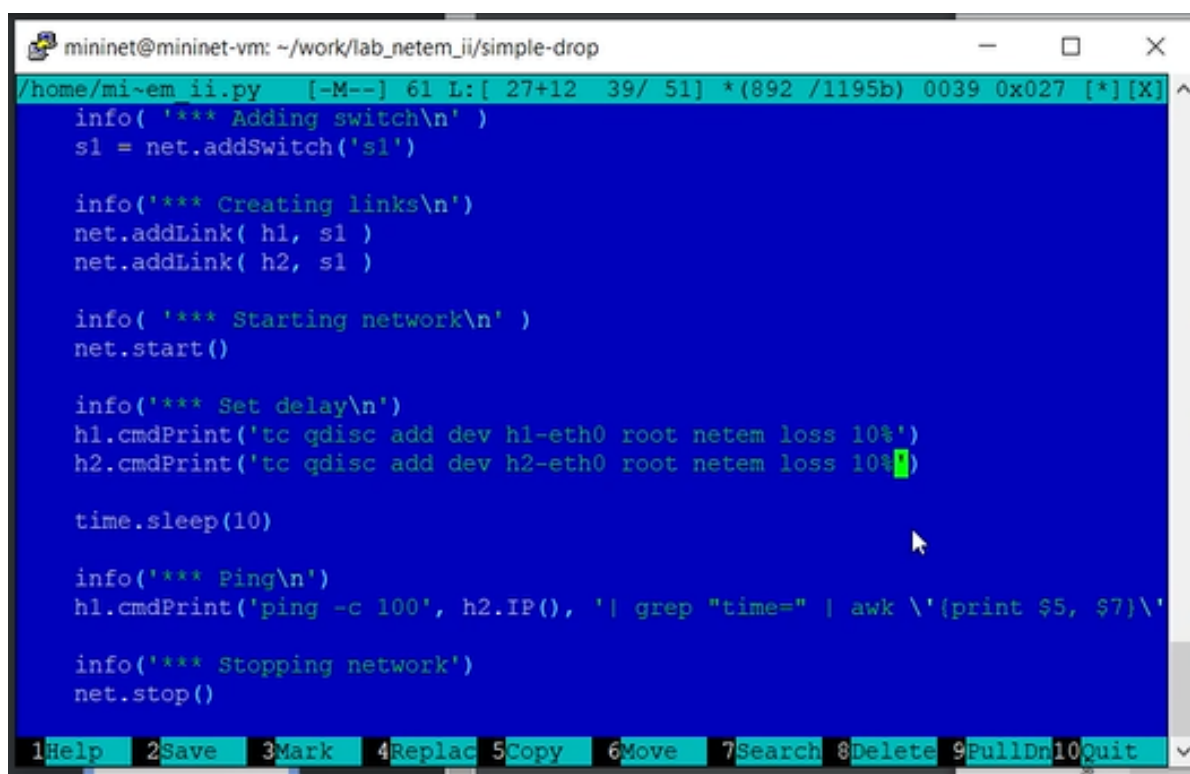
Здесь `expname` может принимать значения `simple-drop`, `correlationdrop` и т.п.

A screenshot of a terminal window showing a series of commands to create directories. The prompt is 'mininet@mininet-vm:~\$'. The commands are: 'mkdir -p ~/work/lab_netem_ii/simple-drop', 'mkdir -p ~/work/lab_netem_ii/correlation-drop', 'mkdir -p ~/work/lab_netem_ii/package-damage', 'mkdir -p ~/work/lab_netem_ii/reordering-packages', 'mkdir -p ~/work/lab_netem_ii/duplicating-packages', and 'ls'. The output of 'ls' is not visible.

```
[2]+  Stopped                  sudo mn --topo=single,2 -x
mininet@mininet-vm:~$ mkdir -p ~/work/lab_netem_ii/simple-drop
mininet@mininet-vm:~$ mkdir -p ~/work/lab_netem_ii/correlation-drop
mininet@mininet-vm:~$ mkdir -p ~/work/lab_netem_ii/package-damage
mininet@mininet-vm:~$ mkdir -p ~/work/lab_netem_ii/reordering-packages
mininet@mininet-vm:~$ mkdir -p ~/work/lab_netem_ii/duplicating-packages
mininet@mininet-vm:~$ ls
```

Рис. 3.1: Создаем каталоги

2. Для каждого случая создайте скрипт для проведения эксперимента `lab_netem_ii.py`.



```
mininet@mininet-vm: ~/work/lab_netem_ii/simple-drop
/home/mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ python ii.py
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)

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

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

Рис. 3.2: Создаем скрипт

Выполните эксперимент:

make

```

*** 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 "time=" | awk \'{print $5, $7}\'' |
sed -e \'/s/time=//g\' -e \'/s/icmp_seq=//g\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
mininet@mininet-vm:~/work/lab_netem_ii/correlation-drop$ make stats
python stats.py
Total packets: 100
Lost packets: 20
Lost packet numbers: [3, 17, 19, 24, 27, 32, 35, 36, 49, 59, 62, 69, 75, 76, 77
, 92, 93, 94, 96, 100]
Loss percentage: 20.00%

```

Рис. 3.3: Выполняем эксперимент

3.2 Задание для самостоятельной работы

Самостоятельно реализуйте воспроизводимые эксперименты по исследованию параметров сети, связанных с потерей, изменением порядка и повреждением пакетов при передаче данных.


```

*** 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 "time=" | awk \'{print $5, $7}\'' |
sed -e \'/s/time=//g\' -e \'/s/icmp_seq=//g\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
mininet@mininet-vm:~/work/lab_netem_ii/correlation-drop$ make stats
python stats.py
Total packets: 100
Lost packets: 20
Lost packet numbers: [3, 17, 19, 24, 27, 32, 35, 36, 49, 59, 62, 69, 75, 76, 77
, 92, 93, 94, 96, 100]
Loss percentage: 20.00%

```

Рис. 3.4: Выполняем эксперимент

Добавление значения корреляции для потери пакетов в эмулируемой глобальной сети

```
mininet@mininet-vm: ~/work/lab_netem_ii/correlation-drop
*** h1 : ('tc qdisc add dev h1-eth0 root netem loss 50% 50%,')
*** h2 : ('tc qdisc add dev h2-eth0 root netem loss 10%,')
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "time=" | awk \'{print $5, $7}\\' |
sed -e \'/time=//g\' -e \'/icmp_seq=//g\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
mininet@mininet-vm:~/work/lab_netem_ii/correlation-drop$ make stats
python stats.py
Total packets: 100
Lost packets: 53
Lost packet numbers: [2, 3, 4, 11, 12, 13, 14, 15, 17, 20, 21, 22, 26, 27, 28,
29, 30, 35, 36, 37, 38, 42, 44, 46, 54, 55, 60, 62, 63, 64, 65, 68, 69, 70, 73,
77, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 91, 93, 95, 98, 99, 100]
Loss percentage: 53.00%
mininet@mininet-vm:~/work/lab_netem_ii/correlation-drop$
```

Рис. 3.5: Выполняем эксперимент

Добавление повреждения пакетов в эмулируемой глобальной сети

```
mininet@mininet-vm: ~/work/lab_netem_ii/package-damage
s1
*** Set delay
*** h1 : ('tc qdisc add dev h1-eth0 root netem corrupt 0.01%',)
*** h2 : ('tc qdisc add dev h2-eth0 root netem loss 10%',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "time=" | awk \'{print $5, $7}\'' |
sed -e \'/s/time=//g\' -e \'/s/icmp_seq=//g\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
mininet@mininet-vm:~/work/lab_netem_ii/package-damage$ make stats
python stats.py
Total packets: 100
Lost packets: 13
Lost packet numbers: [13, 41, 56, 66, 67, 70, 80, 81, 83, 93, 96, 98, 99]
Loss percentage: 13.00%
mininet@mininet-vm:~/work/lab_netem_ii/package-damage$
```

Рис. 3.6: Выполняем эксперимент

Добавление переупорядочивания пакетов в интерфейс подключения к эмулируемой глобальной сети

```
mininet@mininet-vm: ~/work/lab_netem_ii/reordering-packages
s1
*** Set delay
*** h1 : ('tc qdisc add dev h1-eth0 root netem delay 10ms reorder 25% 50%',)
*** h2 : ('tc qdisc add dev h2-eth0 root netem loss 10%',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "time=" | awk \'{print $5, $7}\\' |
sed -e \\'s/time=//g\\' -e \\'s/icmp_seq=//g\\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
mininet@mininet-vm:~/work/lab_netem_ii/reordering-packages$ make stats
python stats.py
Total packets: 100
Lost packets: 8
Lost packet numbers: [11, 58, 65, 66, 67, 72, 73, 95]
Loss percentage: 8.00%
mininet@mininet-vm:~/work/lab_netem_ii/reordering-packages$
```

Рис. 3.7: Выполняем эксперимент

4 Выводы

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

5 Список литературы

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