

Отчёт по лабораторной работе №6

Моделирование сетей передачи данных

Настройка пропускной способности глобальной сети с помощью Token Bucket Filter

Выполнил: Исаев Булат Абубакарович,
НПИбд-01-22, 1132227131

Содержание

1	Цель работы	1
2	Выполнение лабораторной работы.....	1
3	Вывод.....	24
4	Список литературы. Библиография	24

1 Цель работы

Основной целью работы является знакомство с принципами работы дисциплины очереди Token Bucket Filter, которая формирует входящий/исходящий трафик для ограничения пропускной способности, а также получение навыков моделирования и исследования поведения трафика посредством проведения интерактивного и воспроизводимого экспериментов в Mininet.

2 Выполнение лабораторной работы

В виртуальной машине mininet исправим права запуска X-соединения (рис. 1):

```
mininet@mininet-vm:~$ xauth list $DISPLAY
mininet-vm/unix:10  MIT-MAGIC-COOKIE-1  bced047e0c3bcbb01e84e95f729f6f92
mininet@mininet-vm:~$ sudo -i
root@mininet-vm:~# xauth add mininet-vm/unix:10  MIT-MAGIC-COOKIE-1  bced047e0c3bcbb01e84e95f729f6f9
2
root@mininet-vm:~# logout
mininet@mininet-vm:~$
```

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

Зададим топологию сети, состоящую из двух хостов и двух коммутаторов с назначенной по умолчанию mininet сетью 10.0.0.0/8 (рис. 2):

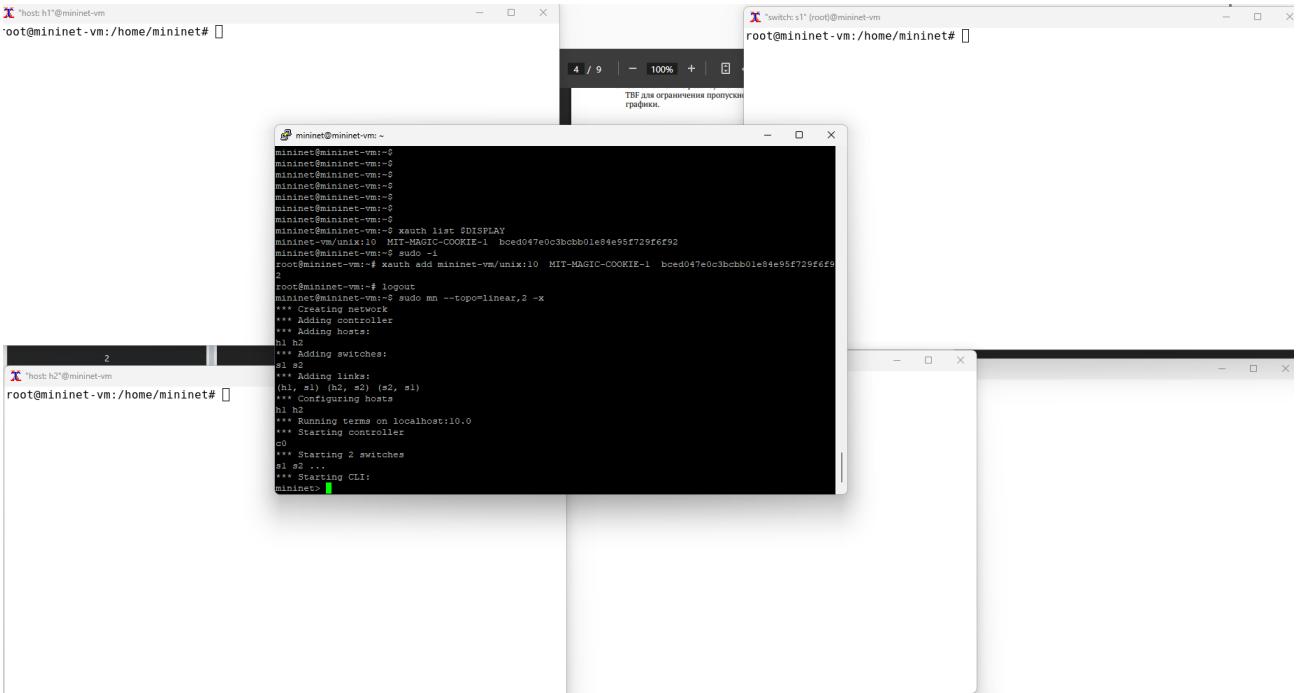


Рис. 2: Создание топологии с двумя хостами и двумя коммутаторами

На хостах h1, h2 и на коммутаторах s1, s2 введём команду ifconfig, чтобы отобразить информацию, относящуюся к их сетевым интерфейсам и назначенным им IP-адресам. В дальнейшем при работе с NETEM и командой tc будем использовать интерфейсы h1-eth0, h2-eth0, s1-eth2 (рис. 3):

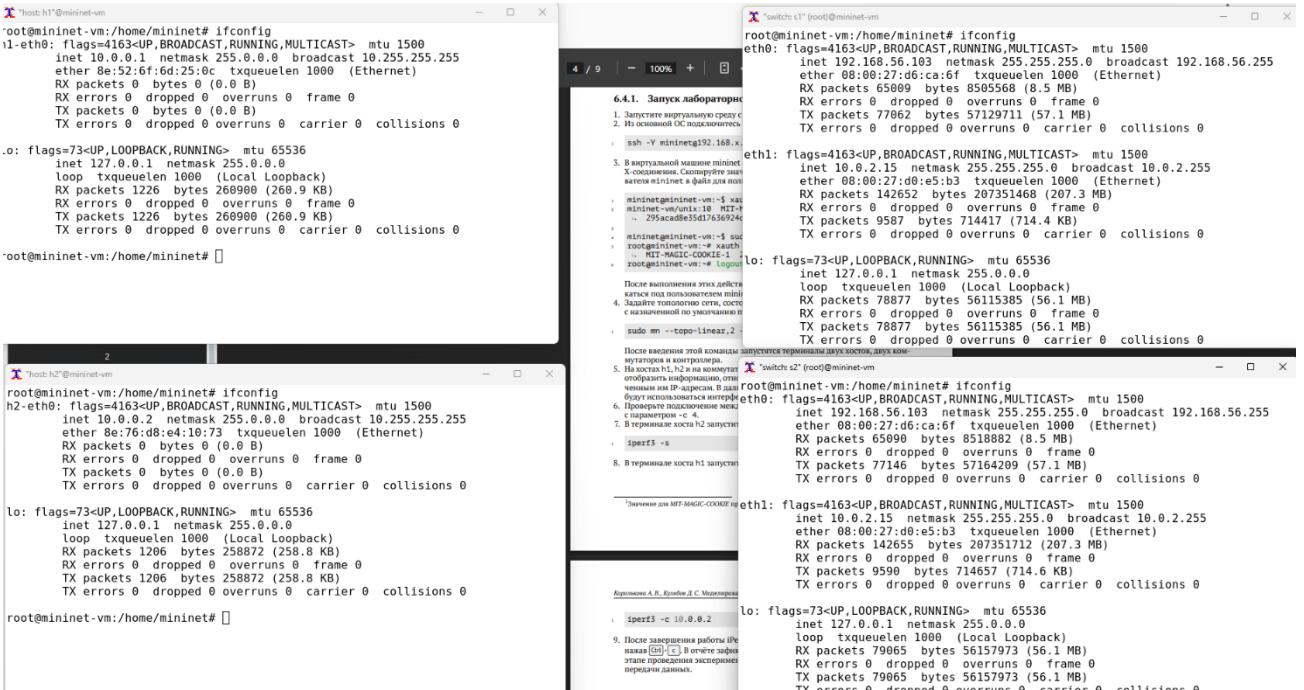


Рис. 3: Отображение информации сетевых интерфейсов и IP-адресов

Проверим подключение между хостами h1 и h2 с помощью команды ping с параметром -c 4 (рис. 4):

```

X "host h1"@mininet-vm
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        loop txqueuelen 1000 (Local Loopback)
        RX packets 1226 bytes 260900 (260.9 KB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 1226 bytes 260900 (260.9 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@mininet-vm:/home/mininet# ping -c 4 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=5.43 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.422 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.093 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.073 ms

--- 10.0.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3056ms
rtt min/avg/max/mdev = 0.073/1.505/5.434/2.272 ms
root@mininet-vm:/home/mininet# □

2

X "host h2"@mininet-vm
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        loop txqueuelen 1000 (Local Loopback)
        RX packets 1206 bytes 258872 (258.8 KB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 1206 bytes 258872 (258.8 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@mininet-vm:/home/mininet# ping -c 4 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=3.56 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.111 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.070 ms
64 bytes from 10.0.0.1: icmp_seq=4 ttl=64 time=0.106 ms

--- 10.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3036ms
rtt min/avg/max/mdev = 0.070/0.962/3.563/1.501 ms

```

Рис. 4: Проверка подключения между хостами h1 и h2

В терминале хоста h2 запустим iPerf3 в режиме сервера (рис. 5):

```

X "host h2"@mininet-vm
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 1206 bytes 258872 (258.8 KB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 1206 bytes 258872 (258.8 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@mininet-vm:/home/mininet# ping -c 4 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=3.56 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.111 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.070 ms
64 bytes from 10.0.0.1: icmp_seq=4 ttl=64 time=0.106 ms

--- 10.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3036ms
rtt min/avg/max/mdev = 0.070/0.962/3.563/1.501 ms
root@mininet-vm:/home/mininet# iperf3 -s
warning: this system does not seem to support IPv6 - trying IPv4
-----
Server listening on 5201
-----

```

Рис. 5: Запуск iPerf3 в режиме сервера на хосте h2

В терминале хоста h1 запустим iPerf3 в режиме клиента (рис. 6):

```

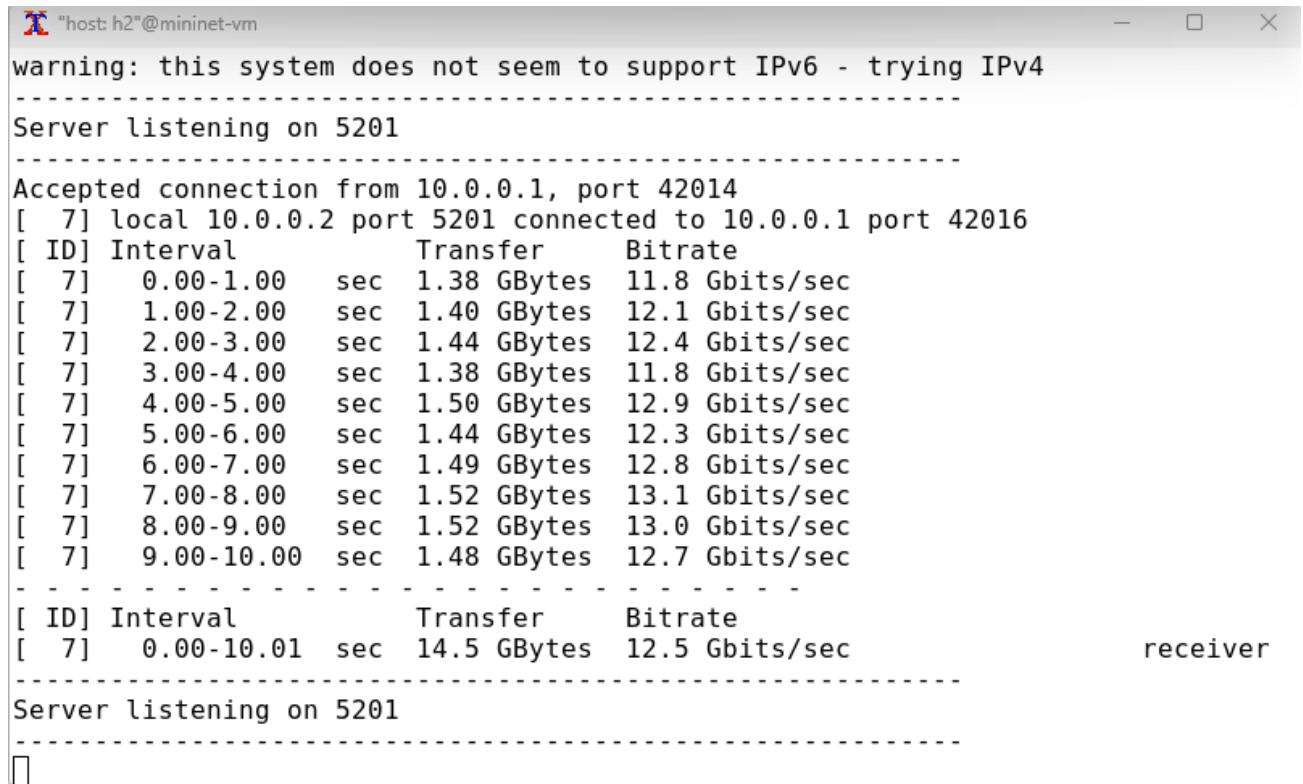
X "host h1"@mininet-vm
--- 10.0.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3056ms
rtt min/avg/max/mdev = 0.073/1.505/5.434/2.272 ms
root@mininet-vm:/home/mininet# iperf3 -c 10.0.0.2
Connecting to host 10.0.0.2, port 5201
[ 7] local 10.0.0.1 port 42016 connected to 10.0.0.2 port 5201
[ ID] Interval          Transfer     Bitrate      Retr  Cwnd
[ 7]  0.00-1.01   sec   1.39 GBytes   11.8 Gbits/sec   0  8.03 MBytes
[ 7]  1.01-2.01   sec   1.39 GBytes   12.0 Gbits/sec   0  8.03 MBytes
[ 7]  2.01-3.00   sec   1.44 GBytes   12.4 Gbits/sec   0  8.03 MBytes
[ 7]  3.00-4.00   sec   1.38 GBytes   11.9 Gbits/sec   0  8.03 MBytes
[ 7]  4.00-5.00   sec   1.49 GBytes   12.8 Gbits/sec   0  8.03 MBytes
[ 7]  5.00-6.01   sec   1.44 GBytes   12.3 Gbits/sec   0  8.03 MBytes
[ 7]  6.01-7.00   sec   1.49 GBytes   12.8 Gbits/sec   0  8.03 MBytes
[ 7]  7.00-8.00   sec   1.52 GBytes   13.1 Gbits/sec   0  8.03 MBytes
[ 7]  8.00-9.01   sec   1.52 GBytes   13.0 Gbits/sec   0  8.03 MBytes
[ 7]  9.01-10.00  sec   1.47 GBytes   12.7 Gbits/sec   0  8.03 MBytes
-----
[ ID] Interval          Transfer     Bitrate      Retr
[ 7]  0.00-10.00  sec   14.5 GBytes   12.5 Gbits/sec   0
[ 7]  0.00-10.01  sec   14.5 GBytes   12.5 Gbits/sec

iperf Done.
root@mininet-vm:/home/mininet# ■

```

Рис. 6: Запуск iPerf3 в режиме клиента на хосте h1

После завершения работы iPerf3 на хосте h1 остановим iPerf3 на хосте h2, нажав Ctrl + c (рис. 7):



```
"host h2"@mininet-vm
warning: this system does not seem to support IPv6 - trying IPv4
-----
Server listening on 5201
-----
Accepted connection from 10.0.0.1, port 42014
[ 7] local 10.0.0.2 port 5201 connected to 10.0.0.1 port 42016
[ ID] Interval Transfer Bitrate
[ 7] 0.00-1.00 sec 1.38 GBytes 11.8 Gbits/sec
[ 7] 1.00-2.00 sec 1.40 GBytes 12.1 Gbits/sec
[ 7] 2.00-3.00 sec 1.44 GBytes 12.4 Gbits/sec
[ 7] 3.00-4.00 sec 1.38 GBytes 11.8 Gbits/sec
[ 7] 4.00-5.00 sec 1.50 GBytes 12.9 Gbits/sec
[ 7] 5.00-6.00 sec 1.44 GBytes 12.3 Gbits/sec
[ 7] 6.00-7.00 sec 1.49 GBytes 12.8 Gbits/sec
[ 7] 7.00-8.00 sec 1.52 GBytes 13.1 Gbits/sec
[ 7] 8.00-9.00 sec 1.52 GBytes 13.0 Gbits/sec
[ 7] 9.00-10.00 sec 1.48 GBytes 12.7 Gbits/sec
-----
[ ID] Interval Transfer Bitrate
[ 7] 0.00-10.01 sec 14.5 GBytes 12.5 Gbits/sec
----- receiver
Server listening on 5201
-----
```

Рис. 7: Остановка iPerf3

Команду tc можно применить к сетевому интерфейсу устройства для формирования исходящего трафика. Требуется ограничить скорость отправки данных с конечного хоста с помощью фильтра Token Bucket Filter (tbf).

Изменим пропускную способность хоста h1, установив пропускную способность на 10 Гбит/с на интерфейсе h1-eth0 и параметры TBF-фильтра (рис. 8):

```

X "host: h1"@mininet-vm
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 42016 connected to 10.0.0.2 port 5201
[ ID] Interval           Transfer     Bitrate      Retr  Cwnd
[ 7]  0.00-1.01   sec  1.39 GBytes  11.8 Gbits/sec    0  8.03 MBytes
[ 7]  1.01-2.01   sec  1.39 GBytes  12.0 Gbits/sec    0  8.03 MBytes
[ 7]  2.01-3.00   sec  1.44 GBytes  12.4 Gbits/sec    0  8.03 MBytes
[ 7]  3.00-4.00   sec  1.38 GBytes  11.9 Gbits/sec    0  8.03 MBytes
[ 7]  4.00-5.00   sec  1.49 GBytes  12.8 Gbits/sec    0  8.03 MBytes
[ 7]  5.00-6.01   sec  1.44 GBytes  12.3 Gbits/sec    0  8.03 MBytes
[ 7]  6.01-7.00   sec  1.49 GBytes  12.8 Gbits/sec    0  8.03 MBytes
[ 7]  7.00-8.00   sec  1.52 GBytes  13.1 Gbits/sec    0  8.03 MBytes
[ 7]  8.00-9.01   sec  1.52 GBytes  13.0 Gbits/sec    0  8.03 MBytes
[ 7]  9.01-10.00  sec  1.47 GBytes  12.7 Gbits/sec   0  8.03 MBytes
[ ID] Interval           Transfer     Bitrate      Retr
[ 7]  0.00-10.00  sec  14.5 GBytes  12.5 Gbits/sec    0
[ 7]  0.00-10.01  sec  14.5 GBytes  12.5 Gbits/sec

iperf Done.
root@mininet-vm:/home/mininet# ^C
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root tbf rate 10gb
it burst 5000000 limit 15000000
root@mininet-vm:/home/mininet#

```

Рис. 8: Изменение пропускной способности хоста h1

Фильтр tbf требует установки значения всплеска при ограничении скорости. Это значение должно быть достаточно высоким, чтобы обеспечить установленную скорость. Она должна быть не ниже указанной частоты, делённой на HZ, где HZ — тактовая частота, настроенная как параметр ядра, и может быть извлечена с помощью следующей команды (рис. 9):

```

X "host: h1"@mininet-vm
[ 7]  0.00-1.01  sec  1.39 GBytes  11.8 Gbits/sec    0  8.03 MBytes
[ 7]  1.01-2.01  sec  1.39 GBytes  12.0 Gbits/sec    0  8.03 MBytes
[ 7]  2.01-3.00  sec  1.44 GBytes  12.4 Gbits/sec    0  8.03 MBytes
[ 7]  3.00-4.00  sec  1.38 GBytes  11.9 Gbits/sec    0  8.03 MBytes
[ 7]  4.00-5.00  sec  1.49 GBytes  12.8 Gbits/sec    0  8.03 MBytes
[ 7]  5.00-6.01  sec  1.44 GBytes  12.3 Gbits/sec    0  8.03 MBytes
[ 7]  6.01-7.00  sec  1.49 GBytes  12.8 Gbits/sec    0  8.03 MBytes
[ 7]  7.00-8.00  sec  1.52 GBytes  13.1 Gbits/sec    0  8.03 MBytes
[ 7]  8.00-9.01  sec  1.52 GBytes  13.0 Gbits/sec    0  8.03 MBytes
[ 7]  9.01-10.00 sec  1.47 GBytes  12.7 Gbits/sec   0  8.03 MBytes
-----
[ ID] Interval          Transfer     Bitrate      Retr
[ 7]  0.00-10.00  sec  14.5 GBytes  12.5 Gbits/sec  0
[ 7]  0.00-10.01  sec  14.5 GBytes  12.5 Gbits/sec

iperf Done.
root@mininet-vm:/home/mininet# ^C
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root tbf rate 10gb
it burst 5000000 limit 15000000
root@mininet-vm:/home/mininet# egrep '^CONFIG_HZ_[0-9]+' /boot/config-'uname -r'
grep: /boot/config-uname -r: No such file or directory
root@mininet-vm:/home/mininet# egrep '^CONFIG_HZ_[0-9]+' /boot/config-'uname -r'
CONFIG_HZ_250=y
root@mininet-vm:/home/mininet# █

```

Рис. 9: Установка значения всплеска при ограничении скорости для фильтра *tbf*

С помощью iPerf3 проверим, что значение пропускной способности изменилось.

В терминале хоста h2 запустим iPerf3 в режиме сервера (рис. 10):

```

X "host: h2"@mininet-vm
[ 7] 2.00-3.00 sec 1.44 GBytes 12.4 Gbits/sec
[ 7] 3.00-4.00 sec 1.38 GBytes 11.8 Gbits/sec
[ 7] 4.00-5.00 sec 1.50 GBytes 12.9 Gbits/sec
[ 7] 5.00-6.00 sec 1.44 GBytes 12.3 Gbits/sec
[ 7] 6.00-7.00 sec 1.49 GBytes 12.8 Gbits/sec
[ 7] 7.00-8.00 sec 1.52 GBytes 13.1 Gbits/sec
[ 7] 8.00-9.00 sec 1.52 GBytes 13.0 Gbits/sec
[ 7] 9.00-10.00 sec 1.48 GBytes 12.7 Gbits/sec
-----
[ ID] Interval Transfer Bitrate
[ 7] 0.00-10.01 sec 14.5 GBytes 12.5 Gbits/sec receiver
-----
Server listening on 5201
-----
^Ciperf3: interrupt - the server has terminated
root@mininet-vm:/home/mininet# egrep '^CONFIG_HZ_[0-9]+' /boot/config-'uname -r'
'
grep: /boot/config-uname -r: No such file or directory
root@mininet-vm:/home/mininet# iperf3 -s
warning: this system does not seem to support IPv6 - trying IPv4
-----
Server listening on 5201
-----

```

Рис. 10: Запуск iPerf3 в режиме сервера на хосте h2

В терминале хоста h1 запустим iPerf3 в режиме клиента (рис. 11):

```

X "host: h1"@mininet-vm
grep: /boot/config-uname -r: No such file or directory
root@mininet-vm:/home/mininet# egrep '^CONFIG_HZ_[0-9]+' /boot/config-'uname -r'
CONFIG_HZ_250=y
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 42020 connected to 10.0.0.2 port 5201
[ ID] Interval Transfer Bitrate Retr Cwnd
[ 7] 0.00-1.00 sec 1.12 GBytes 9.54 Gbits/sec 0 8.25 MBytes
[ 7] 1.00-2.00 sec 1.11 GBytes 9.57 Gbits/sec 0 8.25 MBytes
[ 7] 2.00-3.00 sec 1.08 GBytes 9.30 Gbits/sec 0 8.25 MBytes
[ 7] 3.00-4.00 sec 1.03 GBytes 8.88 Gbits/sec 0 8.25 MBytes
[ 7] 4.00-5.00 sec 1.05 GBytes 9.01 Gbits/sec 0 8.25 MBytes
[ 7] 5.00-6.00 sec 1.04 GBytes 8.96 Gbits/sec 0 8.25 MBytes
[ 7] 6.00-7.00 sec 1.06 GBytes 9.10 Gbits/sec 0 8.25 MBytes
[ 7] 7.00-8.00 sec 1.08 GBytes 9.31 Gbits/sec 0 8.25 MBytes
[ 7] 8.00-9.00 sec 1.05 GBytes 8.97 Gbits/sec 0 8.25 MBytes
[ 7] 9.00-10.00 sec 1.08 GBytes 9.30 Gbits/sec 0 8.25 MBytes
-----
[ ID] Interval Transfer Bitrate Retr
[ 7] 0.00-10.00 sec 10.7 GBytes 9.19 Gbits/sec 0
[ 7] 0.00-10.02 sec 10.7 GBytes 9.18 Gbits/sec
sender
receiver
iperf Done.
root@mininet-vm:/home/mininet# 
```

Рис. 11: Запуск iPerf3 в режиме клиента на хосте h1

После завершения работы iPerf3 на хосте h1 остановим iPerf3 на хосте h2, нажав Ctrl + c (рис. 12):

```
X "host: h2"@mininet-vm
-----
Server listening on 5201
-----
Accepted connection from 10.0.0.1, port 42018
[ 7] local 10.0.0.2 port 5201 connected to 10.0.0.1 port 42020
[ ID] Interval          Transfer     Bitrate
[ 7]  0.00-1.00   sec  1.10 GBytes  9.45 Gbits/sec
[ 7]  1.00-2.00   sec  1.11 GBytes  9.55 Gbits/sec
[ 7]  2.00-3.00   sec  1.09 GBytes  9.36 Gbits/sec
[ 7]  3.00-4.01   sec  1.03 GBytes  8.77 Gbits/sec
[ 7]  4.01-5.00   sec  1.05 GBytes  9.09 Gbits/sec
[ 7]  5.00-6.01   sec  1.04 GBytes  8.90 Gbits/sec
[ 7]  6.01-7.01   sec  1.06 GBytes  9.07 Gbits/sec
[ 7]  7.01-8.00   sec  1.10 GBytes  9.50 Gbits/sec
[ 7]  8.00-9.00   sec  1.05 GBytes  8.99 Gbits/sec
[ 7]  9.00-10.01  sec  1.06 GBytes  9.05 Gbits/sec
[ 7] 10.01-10.02  sec 11.3 MBytes 13.8 Gbits/sec
[ ID] Interval          Transfer     Bitrate
[ 7]  0.00-10.02  sec 10.7 GBytes 9.18 Gbits/sec
-----                                         receiver
Server listening on 5201
-----
```

Рис. 12: Остановка iPerf3

Удалим модифицированную конфигурацию на хосте h1 (рис. 13):

```

X "host h1"@mininet-vm
root@mininet-vm:/home/mininet# egrep '^CONFIG_HZ_[0-9]+' /boot/config-`uname -r` 
CONFIG_HZ_250=y
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 42020 connected to 10.0.0.2 port 5201
[ ID] Interval           Transfer     Bitrate      Retr  Cwnd
[ 7]  0.00-1.00   sec  1.12 GBytes  9.54 Gbits/sec    0  8.25 MBytes
[ 7]  1.00-2.00   sec  1.11 GBytes  9.57 Gbits/sec    0  8.25 MBytes
[ 7]  2.00-3.00   sec  1.08 GBytes  9.30 Gbits/sec    0  8.25 MBytes
[ 7]  3.00-4.00   sec  1.03 GBytes  8.88 Gbits/sec    0  8.25 MBytes
[ 7]  4.00-5.00   sec  1.05 GBytes  9.01 Gbits/sec    0  8.25 MBytes
[ 7]  5.00-6.00   sec  1.04 GBytes  8.96 Gbits/sec    0  8.25 MBytes
[ 7]  6.00-7.00   sec  1.06 GBytes  9.10 Gbits/sec    0  8.25 MBytes
[ 7]  7.00-8.00   sec  1.08 GBytes  9.31 Gbits/sec    0  8.25 MBytes
[ 7]  8.00-9.00   sec  1.05 GBytes  8.97 Gbits/sec    0  8.25 MBytes
[ 7]  9.00-10.00  sec  1.08 GBytes  9.30 Gbits/sec   0  8.25 MBytes
[ 7]  0.00-10.00  sec  10.7 GBytes  9.19 Gbits/sec    0
[ 7]  0.00-10.02  sec  10.7 GBytes  9.18 Gbits/sec

iperf Done.
root@mininet-vm:/home/mininet# sudo tc qdisc del dev h1-eth0 root
root@mininet-vm:/home/mininet#

```

Рис. 13: Удаление модифицированной конфигурации на хосте h1

Применим правило ограничения скорости tbf с параметрами rate = 10gbit, burst = 5,000,000, limit= 15,000,000 к интерфейсу s1-eth2 коммутатора s1, который соединяет его с коммутатором s2 (рис. 14):

```
X "switch: s1" (root)@mininet-vm
-
s1-eth2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    ether aa:78:c6:b5:bc:51 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

s2-eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    ether 62:37:5f:4d:86:17 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

s2-eth2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    ether 7e:de:87:d5:44:a6 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

root@mininet-vm:/home/mininet# sudo tc qdisc add dev s1-eth2 root tbf rate 10gb
it burst 5000000 limit 15000000
root@mininet-vm:/home/mininet#
```

Рис. 14: Применение правила ограничения скорости *tbf*

Проверим конфигурацию с помощью инструмента iperf3 для измерения пропускной способности.

В терминале хоста h2 запустим iPerf3 в режиме сервера (рис. 15):

```

X "host: h2"@mininet-vm
[ 7] 0.00-1.00    sec 1.10 GBytes 9.45 Gbits/sec
[ 7] 1.00-2.00    sec 1.11 GBytes 9.55 Gbits/sec
[ 7] 2.00-3.00    sec 1.09 GBytes 9.36 Gbits/sec
[ 7] 3.00-4.01    sec 1.03 GBytes 8.77 Gbits/sec
[ 7] 4.01-5.00    sec 1.05 GBytes 9.09 Gbits/sec
[ 7] 5.00-6.01    sec 1.04 GBytes 8.90 Gbits/sec
[ 7] 6.01-7.01    sec 1.06 GBytes 9.07 Gbits/sec
[ 7] 7.01-8.00    sec 1.10 GBytes 9.50 Gbits/sec
[ 7] 8.00-9.00    sec 1.05 GBytes 8.99 Gbits/sec
[ 7] 9.00-10.01   sec 1.06 GBytes 9.05 Gbits/sec
[ 7] 10.01-10.02  sec 11.3 MBytes 13.8 Gbits/sec
-----
[ ID] Interval      Transfer     Bitrate
[ 7] 0.00-10.02  sec 10.7 GBytes 9.18 Gbits/sec           receiver
-----
Server listening on 5201
-----
^Ciperf3: interrupt - the server has terminated
root@mininet-vm:/home/mininet# iperf3 -s
warning: this system does not seem to support IPv6 - trying IPv4
-----
Server listening on 5201
-----

```

Рис. 15: Запуск iPerf3 в режиме сервера на хосте h2

В терминале хоста h1 запустим iPerf3 в режиме клиента (рис. 16):

```

X "host: h1"@mininet-vm
Cannot find device "s1-eth2"
root@mininet-vm:/home/mininet# sudo tc qdisc add dev s1-eth2 root tbf rate 10gbps burst 5000000 limit 15000000^C
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 42024 connected to 10.0.0.2 port 5201
[ ID] Interval      Transfer     Bitrate      Retr  Cwnd
[ 7] 0.00-1.00    sec 1.07 GBytes 9.17 Gbits/sec  27  19.8 MBytes
[ 7] 1.00-2.00    sec 1.09 GBytes 9.37 Gbits/sec  0  19.8 MBytes
[ 7] 2.00-3.00    sec 1.11 GBytes 9.50 Gbits/sec  0  19.8 MBytes
[ 7] 3.00-4.00    sec 1.11 GBytes 9.49 Gbits/sec  0  19.8 MBytes
[ 7] 4.00-5.00    sec 1.10 GBytes 9.51 Gbits/sec  0  19.8 MBytes
[ 7] 5.00-6.00    sec 1.11 GBytes 9.51 Gbits/sec  0  19.8 MBytes
[ 7] 6.00-7.00    sec 1.10 GBytes 9.48 Gbits/sec  0  19.8 MBytes
[ 7] 7.00-8.00    sec 1.11 GBytes 9.52 Gbits/sec  0  19.8 MBytes
[ 7] 8.00-9.00    sec 1.11 GBytes 9.57 Gbits/sec  0  19.8 MBytes
[ 7] 9.00-10.00   sec 1.11 GBytes 9.53 Gbits/sec  0  19.8 MBytes
-----
[ ID] Interval      Transfer     Bitrate      Retr
[ 7] 0.00-10.00   sec 11.0 GBytes 9.46 Gbits/sec  27           sender
[ 7] 0.00-10.02   sec 11.0 GBytes 9.44 Gbits/sec           receiver
-----
iperf Done.
root@mininet-vm:/home/mininet# █

```

Рис. 16: Запуск iPerf3 в режиме клиента на хосте h1

После завершения работы iPerf3 на хосте h1 остановим iPerf3 на хосте h2, нажав Ctrl + c (рис. 17):

```
X "host: h2"@mininet-vm
-----
Server listening on 5201
-----
Accepted connection from 10.0.0.1, port 42022
[ 7] local 10.0.0.2 port 5201 connected to 10.0.0.1 port 42024
[ ID] Interval          Transfer     Bitrate
[ 7]  0.00-1.00   sec  1.05 GBytes  9.02 Gbits/sec
[ 7]  1.00-2.00   sec  1.10 GBytes  9.39 Gbits/sec
[ 7]  2.00-3.00   sec  1.11 GBytes  9.53 Gbits/sec
[ 7]  3.00-4.00   sec  1.11 GBytes  9.51 Gbits/sec
[ 7]  4.00-5.00   sec  1.10 GBytes  9.42 Gbits/sec
[ 7]  5.00-6.00   sec  1.11 GBytes  9.58 Gbits/sec
[ 7]  6.00-7.01   sec  1.10 GBytes  9.43 Gbits/sec
[ 7]  7.01-8.00   sec  1.10 GBytes  9.50 Gbits/sec
[ 7]  8.00-9.00   sec  1.12 GBytes  9.56 Gbits/sec
[ 7]  9.00-10.01  sec  1.11 GBytes  9.47 Gbits/sec
[ 7] 10.01-10.02  sec  7.95 MBytes  18.1 Gbits/sec
[ ID] Interval          Transfer     Bitrate
[ 7]  0.00-10.02  sec  11.0 GBytes  9.44 Gbits/sec
----- receiver
Server listening on 5201
-----
```

Рис. 17: Остановка iPerf3

Удалим модифицированную конфигурацию на коммутаторе s1 (рис. 18):

```

X "switch: s1" (root)@mininet-vm
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

s2-eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
ether 62:37:5f:4d:86:17 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

s2-eth2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
ether 7e:de:87:d5:44:a6 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

root@mininet-vm:/home/mininet# sudo tc qdisc add dev s1-eth2 root tbf rate 10gb
it burst 5000000 limit 15000000
root@mininet-vm:/home/mininet# sudo tc qdisc add dev s1-eth2 root
Error: Exclusivity flag on, cannot modify.
root@mininet-vm:/home/mininet# sudo tc qdisc del dev s1-eth2 root
root@mininet-vm:/home/mininet#

```

Рис. 18: Удаление модифицированной конфигурации на коммутаторе s1

NETEM используется для изменения задержки, джиттера, повреждения пакетов и т.д. TBF может использоваться для ограничения скорости. Утилита tc позволяет комбинировать несколько модулей. При этом первая дисциплина очереди (qdisc1) присоединяется к корневой метке, последующие дисциплины очереди можно прикрепить к своим родителям, указав правильную метку.

Объединим NETEM и TBF, введя на интерфейсе s1-eth2 коммутатора s1 задержку, джиттер, повреждение пакетов и указав скорость (рис. 19):

```

X "switch: s1" (root)@mininet-vm
root@mininet-vm:/home/mininet# sudo tc qdisc add dev s1-eth2 root handle 1: natem
delay 10ms

```

Рис. 19: Объединение NETEM и TBF

Убедимся, что соединение от хоста h1 к хосту h2 имеет заданную задержку. Для этого запустим команду ping с параметром -c 4 с терминала хоста h1 (рис. 20):

```
X "host: h1"@mininet-vm
root@mininet-vm:/home/mininet# ping -c 4 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=5.22 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.839 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.092 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.130 ms

--- 10.0.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3010ms
rtt min/avg/max/mdev = 0.092/1.569/5.218/2.127 ms
root@mininet-vm:/home/mininet#
```

Рис. 20: Проверка задержки

Добавим второе правило на коммутаторе s1, которое задаёт ограничение скорости с помощью tbf с параметрами rate=2gbit, burst=1,000,000, limit=2,000,000 (рис. 21):

```
root@mininet-vm:/home/mininet# sudo tc qdisc add dev s1-eth2 parent 1: handle 2:
tbf rate 2gbit burst 1000000 limit 2000000
root@mininet-vm:/home/mininet#
```

Рис. 21: Добавление второго правила на коммутаторе s1

Проверим конфигурацию с помощью инструмента iperf3 для измерения пропускной способности.

В терминале хоста h2 запустим iPerf3 в режиме сервера (рис. 22):

```
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
-----
^Ciperf3: interrupt - the server has terminated
root@mininet-vm:/home/mininet# iperf3 -s
warning: this system does not seem to support IPv6 - trying IPv4
-----
Server listening on 5201
-----
```

Рис. 22: Запуск iPerf3 в режиме сервера на хосте h2

В терминале хоста h1 запустим iPerf3 в режиме клиента (рис. 23):

```
X "host: h1"@mininet-vm
--- 10.0.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3010ms
rtt min/avg/max/mdev = 0.092/1.569/5.218/2.127 ms
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 42048 connected to 10.0.0.2 port 5201
[ ID] Interval           Transfer     Bitrate      Retr  Cwnd
[ 7]  0.00-1.01   sec    210 MBytes   1.75 Gbits/sec  1822  2.93 MBytes
[ 7]  1.01-2.00   sec    194 MBytes   1.63 Gbits/sec   90  2.17 MBytes
[ 7]  2.00-3.00   sec    192 MBytes   1.62 Gbits/sec    0  2.28 MBytes
[ 7]  3.00-4.00   sec    136 MBytes   1.14 Gbits/sec   90  1.70 MBytes
[ 7]  4.00-5.00   sec    146 MBytes   1.23 Gbits/sec    0  1.79 MBytes
[ 7]  5.00-6.00   sec    158 MBytes   1.32 Gbits/sec    0  1.85 MBytes
[ 7]  6.00-7.00   sec    161 MBytes   1.35 Gbits/sec    0  1.89 MBytes
[ 7]  7.00-8.00   sec    111 MBytes   934 Mbites/sec   21  1.41 MBytes
[ 7]  8.00-9.00   sec    119 MBytes   997 Mbites/sec    0  1.48 MBytes
[ 7]  9.00-10.00  sec    121 MBytes   1.02 Gbits/sec    0  1.54 MBytes
[ 7]  10.00-10.02 sec    1.51 GBytes  1.30 Gbits/sec  2023
                                         sender
                                         receiver

iperf Done.
root@mininet-vm:/home/mininet#
```

Рис. 23: Запуск iPerf3 в режиме клиента на хосте h1

После завершения работы iPerf3 на хосте h1 остановим iPerf3 на хосте h2, нажав Ctrl + c (рис. 24):

```
X "host: h2"@mininet-vm
-----
Server listening on 5201
-----
Accepted connection from 10.0.0.1, port 42046
[ 7] local 10.0.0.2 port 5201 connected to 10.0.0.1 port 42046
[ ID] Interval Transfer Bitrate
[ 7] 0.00-1.01 sec 198 MBytes 1.64 Gbits/sec
[ 7] 1.01-2.00 sec 194 MBytes 1.64 Gbits/sec
[ 7] 2.00-3.00 sec 193 MBytes 1.62 Gbits/sec
[ 7] 3.00-4.00 sec 137 MBytes 1.14 Gbits/sec
[ 7] 4.00-5.00 sec 145 MBytes 1.21 Gbits/sec
[ 7] 5.00-6.00 sec 159 MBytes 1.33 Gbits/sec
[ 7] 6.00-7.01 sec 161 MBytes 1.34 Gbits/sec
[ 7] 7.01-8.00 sec 111 MBytes 935 Mbytes/sec
[ 7] 8.00-9.01 sec 119 MBytes 993 Mbytes/sec
[ 7] 9.01-10.00 sec 121 MBytes 1.02 Gbits/sec
[ 7] 10.00-10.02 sec 128 KBytes 53.5 Mbytes/sec
-----
[ ID] Interval Transfer Bitrate
[ 7] 0.00-10.02 sec 1.50 GBytes 1.29 Gbits/sec
----- receiver
-----
Server listening on 5201
-----

```

Рис. 24: Остановка iPerf3

Удалим модифицированную конфигурацию на коммутаторе s1 (рис. 25):

```
X "switch: s1" (root)@mininet-vm
-----
root@mininet-vm:/home/mininet# sudo tc qdisc add dev s1-eth2 parent 1: handle 2: tbf rate 2gbit burst 1000000 limit 2000000
Error: Failed to find specified qdisc.
root@mininet-vm:/home/mininet# sudo tc qdisc add dev s1-eth2 parent 1: handle 2: tbf rate 2gbit burst 1000000 limit 2000000
Error: Failed to find specified qdisc.
root@mininet-vm:/home/mininet# ^C
root@mininet-vm:/home/mininet# sudo tc qdisc add dev s1-eth2 root handle 1: netem delay 10ms
root@mininet-vm:/home/mininet# sudo tc qdisc show dev s1-eth2
qdisc netem 1: root refcnt 2 limit 1000 delay 10.0ms
root@mininet-vm:/home/mininet# sudo tc qdisc add dev s1-eth2 parent 1: handle 2: tbf rate 2gbit burst 1000000 limit 2000000
root@mininet-vm:/home/mininet# sudo tc qdisc del dev s1-eth2 root
root@mininet-vm:/home/mininet#
```

Рис. 25: Удаление модифицированной конфигурации на коммутаторе s1

Для самостоятельного задания создадим необходимые каталоги (рис. 26):

```
mininet@mininet-vm:~$ ls
mininet  mininet.orig  oflops  oftest  openflow  pox  work
mininet@mininet-vm:~$ cd work/
mininet@mininet-vm:~/work$ ls
lab_iperf3  lab_netem_i  lab_netem_ii  lesson1.mn
mininet@mininet-vm:~/work$ mkdir lab6
mininet@mininet-vm:~/work$ ls
lab6  lab_iperf3  lab_netem_i  lab_netem_ii  lesson1.mn
mininet@mininet-vm:~/work$ cd lab6
mininet@mininet-vm:~/work/lab6$ mkdir expl
mininet@mininet-vm:~/work/lab6$ mkdir exp2
mininet@mininet-vm:~/work/lab6$ ls
expl  exp2
mininet@mininet-vm:~/work/lab6$
```

Рис. 26: Создание необходимых каталогов

Затем напишем скрипты по примеру из прошлых лабораторных работ (рис. 27 - рис. 29):

```
mininet@mininet-vm: ~/work/lab6/exp1
GNU nano 4.8
#!/usr/bin/env python

"""
This example shows how to create an empty Mininet object
(without a topology object) and add nodes to it manually.
"""

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

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' )
    s2 = net.addSwitch( 's2' )

    s1.cmd('ip link del s1-eth2')
    s2.cmd('ip link del s2-eth1')

    info( '*** Creating links\n' )
    net.addLink( h1, s1 )
    net.addLink( h2, s1 )
    net.addLink( s1, s2 )

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

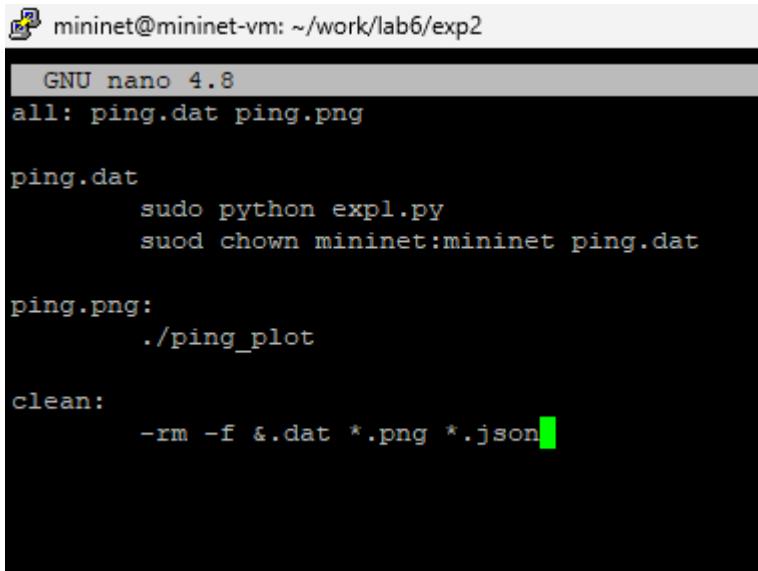
    s1.cmdPrint('sudo tc qdisc add dev s1-eth2 root handle 1: netem delay 10ms')
    s2.cmdPrint('sudo tc qdisc add dev s1-eth2 parent 1: handle 2: tbf rate 2gbit burst 1000000 limit 2000000')

    info( '*** Running CLI\n' )
    h2.cmdPrint('iperf3 -s -D -l')
    time.sleep(10) # Wait 10 seconds
    h1.cmdPrint('iperf3 -s -D -l')
    h1.cmdPrint('iperf3 -s -D -l')

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

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

Рис. 27: Написание 1 скрипта



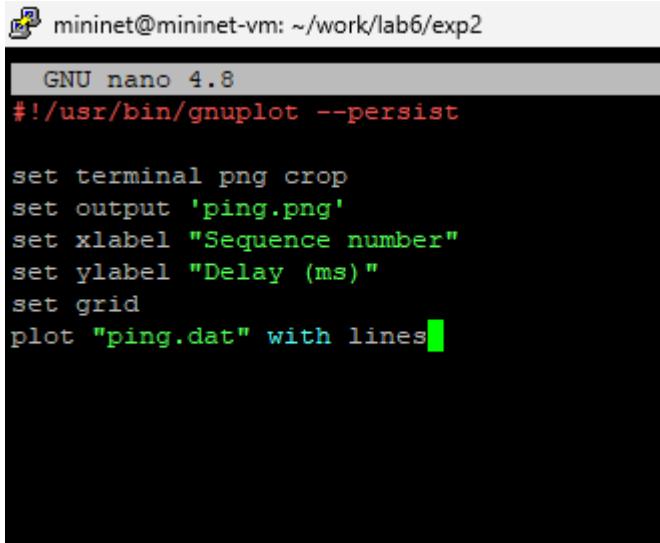
```
mininet@mininet-vm: ~/work/lab6/exp2
GNU nano 4.8
all: ping.dat ping.png

ping.dat
    sudo python expl.py
    suod chown mininet:mininet ping.dat

ping.png:
    ./ping_plot

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

Рис. 28: Написание 2 скрипта



```
mininet@mininet-vm: ~/work/lab6/exp2
GNU nano 4.8
#!/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. 29: Написание 3 скрипта

Запустим на выполнение скрипты для первой части самостоятельного задания (рис. 30):

```
mininet@mininet-vm:~/work/lab6/exp1$ make
sudo python lab6_exp1.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
*** Starting network
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 2 switches
s1 s2 ...
*** Waiting for switches to connect
s1 s2
*** Set delay
*** s1 : ('tc qdisc add dev s1-eth2 root handle 1: netem delay 10ms',)
*** s2 : ('tc qdisc add dev s1-eth2 parent 1: handle 2: tbf rate 2gbit burst 1000000 limit 2000000',)
*** Traffic generation
*** h2 : ('iperf3 -s -D -l',)
*** h1 : ('iperf3 -c', '10.0.0.2', '-J > iperf_result.json')
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "time=' | awk \'(print $5, $7)\' | sed -e \'s/time=/g\' -e \'s/icmp_seq=/g\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 3 links
...
*** Stopping 2 switches
s1 s2
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
./ping_plot
mininet@mininet-vm:~/work/lab6/exp1$ ls
iperf_result.json lab6_exp1.py Makefile ping.dat ping_plot ping.png
mininet@mininet-vm:~/work/lab6/exp1$
```

Рис. 30: Выполнение скриптов

Изменим параметры в скрипте для первого задания и запустим на выполнение (рис. 31 - рис. 32):

```

mininet@mininet-vm: ~/work/lab6/exp1
/home/mininet/work/lab6/exp1/lab6_exp1.py [-M--] 92 L:[ 12+32 44/ 58 ] *(1029/1469b) 39 0x027
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' )
    s2 = net.addSwitch( 's2' )

    s1.cmd('ip link del s1-eth2')
    s2.cmd('ip link del s2-eth1')

    info( "*** Creating links\n" )
    net.addLink( h1, s1 )
    net.addLink( h2, s1 )
    net.addLink( s1, s2 )

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

    info( "*** Set delay\n" )
    s1.cmdPrint( 'tc qdisc add dev h1-eth0 root tbf rate 10gbit burst 5000000 limit 15000000' )

    info( "*** Traffic generation\n" )
    h2.cmdPrint( 'iperf3 -s -D -1' )
    time.sleep(10) # Wait 10 seconds
    h1.cmdPrint( 'iperf3 -c', h2.IP(), '-J > iperf_result.json' )
    h1.cmdPrint('ping -c 100', h2.IP(), '| grep "time=" | awk \'(print $5, $7)\' | sed -e \'$s/time=/g\' -e \'$s/icmp_seq=/g\' > ping.dat')

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

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

```

Рис. 31: Изменение параметров

```

mininet@mininet-vm: ~/work/lab6/exp1
mininet@mininet-vm:~/work/lab6/exp1$ make
sudo python lab6_exp1.py
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
*** Starting network
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 2 switches
s1 s2 ...
*** Waiting for switches to connect
s1 s2
*** Set delay
*** s1 : ('tc qdisc add dev h1-eth0 root tbf rate 10gbit burst 5000000 limit 15000000',)
Cannot find device "h1-eth0"
*** Traffic generation
*** h2 : ('iperf3 -s -D -1',)
*** h1 : ('iperf3 -c', '10.0.0.2', '-J > iperf_result.json')
*** h1 : ('ping -c 100', '10.0.0.2', '| grep "time=" | awk \'(print $5, $7)\' | sed -e \'$s/time=/g\' -e \'$s/icmp_seq=/g\' > ping.dat')
*** Stopping network*** Stopping 1 controllers
c0
*** Stopping 3 links
...
*** Stopping 2 switches
s1 s2
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
./ping_plot
mininet@mininet-vm:~/work/lab6/exp1$ ls
iperf_result.json lab6_exp1.py Makefile ping.dat ping_plot ping.png
mininet@mininet-vm:~/work/lab6/exp1$ 

```

Рис. 32: Выполнение скриптов

Просмотрим полученные графики (рис. 33 - рис. 34):

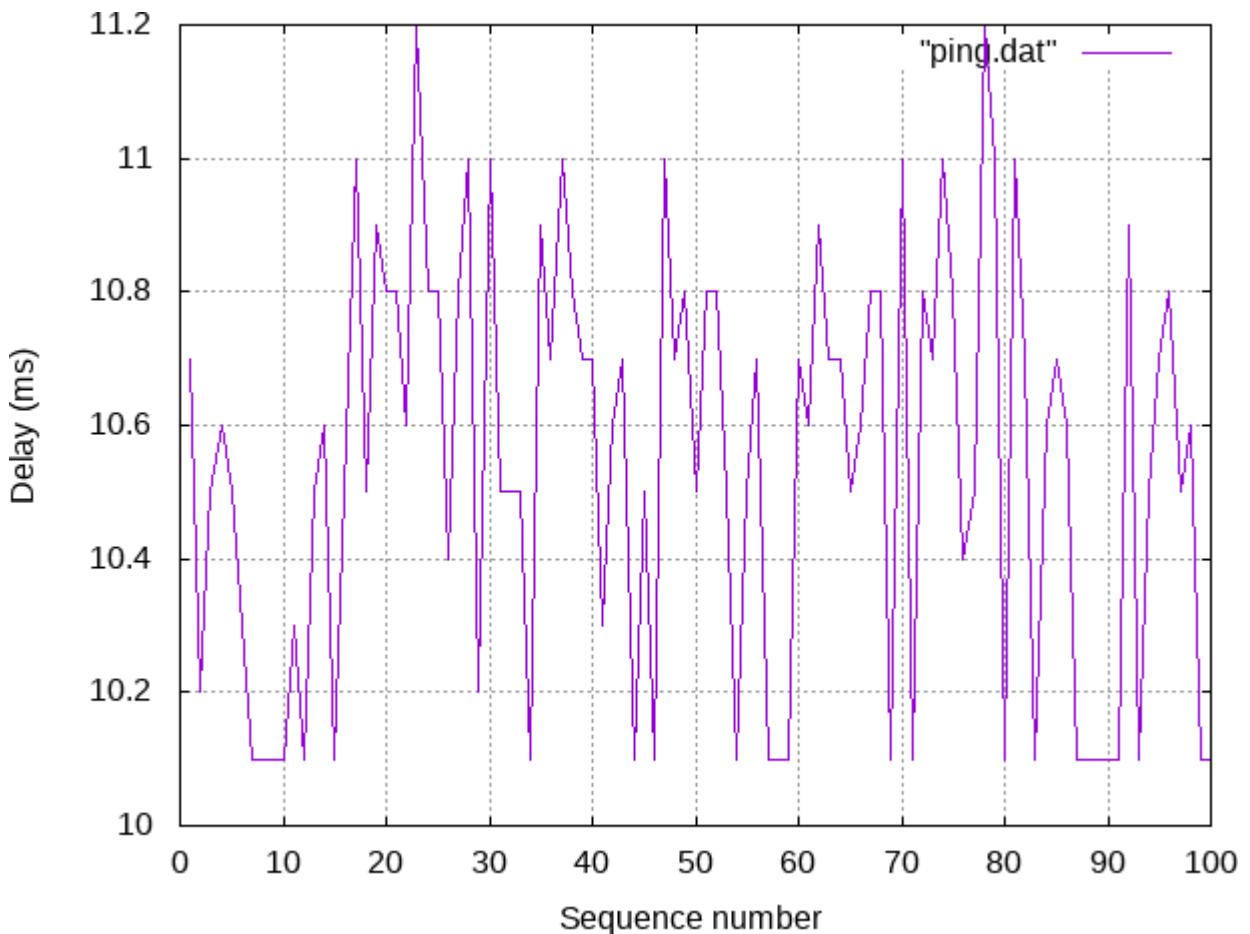


Рис. 33: График №1

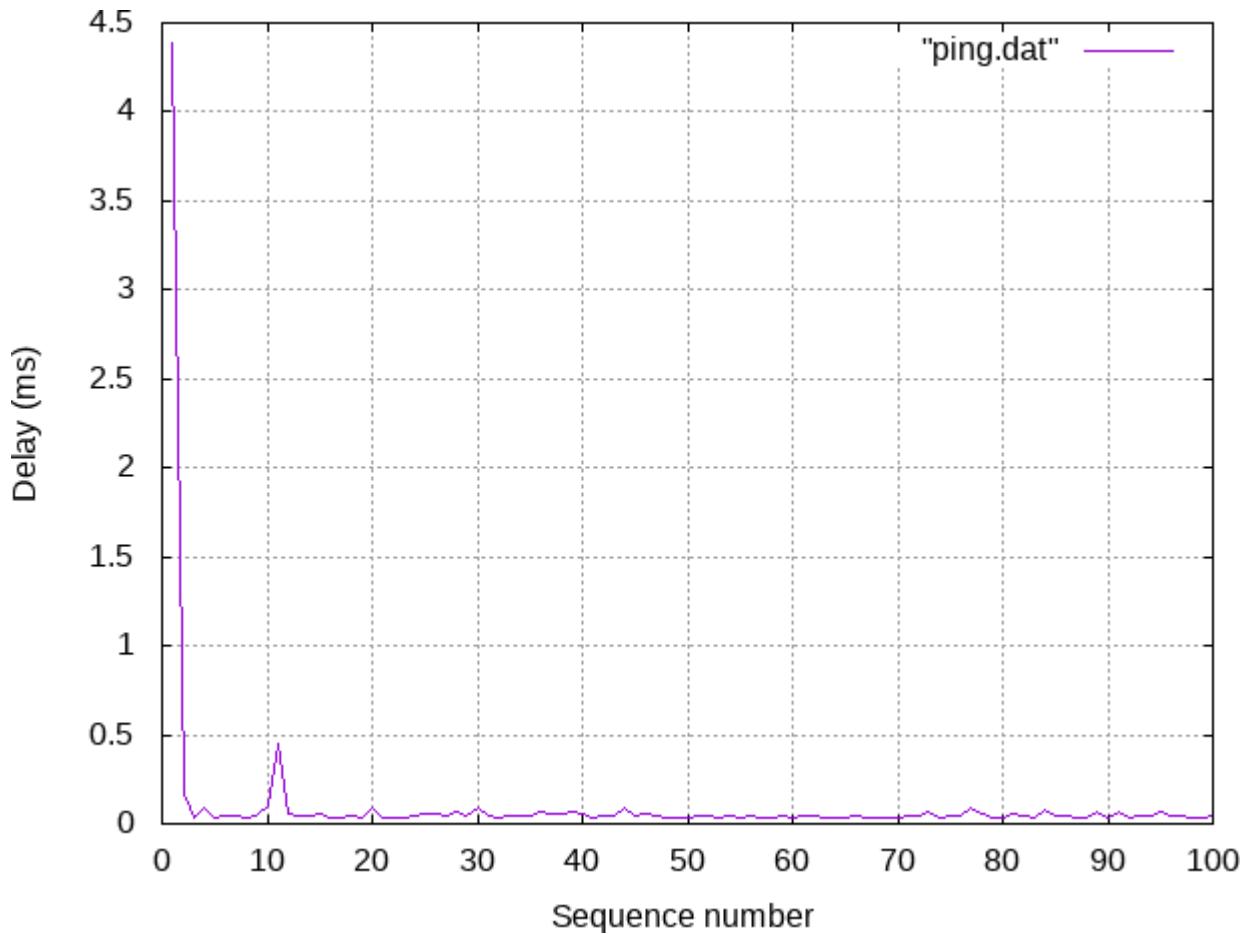


Рис. 34: График №2

3 Вывод

В ходе выполнения лабораторной работы познакомились с принципами работы дисциплины очереди Token Bucket Filter, которая формирует входящий/исходящий трафик для ограничения пропускной способности, а также получили навыки моделирования и исследования поведения трафика посредством проведения интерактивного и воспроизводимого экспериментов в Mininet.

4 Список литературы. Библиография

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