

Лабораторная работа №4  
Моделирование сетей передачи данных

---

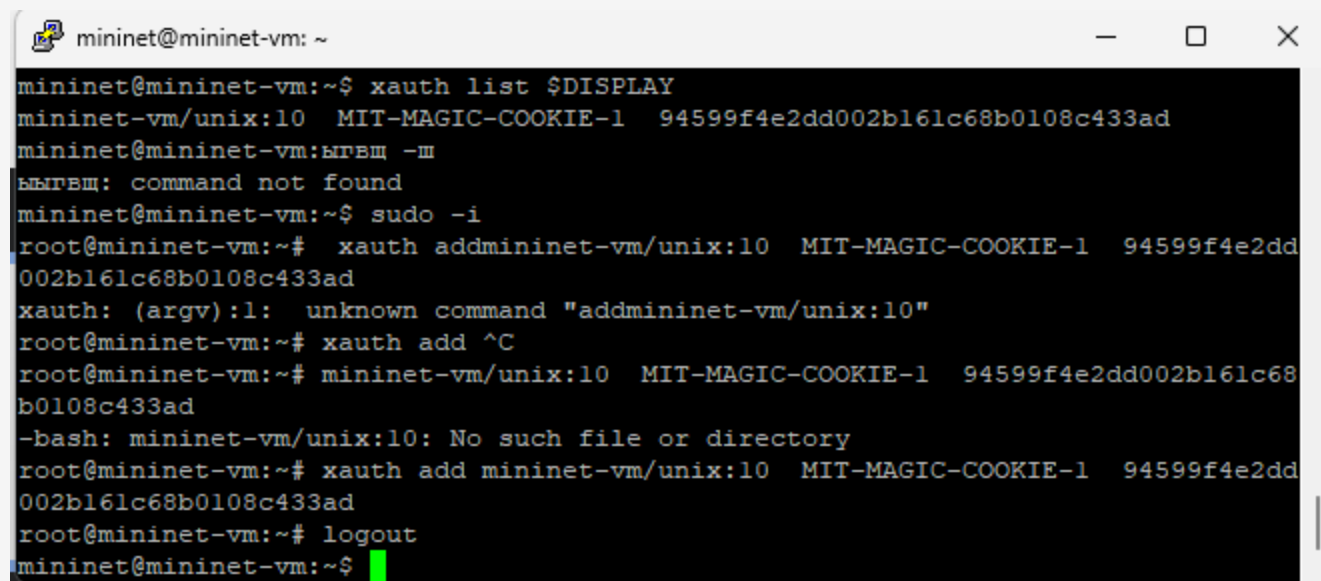
Исаев Б. А.

2025

Российский университет дружбы народов имени Патриса Лумумбы, Москва, Россия

## Запуск лабораторной топологии

---

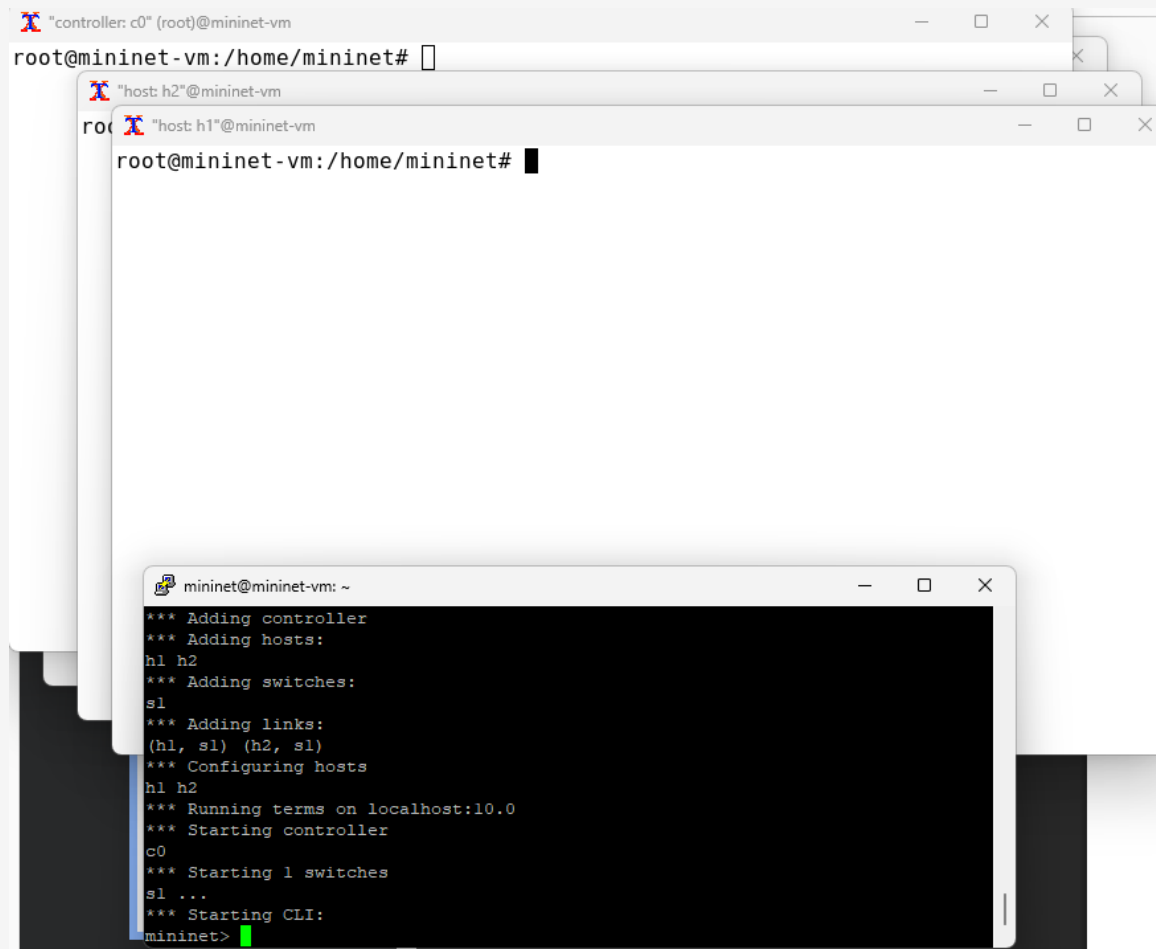


```
mininet@mininet-vm: ~  
mininet@mininet-vm:~$ xauth list $DISPLAY  
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 94599f4e2dd002b161c68b0108c433ad  
mininet@mininet-vm:~$ xauth -ш  
xauth: command not found  
mininet@mininet-vm:~$ sudo -i  
root@mininet-vm:~# xauth addmininet-vm/unix:10 MIT-MAGIC-COOKIE-1 94599f4e2dd002b161c68b0108c433ad  
xauth: (argv):1: unknown command "addmininet-vm/unix:10"  
root@mininet-vm:~# xauth add ^C  
root@mininet-vm:~# mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 94599f4e2dd002b161c68b0108c433ad  
-bash: mininet-vm/unix:10: No such file or directory  
root@mininet-vm:~# xauth add mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 94599f4e2dd002b161c68b0108c433ad  
root@mininet-vm:~# logout  
mininet@mininet-vm:~$
```

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

## Запуск лабораторной топологии

---

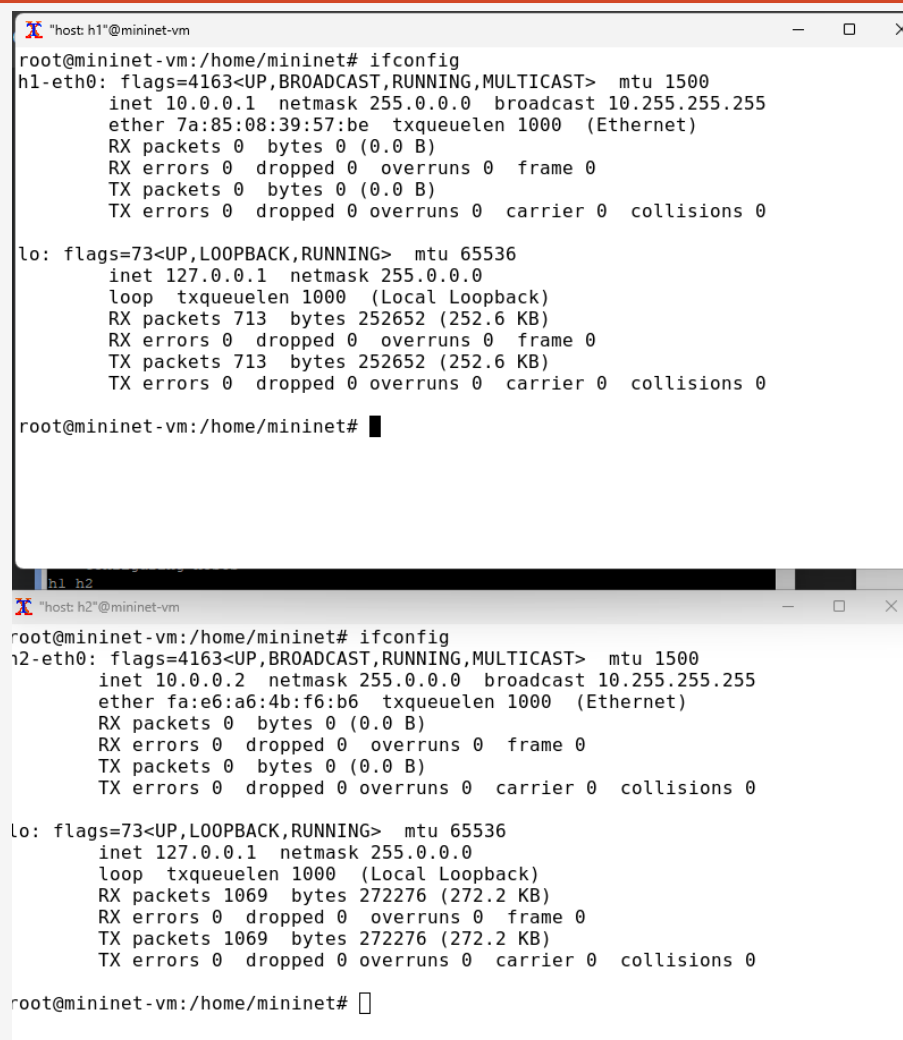


The image shows three overlapping terminal windows from the Mininet environment. The top window is titled "controller: c0" (root)@mininet-vm and shows the prompt root@mininet-vm:/home/mininet#. The middle window is titled "host: h2" (root)@mininet-vm and also shows the prompt root@mininet-vm:/home/mininet#. The bottom window is titled "mininet@mininet-vm: ~" and displays the output of the mininet command, which sets up a network topology with one controller (c0), two hosts (h1, h2), and one switch (s1).

```
*** 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>
```

Рис. 2: Создание простейшей топологии

## Запуск лабораторной топологии



The image displays two terminal windows from a Mininet VM. The top window, titled "host: h1@mininet-vm", shows the output of the 'ifconfig' command for host h1. It lists the configuration for the h1-eth0 interface (10.0.0.1) and the loopback interface lo (127.0.0.1). The bottom window, titled "host: h2@mininet-vm", shows the output of the 'ifconfig' command for host h2. It lists the configuration for the h2-eth0 interface (10.0.0.2) and the loopback interface lo (127.0.0.1). Both windows show standard network statistics like RX/TX packets, bytes, and errors.

```
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 7a:85:08:39:57:be 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 713 bytes 252652 (252.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 713 bytes 252652 (252.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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

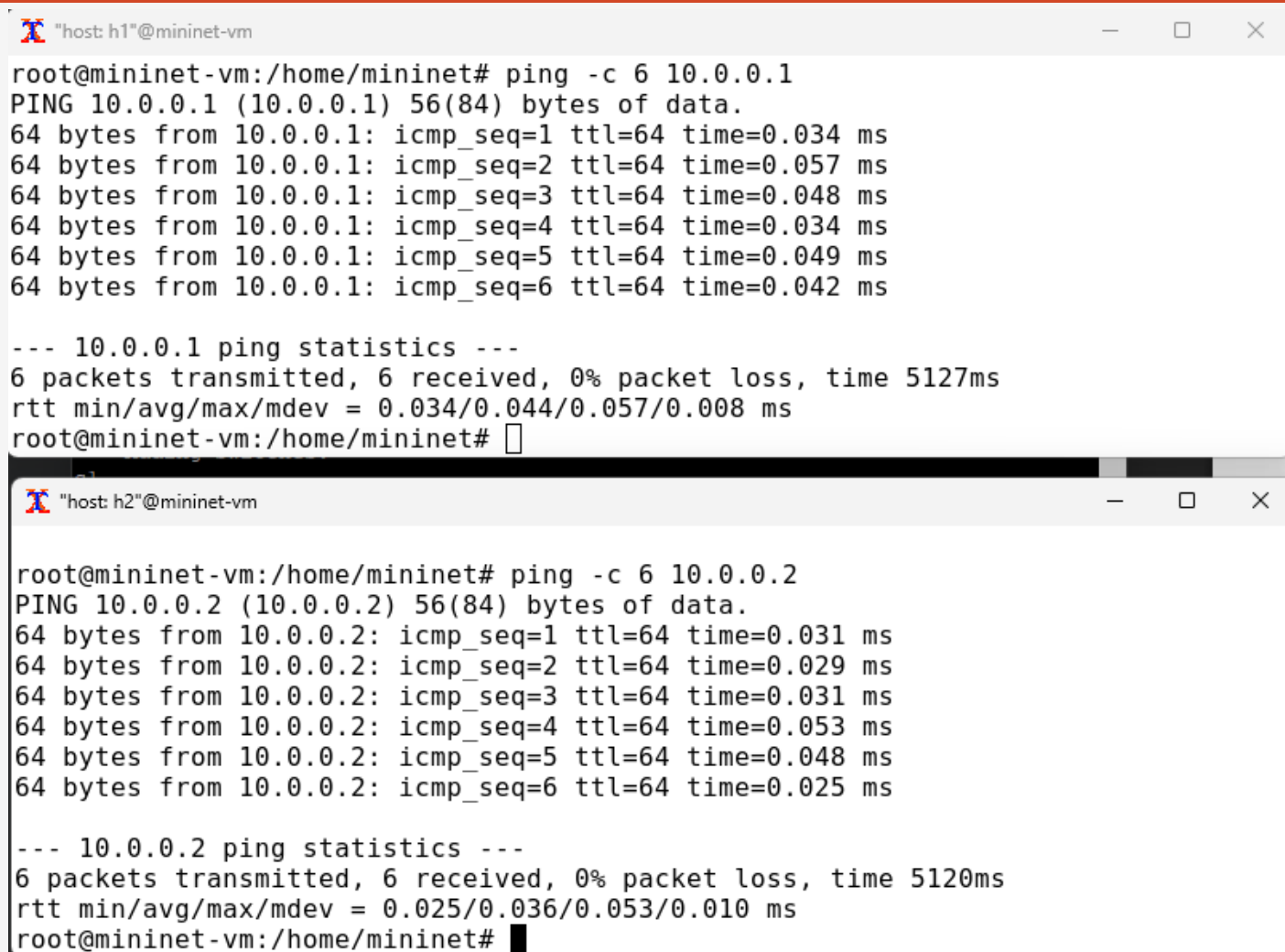
```
root@mininet-vm:/home/mininet# ifconfig
h2-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.2 netmask 255.0.0.0 broadcast 10.255.255.255
    ether fa:e6:a6:4b:f6:b6 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    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 1069 bytes 272276 (272.2 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1069 bytes 272276 (272.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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

Рис. 3: Отображение информации их сетевых интерфейсов и IP-адресов

## Запуск лабораторной топологии



The image shows two terminal windows from a Mininet virtual machine. The top window, titled '"host: h1"@mininet-vm', shows a ping command being executed from the root user at the mininet-vm prompt. The command is 'ping -c 6 10.0.0.1'. The output shows six successful ping packets with varying round-trip times (0.034 ms to 0.057 ms). Below the individual results, a summary line shows '--- 10.0.0.1 ping statistics ---' with '6 packets transmitted, 6 received, 0% packet loss, time 5127ms' and 'rtt min/avg/max/mdev = 0.034/0.044/0.057/0.008 ms'. The bottom window, titled '"host: h2"@mininet-vm', shows a similar ping command 'ping -c 6 10.0.0.2'. The output also shows six successful ping packets with round-trip times (0.029 ms to 0.053 ms). Its summary line shows '--- 10.0.0.2 ping statistics ---' with '6 packets transmitted, 6 received, 0% packet loss, time 5120ms' and 'rtt min/avg/max/mdev = 0.025/0.036/0.053/0.010 ms'. Both windows have a black scrollbar on the right side.

```
"host: h1"@mininet-vm
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=0.034 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.057 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.048 ms
64 bytes from 10.0.0.1: icmp_seq=4 ttl=64 time=0.034 ms
64 bytes from 10.0.0.1: icmp_seq=5 ttl=64 time=0.049 ms
64 bytes from 10.0.0.1: icmp_seq=6 ttl=64 time=0.042 ms

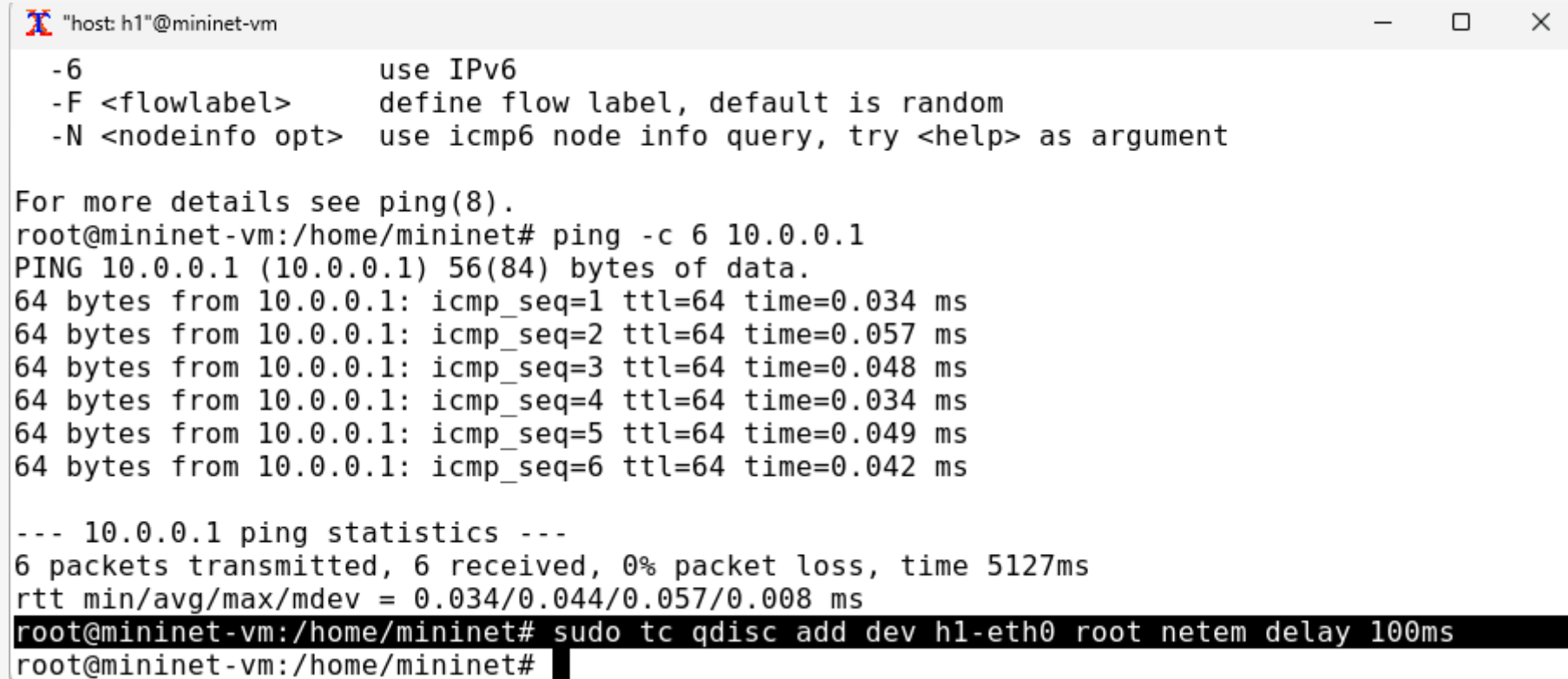
--- 10.0.0.1 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5127ms
rtt min/avg/max/mdev = 0.034/0.044/0.057/0.008 ms
root@mininet-vm:/home/mininet#

"host: h2"@mininet-vm
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=0.031 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.029 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.031 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=0.025 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5120ms
rtt min/avg/max/mdev = 0.025/0.036/0.053/0.010 ms
root@mininet-vm:/home/mininet#
```

Рис. 4: Проверка подключения между хостами h1 и h2

## Интерактивные эксперименты

A terminal window titled '"host: h1"@mininet-vm' with standard window controls. It displays the help for the 'ping' command, followed by a successful ping of 10.0.0.1. The ping statistics show 6 packets transmitted with 0% loss. The final command, 'sudo tc qdisc add dev h1-eth0 root netem delay 100ms', is highlighted in black.

```
"host: h1"@mininet-vm
-6                use IPv6
-F <flowlabel>    define flow label, default is random
-N <nodeinfo opt> use icmp6 node info query, try <help> as argument

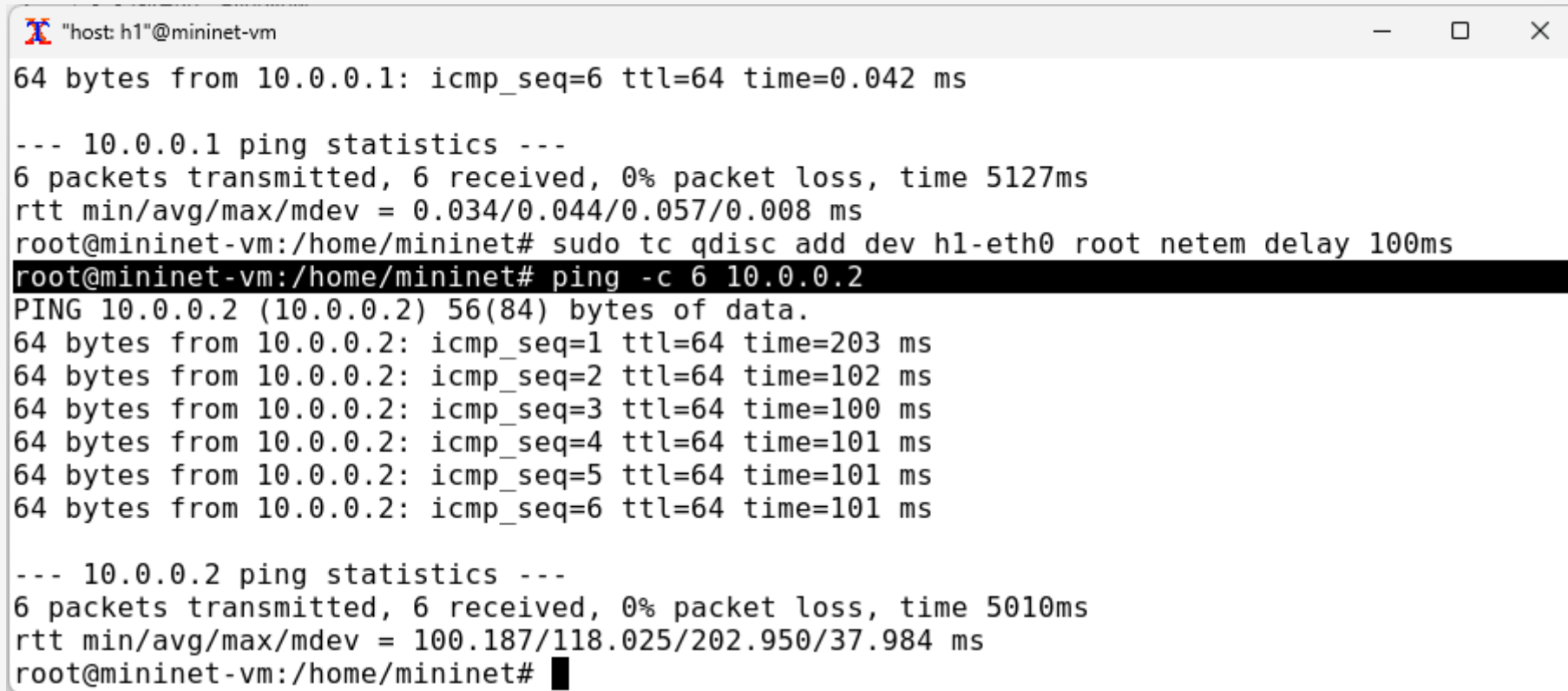
For more details see ping(8).
root@mininet-vm:/home/mininet# ping -c 6 10.0.0.1
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=0.034 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.057 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.048 ms
64 bytes from 10.0.0.1: icmp_seq=4 ttl=64 time=0.034 ms
64 bytes from 10.0.0.1: icmp_seq=5 ttl=64 time=0.049 ms
64 bytes from 10.0.0.1: icmp_seq=6 ttl=64 time=0.042 ms

--- 10.0.0.1 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5127ms
rtt min/avg/max/mdev = 0.034/0.044/0.057/0.008 ms
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 100ms
root@mininet-vm:/home/mininet#
```

Рис. 5: Добавление задержки в 100 мс к выходному интерфейсу на хосте h1

## Интерактивные эксперименты

---



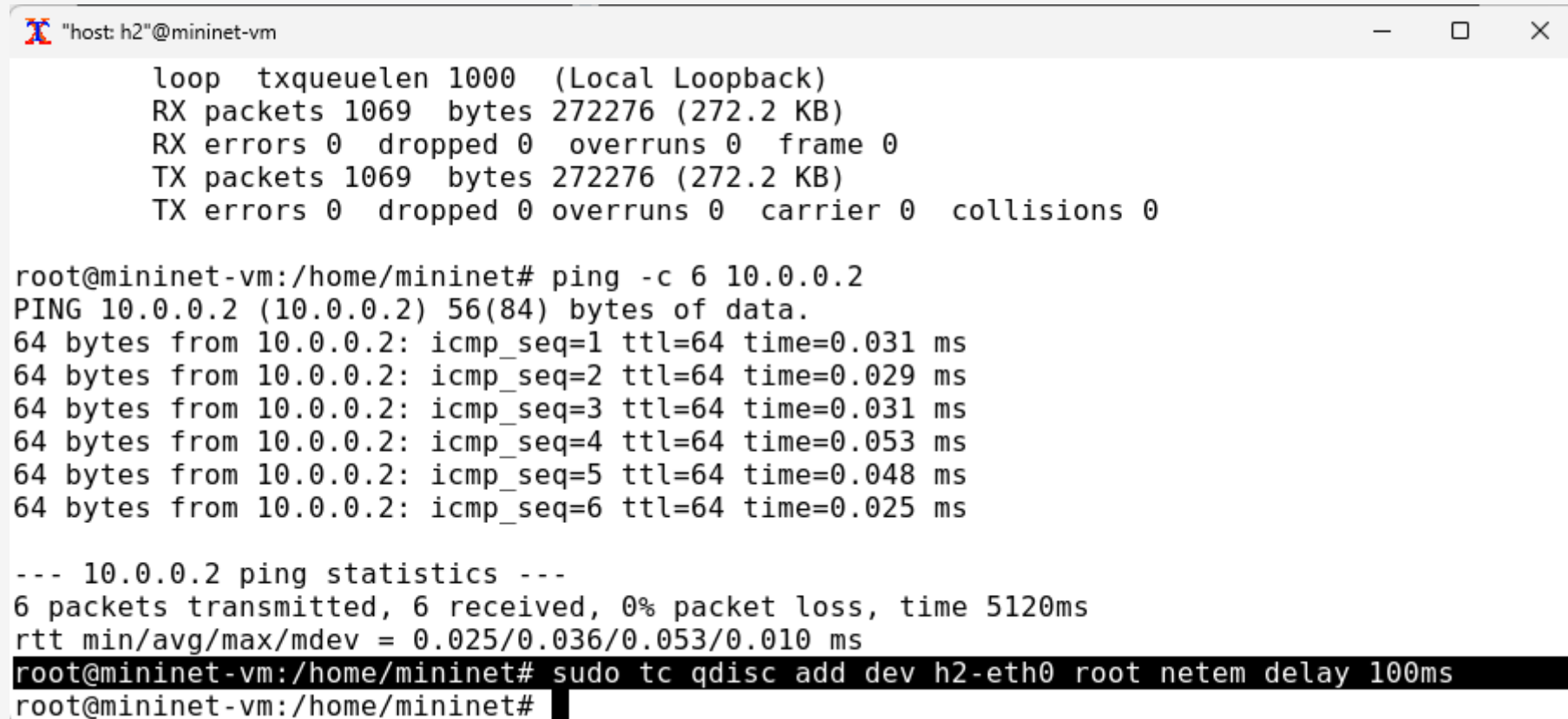
```
"host: h1"@mininet-vm
64 bytes from 10.0.0.1: icmp_seq=6 ttl=64 time=0.042 ms

--- 10.0.0.1 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5127ms
rtt min/avg/max/mdev = 0.034/0.044/0.057/0.008 ms
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 100ms
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=203 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=102 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=100 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=101 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=101 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=101 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5010ms
rtt min/avg/max/mdev = 100.187/118.025/202.950/37.984 ms
root@mininet-vm:/home/mininet#
```

Рис. 6: Проверка

## Интерактивные эксперименты



```
"host: h2"@mininet-vm
loop txqueuelen 1000 (Local Loopback)
RX packets 1069 bytes 272276 (272.2 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 1069 bytes 272276 (272.2 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

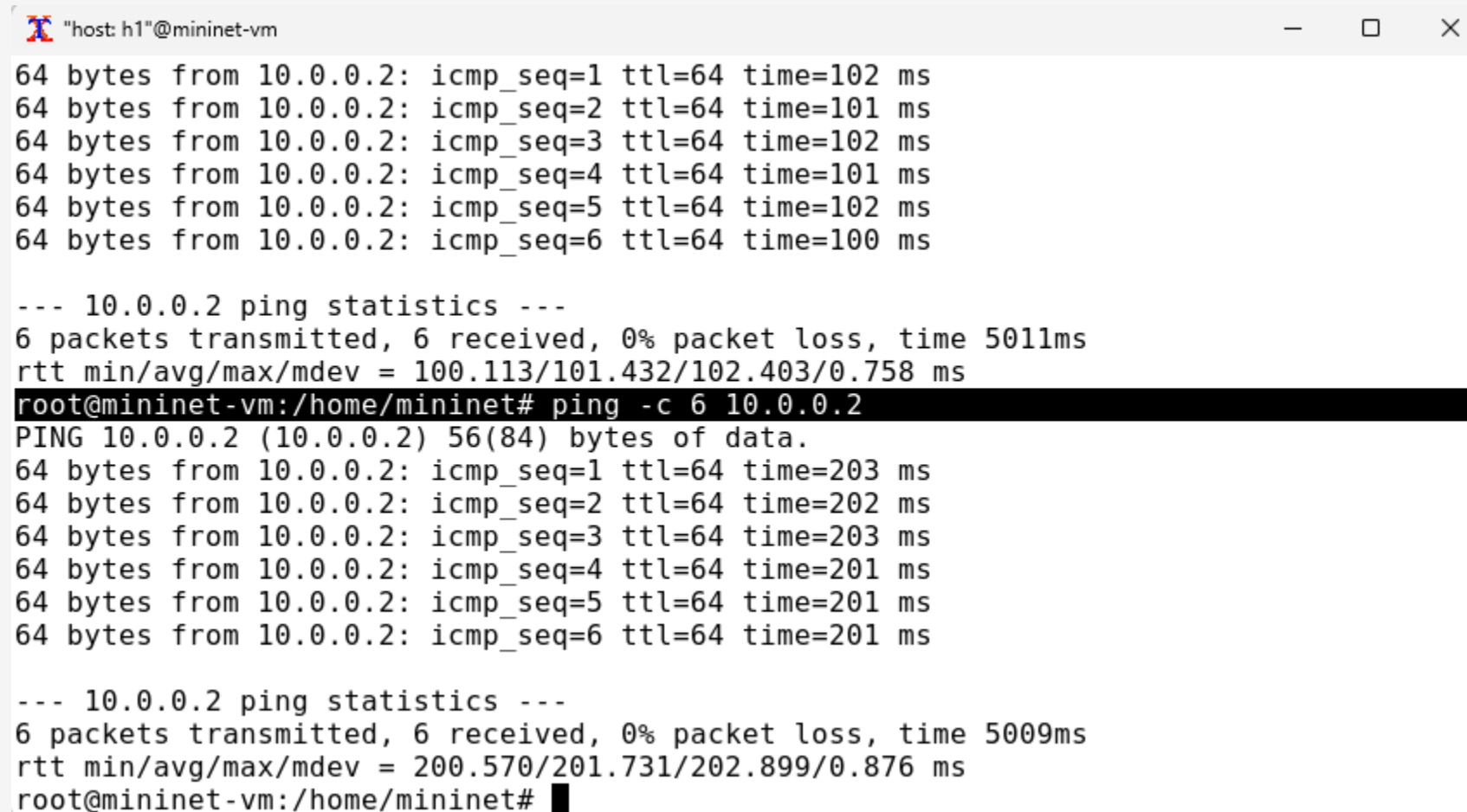
root@mininet-vm:/home/mininet# ping -c 6 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data:
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.031 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.029 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.031 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 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=0.025 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5120ms
rtt min/avg/max/mdev = 0.025/0.036/0.053/0.010 ms
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h2-eth0 root netem delay 100ms
root@mininet-vm:/home/mininet#
```

Рис. 7: Добавление задержки в 100 мс к выходному интерфейсу на хосте h2



## Интерактивные эксперименты



```
"host: h1"@mininet-vm
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=102 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=101 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=102 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=101 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=102 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=100 ms

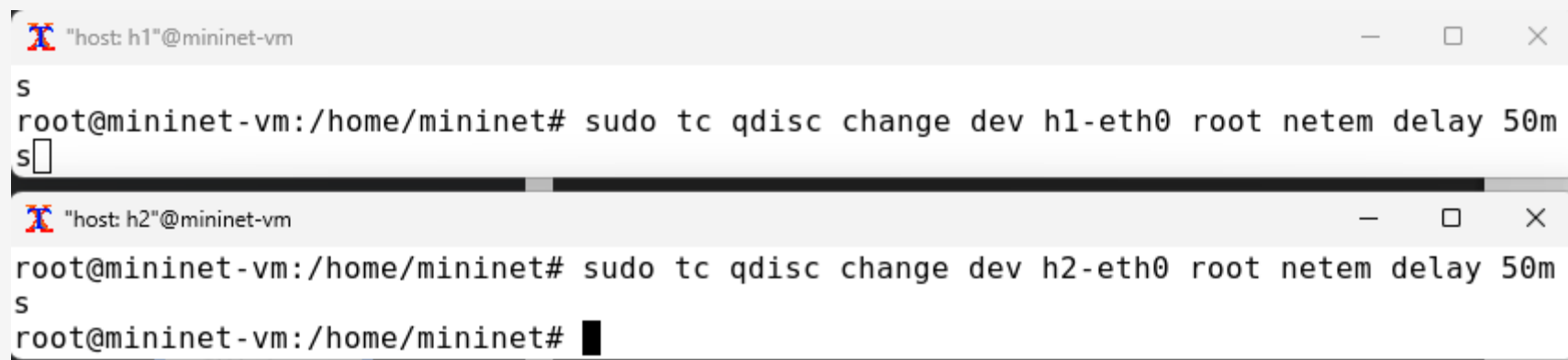
--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5011ms
rtt min/avg/max/mdev = 100.113/101.432/102.403/0.758 ms
root@mininet-vm:/home/mininet# ping -c 6 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=203 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=202 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=203 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=201 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=201 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=201 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5009ms
rtt min/avg/max/mdev = 200.570/201.731/202.899/0.876 ms
root@mininet-vm:/home/mininet#
```

Рис. 8: Проверка

## Интерактивные эксперименты

---



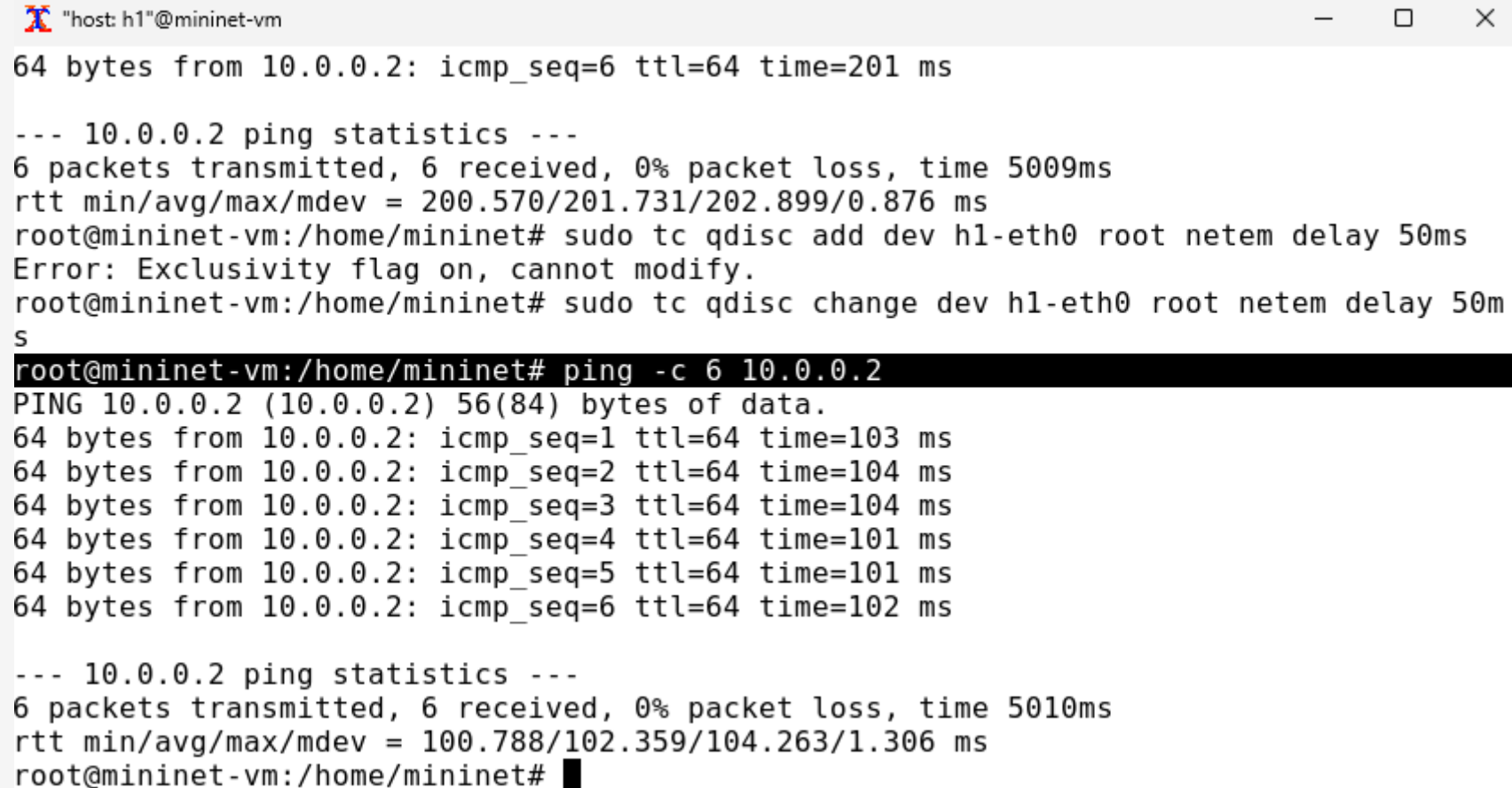
The image shows two terminal windows stacked vertically. The top window is titled '"host: h1"@mininet-vm' and contains the command `sudo tc qdisc change dev h1-eth0 root netem delay 50ms` followed by a cursor. The bottom window is titled '"host: h2"@mininet-vm' and contains the same command, followed by a cursor. Both windows have standard window control buttons (minimize, maximize, close) in the top right corner.

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

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

Рис. 9: Изменение задержки со 100 мс до 50 мс

## Интерактивные эксперименты



A terminal window titled "host: h1"@mininet-vm with standard window controls. The terminal shows a sequence of commands and their outputs. It starts with a ping to 10.0.0.2, followed by ping statistics. Then, it attempts to configure traffic control (tc) on the h1-eth0 interface, first with 'add' and then with 'change' to set a 50ms delay. Finally, it performs a second ping to 10.0.0.2 with a count of 6.

```
"host: h1"@mininet-vm
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=201 ms

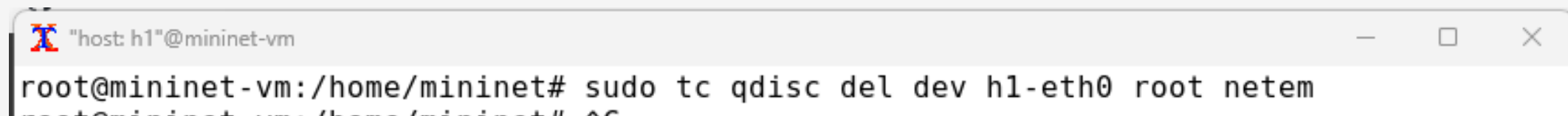
--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5009ms
rtt min/avg/max/mdev = 200.570/201.731/202.899/0.876 ms
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 50ms
Error: Exclusivity flag on, cannot modify.
root@mininet-vm:/home/mininet# sudo tc qdisc change dev h1-eth0 root netem delay 50ms
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=103 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=104 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=104 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=101 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=101 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=102 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5010ms
rtt min/avg/max/mdev = 100.788/102.359/104.263/1.306 ms
root@mininet-vm:/home/mininet#
```

Рис. 10: Проверка

## Интерактивные эксперименты

---



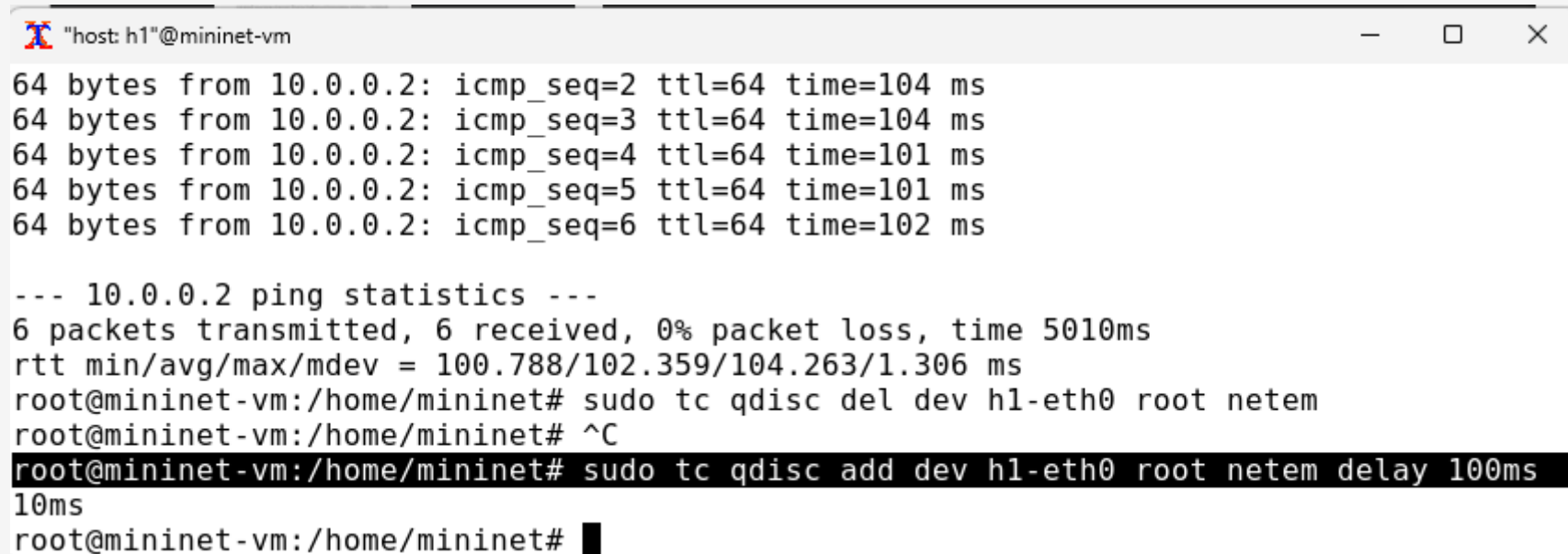
A terminal window titled "host: h1" @mininet-vm. The prompt is root@mininet-vm:/home/mininet#. The command entered is sudo tc qdisc del dev h1-eth0 root netem. The command is partially executed, with the prompt changing to root@mininet-vm:/home/mininet# ^C.

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

Рис. 11: Восстановление конфигураций по умолчанию

## Интерактивные эксперименты

---



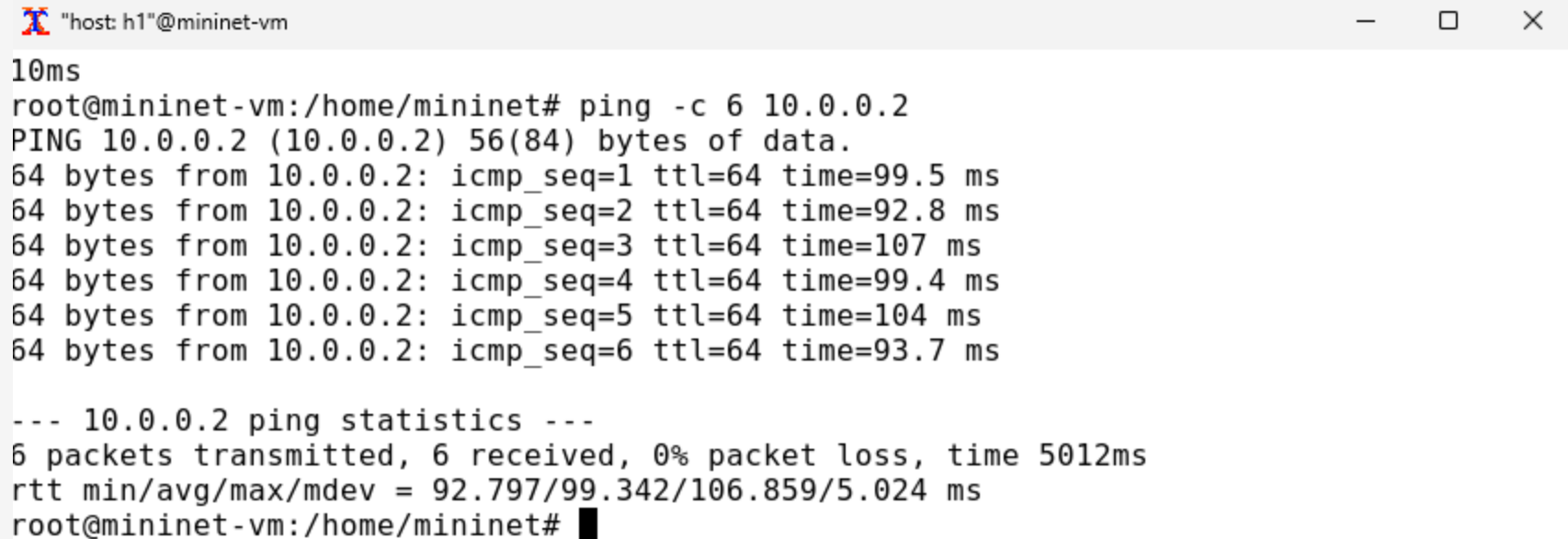
```
"host: h1"@mininet-vm
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=104 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=104 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=101 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=101 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=102 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5010ms
rtt min/avg/max/mdev = 100.788/102.359/104.263/1.306 ms
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet# ^C
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 100ms
10ms
root@mininet-vm:/home/mininet#
```

Рис. 12: Добавление на узле h1 задержки в 100 мс со случайным отклонением 10 мс

## Интерактивные эксперименты

---



A terminal window titled "host: h1"@mininet-vm with standard window controls (minimize, maximize, close). The terminal shows a ping command being executed from the root user at the mininet-vm host. The output displays the results of a ping to 10.0.0.2, including individual packet statistics and a summary of the ping statistics.

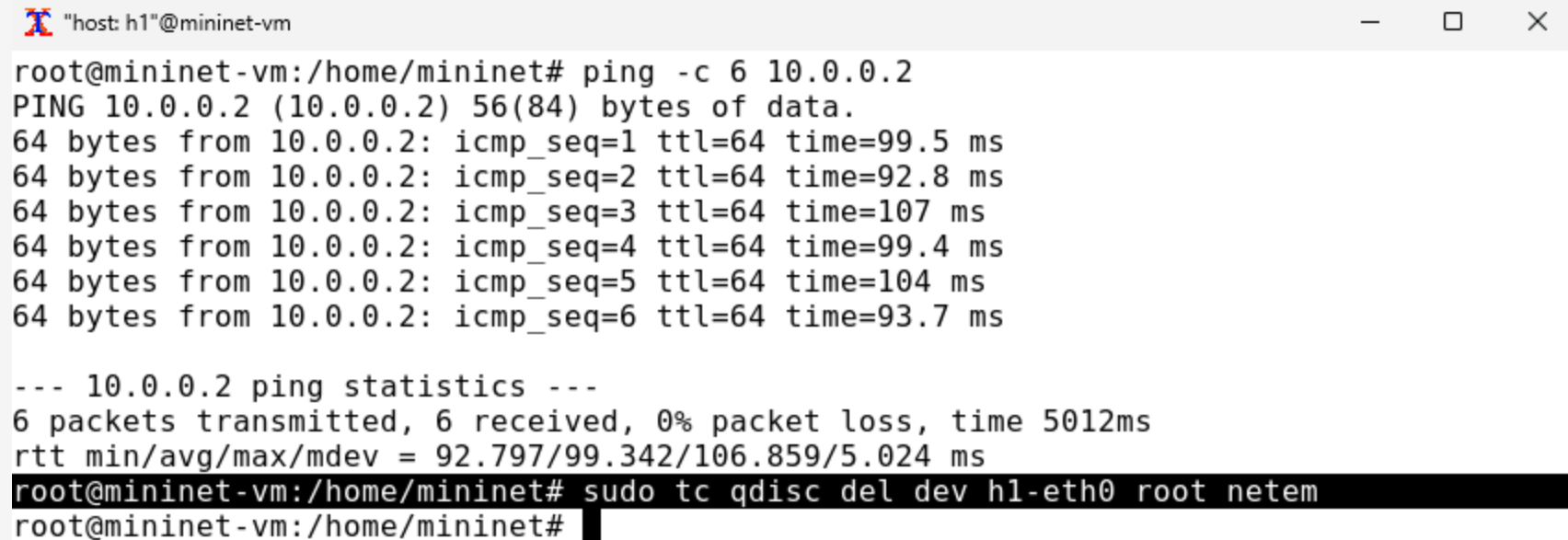
```
"host: h1"@mininet-vm
10ms
root@mininet-vm:/home/mininet# ping -c 6 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=99.5 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=92.8 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=107 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=99.4 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=104 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=93.7 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5012ms
rtt min/avg/max/mdev = 92.797/99.342/106.859/5.024 ms
root@mininet-vm:/home/mininet#
```

Рис. 13: Проверка

## Интерактивные эксперименты

---

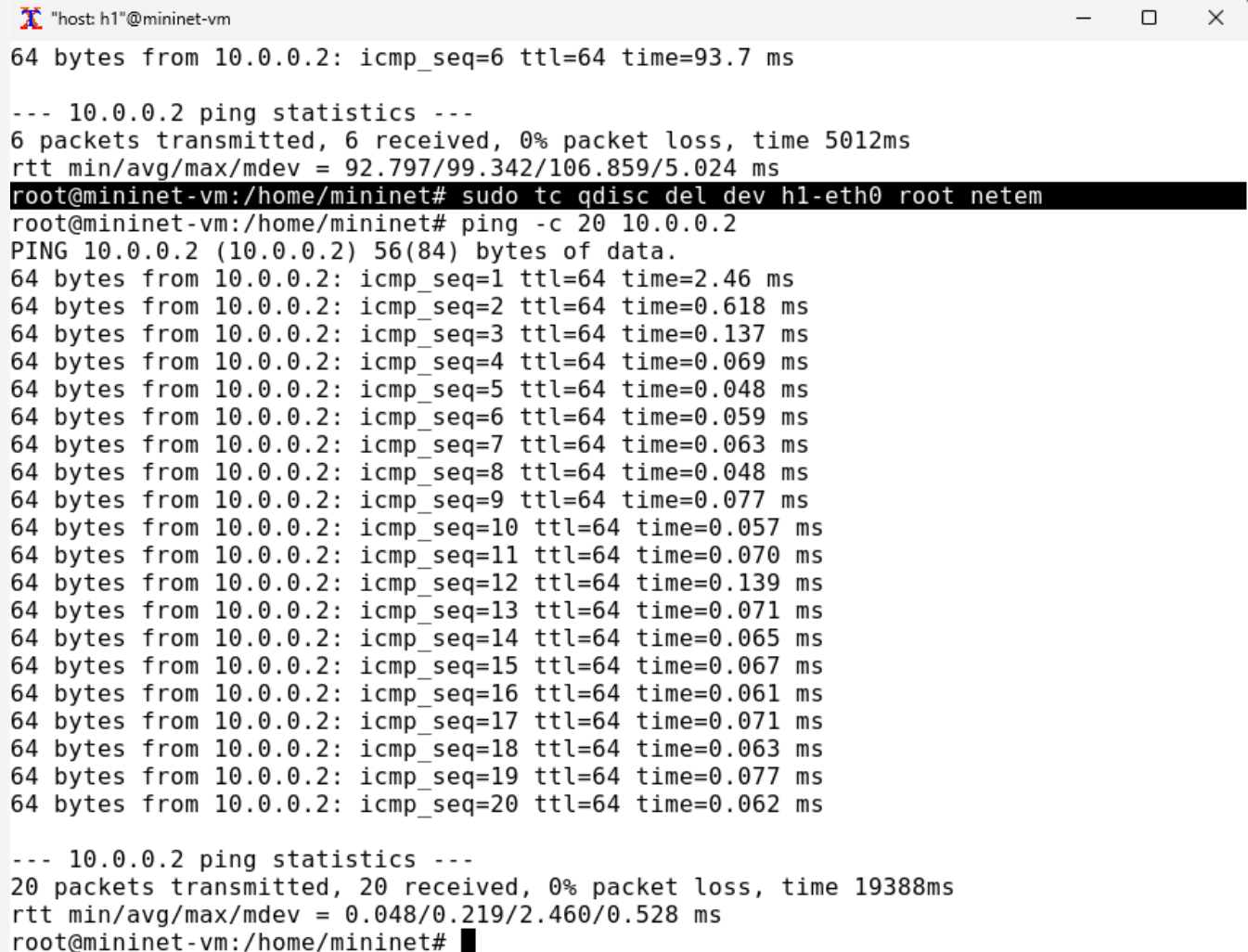


```
"host: h1"@mininet-vm
root@mininet-vm:/home/mininet# ping -c 6 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=99.5 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=92.8 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=107 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=99.4 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=104 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=93.7 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5012ms
rtt min/avg/max/mdev = 92.797/99.342/106.859/5.024 ms
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet#
```

Рис. 14: Восстановление конфигурации интерфейса по умолчанию

## Интерактивные эксперименты

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, followed by statistics, and then a tc command to delete the h1-eth0 root netem queue. This is followed by another ping command to 10.0.0.2 with a count of 20, its statistics, and the prompt root@mininet-vm:/home/mininet#. The terminal text is as follows:

```
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=93.7 ms

--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5012ms
rtt min/avg/max/mdev = 92.797/99.342/106.859/5.024 ms
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
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=2.46 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.618 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.137 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=0.048 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=0.063 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=0.077 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.057 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=0.070 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.139 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.071 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=0.065 ms
64 bytes from 10.0.0.2: icmp_seq=15 ttl=64 time=0.067 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=0.061 ms
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=0.071 ms
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=0.063 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=0.077 ms
64 bytes from 10.0.0.2: icmp_seq=20 ttl=64 time=0.062 ms

--- 10.0.0.2 ping statistics ---
20 packets transmitted, 20 received, 0% packet loss, time 19388ms
rtt min/avg/max/mdev = 0.048/0.219/2.460/0.528 ms
root@mininet-vm:/home/mininet#
```

Рис. 15: Проверка



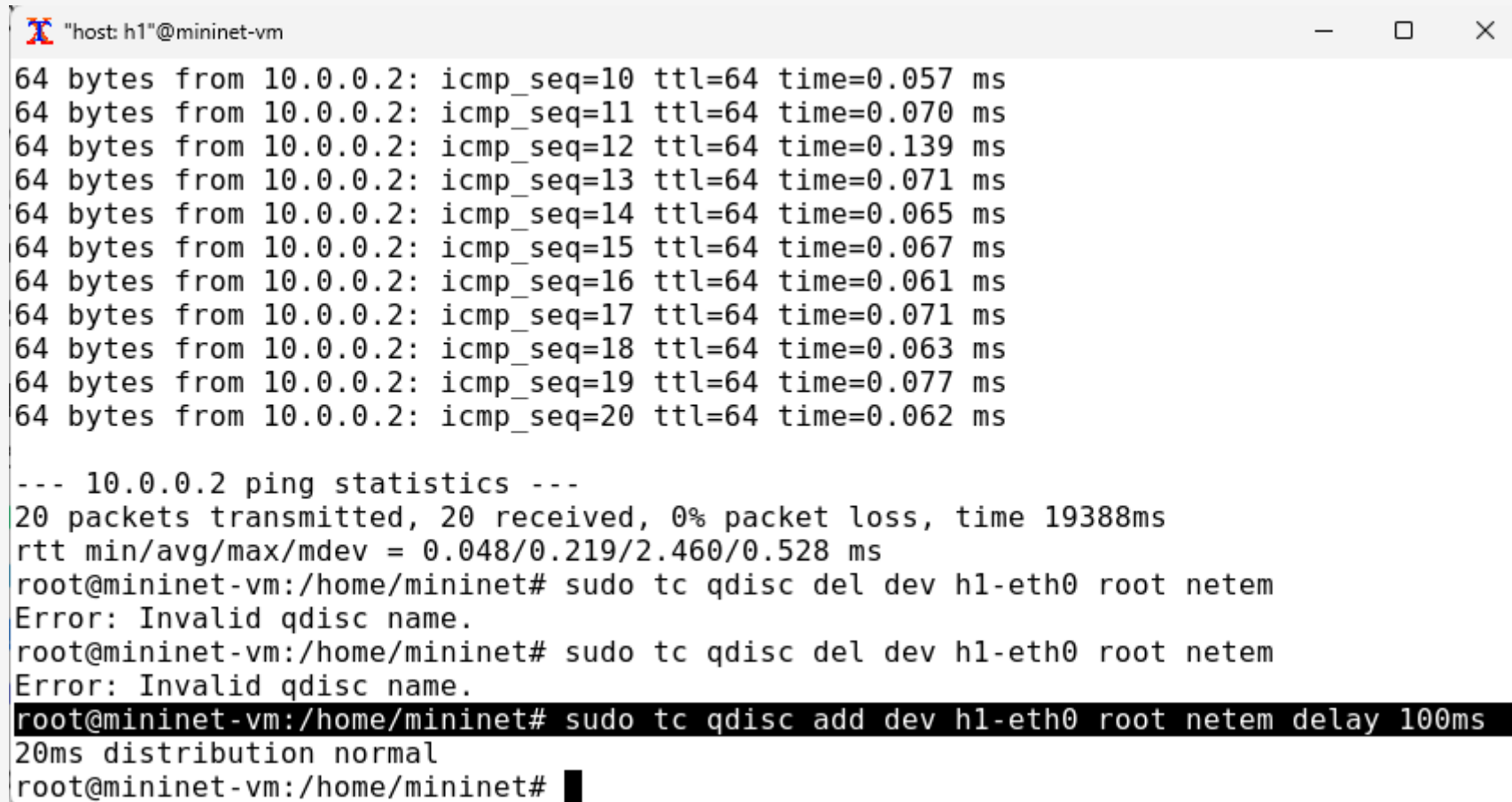
## Интерактивные эксперименты

---

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

Рис. 16: Восстановление конфигурации интерфейса по умолчанию

## Интерактивные эксперименты

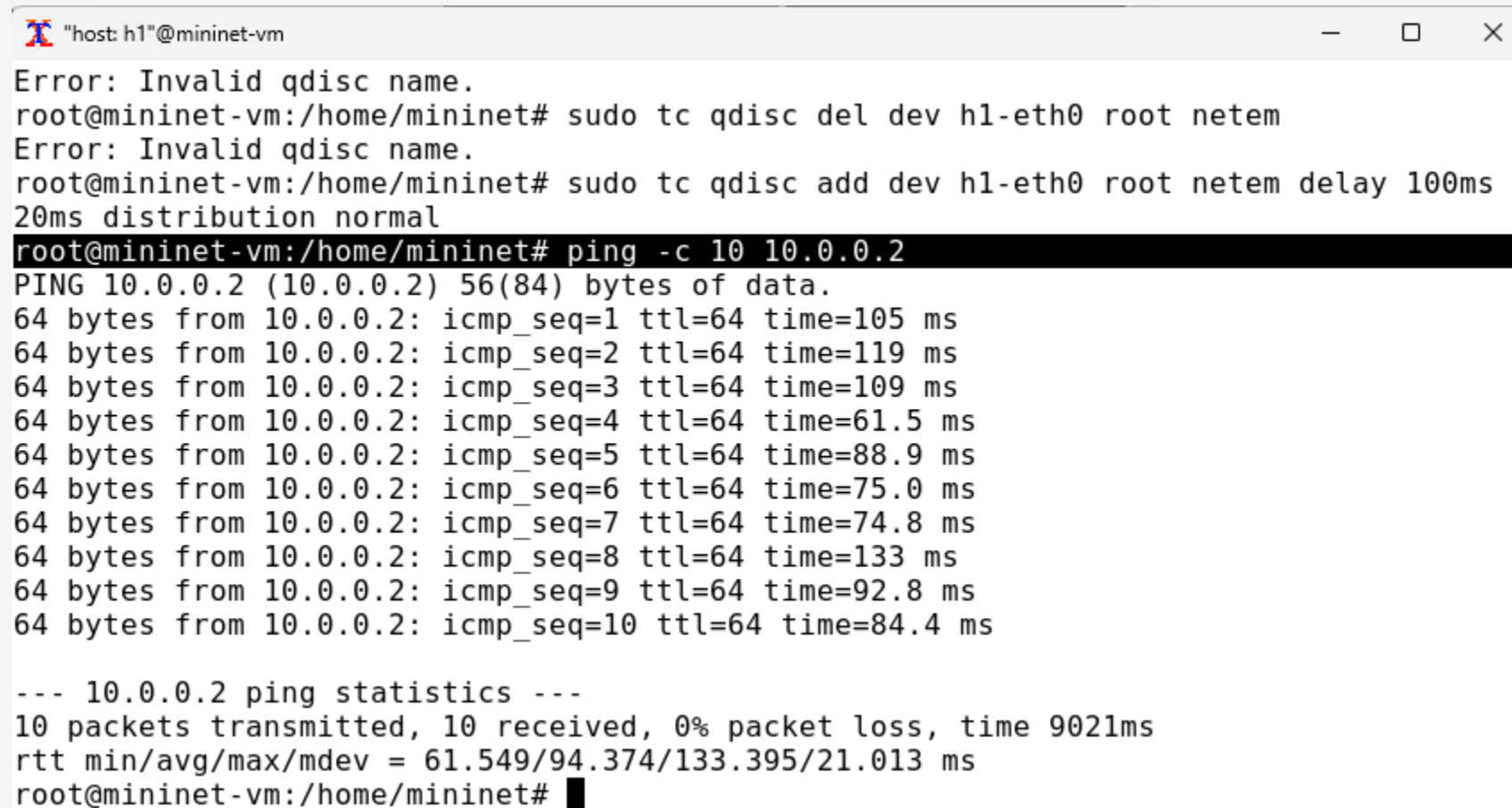


```
"host: h1"@mininet-vm
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.057 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=0.070 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.139 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.071 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=0.065 ms
64 bytes from 10.0.0.2: icmp_seq=15 ttl=64 time=0.067 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=0.061 ms
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=0.071 ms
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=0.063 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=0.077 ms
64 bytes from 10.0.0.2: icmp_seq=20 ttl=64 time=0.062 ms

--- 10.0.0.2 ping statistics ---
20 packets transmitted, 20 received, 0% packet loss, time 19388ms
rtt min/avg/max/mdev = 0.048/0.219/2.460/0.528 ms
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
Error: Invalid qdisc name.
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
Error: Invalid qdisc name.
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 100ms
20ms distribution normal
root@mininet-vm:/home/mininet#
```

Рис. 17: Настройка нормального распределения задержки на узле h1 в эмулируемой сети

## Интерактивные эксперименты



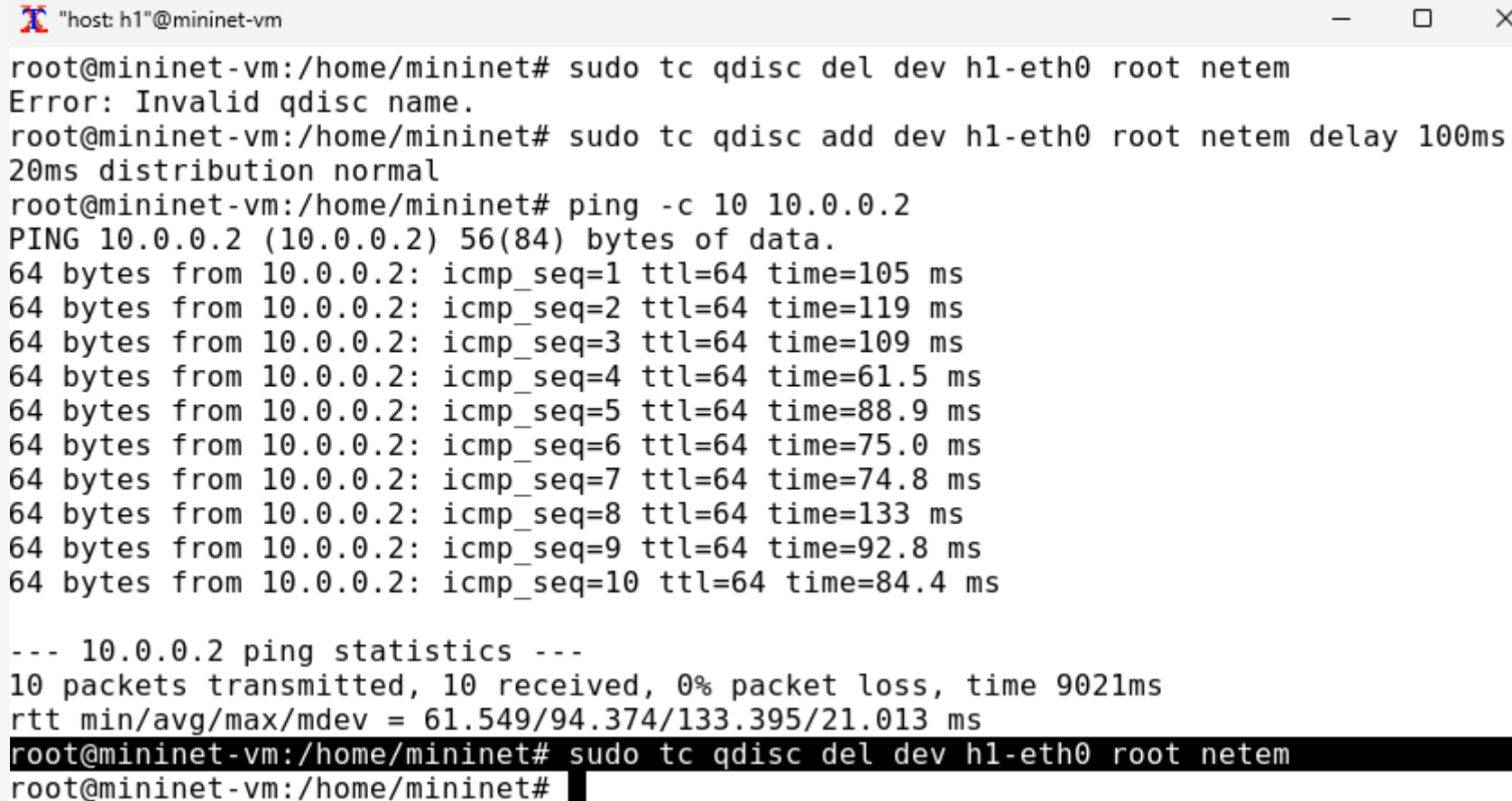
A terminal window titled "host: h1" @mininet-vm. The window shows the following commands and output:

```
Error: Invalid qdisc name.
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
Error: Invalid qdisc name.
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 100ms
20ms distribution normal
root@mininet-vm:/home/mininet# ping -c 10 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data:
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=105 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=119 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=109 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=61.5 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=88.9 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=75.0 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=74.8 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=133 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=92.8 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=84.4 ms

--- 10.0.0.2 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9021ms
rtt min/avg/max/mdev = 61.549/94.374/133.395/21.013 ms
root@mininet-vm:/home/mininet#
```

Рис. 18: Проверка

## Интерактивные эксперименты



A terminal window titled "host: h1"@mininet-vm with standard window controls. It displays a series of commands and their outputs to restore network configuration. The commands include deleting and adding a qdisc to h1-eth0, performing a ping test to 10.0.0.2, and finally deleting the qdisc again. The ping output shows 10 successful packets with various round-trip times.

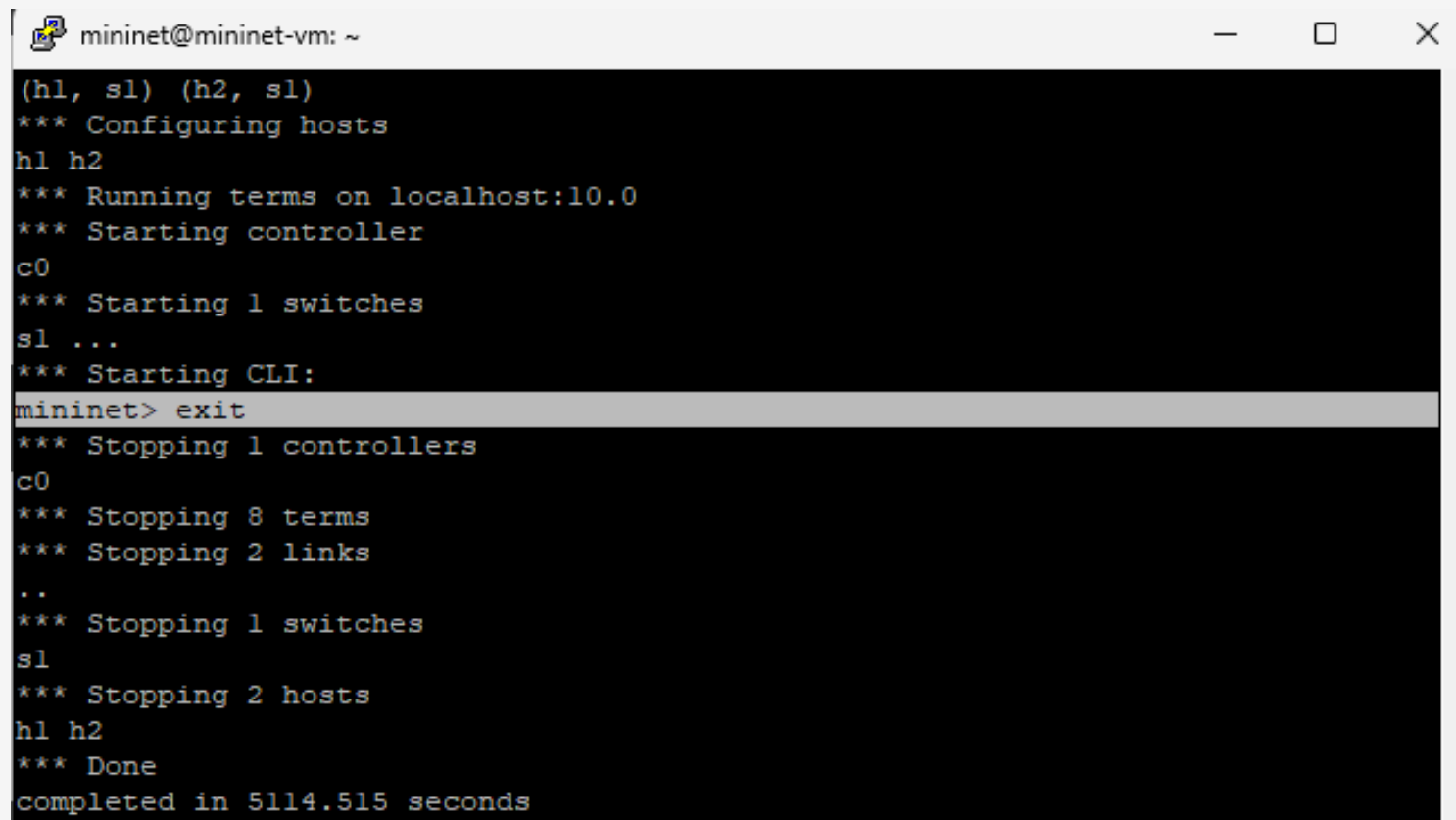
```
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
Error: Invalid qdisc name.
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root netem delay 100ms
20ms distribution normal
root@mininet-vm:/home/mininet# ping -c 10 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=105 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=119 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=109 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=61.5 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=88.9 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=75.0 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=74.8 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=133 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=92.8 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=84.4 ms

--- 10.0.0.2 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9021ms
rtt min/avg/max/mdev = 61.549/94.374/133.395/21.013 ms
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root netem
root@mininet-vm:/home/mininet#
```

Рис. 19: Восстановление конфигурации интерфейса по умолчанию

## Интерактивные эксперименты

---



```
mininet@mininet-vm: ~  
(h1, s1) (h2, s1)  
*** Configuring hosts  
h1 h2  
*** Running terms on localhost:10.0  
*** Starting controller  
c0  
*** Starting 1 switches  
s1 ...  
*** Starting CLI:  
mininet> exit  
*** Stopping 1 controllers  
c0  
*** Stopping 8 terms  
*** Stopping 2 links  
..  
*** Stopping 1 switches  
s1  
*** Stopping 2 hosts  
h1 h2  
*** Done  
completed in 5114.515 seconds
```

Рис. 20: Завершение работы mininet в интерактивном режиме

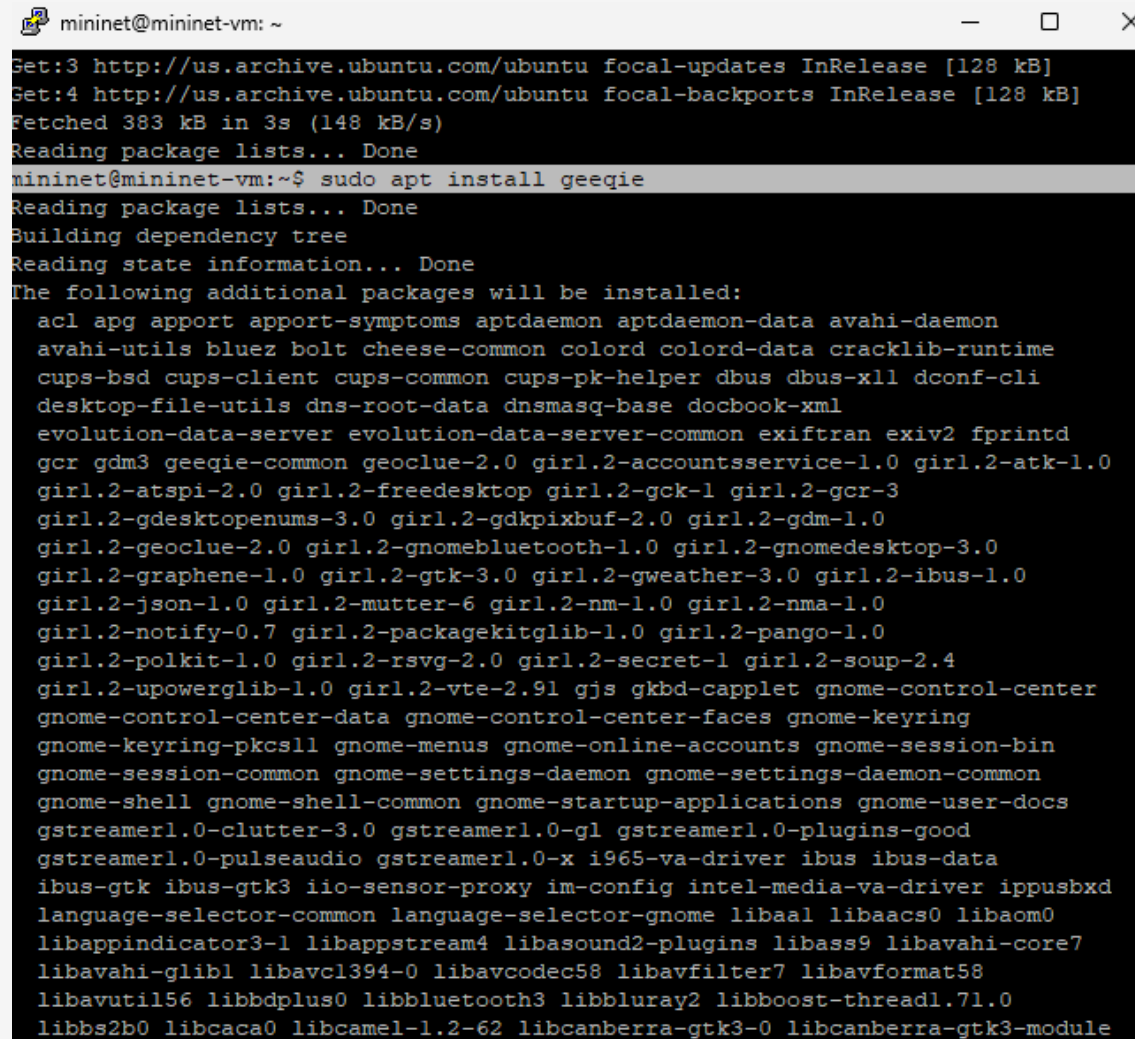
## Воспроизведение экспериментов

---

```
mininet@mininet-vm:~$ sudo apt-get update
Hit:1 http://us.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://security.ubuntu.com/ubuntu focal-security InRelease [128 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease [128 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease [128 kB]
Fetched 383 kB in 3s (148 kB/s)
Reading package lists... Done
mininet@mininet-vm:~$
```

Рис. 21: Обновление репозитория программного обеспечения на виртуальной машине

## Воспроизведение экспериментов



```
mininet@mininet-vm: ~  
Get:3 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease [128 kB]  
Get:4 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease [128 kB]  
Fetched 383 kB in 3s (148 kB/s)  
Reading package lists... Done  
mininet@mininet-vm:~$ sudo apt install geeqie  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following additional packages will be installed:  
  acl apg appport appport-symptoms aptdaemon aptdaemon-data avahi-daemon  
  avahi-utils bluez bolt cheese-common colord colord-data cracklib-runtime  
  cups-bsd cups-client cups-common cups-pk-helper dbus dbus-x11 dconf-cli  
  desktop-file-utils dns-root-data dnsmasq-base docbook-xml  
  evolution-data-server evolution-data-server-common exiftran exiv2 fprintd  
  gcr gdm3 geeqie-common geoclue-2.0 gir1.2-accountsservice-1.0 gir1.2-atk-1.0  
  gir1.2-atspi-2.0 gir1.2-freedesktop gir1.2-gck-1 gir1.2-gcr-3  
  gir1.2-gdesktopenums-3.0 gir1.2-gdkpixbuf-2.0 gir1.2-gdm-1.0  
  gir1.2-geoclue-2.0 gir1.2-gnomebluetooth-1.0 gir1.2-gnomedesktop-3.0  
  gir1.2-graphene-1.0 gir1.2-gtk-3.0 gir1.2-gweather-3.0 gir1.2-ibus-1.0  
  gir1.2-json-1.0 gir1.2-mutter-6 gir1.2-nm-1.0 gir1.2-nma-1.0  
  gir1.2-notify-0.7 gir1.2-packagekitglib-1.0 gir1.2-pango-1.0  
  gir1.2-polkit-1.0 gir1.2-rsvg-2.0 gir1.2-secret-1 gir1.2-soup-2.4  
  gir1.2-upowerglib-1.0 gir1.2-vte-2.91 gjs gkbd-capplet gnome-control-center  
  gnome-control-center-data gnome-control-center-faces gnome-keyring  
  gnome-keyring-pkcs11 gnome-menus gnome-online-accounts gnome-session-bin  
  gnome-session-common gnome-settings-daemon gnome-settings-daemon-common  
  gnome-shell gnome-shell-common gnome-startup-applications gnome-user-docs  
  gstreamer1.0-clutter-3.0 gstreamer1.0-gi gstreamer1.0-plugins-good  
  gstreamer1.0-pulseaudio gstreamer1.0-x i965-va-driver ibus ibus-data  
  ibus-gtk ibus-gtk3 iio-sensor-proxy im-config intel-media-va-driver ipusbxd  
  language-selector-common language-selector-gnome libaal libaacs0 libaom0  
  libappindicator3-1 libappstream4 libasound2-plugins libass9 libavahi-core7  
  libavahi-glib1 libavc1394-0 libavcodec58 libavfilter7 libavformat58  
  libavutil56 libbdplus0 libbluetooth3 libbluray2 libboost-thread1.71.0  
  libbs2b0 libcac0 libcamel-1.2-62 libcanberra-gtk3-0 libcanberra-gtk3-module
```

Рис. 22: Установка пакета geeqie

## Воспроизведение экспериментов

---

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

Рис. 23: Создание нового каталога



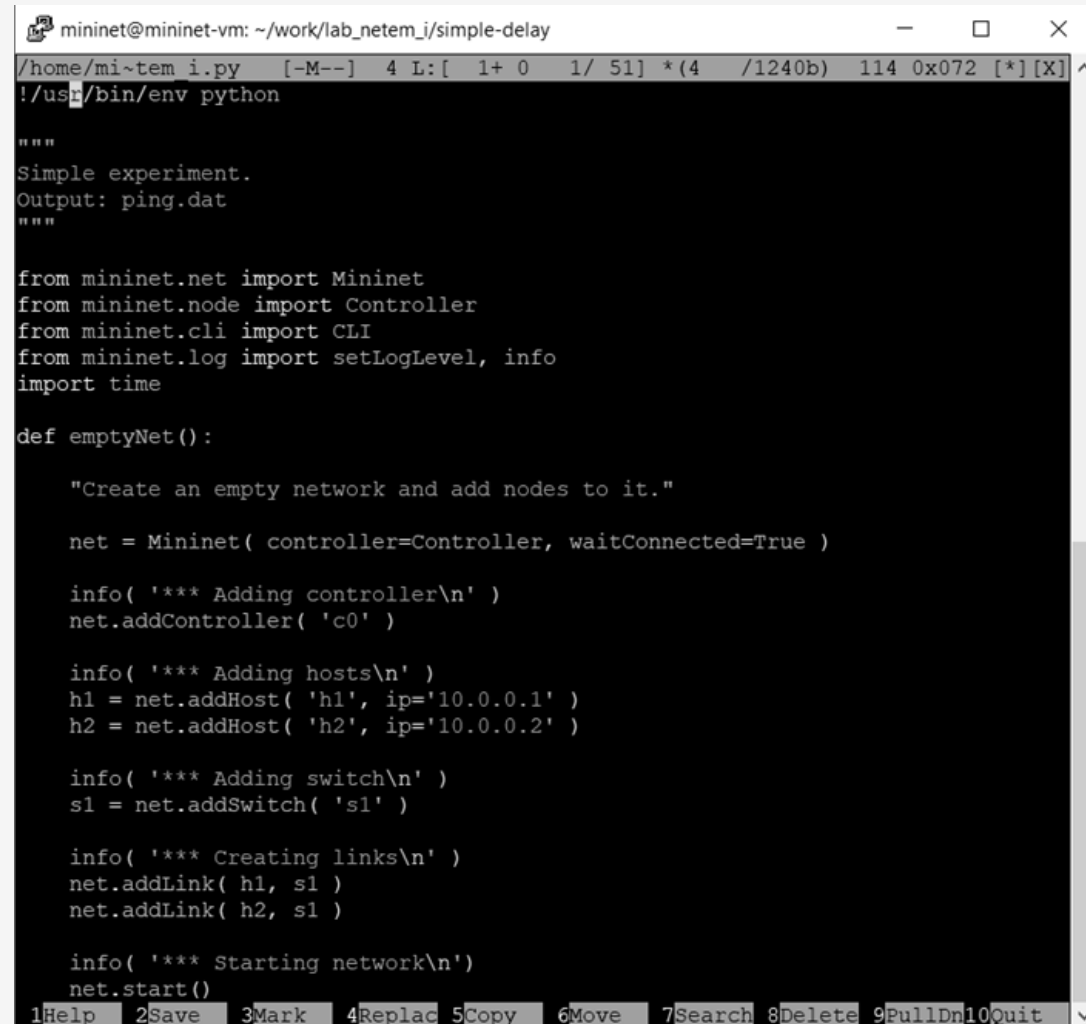
## Воспроизведение экспериментов

---

```
mininet@mininet-vm:~$ mkdir -p ~/work/lab_netem_i/simple-delay
mininet@mininet-vm:~$ cd ~/work/lab_netem_i/simple-delay
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ ls
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ touch lab_netem_i.py
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ ls
lab_netem_i.py
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ nano lab_netem_i.py
```

Рис. 24: Создание каталога simple-delay

## Воспроизведение экспериментов



The image shows a terminal window with a dark background. The title bar reads "mininet@mininet-vm: ~/work/lab\_netem\_i/simple-delay". The terminal content shows the execution of a shell script to create a Python file named "i.py". The script includes a docstring for a "Simple experiment" and Python code to set up a Mininet network with two hosts and a switch.

```
mininet@mininet-vm: ~/work/lab_netem_i/simple-delay
/home/mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ cat >i.py
#!/usr/bin/env python

"""
Simple experiment.
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()

1Help 2Save 3Mark 4Replac 5Copy 6Move 7Search 8Delete 9PullDn10Quit
```

Рис. 25: Создание скрипта lab\_netem\_i.py для эксперимента

## Воспроизведение экспериментов

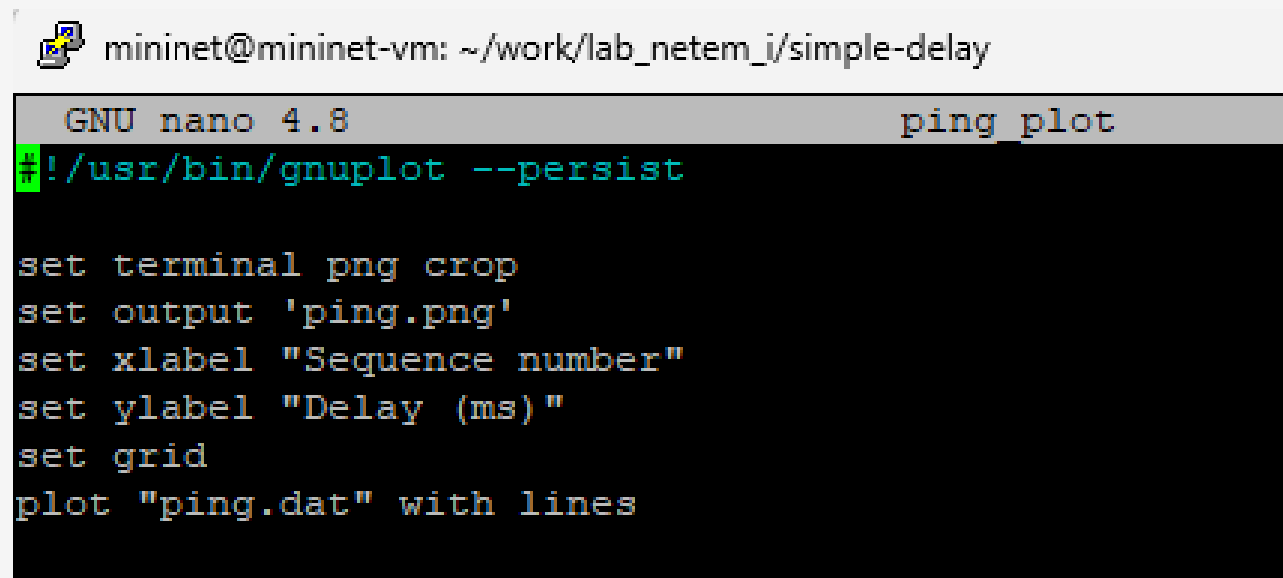
---

```
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ touch ping_plot  
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ nano ping_plot  
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
```

Рис. 26: Создание файла ping\_plot

## Воспроизведение экспериментов

---



```
mininet@mininet-vm: ~/work/lab_netem_i/simple-delay
GNU nano 4.8 ping_plot
#!/usr/bin/gnuplot --persist

set terminal png crop
set output 'ping.png'
set xlabel "Sequence number"
set ylabel "Delay (ms)"
set grid
plot "ping.dat" with lines
```

Рис. 27: Создание скрипта ping\_plot для визуализации результатов эксперимента

## Воспроизведение экспериментов

---

```
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ chmod +x ping_plot  
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
```

Рис. 28: Настройка прав доступа к файлу скрипта

## Воспроизведение экспериментов

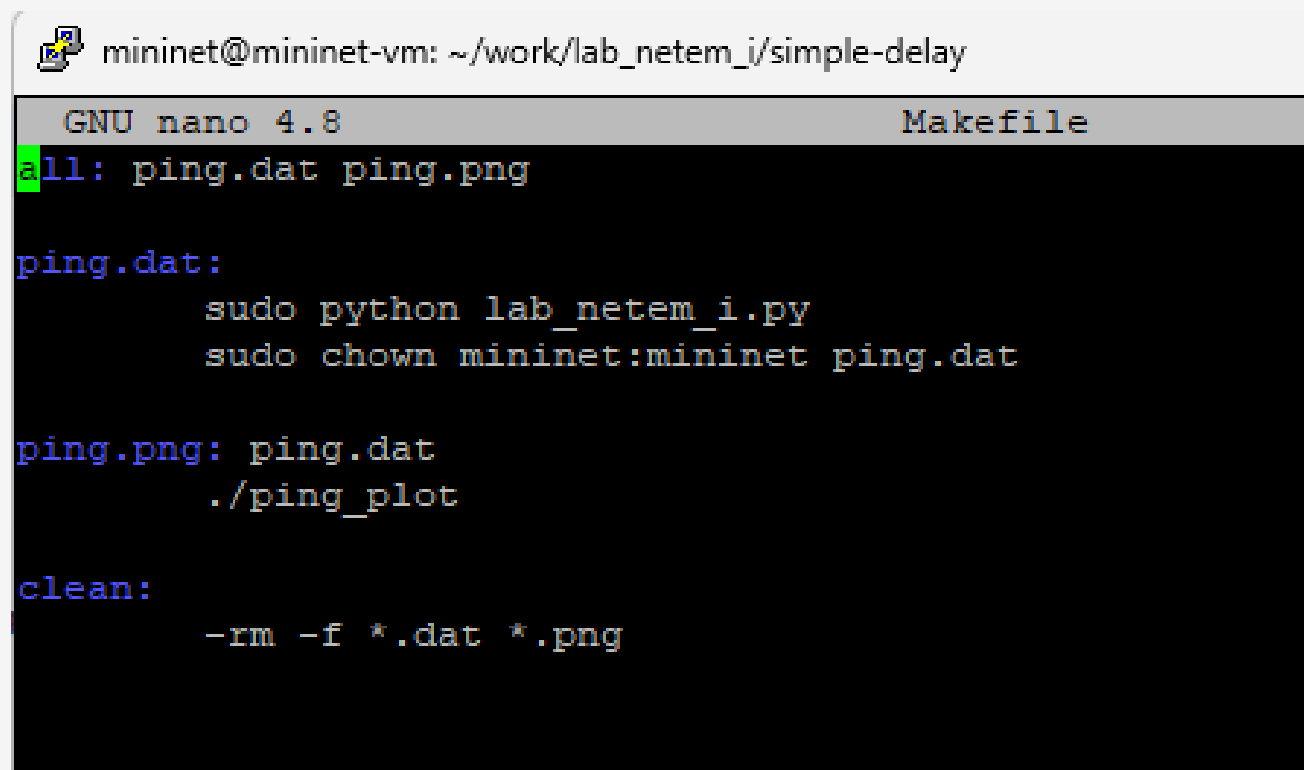
---

```
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ touch Makefile
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ nano m
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ ls
lab_netem_i.py  Makefile  ping_plot
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ nano Makefile
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ nano Makefile
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
```

Рис. 29: Создание файла Makefile

## Воспроизведение экспериментов

---



The screenshot shows a terminal window with a terminal icon in the title bar. The title bar text is "mininet@mininet-vm: ~/work/lab\_netem\_i/simple-delay". The terminal content shows the GNU nano 4.8 editor editing a file named "Makefile". The Makefile content is as follows:

```
GNU nano 4.8 Makefile
all: ping.dat ping.png

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

ping.png: ping.dat
    ./ping_plot

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

Рис. 30: Добавления скрипта в Makefile для управления процессом проведения эксперимента

## Воспроизведение экспериментов

```
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ make
sudo python lab_netem_i.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
*** Starting network
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Waiting for switches to connect
s1
*** Set delay
*** h1 : ('tc qdisc add dev h1-eth0 root netem delay 100ms',)
*** h2 : ('tc qdisc add dev h2-eth0 root netem delay 100ms',)
*** 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
./ping_plot
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
```

Рис. 31: Выполнение эксперимента



## Воспроизведение экспериментов

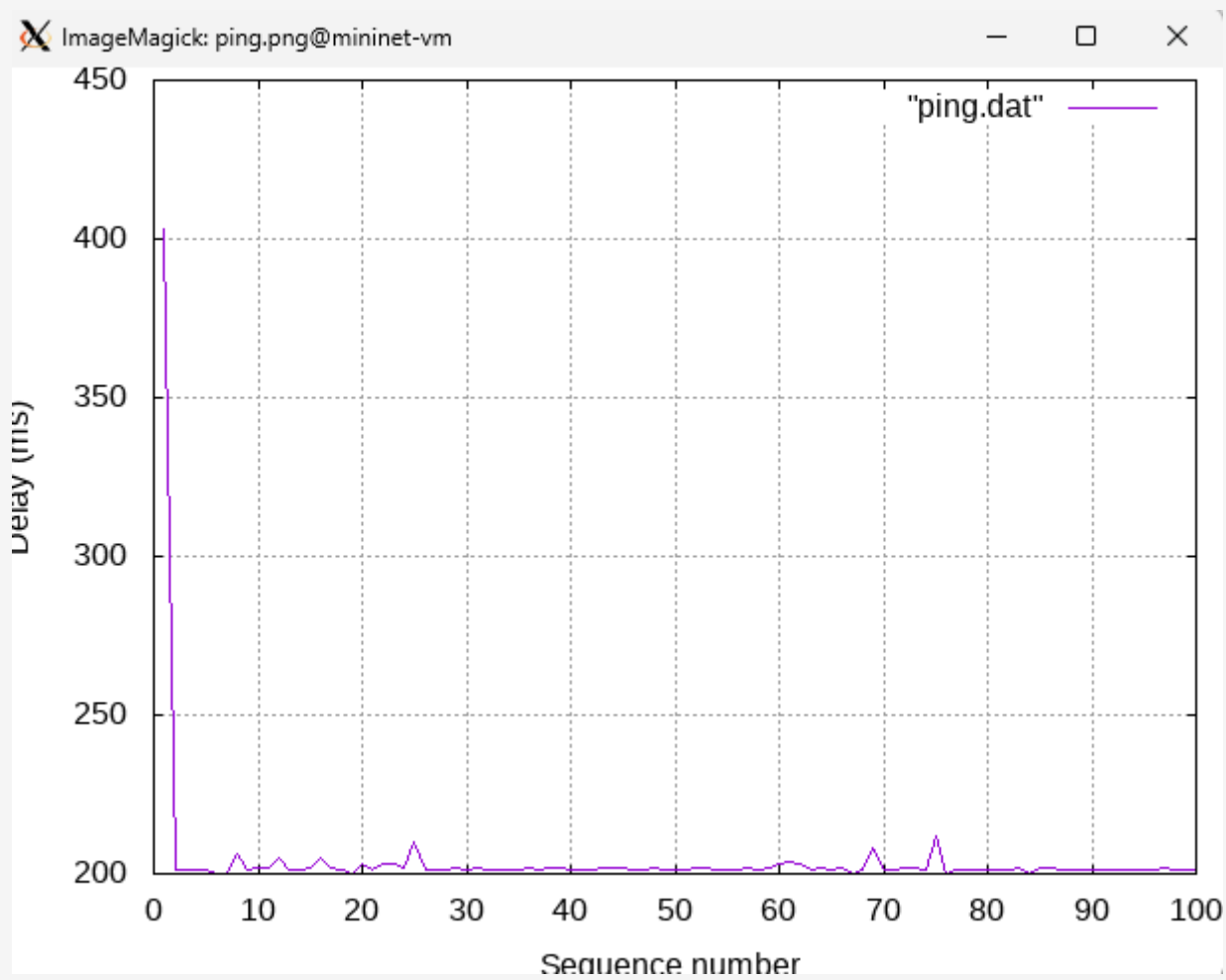
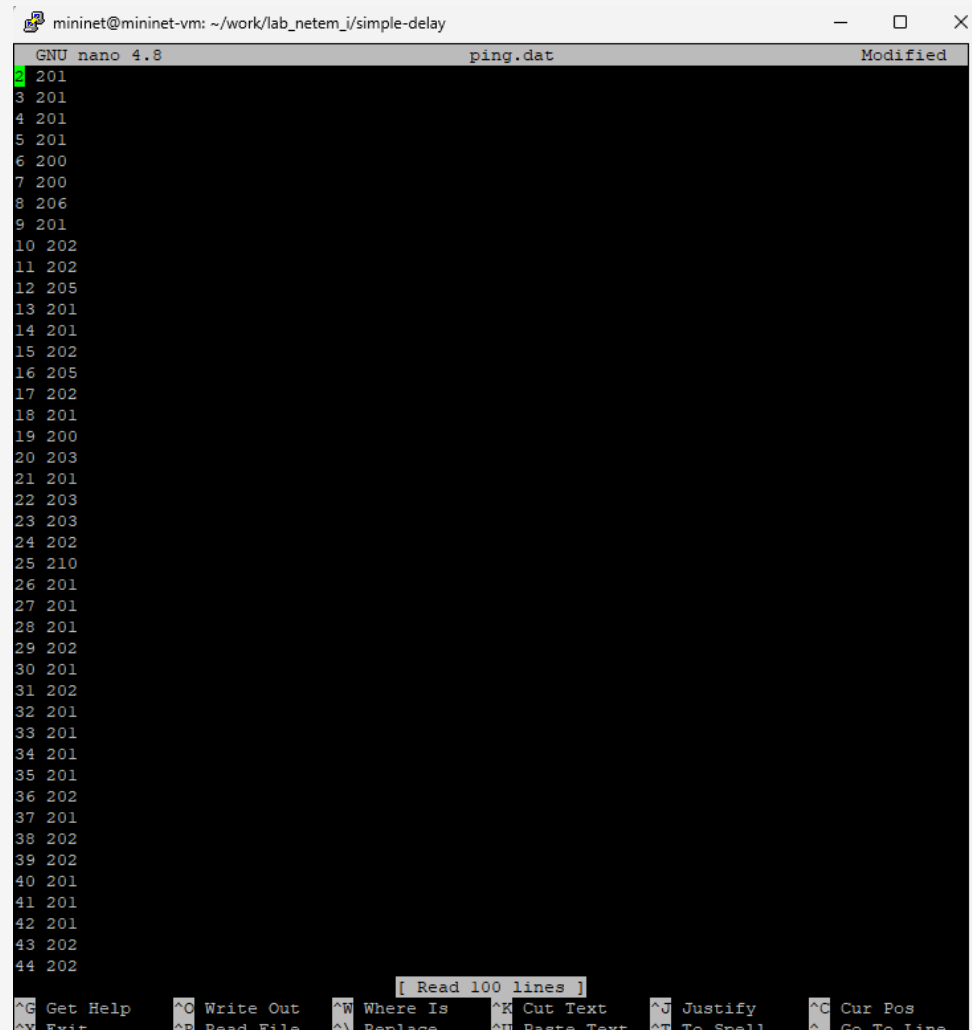


Рис. 32: Просмотр графика

## Воспроизведение экспериментов



```
mininet@mininet-vm: ~/work/lab_netem_i/simple-delay
GNU nano 4.8 ping.dat Modified
2 201
3 201
4 201
5 201
6 200
7 200
8 206
9 201
10 202
11 202
12 205
13 201
14 201
15 202
16 205
17 202
18 201
19 200
20 203
21 201
22 203
23 203
24 202
25 210
26 201
27 201
28 201
29 202
30 201
31 202
32 201
33 201
34 201
35 201
36 202
37 201
38 202
39 202
40 201
41 201
42 201
43 202
44 202
[ Read 100 lines ]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Paste Text ^T To Spell ^_ Go To Line
```

Рис. 33: Удаление первой строчки из файла ping.dat

## Воспроизведение экспериментов

---

```
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ make ping.png
./ping_plot
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ make ping.png
make: 'ping.png' is up to date.
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ display ping.png
█
```

Рис. 34: Повторное построение графика

## Воспроизведение экспериментов

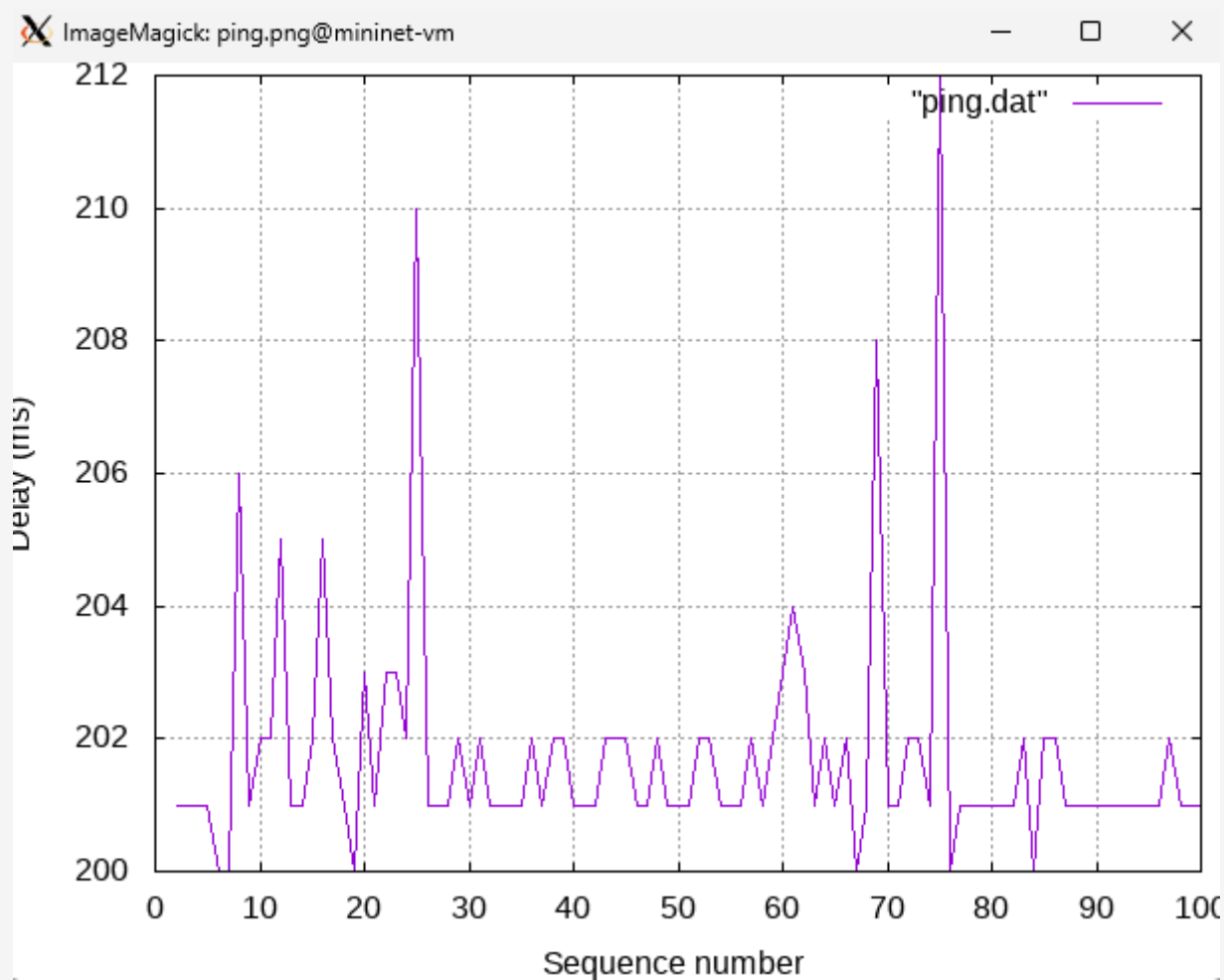
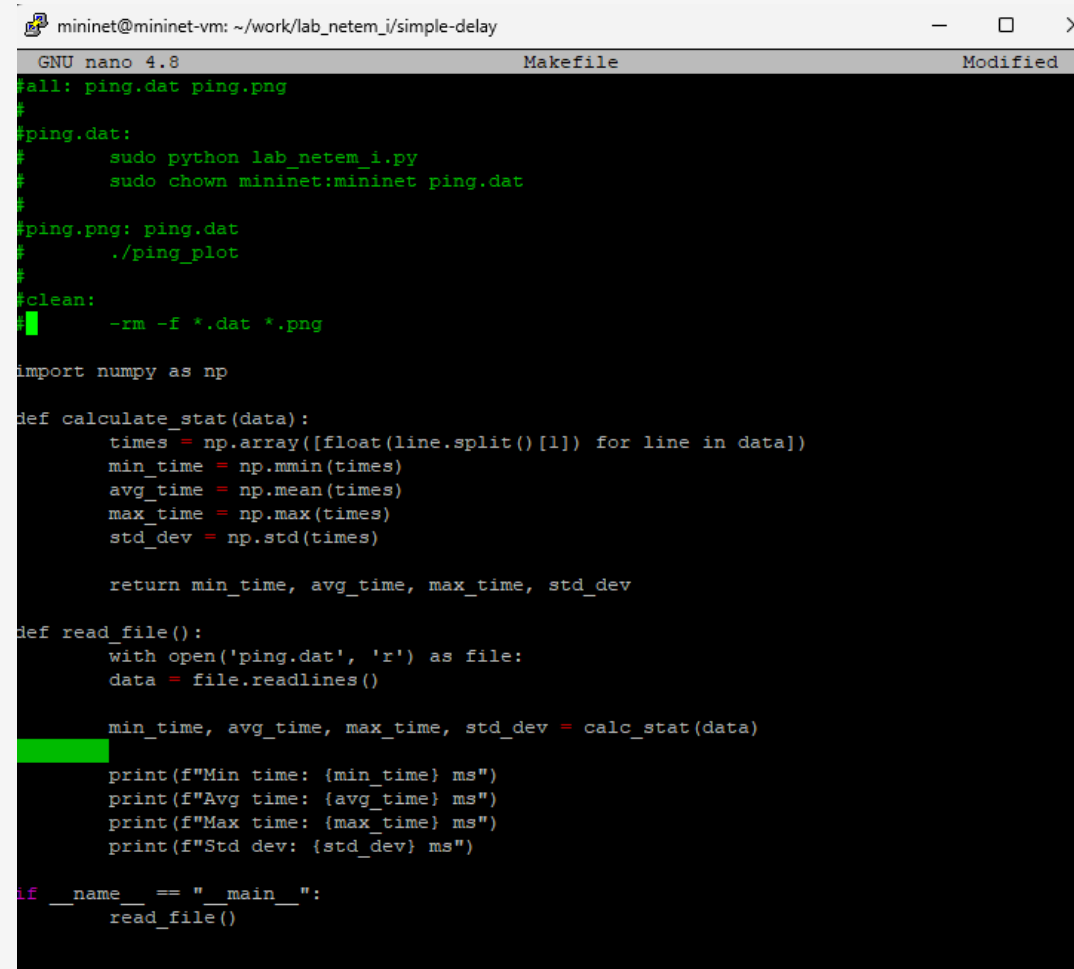


Рис. 35: Просмотр графика

## Воспроизведение экспериментов



```
mininet@mininet-vm: ~/work/lab_netem_i/simple-delay
GNU nano 4.8 Makefile Modified
#all: ping.dat ping.png
#
#ping.dat:
#    sudo python lab_netem_i.py
#    sudo chown mininet:mininet ping.dat
#
#ping.png: ping.dat
#    ./ping_plot
#
#clean:
#    -rm -f *.dat *.png

import numpy as np

def calculate_stat(data):
    times = np.array([float(line.split()[1]) for line in data])
    min_time = np.min(times)
    avg_time = np.mean(times)
    max_time = np.max(times)
    std_dev = np.std(times)

    return min_time, avg_time, max_time, std_dev

def read_file():
    with open('ping.dat', 'r') as file:
        data = file.readlines()

    min_time, avg_time, max_time, std_dev = calc_stat(data)

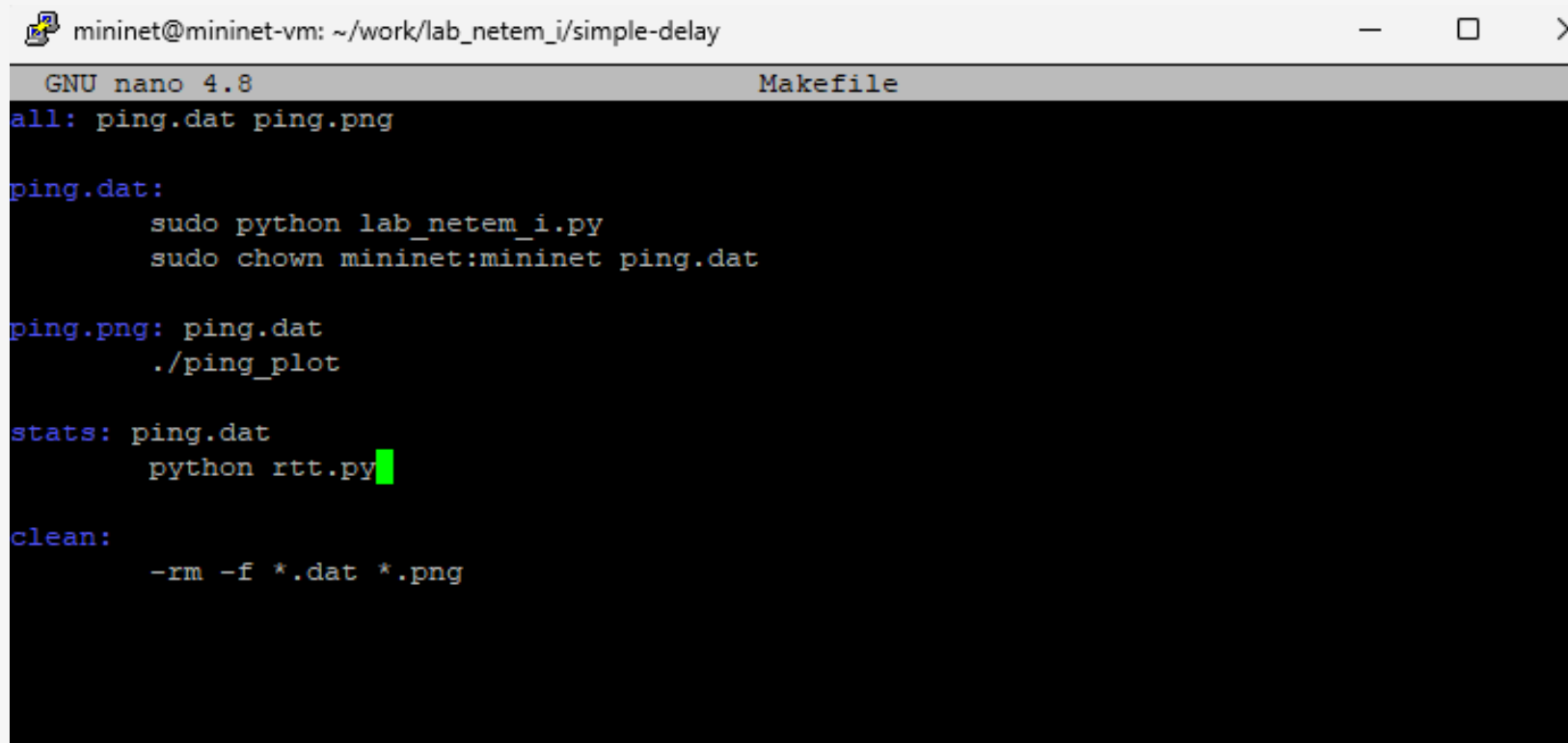
    print(f"Min time: {min_time} ms")
    print(f"Avg time: {avg_time} ms")
    print(f"Max time: {max_time} ms")
    print(f"Std dev: {std_dev} ms")

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

Рис. 36: Разработка скрипта для вычисления на основе данных файла ping.dat минимального, среднего, максимального и стандартного отклонения времени приёма-передачи

## Воспроизведение экспериментов

---



The image shows a terminal window with a title bar indicating the user is 'mininet' on a 'mininet-vm' machine, in the directory '~/work/lab\_netem\_i/simple-delay'. The terminal is running the GNU nano 4.8 text editor, editing a file named 'Makefile'. The content of the Makefile is as follows:

```
all: ping.dat ping.png

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

ping.png: ping.dat
    ./ping_plot

stats: ping.dat
    python rtt.py

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

The cursor is positioned at the end of the line 'python rtt.py' in the 'stats' target.

Рис. 37: Добавление правила запуска скрипта в Makefil

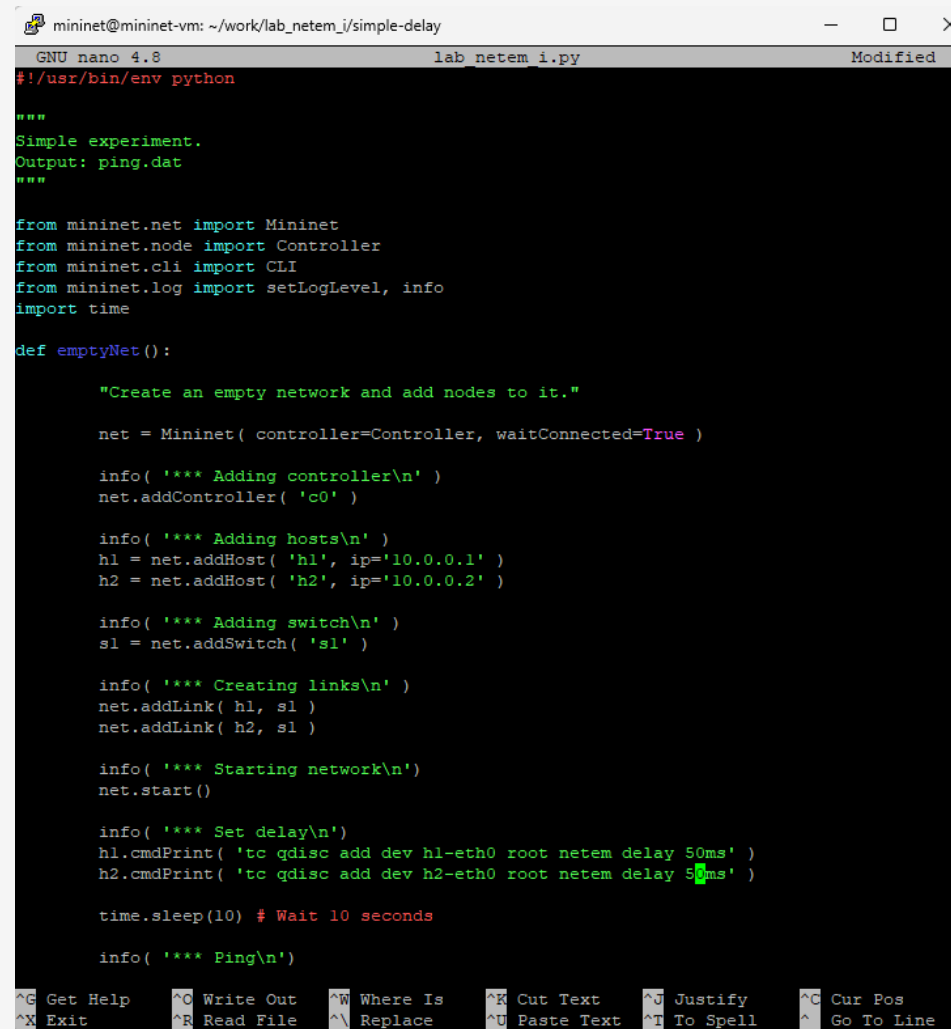
## Воспроизведение экспериментов

---

```
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ make stats  
python rtt.py  
Min time: 200.0 ms  
Avg time: 201.73737373737373 ms  
Max time: 212.0 ms  
Std dev: 1.7842512992020527 ms  
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
```

Рис. 38: Проверка

## Воспроизведение экспериментов



```
mininet@mininet-vm: ~/work/lab_netem_i/simple-delay
GNU nano 4.8 lab_netem_i.py Modified
#!/usr/bin/env python

"""
Simple experiment.
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 delay 50ms' )
    h2.cmdPrint( 'tc qdisc add dev h2-eth0 root netem delay 50ms' )

    time.sleep(10) # Wait 10 seconds

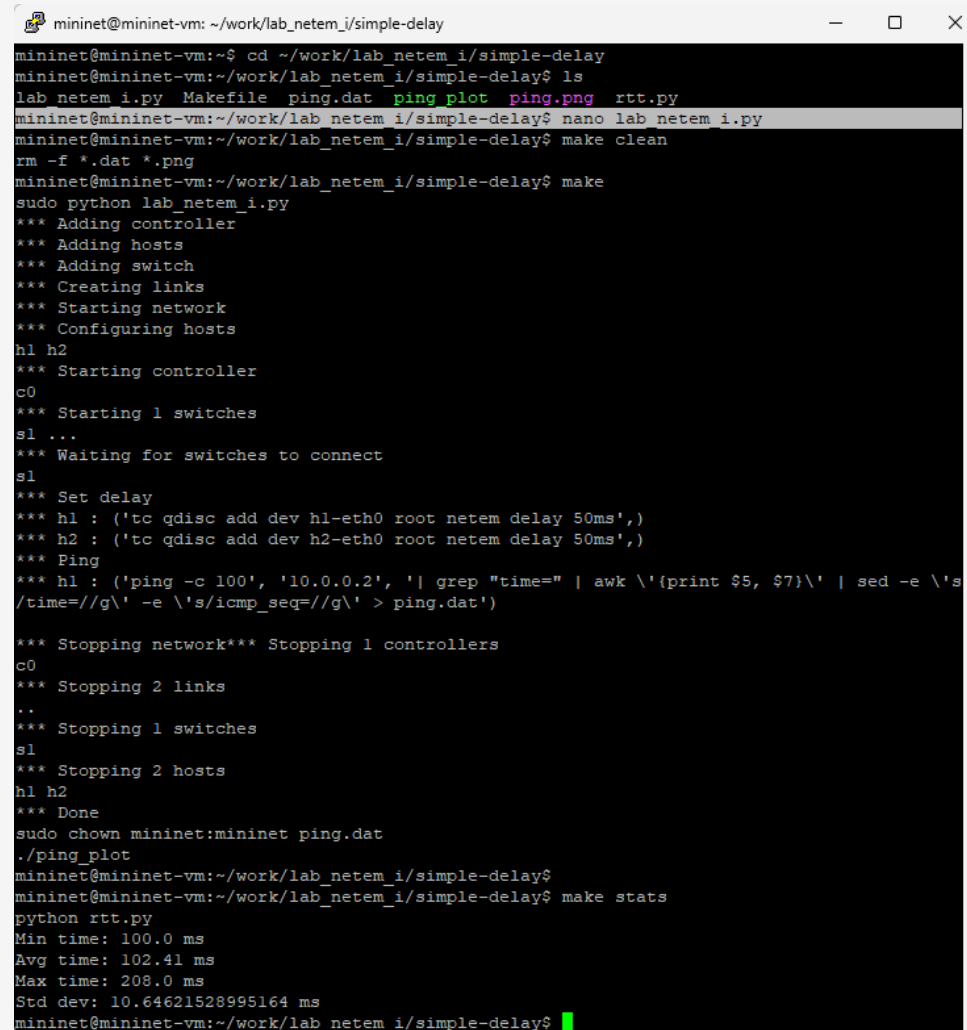
    info( '*** Ping\n' )

^G Get Help  ^O Write Out  ^W Where Is  ^R Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File  ^\ Replace   ^U Paste Text ^T To Spell  ^_ Go To Line
```

Рис. 39: Воспроизводимый эксперимент по изменению задержки



## Воспроизведение экспериментов



```
mininet@mininet-vm: ~/work/lab_netem_i/simple-delay
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ ls
lab_netem_i.py Makefile ping.dat ping_plot ping.png rtt.py
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ nano lab_netem_i.py
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ make clean
rm -f *.dat *.png
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ make
sudo python lab_netem_i.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
*** Starting network
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Waiting for switches to connect
s1
*** Set delay
*** h1 : ('tc qdisc add dev h1-eth0 root netem delay 50ms',)
*** h2 : ('tc qdisc add dev h2-eth0 root netem delay 50ms',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "time=" | awk '{print $5, $7}' | sed -e 's
/time=//g\' -e 's/icmp_seq=//g\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
./ping_plot
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ make stats
python rtt.py
Min time: 100.0 ms
Avg time: 102.41 ms
Max time: 208.0 ms
Std dev: 10.64621528995164 ms
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
```

Рис. 40: Воспроизводимый эксперимент по изменению задержки

## Воспроизведение экспериментов

---

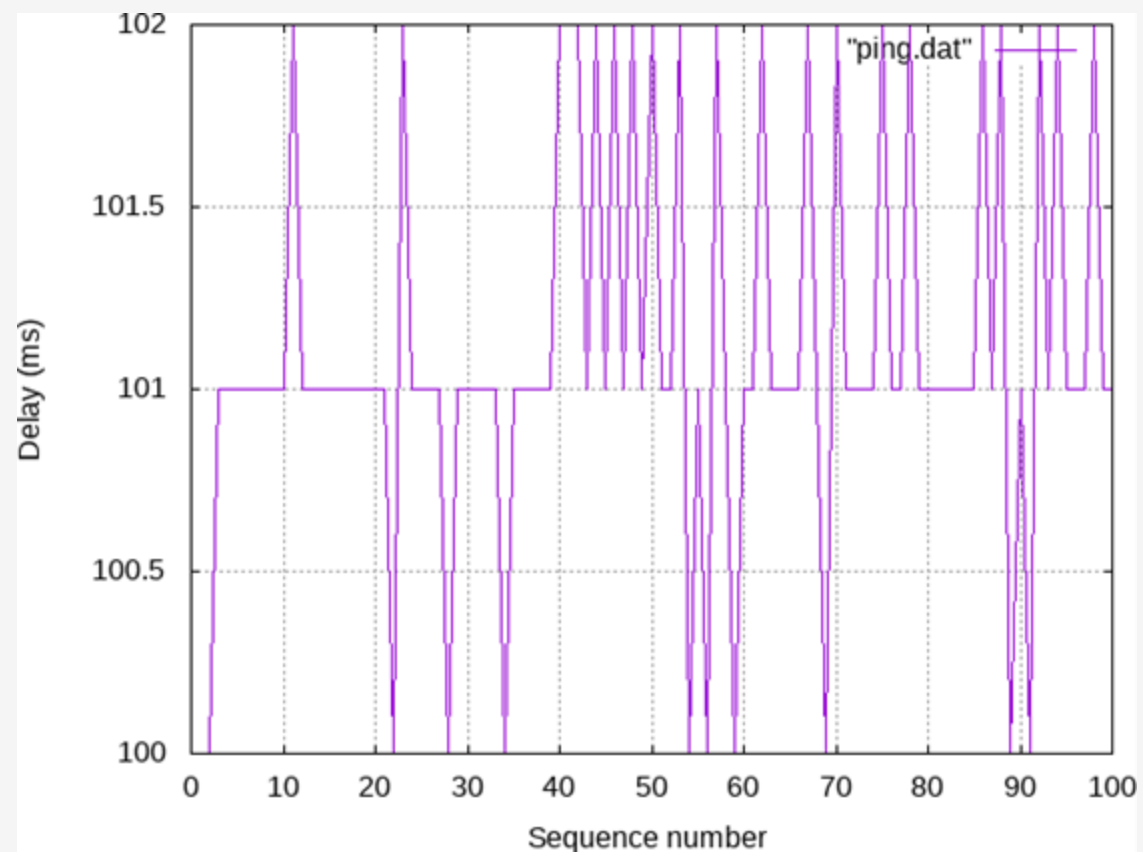
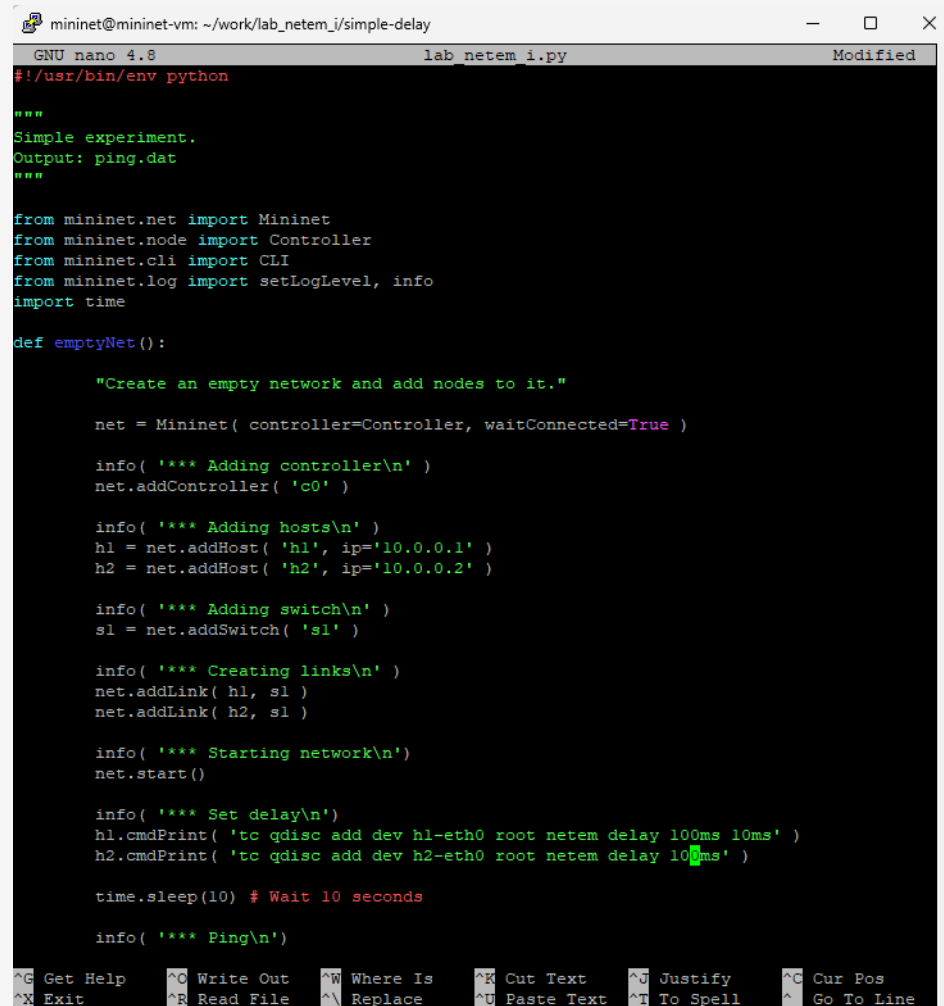


Рис. 41: Просмотр графика

## Воспроизведение экспериментов



```
mininet@mininet-vm: ~/work/lab_netem_i/simple-delay
GNU nano 4.8 lab_netem_i.py Modified
#!/usr/bin/env python

"""
Simple experiment.
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()

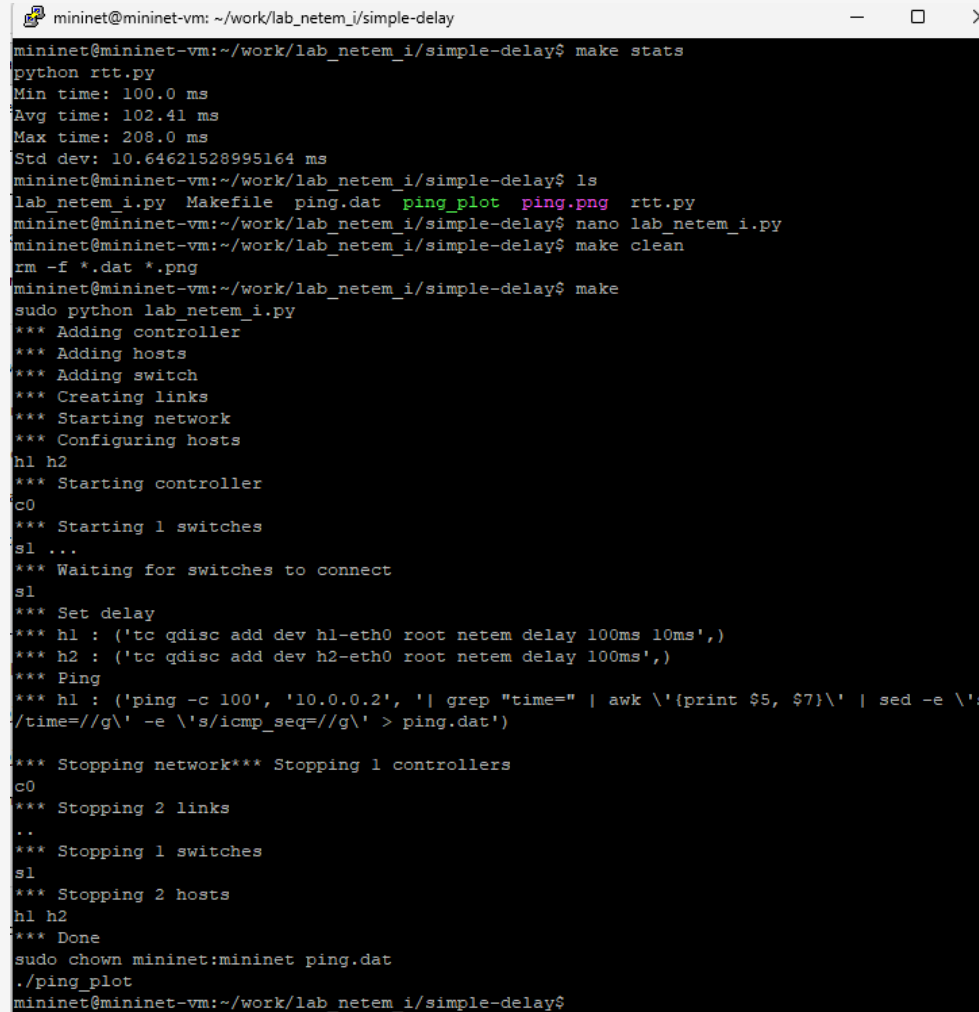
    info( '*** Set delay\n' )
    h1.cmdPrint( 'tc qdisc add dev h1-eth0 root netem delay 100ms 10ms' )
    h2.cmdPrint( 'tc qdisc add dev h2-eth0 root netem delay 100ms' )

    time.sleep(10) # Wait 10 seconds

    info( '*** Ping\n' )
```

Рис. 42: Воспроизводимый эксперимент по изменению джиттера

## Воспроизведение экспериментов



```
mininet@mininet-vm: ~/work/lab_netem_i/simple-delay
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ make stats
python rtt.py
Min time: 100.0 ms
Avg time: 102.41 ms
Max time: 208.0 ms
Std dev: 10.64621528995164 ms
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ ls
lab_netem_i.py Makefile ping.dat ping_plot ping.png rtt.py
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ nano lab_netem_i.py
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ make clean
rm -f *.dat *.png
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ make
sudo python lab_netem_i.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
*** Starting network
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Waiting for switches to connect
s1
*** Set delay
*** h1 : ('tc qdisc add dev h1-eth0 root netem delay 100ms 10ms',)
*** h2 : ('tc qdisc add dev h2-eth0 root netem delay 100ms',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "time=" | awk \'{print $5, $7}\'' | sed -e \'s/time=//g\' -e \'s/icmp_seq=//g\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
./ping_plot
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
```

Рис. 43: Воспроизводимый эксперимент по изменению джиттера

## Воспроизведение экспериментов

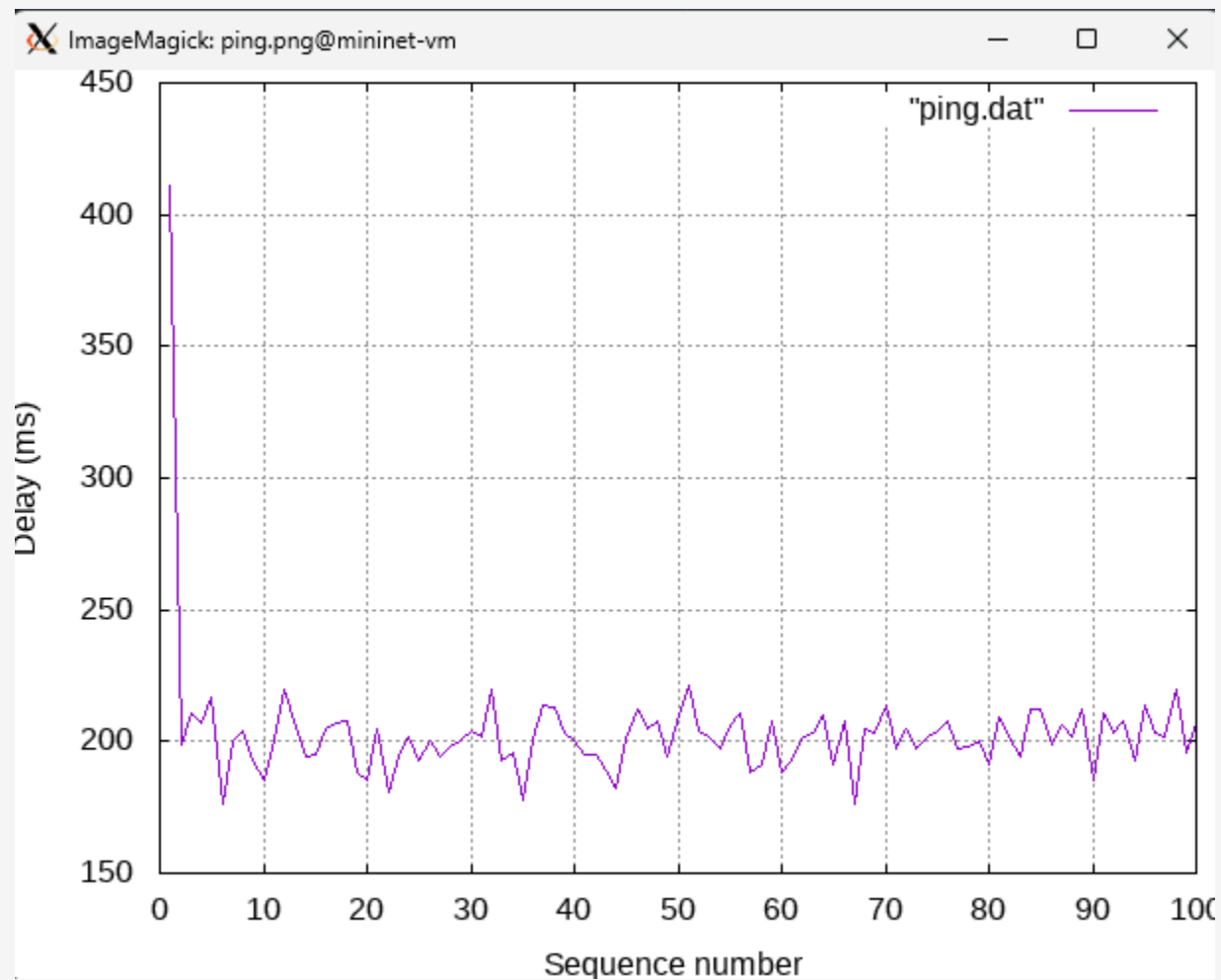
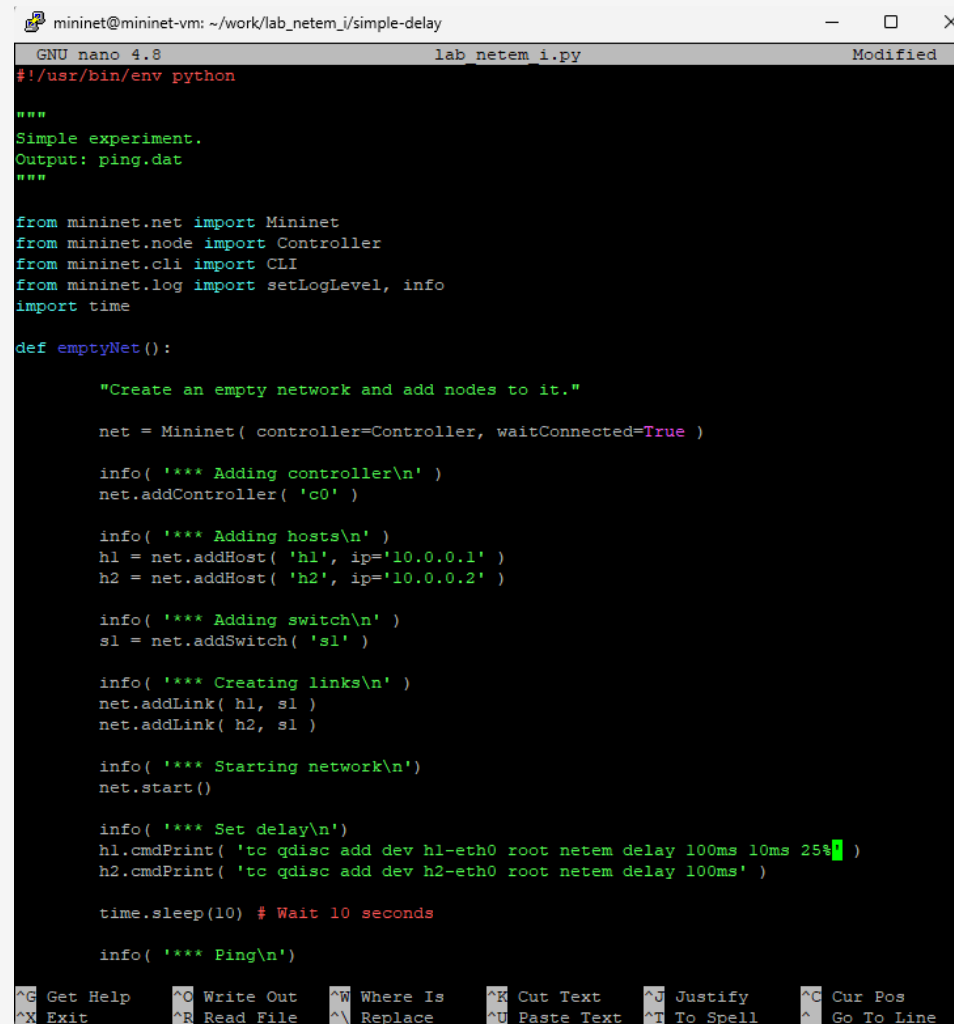


Рис. 44: Просмотр графика

## Воспроизведение экспериментов



```
mininet@mininet-vm: ~/work/lab_netem_i/simple-delay
GNU nano 4.8 lab_netem_i.py Modified
#!/usr/bin/env python

"""
Simple experiment.
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 delay 100ms 10ms 25%' )
    h2.cmdPrint( 'tc qdisc add dev h2-eth0 root netem delay 100ms' )

    time.sleep(10) # Wait 10 seconds

    info( '*** Ping\n' )
```

Рис. 45: Воспроизводимый эксперимент по изменению значения корреляции для джиттера и задержки

## Воспроизведение экспериментов

```
mininet@mininet-vm: ~/work/lab_netem_i/simple-delay
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
./ping_plot
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ display ping.png
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ nano lab_netem_i.py
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ make clean
rm -f *.dat *.png
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ make
sudo python lab_netem_i.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
*** Starting network
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Waiting for switches to connect
s1
*** Set delay
*** h1 : ('tc qdisc add dev h1-eth0 root netem delay 100ms 10ms 25%,)
*** h2 : ('tc qdisc add dev h2-eth0 root netem delay 100ms',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "time=" | awk \'{print $5, $7}\'' | sed -e \'/time=//g\' -e \'/icmp_seq=//g\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
./ping_plot
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
```

Рис. 46: Воспроизводимый эксперимент по изменению значения корреляции для джиттера и задержки

## Воспроизведение экспериментов

---

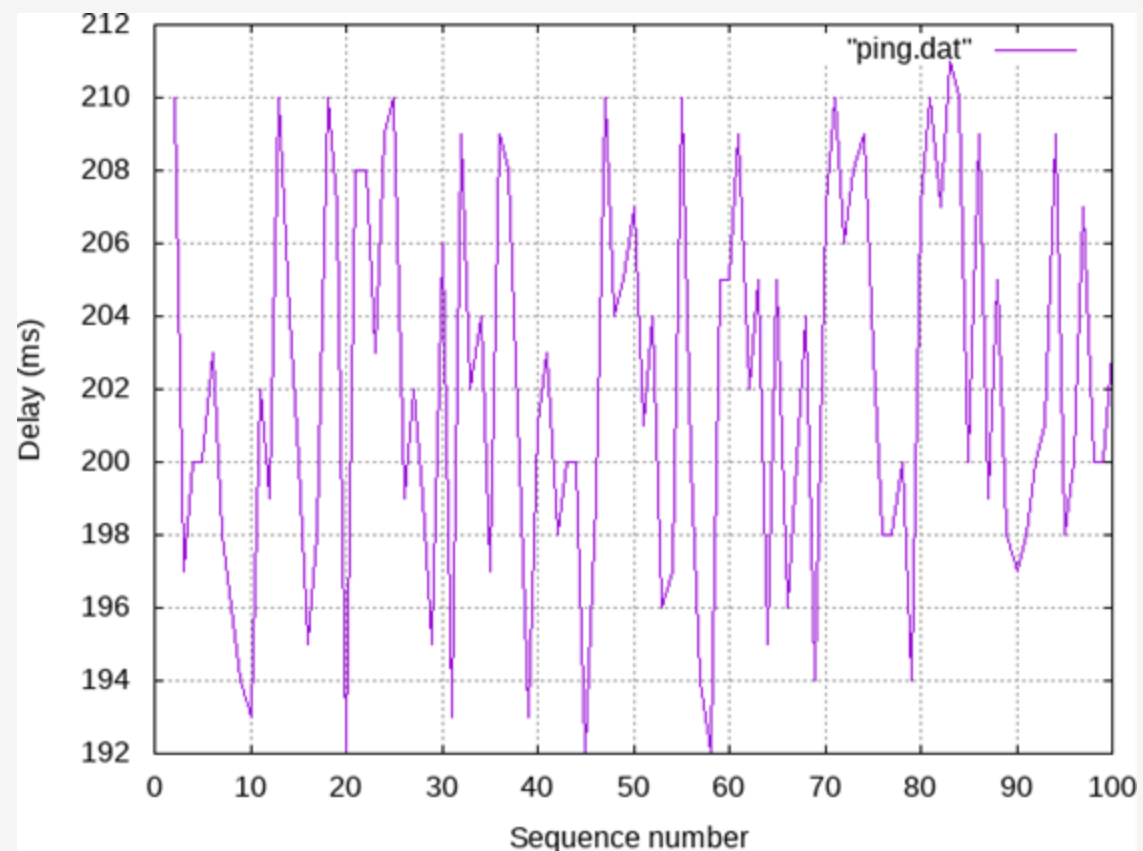
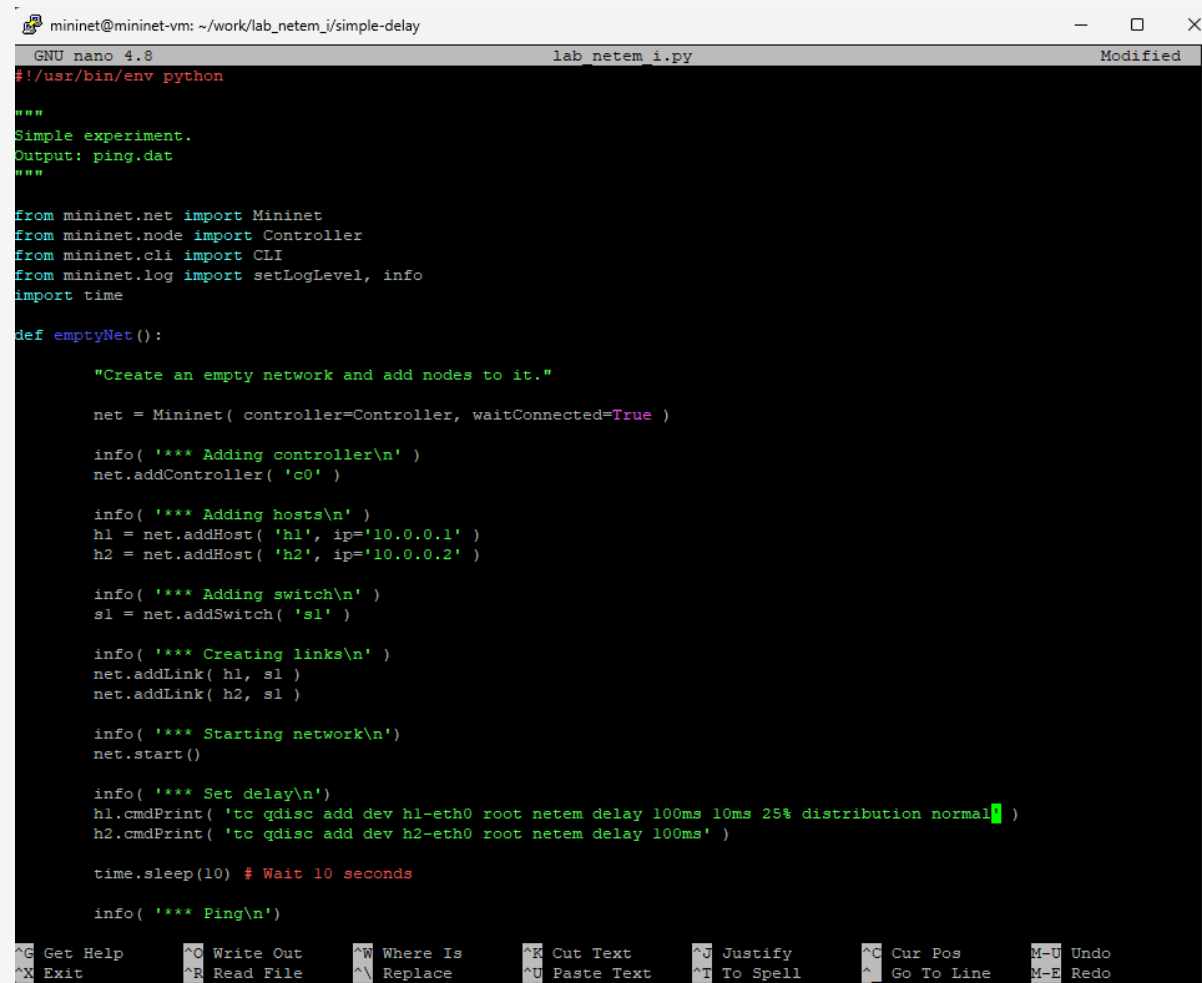


Рис. 47: Просмотр графика



## Воспроизведение экспериментов



```
mininet@mininet-vm: ~/work/lab_netem_i/simple-delay
GNU nano 4.8 lab_netem_i.py Modified
#!/usr/bin/env python

"""
Simple experiment.
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()

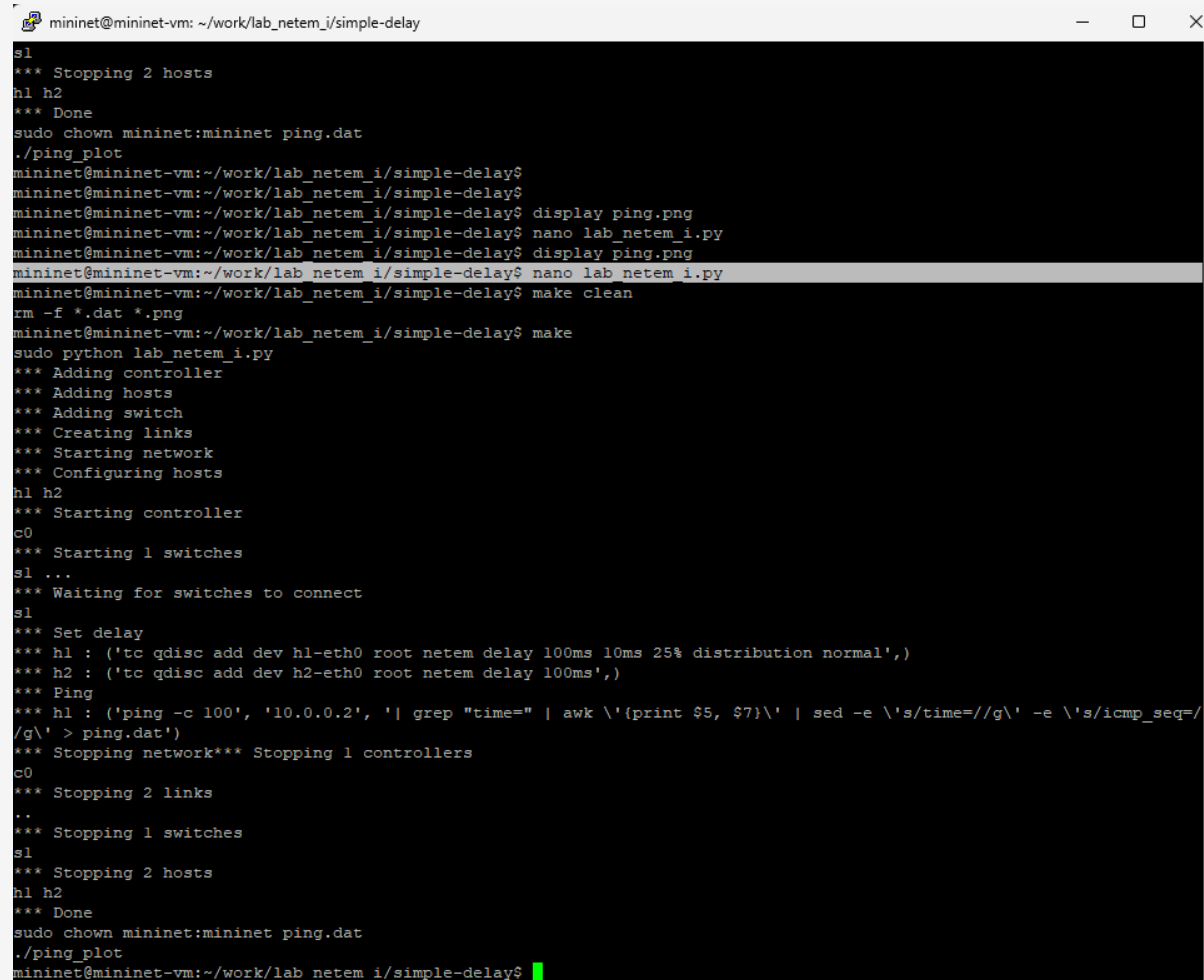
    info( '*** Set delay\n' )
    h1.cmdPrint( 'tc qdisc add dev h1-eth0 root netem delay 100ms 10ms 25% distribution normal' )
    h2.cmdPrint( 'tc qdisc add dev h2-eth0 root netem delay 100ms' )

    time.sleep(10) # Wait 10 seconds

    info( '*** Ping\n' )
```

Рис. 48: Воспроизводимый эксперимент по изменению распределения времени задержки в эмулируемой глобальной сети

## Воспроизведение экспериментов



```
mininet@mininet-vm: ~/work/lab_netem_i/simple-delay
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
./ping_plot
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ display ping.png
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ nano lab_netem_i.py
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ display ping.png
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ nano lab_netem_i.py
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ make clean
rm -f *.dat *.png
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$ make
sudo python lab_netem_i.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
*** Starting network
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Waiting for switches to connect
s1
*** Set delay
*** h1 : ('tc qdisc add dev h1-eth0 root netem delay 100ms 10ms 25% distribution normal',)
*** h2 : ('tc qdisc add dev h2-eth0 root netem delay 100ms',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "time=" | awk '{print $5, $7}' | sed -e 's/time=//g' -e 's/icmp_seq=//g' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
./ping_plot
mininet@mininet-vm:~/work/lab_netem_i/simple-delay$
```

Рис. 49: Воспроизводимый эксперимент по изменению распределения времени задержки в эмулируемой глобальной сети

## Воспроизведение экспериментов

---

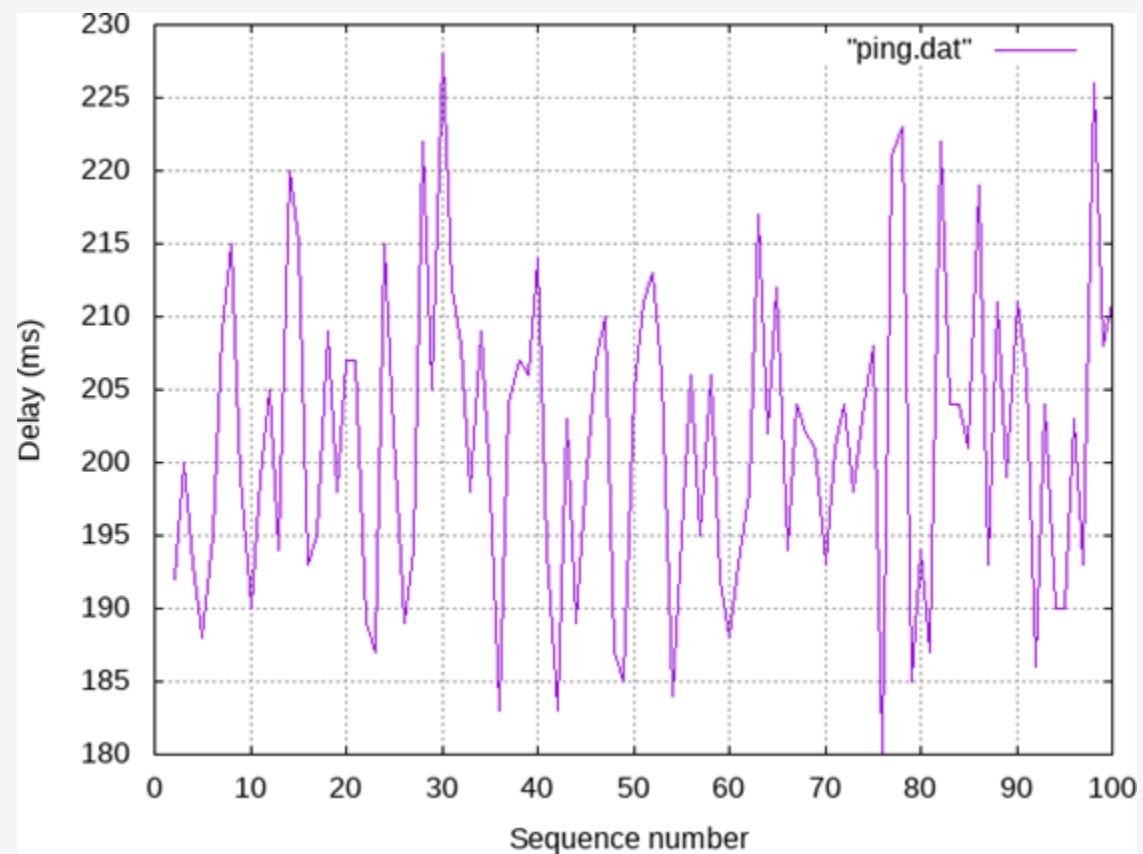


Рис. 50: Просмотр графика

## Вывод

---

- В ходе выполнения лабораторной работы познакомились с NETEM — инструментом для тестирования производительности приложений в виртуальной сети, а также получили навыки проведения интерактивного и воспроизводимого экспериментов по измерению задержки и её дрожания (jitter) в моделируемой сети в среде Mininet.

## Список литературы. Библиография

---

[1] Julia Documentation: <https://docs.julialang.org/en/v1/>