

Лабораторная Работа №5.

Моделирование сетей передачи данных

Боровиков Д.А.

Российский университет дружбы народов им. Патриса Лумумбы, Москва, Россия

- Боровиков Даниил Александрович
- НПИБд-01-22
- Российский университет дружбы народов
- [1132222006@pfur.ru]

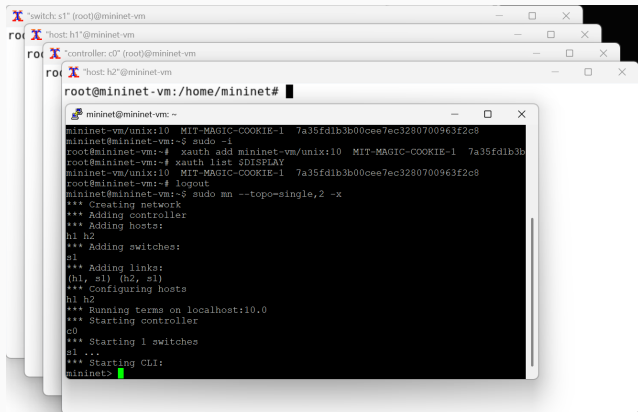
Основной целью работы является получение навыков проведения интерактивных экспериментов в среде Mininet по исследованию параметров сети, связанных с потерей, дублированием, изменением порядка и повреждением пакетов при передаче данных. Эти параметры влияют на производительность протоколов и сетей.

Исправление прав запуска

```
mininet@mininet-vm:~$ xauth list $DISPLAY
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 7a35fd1b3b00cee7ec3280700963f2c8
mininet@mininet-vm:~$ sudo -i
root@mininet-vm:~# xauth add mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 7a35fd1b3b00cee7ec3280700963f2c8
root@mininet-vm:~# xauth list $DISPLAY
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 7a35fd1b3b00cee7ec3280700963f2c8
root@mininet-vm:~#
```

Figure 1: Исправление прав запуска X-соединения в виртуальной машине mininet

Создание простейшей топологии

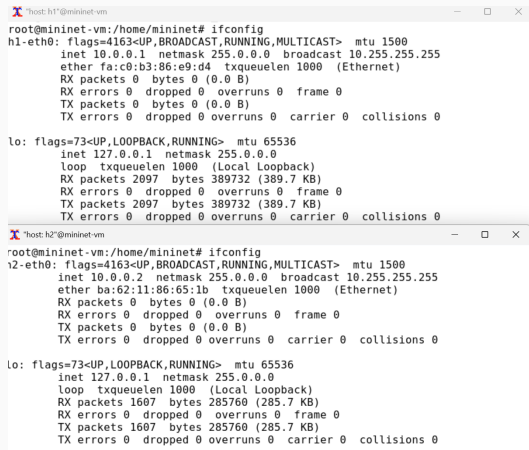


The image shows a terminal window titled "mininet@mininet-vm: ~" with a green cursor. The terminal output displays the steps to create a network topology using Mininet. It starts with adding a user, then running the 'mn' command with specific options to create a single-topology network with two hosts (h1, h2) and one switch (s1). The output shows the successful creation and configuration of the network.

```
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 7a35fdb3b00cee7ec3280700963f2c8
mininet@mininet-vm:~$ sudo -i
root@mininet-vm:~# xauth add mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 7a35fdb3b
root@mininet-vm:~# xauth list $DISPLAY
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 7a35fdb3b00cee7ec3280700963f2c8
root@mininet-vm:~# logout
mininet@mininet-vm:~$ sudo mn --topo=single,2 -x
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
*** Configuring hosts
h1 h2
*** Running terms on localhost:10.0
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet>
```

Figure 2: Создание простейшей топологии

Отображение информации их сетевых интерфейсов и IP-адресов



```
host: h1@mininet-vm
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 fa:c0:b3:86:e9:d4 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

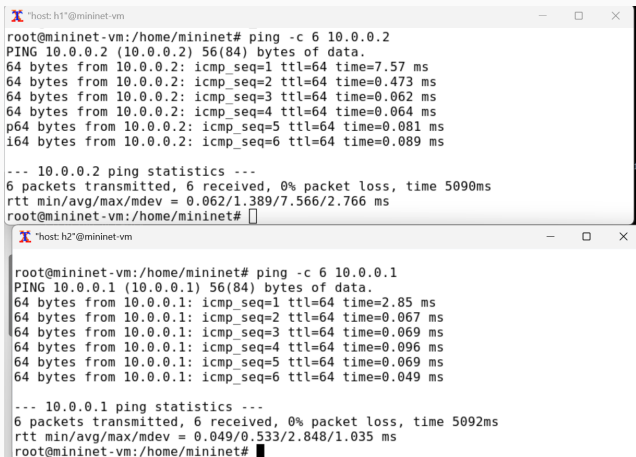
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    loop txqueuelen 1000 (Local Loopback)
    RX packets 2097 bytes 389732 (389.7 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2097 bytes 389732 (389.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

host: h2@mininet-vm
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 ba:62:11:86:65:1b 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 1607 bytes 285760 (285.7 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1607 bytes 285760 (285.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Figure 3: Отображение информации их сетевых интерфейсов и IP-адресов

Проверка подключения между хостами h1 и h2



The image displays two terminal windows from a Mininet VM. The top window, titled "host: h1" and "mininet-vm", shows a successful ping to 10.0.0.2. The output includes details for 6 ICMP packets, showing TTL=64 and various response times. The statistics section indicates 6 packets transmitted and received with 0% packet loss and a total time of 5090ms. The bottom window, titled "host: h2" and "mininet-vm", shows a successful ping to 10.0.0.1. Similar to the first, it shows details for 6 ICMP packets and statistics indicating 6 packets transmitted and received with 0% packet loss and a total time of 5092ms.

```
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=7.57 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.473 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.062 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.064 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.081 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.089 ms

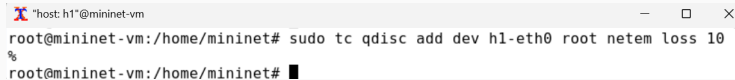
--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5090ms
rtt min/avg/max/mdev = 0.062/1.389/7.566/2.766 ms
root@mininet-vm:/home/mininet#
```

```
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=2.85 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.067 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.069 ms
64 bytes from 10.0.0.1: icmp_seq=4 ttl=64 time=0.096 ms
64 bytes from 10.0.0.1: icmp_seq=5 ttl=64 time=0.069 ms
64 bytes from 10.0.0.1: icmp_seq=6 ttl=64 time=0.049 ms

--- 10.0.0.1 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5092ms
rtt min/avg/max/mdev = 0.049/0.533/2.848/1.035 ms
root@mininet-vm:/home/mininet#
```

Figure 4: Проверка подключения между хостами h1 и h2

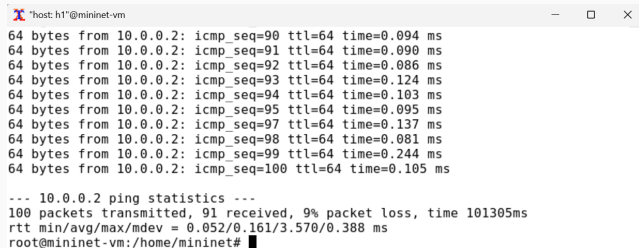
Добавление 10% потерь пакетов на хосте h1



A terminal window titled "host: h1"@mininet-vm. The prompt is root@mininet-vm:/home/mininet#. The command entered is sudo tc qdisc add dev h1-eth0 root netem loss 10%. The cursor is at the end of the command line.

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

Figure 5: Добавление 10% потерь пакетов на хосте h1

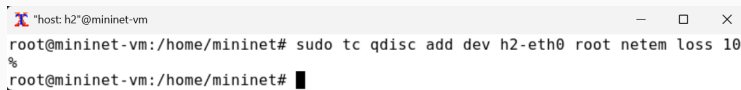


```
*host: h1"@mininet-vm
64 bytes from 10.0.0.2: icmp_seq=90 ttl=64 time=0.094 ms
64 bytes from 10.0.0.2: icmp_seq=91 ttl=64 time=0.090 ms
64 bytes from 10.0.0.2: icmp_seq=92 ttl=64 time=0.086 ms
64 bytes from 10.0.0.2: icmp_seq=93 ttl=64 time=0.124 ms
64 bytes from 10.0.0.2: icmp_seq=94 ttl=64 time=0.103 ms
64 bytes from 10.0.0.2: icmp_seq=95 ttl=64 time=0.095 ms
64 bytes from 10.0.0.2: icmp_seq=97 ttl=64 time=0.137 ms
64 bytes from 10.0.0.2: icmp_seq=98 ttl=64 time=0.081 ms
64 bytes from 10.0.0.2: icmp_seq=99 ttl=64 time=0.244 ms
64 bytes from 10.0.0.2: icmp_seq=100 ttl=64 time=0.105 ms

--- 10.0.0.2 ping statistics ---
100 packets transmitted, 91 received, 9% packet loss, time 101305ms
rtt min/avg/max/mdev = 0.052/0.161/3.570/0.388 ms
root@mininet-vm:/home/mininet#
```

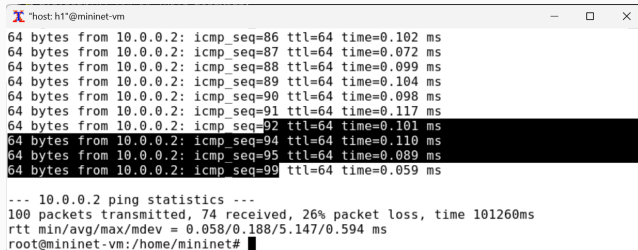
Figure 6: Проверка

Добавление 10% потерь пакетов на хосте h2



```
"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#
```

Figure 7: Добавление 10% потерь пакетов на хосте h2

A terminal window titled "host: h1@mininet-vm" with standard window controls. It displays the output of a ping command to 10.0.0.2. The output shows 10 successful ping requests, each with a 64-byte payload, TTL of 64, and response times ranging from 0.059 ms to 0.117 ms. The last two lines of the ping output are highlighted with a black background. Below the ping results, the statistics for 10.0.0.2 are shown: 100 packets transmitted, 74 received, 26% packet loss, and a total time of 101260ms. The RTT statistics are also displayed: min/avg/max/mdev = 0.058/0.188/5.147/0.594 ms. The prompt at the bottom is root@mininet-vm:/home/mininet#.

```

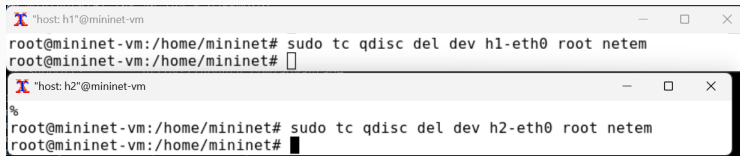
host: h1@mininet-vm
64 bytes from 10.0.0.2: icmp_seq=86 ttl=64 time=0.102 ms
64 bytes from 10.0.0.2: icmp_seq=87 ttl=64 time=0.072 ms
64 bytes from 10.0.0.2: icmp_seq=88 ttl=64 time=0.099 ms
64 bytes from 10.0.0.2: icmp_seq=89 ttl=64 time=0.104 ms
64 bytes from 10.0.0.2: icmp_seq=90 ttl=64 time=0.098 ms
64 bytes from 10.0.0.2: icmp_seq=91 ttl=64 time=0.117 ms
64 bytes from 10.0.0.2: icmp_seq=92 ttl=64 time=0.101 ms
64 bytes from 10.0.0.2: icmp_seq=94 ttl=64 time=0.110 ms
64 bytes from 10.0.0.2: icmp_seq=95 ttl=64 time=0.089 ms
64 bytes from 10.0.0.2: icmp_seq=99 ttl=64 time=0.059 ms

--- 10.0.0.2 ping statistics ---
100 packets transmitted, 74 received, 26% packet loss, time 101260ms
rtt min/avg/max/mdev = 0.058/0.188/5.147/0.594 ms
root@mininet-vm:/home/mininet#

```

Figure 8: Проверка

Восстановление конфигурации

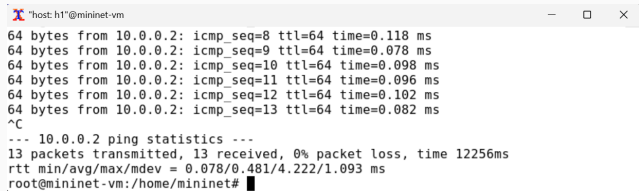


The image shows two terminal windows. The top window is titled "host: h1"@mininet-vm and shows the command `sudo tc qdisc del dev h1-eth0 root netem` being executed. The bottom window is titled "host: h2"@mininet-vm and shows the command `sudo tc qdisc del dev h2-eth0 root netem` being executed. Both windows show the prompt `root@mininet-vm:/home/mininet#`.

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

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

Figure 9: Восстановление конфигурации по умолчанию для хоста h1 и хоста h2



```
root@mininet-vm:~# ping 10.0.0.2
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.118 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=0.078 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.098 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=0.096 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.102 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.082 ms
^C
--- 10.0.0.2 ping statistics ---
13 packets transmitted, 13 received, 0% packet loss, time 12256ms
rtt min/avg/max/mdev = 0.078/0.481/4.222/1.093 ms
root@mininet-vm:/home/mininet#
```

Figure 10: Проверка

Добавление коэффициента потери пакетов 50%

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

Figure 11: Добавление на узле h1 коэффициента потери пакетов 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=4 ttl=64 time=2.76 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.949 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.293 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.078 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.121 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=0.100 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.075 ms
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=0.100 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=0.106 ms
64 bytes from 10.0.0.2: icmp_seq=22 ttl=64 time=0.110 ms
64 bytes from 10.0.0.2: icmp_seq=23 ttl=64 time=0.090 ms
64 bytes from 10.0.0.2: icmp_seq=24 ttl=64 time=0.218 ms
64 bytes from 10.0.0.2: icmp_seq=29 ttl=64 time=0.093 ms
64 bytes from 10.0.0.2: icmp_seq=32 ttl=64 time=0.085 ms
64 bytes from 10.0.0.2: icmp_seq=33 ttl=64 time=0.108 ms
64 bytes from 10.0.0.2: icmp_seq=34 ttl=64 time=0.088 ms
64 bytes from 10.0.0.2: icmp_seq=35 ttl=64 time=0.088 ms
64 bytes from 10.0.0.2: icmp_seq=36 ttl=64 time=0.168 ms
64 bytes from 10.0.0.2: icmp_seq=37 ttl=64 time=0.110 ms
64 bytes from 10.0.0.2: icmp_seq=39 ttl=64 time=0.104 ms
64 bytes from 10.0.0.2: icmp_seq=40 ttl=64 time=0.094 ms
64 bytes from 10.0.0.2: icmp_seq=41 ttl=64 time=0.059 ms
64 bytes from 10.0.0.2: icmp_seq=42 ttl=64 time=0.113 ms
64 bytes from 10.0.0.2: icmp_seq=43 ttl=64 time=0.109 ms
64 bytes from 10.0.0.2: icmp_seq=44 ttl=64 time=0.090 ms
64 bytes from 10.0.0.2: icmp_seq=45 ttl=64 time=0.094 ms
64 bytes from 10.0.0.2: icmp_seq=46 ttl=64 time=0.135 ms
64 bytes from 10.0.0.2: icmp_seq=47 ttl=64 time=0.104 ms
64 bytes from 10.0.0.2: icmp_seq=48 ttl=64 time=0.099 ms

--- 10.0.0.2 ping statistics ---
50 packets transmitted, 29 received, 42% packet loss, time 50121ms
rtt min/avg/max/mdev = 0.059/0.232/2.759/0.503 ms
root@mininet-vm:/home/mininet#
```

[Свернуть все окна](#)

Figure 12: Проверка

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

Figure 13: Восстановление конфигурации интерфейса по умолчанию

Добавление повреждения пакетов

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

Figure 14: Добавление на узле h1 0.01% повреждения пакетов

Проверка конфигурации

```

X "host: h1" @mininet-vm
1%: command not found
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 41694 connected to 10.0.0.2 port 5201
[ ID] Interval      Transfer    Bitrate    Retr  Cwnd
[ 7]  0.00-1.00    sec    985 MBytes  8.24 Gbits/sec    26   1.66 MBytes
[ 7]  1.00-2.00    sec    1.43 GBytes 12.3 Gbits/sec     3   1.41 MBytes
[ 7]  2.00-3.00    sec    1.45 GBytes 12.5 Gbits/sec     4   1.27 MBytes
[ 7]  3.00-4.00    sec    1.28 GBytes 11.0 Gbits/sec     1   1.26 MBytes
[ 7]  4.00-5.00    sec    1.63 GBytes 13.9 Gbits/sec     3   2.23 MBytes
[ 7]  5.00-6.01    sec    1.38 GBytes 11.8 Gbits/sec     1   1.89 MBytes
[ 7]  6.01-7.00    sec    1.35 GBytes 11.6 Gbits/sec     6   1.41 MBytes
[ 7]  7.00-8.01    sec    1.54 GBytes 13.1 Gbits/sec     3   1.26 MBytes
[ 7]  8.01-9.00    sec    1.60 GBytes 13.9 Gbits/sec     5   1.52 MBytes
[ 7]  9.00-10.00   sec    1.57 GBytes 13.5 Gbits/sec     4   2.21 MBytes
-----
[ ID] Interval      Transfer    Bitrate    Retr
[ 7]  0.00-10.00   sec    14.2 GBytes 12.2 Gbits/sec    56
[ 7]  0.00-10.01   sec    14.2 GBytes 12.2 Gbits/sec
-----
iperf Done.
root@mininet-vm:/home/mininet#

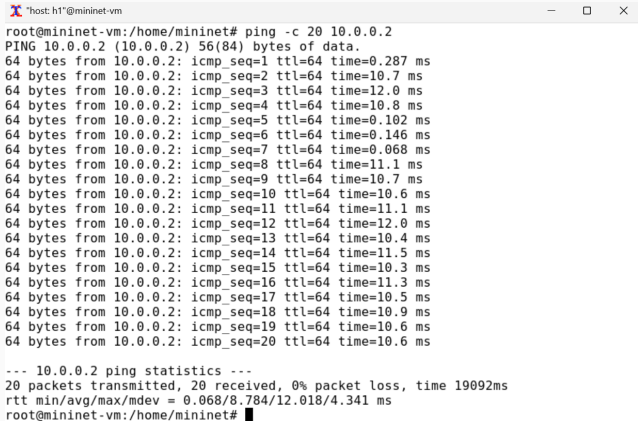
X "host: h2" @mininet-vm
root@mininet-vm:/home/mininet# iperf3 -s
warning: this system does not seem to support IPv6 - trying IPv4
Server listening on 5201
-----
Accepted connection from 10.0.0.1, port 41692
[ 7] local 10.0.0.2 port 5201 connected to 10.0.0.1 port 41694
[ ID] Interval      Transfer    Bitrate
[ 7]  0.00-1.00    sec    985 MBytes  8.26 Gbits/sec
[ 7]  1.00-2.00    sec    1.42 GBytes 12.2 Gbits/sec
[ 7]  2.00-3.00    sec    1.44 GBytes 12.3 Gbits/sec
[ 7]  3.00-4.00    sec    1.30 GBytes 11.2 Gbits/sec
[ 7]  4.00-5.00    sec    1.62 GBytes 13.9 Gbits/sec
[ 7]  5.00-6.00    sec    1.37 GBytes 11.8 Gbits/sec
[ 7]  6.00-7.00    sec    1.36 GBytes 11.7 Gbits/sec
[ 7]  7.00-8.00    sec    1.51 GBytes 12.9 Gbits/sec
[ 7]  8.00-9.00    sec    1.62 GBytes 13.9 Gbits/sec
[ 7]  9.00-10.00   sec    1.57 GBytes 13.6 Gbits/sec
[ 7] 10.00-10.01   sec     7.25 MBytes 11.3 Gbits/sec
-----
[ ID] Interval      Transfer    Bitrate
[ 7]  0.00-10.01   sec    14.2 GBytes 12.2 Gbits/sec
-----
Server listening on 5201
-----
receiver
```

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

Figure 16: Восстановление конфигурации интерфейса по умолчанию

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 10m  
s reorder 25% 50%  
root@mininet-vm:/home/mininet# █
```

Figure 17: Добавление на узле h1 правила из лабораторной работы



```
*host: h1"@mininet-vm
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=0.287 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=12.0 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=0.102 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.146 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.068 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=11.1 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=10.6 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=11.1 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=12.0 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=10.4 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=11.5 ms
64 bytes from 10.0.0.2: icmp_seq=15 ttl=64 time=10.3 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=11.3 ms
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=10.5 ms
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=10.9 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=10.6 ms
64 bytes from 10.0.0.2: icmp_seq=20 ttl=64 time=10.6 ms

--- 10.0.0.2 ping statistics ---
20 packets transmitted, 20 received, 0% packet loss, time 19092ms
rtt min/avg/max/mdev = 0.068/8.784/12.018/4.341 ms
root@mininet-vm:/home/mininet#
```

Figure 18: Проверка

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

Figure 19: Восстановление конфигурации интерфейса по умолчанию

Добавление правила с дублированием

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

Figure 20: Добавление на узле h1 правила с дублированием 50% пакетов

```
*host: h1*@mininet-vm
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=2.92 ms
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=2.96 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=1.14 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=1.75 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.299 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.073 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.084 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.085 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.106 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.085 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.075 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=0.077 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.116 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=0.071 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=0.072 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.103 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.104 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.083 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.083 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=0.072 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=0.073 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=15 ttl=64 time=0.105 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=0.112 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=0.112 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=0.084 ms
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=0.086 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=0.122 ms
64 bytes from 10.0.0.2: icmp_seq=20 ttl=64 time=0.074 ms

--- 10.0.0.2 ping statistics ---
20 packets transmitted, 20 received, +8 duplicates, 0% packet loss, time 19421ms
rtt min/avg/max/mdev = 0.071/0.397/2.961/0.790 ms
root@mininet-vm:/home/mininet#
```

Figure 21: Проверка


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

Figure 22: Восстановление конфигурации интерфейса по умолчанию

Создание каталога expname

```
mininet@mininet-vm:~$ mkdir -p ~/work/lab_netem_i1/expname  
mininet@mininet-vm:~$ |
```

Figure 23: Создание каталога expname

Создание каталога simple-drop

```
mininet@mininet-vm:~$ mkdir -p ~/work/lab_netem_ii/simple-drop
mininet@mininet-vm:~$ cd ~/work/lab_netem_ii/simple-drop
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ touch lab_netem_ii.py
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ ls
lab_netem_ii.py
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ nano lab_netem_ii.py |
```

Figure 24: Создание каталога simple-drop и дальнейшее его открытие

Создание скрипта для эксперимента

```
mininet@mininet-vm: ~/work  x  +  -  x
GNU nano 4.8 lab_netem_ii.py Modified
Output: ping.dat
***

from mininet.net import Mininet
from mininet.node import Controller
from mininet.cli import CLI
from mininet.log import setLogLevel, info
import time

def emptyNet():

    "Create an empty network and add nodes to it."

    net = Mininet( controller=Controller, waitConnected=True )
    info( '*** Adding controller\n' )
    net.addController( 'c0' )

    info( '*** Adding hosts\n' )
    h1 = net.addHost( 'h1', ip='10.0.0.1' )
    h2 = net.addHost( 'h2', ip='10.0.0.2' )

    info( '*** Adding switch\n' )
    s1 = net.addSwitch( 's1' )

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

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

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

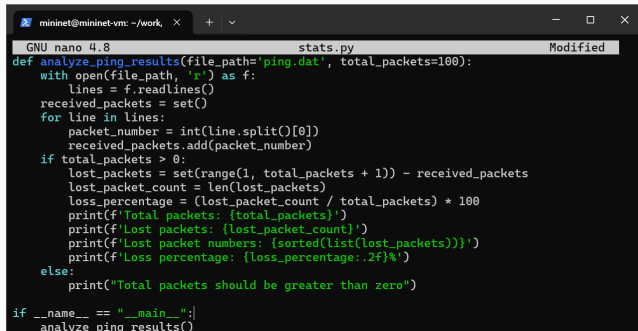
    time.sleep(10) # Wait 10 seconds

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

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

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

Создание скрипта

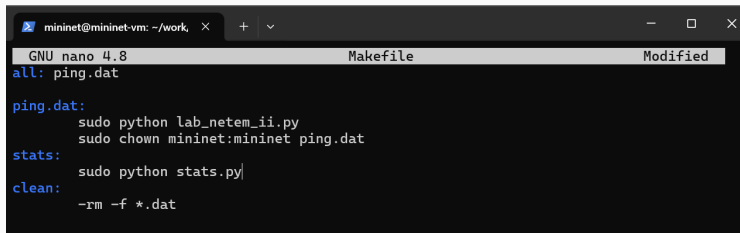


```
mininet@mininet-vm: ~/work, x + v
GNU nano 4.8 stats.py Modified
def analyze_ping_results(file_path='ping.dat', total_packets=100):
    with open(file_path, 'r') as f:
        lines = f.readlines()
        received_packets = set()
        for line in lines:
            packet_number = int(line.split()[0])
            received_packets.add(packet_number)
        if total_packets > 0:
            lost_packets = set(range(1, total_packets + 1)) - received_packets
            lost_packet_count = len(lost_packets)
            loss_percentage = (lost_packet_count / total_packets) * 100
            print(f'Total packets: {total_packets}')
            print(f'Lost packets: {lost_packet_count}')
            print(f'Lost packet numbers: {sorted(list(lost_packets))}')
            print(f'Loss percentage: {loss_percentage:.2f}%')
        else:
            print("Total packets should be greater than zero")

if __name__ == "__main__":
    analyze_ping_results()
```

Figure 26: Создание нового скрипта для вывода информации о потере пакетов

Создание Makefile



The screenshot shows a terminal window with a dark theme. The title bar indicates the user is 'mininet@mininet-vm' in the directory '~/work'. The editor is 'GNU nano 4.8' editing a file named 'Makefile'. The content of the Makefile is as follows:

```
all: ping.dat

ping.dat:
    sudo python lab_netem_ii.py
    sudo chown mininet:mininet ping.dat

stats:
    sudo python stats.py

clean:
    -rm -f *.dat
```

Figure 27: Создание Makefile и помещение в него скрипта

Выполнение эксперимента

```
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ 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 "time=" | awk \{print $5, $7\}\}' | sed -e '\s/time=//g\' -e '\s/icnp_seq=//g\' >
ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ make stats
sudo python stats.py
Total packets: 100
Lost packets: 24
Lost packet numbers: [11, 14, 20, 21, 23, 24, 25, 28, 30, 33, 34, 38, 39, 49, 52, 72, 74, 77, 78, 83, 86, 89, 93, 95]
Loss percentage: 24.00%
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ |
```

Figure 28: Выполнение эксперимента и последующая очистка каталога

Реализация воспроизводимого эксперимента

```
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ 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%',)
*** h2 : ('tc qdisc add dev h2-eth0 root netem',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "time=" | awk \'{print $5, $7}\'} | sed -e \'s/time=//g\' -e \'s/icmp_seq=//g\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ make stats
sudo python stats.py
Total packets: 100
Lost packets: 63
Lost packet numbers: [1, 6, 8, 9, 10, 11, 14, 15, 16, 17, 26, 27, 29, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 46, 49, 50, 51, 52, 53, 54, 55, 57, 58, 59, 65, 67, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 82, 83, 84, 87, 89, 90, 91, 94, 95, 97, 98, 99, 100]
Loss percentage: 63.00%
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ |
```

Figure 29: Реализация воспроизводимого эксперимента по исследованию параметров сети

Реализация воспроизводимого эксперимента

```
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ 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%',)
*** h2 : ('tc qdisc add dev h2-eth0 root netem',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "time=" | awk '{print $5, $7}' | sed -e 's/time=//g' -e 's/icmp_seq=//g' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ make stats
sudo python stats.py
Total packets: 100
Lost packets: 0
Lost packet numbers: []
Loss percentage: 0.00%
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ |
```

Figure 30: Реализация воспроизводимого эксперимента по исследованию параметров сети

Реализация воспроизводимого эксперимента

```
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ 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%,)
*** h2 : ('tc qdisc add dev h2-eth0 root netem',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "time=" | awk '{print $5, $7}' | sed -e 's/time=//g' -e 's/icnp_seq=//g' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ make stats
sudo python stats.py
Total packets: 100
Lost packets: 0
Lost packet numbers: []
Loss percentage: 0.00%
mininet@mininet-vm:~/work/lab_netem_ii/simple-drop$ make clean
```

Figure 31: Реализация воспроизводимого эксперимента по исследованию параметров сети

Реализация воспроизводимого эксперимента

```
mininet@mininet-va:~/work/lab_neten_ii/simple-drop$ make
sudo python lab_neten_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%,)
*** h2 : ('tc qdisc add dev h2-eth0 root netem',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "time=" | awk '{print $5, $7}' | sed -e 's/time=//g' -e 's/icmp_seq=//g' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
mininet@mininet-va:~/work/lab_neten_ii/simple-drop$ make stats
sudo python stats.py
Total packets: 100
Lost packets: 0
Last packet numbers: []
Loss percentage: 0.00%
mininet@mininet-va:~/work/lab_neten_ii/simple-drop$ |
```

Figure 32: Реализация воспроизводимого эксперимента по исследованию параметров сети

В ходе выполнения лабораторной работы 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. Эти параметры влияют на производительность протоколов и сетей.