

Лабораторная работа № 5

Моделирование сетей передачи данных

Оразгелдиев Язгелди

2025-11-08

Содержание I

Информация

Докладчик

► Оразгелдиев Язгелди

Информация

Докладчик

- ▶ Оразгелдиев Язгелди
- ▶ 1032225075@pfur.ru

Информация

Докладчик

- ▶ Оразгелдиев Язгелди
- ▶ 1032225075@pfur.ru
- ▶ <https://github.com/YazgeldiOrazgeldiyev>

Цель

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.

Задание

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

Задание

1. Задайте простейшую топологию, состоящую из двух хостов и коммутатора с назначенной по умолчанию mininet сетью 10.0.0.0/8.
2. Проведите интерактивные эксперименты по исследованию параметров сети, связанных с потерей, дублированием, изменением порядка и повреждением пакетов при передаче данных.

Задание

1. Задайте простейшую топологию, состоящую из двух хостов и коммутатора с назначенной по умолчанию mininet сетью 10.0.0.0/8.
2. Проведите интерактивные эксперименты по исследованию параметров сети, связанных с потерей, дублированием, изменением порядка и повреждением пакетов при передаче данных.
3. Реализуйте воспроизводимый эксперимент по добавлению правила отбрасывания пакетов в эмулируемой глобальной сети. На экран выведите сводную информацию о потерянных пакетах.

Задание

1. Задайте простейшую топологию, состоящую из двух хостов и коммутатора с назначенной по умолчанию mininet сетью 10.0.0.0/8.
2. Проведите интерактивные эксперименты по исследованию параметров сети, связанных с потерей, дублированием, изменением порядка и повреждением пакетов при передаче данных.
3. Реализуйте воспроизводимый эксперимент по добавлению правила отбрасывания пакетов в эмулируемой глобальной сети. На экран выведите сводную информацию о потерянных пакетах.
4. Самостоятельно реализуйте воспроизводимые эксперименты по исследованию параметров сети, связанных с потерей, изменением порядка и повреждением пакетов при передаче данных. На экран выведите сводную информацию о потерянных пакетах.

Выполнение

```
yazik@fedora:~$ ssh -Y mininet@192.168.56.104
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-42-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

Last login: Sat Oct 25 11:20:14 2025 from 192.168.56.1
mininet@mininet-vm:~$ xauth list $DISPLAY
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 e6c84167b31fc7dd2fd2b69be88a4dd
mininet@mininet-vm:~$ sudo -i
root@mininet-vm:~# xauth add mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 e6c84167b31fc7dd2fd2b69be88a4dd
root@mininet-vm:~# logout
```

Рисунок 1: Изменение прав запуска X-соединения.

Выполнение

```
mininet@mininet-vm: $ sudo mn --topo=single,2 -x
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
*** Configuring hosts
h1 h2
*** Running terms on localhost:10.0
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet> █
```

Рисунок 2: Создание топологии

Выполнение

```
"host: h1"
root@mininet-vm:/home/mininet# ifconfig
h1-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 10.0.0.1 netmask 255.0.0.0 broadcast 10.255.255.255
          ether e6:c9:b1:82:fc:45 txqueuelen 1000 (Ethernet)
            RX packets 0 bytes 0 (0.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 0 bytes 0 (0.0 B)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
          loop txqueuelen 1000 (Local Loopback)
            RX packets 902 bytes 308540 (308.5 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 902 bytes 308540 (308.5 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
root@mininet-vm:/home/mininet#
```

Рисунок 3: Команда *ifconfig* на хосте *h1*

Выполнение

```
"host: h2"
root@mininet-vm:/home/mininet# ifconfig
h2-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
          inet 10.0.0.2  netmask 255.0.0.0  broadcast 10.255.255.255
            ether a2:c3:5a:4f:d8:bd  txqueuelen 1000  (Ethernet)
            RX packets 0  bytes 0 (0.0 B)
            RX errors 0  dropped 0  overruns 0  frame 0
            TX packets 0  bytes 0 (0.0 B)
            TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
          inet 127.0.0.1  netmask 255.0.0.0
            loop  txqueuelen 1000  (Local Loopback)
            RX packets 984  bytes 309672 (309.6 KB)
            RX errors 0  dropped 0  overruns 0  frame 0
            TX packets 984  bytes 309672 (309.6 KB)
            TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
root@mininet-vm:/home/mininet# █
```

Рисунок 4: Команда *ifconfig* на хосте *h2*

Выполнение

```
root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 6
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=1.47 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.153 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.040 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.053 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.048 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.057 ms
--- 10.0.0.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5088ms
rtt min/avg/max/mdev = 0.040/0.303/1.468/0.522 ms
```

Рисунок 5: Пингование

Выполнение

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

Рисунок 6: *Добавление процента потерь*

Выполнение

```
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.051 ms
--- 10.0.0.2 ping statistics ---
100 packets transmitted, 86 received, 14% packet loss, time 101382ms
rtt min/avg/max/mdev = 0.043/0.063/0.710/0.074 ms
```

Рисунок 7: Пингование

Выполнение

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

Рисунок 8: *Добавление процента потерь*

Выполнение

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

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

Рисунок 9: Пингование

Выполнение

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

Рисунок 10: Восстановление конфигурации

Выполнение

```
root@mininet-vm:/home/mininet# ping 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.448 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.249 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.115 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.045 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.059 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=0.045 ms
^C
--- 10.0.0.2 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8173ms
rtt min/avg/max/mdev = 0.045/0.123/0.448/0.130 ms
```

Рисунок 11: Пингование

Выполнение

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

Рисунок 12: Добавление коэффициента потерь с корреляцией

Выполнение

```
root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 50
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.506 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.285 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.125 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.043 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.048 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=0.056 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.049 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=0.047 ms
64 bytes from 10.0.0.2: icmp_seq=21 ttl=64 time=0.058 ms
64 bytes from 10.0.0.2: icmp_seq=22 ttl=64 time=0.047 ms
64 bytes from 10.0.0.2: icmp_seq=34 ttl=64 time=0.180 ms
64 bytes from 10.0.0.2: icmp_seq=36 ttl=64 time=0.059 ms
64 bytes from 10.0.0.2: icmp_seq=38 ttl=64 time=0.066 ms
64 bytes from 10.0.0.2: icmp_seq=39 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=40 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=41 ttl=64 time=0.048 ms
th0 root 64 bytes from 10.0.0.2: icmp_seq=44 ttl=64 time=0.046 ms
64 bytes from 10.0.0.2: icmp_seq=45 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=46 ttl=64 time=0.060 ms
64 bytes from 10.0.0.2: icmp_seq=47 ttl=64 time=0.049 ms
64 bytes from 10.0.0.2: icmp_seq=48 ttl=64 time=0.054 ms
64 bytes from 10.0.0.2: icmp_seq=50 ttl=64 time=0.050 ms

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

Рисунок 13: Пингование

Выполнение

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

Рисунок 14: Добавление повреждения

Выполнение

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

Рисунок 15: Запуск сервера

Выполнение

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

Рисунок 16: Запуск клиента

Выполнение

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

Рисунок 17: Добавление переупорядочивания пакетов

Выполнение

```
root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 20
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=11.4 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.333 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.127 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=10.8 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=10.8 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=10.6 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=10.8 ms
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=10.6 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.049 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.046 ms
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=0.048 ms
64 bytes from 10.0.0.2: icmp_seq=15 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=10.7 ms
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=0.047 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=0.050 ms
64 bytes from 10.0.0.2: icmp_seq=20 ttl=64 time=0.040 ms

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

Рисунок 18: Пингование

Выполнение

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

Рисунок 19: Добавление дублирования

Выполнение

```
root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 20
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.465 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.270 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.117 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.047 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.059 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.059 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.044 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.044 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=7 ttl=64 time=0.049 ms
64 bytes from 10.0.0.2: icmp_seq=8 ttl=64 time=0.049 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=9 ttl=64 time=0.079 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.054 ms
64 bytes from 10.0.0.2: icmp_seq=10 ttl=64 time=0.054 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=11 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.053 ms
64 bytes from 10.0.0.2: icmp_seq=12 ttl=64 time=0.054 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.049 ms
64 bytes from 10.0.0.2: icmp_seq=13 ttl=64 time=0.049 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=14 ttl=64 time=0.046 ms
64 bytes from 10.0.0.2: icmp_seq=15 ttl=64 time=0.046 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=0.056 ms
64 bytes from 10.0.0.2: icmp_seq=16 ttl=64 time=0.057 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=0.048 ms
64 bytes from 10.0.0.2: icmp_seq=17 ttl=64 time=0.048 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=18 ttl=64 time=0.052 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=0.051 ms
64 bytes from 10.0.0.2: icmp_seq=19 ttl=64 time=0.051 ms (DUP!)
64 bytes from 10.0.0.2: icmp_seq=20 ttl=64 time=0.050 ms

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

Рисунок 20: Пингование

Выполнение

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

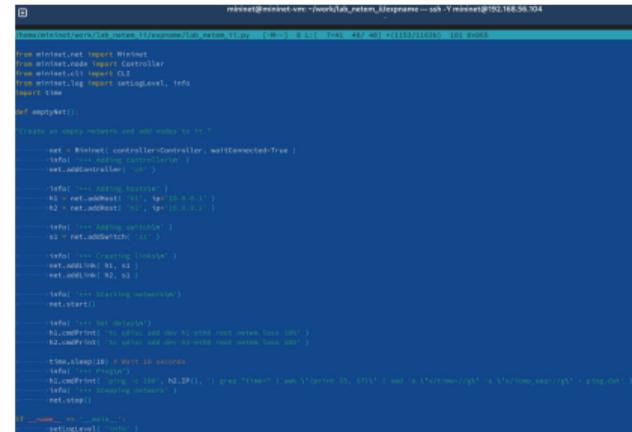
Рисунок 21: *Создание каталога*

Выполнение

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

Рисунок 22: Создание подкаталога

Выполнение



```
root@mininet-v3:/work/lab_neter_ii# python lab_neter_ii.py
[INFO] 2018-07-01T14:48:40Z mininet@102.168.56.50:304
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 host0\n')
    h0 = net.addHost('h0', ip='192.168.0.1')
    h2 = net.addHost('h2', ip='192.168.0.2')

    info('*** adding switch0\n')
    s0 = net.addSwitch('s0')

    info('*** creating links\n')
    net.addLink(h0, s0)
    net.addLink(h2, s0)

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

    info('*** test delay\n')
    h0.cmd('ping -c 2 h2 & sleep 1000 & route del default dev h0eth0')
    h2.cmd('ping -c 2 h0 & sleep 1000 & route del default dev h2eth0')

    time.sleep(10) # Wait 10 seconds

    info('*** stop\n')
    h0.cmdPrint("ifconfig h0 eth0:0 down")
    h2.cmdPrint("ifconfig h2 eth0:0 down")
    net.stop()

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

Рисунок 23: Скрипт *lab_neter_ii.py*

Выполнение

```
info: --- ping[0]
hi,cmdPrint('ping < 100', h2_ip(), '| grep "packet loss" | sed \'(print $1, $2, $3)\'\> ping.dat')
info: --- ping[1]
```

Рисунок 24: Изменение в скрипте *lab_netem_ii.py*

Выполнение

```
all: ping.dat

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

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

Рисунок 25: Создание Makefile

Выполнение

```
mininet@mininet-vm:~/work/lab_network_1/expname$ make
sudo python lab.netem_14.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
*** Starting network
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Waiting for switches to connect
s1
*** Set delay
*** h1 : ('tc qdisc add dev h1-eth0 root netem loss 10%',)
*** h2 : ('tc qdisc add dev h2-eth0 root netem loss 10%',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "packet loss" | awk \'{print $5, $7, $8}\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
...
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
```

Рисунок 26: Выполнение эксперимента

Выполнение

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

Рисунок 27: Добавление коэффициента потерь с корреляцией

Выполнение

```
mininet@mininet-vm:~/work/lab_neter_11/experiments$ make
sudo python lab_neter_11.py
*** Adding controller
c0
*** Adding hosts
h1 h2
*** Adding switch
s1
*** Creating links
*** Starting network
*** Starting hosts
h1 h2
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Waiting for switches to connect
s1
*** Set delay
*** h1 : ('tc qdisc add dev h1-eth0 root netem loss 50% 50%',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "packet loss" | awk \'{print $5, $7, $8}\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
...
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
```

Рисунок 28: Выполнение эксперимента

Выполнение

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

Рисунок 29: Добавление повреждения

Выполнение

```
admin@mininet-wm:~/work/lab_reseau_11/dynamips$ make
sudo python lab_neterm_if.py
*** Adding controller
*** Adding hosts
*** Adding switches
*** Creating links
*** Starting network
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Waiting for switches to connect
s1
*** Set delay
*** h1 : ('tc qdisc add dev h1-eth0 root netem corrupt 0.01%',)
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "packet loss" | awk \'{print $5, $7, $8}\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
```

Рисунок 30: Выполнение эксперимента

Выполнение

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

Рисунок 31: Добавление переупорядочивания пакетов

Выполнение

```
pi@raspberrypi:~/work/lab_experiments$ make
sudo python lab_netem_ii.py
*** Adding controller
*** Adding hosts
*** Adding switches
*** Creating links
*** Starting network
*** Configuring hosts
:1 h2
*** Starting controller
:20
*** Starting 1 switches
:1 ...
*** Waiting for switches to connect
:1
*** Set delay
*** h1 : ('tc qdisc add dev h1-eth0 root netem delay 10ms reorder 25% 50%')
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "packet loss" | awk \'(print $5, $7, $8)\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
:20
*** Stopping 2 links
:...
*** Stopping 1 switches
:1
*** Stopping 2 hosts
:1 h2
*** Done
sudo chown mininet:mininet ping.dat
```

Рисунок 32: Выполнение эксперимента

Выполнение

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

Рисунок 33: Добавление дублирования

Выполнение

```
mininet@mininet-vm:~/work/lab_netsim_11/experiments$ make
sudo python lab_netsim_11.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
*** Starting network
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Waiting for switches to connect
s1
*** Set delay
*** h1 : ('tc qdisc add dev h1-eth0 root netem duplicate 50%')
*** Ping
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "packet loss" | awk \'{print $5, $7, $8}\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 2 links
...
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
```

Рисунок 34: Выполнение эксперимента

Вывод

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