

# **Отчёт по лабораторной работе №6 по дисциплине Моделирование сетей передачи данных**

**Настройка пропускной способности глобальной сети с помощью  
Token Bucket Filter**

Шаповалова Диана Дмитриевна, НПИбд-02-21, 1032211220

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение работы</b>	<b>6</b>
2.1	Запуск лабораторной топологии . . . . .	6
2.2	Интерактивные эксперименты . . . . .	6
2.3	Ограничение скорости на конечных хостах . . . . .	6
2.4	Ограничение скорости на коммутаторах . . . . .	8
2.5	Объединение NETEM и TBF . . . . .	9
2.6	Воспроизводимые эксперименты . . . . .	11
<b>3</b>	<b>Выводы</b>	<b>16</b>
<b>4</b>	<b>Список литературы</b>	<b>17</b>

# Список иллюстраций

2.1	Ограничение скорости на конечных хостах . . . . .	8
2.2	Ограничение скорости на коммутаторах . . . . .	9
2.3	ping с параметром -c 4 с терминала хоста h1 . . . . .	10
2.4	Объединение NETEM и TBF . . . . .	11
2.5	Каталоги для экспериментов . . . . .	11
2.6	Объединение NETEM и TBF . . . . .	12
2.7	Объединение NETEM и TBF . . . . .	13
2.8	Ограничение скорости на коммутаторах . . . . .	14
2.9	Ограничение скорости на коммутаторах . . . . .	15

## **Список таблиц**

# 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 работы

### 2.1 Запуск лабораторной топологии

1. Запустите виртуальную среду с mininet.
2. Из основной ОС подключитесь к виртуальной машине:
3. В виртуальной машине mininet при необходимости исправьте права запуска X-соединения. Скопируйте значение куки (MIT magic cookie)<sup>1</sup> своего пользователя mininet в файл для пользователя root:

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

4. Задайте топологию сети, состоящую из двух хостов и двух коммутаторов с назначенной по умолчанию mininet сетью 10.0.0.0/8:

```
sudo mn -topo=linear,2 -x
```

### 2.2 Интерактивные эксперименты

### 2.3 Ограничение скорости на конечных хостах

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).

1. Измените пропускную способность хоста h1, установив пропускную способность на 10 Гбит/с на интерфейсе h1-eth0 и параметры TBF-фильтра:

```
sudo tc qdisc add dev h1-eth0 root tbf rate 10gbit burst 5000000 limit 15000000
```

Здесь:

- sudo: включить выполнение команды с более высокими привилегиями без-опасности;
- tc: вызвать управление трафиком Linux;
- qdisc: изменить дисциплину очередей сетевого планировщика;
- add (добавить): создать новое правило;
- dev h1-eth0 root: интерфейс, на котором будет применяться правило;
- tbf: использовать алгоритм Token Bucket Filter;
- rate: указать скорость передачи (10 Гбит/с);
- burst: количество байтов, которое может поместиться в корзину (5 000 000);
- limit: размер очереди в байтах (15 000 000).

С помощью iPerf3 проверьте, что значение пропускной способности изменилось:

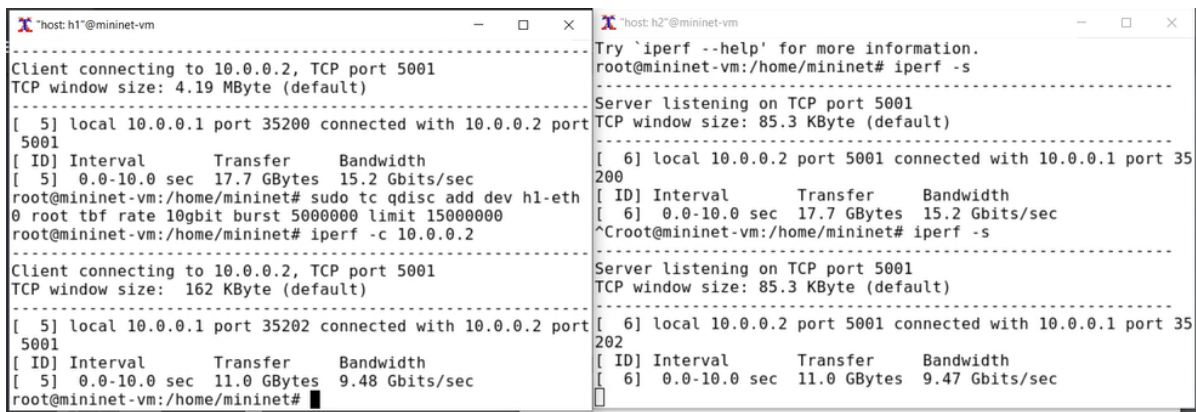
- В терминале хоста h2 запустите iPerf3 в режиме сервера:

```
iperf3 -s
```

- В терминале хоста h2 запустите iPerf3 в режиме клиента:

```
iperf3 -c 10.0.0.2
```

- После завершения работы iPerf3 на хосте h1 остановите iPerf3 на хосте h2, нажав Ctrl + c . В отчёте зафиксируйте результат отработки iPerf3 на данном этапе проведения эксперимента.



```
host: h1@mininet-vm
Client connecting to 10.0.0.2, TCP port 5001
TCP window size: 4.19 MByte (default)
[ 5] local 10.0.0.1 port 35200 connected with 10.0.0.2 port 5001
[ ID] Interval      Transfer      Bandwidth
[ 5]  0.0-10.0 sec  17.7 GBytes  15.2 Gbits/sec
root@mininet-vm:/home/mininet# sudo tc qdisc add dev h1-eth0 root tbf rate 10gbit burst 5000000 limit 15000000
root@mininet-vm:/home/mininet# iperf -c 10.0.0.2
Client connecting to 10.0.0.2, TCP port 5001
TCP window size: 162 KByte (default)
[ 5] local 10.0.0.1 port 35202 connected with 10.0.0.2 port 5001
[ ID] Interval      Transfer      Bandwidth
[ 5]  0.0-10.0 sec  11.0 GBytes  9.48 Gbits/sec
root@mininet-vm:/home/mininet#

host: h2@mininet-vm
Try 'iperf --help' for more information.
root@mininet-vm:/home/mininet# iperf -s
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
[ 6] local 10.0.0.2 port 5001 connected with 10.0.0.1 port 35200
[ ID] Interval      Transfer      Bandwidth
[ 6]  0.0-10.0 sec  17.7 GBytes  15.2 Gbits/sec
^Croot@mininet-vm:/home/mininet# iperf -s
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
[ 6] local 10.0.0.2 port 5001 connected with 10.0.0.1 port 35202
[ ID] Interval      Transfer      Bandwidth
[ 6]  0.0-10.0 sec  11.0 GBytes  9.47 Gbits/sec
```

Рис. 2.1: Ограничение скорости на конечных хостах

## 2.4 Ограничение скорости на коммутаторах

При ограничении скорости на интерфейсе s1-eth2 коммутатора s1 все сеансы связи между коммутатором s1 и коммутатором s2 будут фильтроваться в соответствии с применяемыми правилами.

1. Примените 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:

```
sudo tc qdisc add dev s1-eth2 root tbf rate 10gbit burst 5000000 limit 15000000
```

2. Проверьте конфигурацию с помощью инструмента iperf3 для измерения пропускной способности:

- В терминале хоста h2 запустите iPerf3 в режиме сервера:

```
iperf3 -s
```

- В терминале хоста h2 запустите iPerf3 в режиме клиента:

```
iperf3 -c 10.0.0.2
```

- После завершения работы iPerf3 на хосте h1 остановите iPerf3 на хосте h2, нажав Ctrl + c. В отчёте зафиксируйте результат отработки iPerf3 на данном этапе проведения эксперимента.



```

[ 5] local 10.0.0.1 port 35202 connected with 10.0.0.2 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 5] 0.0-10.0 sec  11.0 GBytes 9.48 Gbits/sec
root@mininet-virtual-machine:~# sudo tc qdisc del dev s1-eth2
Cannot find device "s1-eth2"
root@mininet-virtual-machine:~# sudo tc qdisc del dev h1-eth0
root@mininet-virtual-machine:~# iperf -c 10.0.0.2
Client connecting to 10.0.0.2, TCP port 5001
TCP window size: 570 KByte (default)
[ 5] local 10.0.0.1 port 35204 connected with 10.0.0.2 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 5] 0.0-10.0 sec  11.1 GBytes 9.50 Gbits/sec
root@mininet-virtual-machine:~#

[ 6] 0.0-10.0 sec  17.7 GBytes 15.2 Gbits/sec
^Croot@mininet-virtual-machine:~# iperf -s
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
[ 6] local 10.0.0.2 port 5001 connected with 10.0.0.1 port 35202
[ ID] Interval      Transfer    Bandwidth
[ 6] 0.0-10.0 sec  11.0 GBytes 9.47 Gbits/sec
^Croot@mininet-virtual-machine:~# iperf -s
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
[ 6] local 10.0.0.2 port 5001 connected with 10.0.0.1 port 35204
[ ID] Interval      Transfer    Bandwidth
[ 6] 0.0-10.0 sec  11.1 GBytes 9.49 Gbits/sec

```

Рис. 2.2: Ограничение скорости на коммутаторах

## 2.5 Объединение NETEM и TBF

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.

1. Объедините NETEM и TBF, введя на интерфейсе s1-eth2 коммутатора s1 задержку, джиттер, повреждение пакетов и указав скорость:

```
sudo tc qdisc add dev s1-eth2 root handle 1: netem delay 10ms
```

Здесь ключевое слово handle задаёт дескриптор подключения, имеющий смысл очередности подключения разных дисциплин qdisc.

2. Убедитесь, что соединение от хоста h1 к хосту h2 имеет заданную задержку. Для этого запустите команду ping с параметром -c 4 с терминала хоста h1.

```
host: h1" @mininet-vm
-----
Client connecting to 10.0.0.2, TCP port 5001
TCP window size: 570 KByte (default)
-----
[ 5] local 10.0.0.1 port 35204 connected with 10.0.0.2 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 5]  0.0-10.0 sec  11.1 GBytes 9.50 Gbits/sec
root@mininet-vm:/home/mininet# ping 10.0.0.2 -c 4
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=13.6 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=11.1 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=10.6 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=10.1 ms

--- 10.0.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005 ms
rtt min/avg/max/mdev = 10.108/11.335/13.553/1.326 ms
root@mininet-vm:/home/mininet#
```

Рис. 2.3: ping с параметром -c 4 с терминала хоста h1

4. Добавьте второе правило на коммутаторе s1, которое задаёт ограничение скорости с помощью tbf с параметрами rate=2gbit, burst=1,000,000, limit=2,000,000:

```
sudo tc qdisc add dev s1-eth2 parent 1: handle 2: tbf rate 2gbit burst 1000000 limit 2000000
```

5. Проверьте конфигурацию с помощью инструмента iperf3 для измерения пропускной способности:

– В терминале хоста h2 запустите iPerf3 в режиме сервера:

```
iperf3 -s
```

– В терминале хоста h2 запустите iPerf3 в режиме клиента:

```
iperf3 -c 10.0.0.2
```

– После завершения работы iPerf3 на хосте h1 остановите iPerf3 на хосте h2, нажав Ctrl + c . В отчёте зафиксируйте результат отработки iPerf3 на данном этапе проведения эксперимента.

```

host: h1@mininet-vm
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=13.6 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=11.1 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=10.6 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=10.1 ms

--- 10.0.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005 ms
rtt min/avg/max/mdev = 10.108/11.335/13.553/1.326 ms
root@mininet-vm:/home/mininet# iperf -c 10.0.0.2

Client connecting to 10.0.0.2, TCP port 5001
TCP window size: 85.3 KByte (default)

[ 5] local 10.0.0.1 port 35206 connected with 10.0.0.2 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 5] 0.0-10.0 sec  2.19 GBytes  1.88 Gbits/sec

host: h2@mininet-vm
[ 6] 0.0-10.0 sec  11.0 GBytes  9.47 Gbits/sec
^Croot@mininet-vm:/home/mininet# iperf -s

Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)

[ 6] local 10.0.0.2 port 5001 connected with 10.0.0.1 port 35206
[ ID] Interval      Transfer    Bandwidth
[ 6] 0.0-10.0 sec  11.1 GBytes  9.49 Gbits/sec
^Croot@mininet-vm:/home/mininet# iperf -s

Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)

[ 6] local 10.0.0.2 port 5001 connected with 10.0.0.1 port 35206
[ ID] Interval      Transfer    Bandwidth
[ 6] 0.0-10.0 sec  2.19 GBytes  1.87 Gbits/sec

```

Рис. 2.4: Объединение NETEM и TBF

## 2.6 Воспроизводимые эксперименты

Самостоятельно реализуйте воспроизводимые эксперименты по использованию TBF для ограничения пропускной способности. Постройте соответствующие графики.

1. Для каждого эксперимента создала каталог

```

lab1.mn lab_iperf3 lab_iperf3_topo lab_n
mininet@mininet-vm:~/work$ mkdir exp1
mininet@mininet-vm:~/work$ mkdir exp2
mininet@mininet-vm:~/work$ ls
exp1 exp2 lab1.mn lab_iperf3 lab_iperf3
mininet@mininet-vm:~/work$ cd lab_iperf3

```

Рис. 2.5: Каталоги для экспериментов

Создала скрипты lab.py, Makefile, ping\_plot

Объединение NETEM и TBF

```

*** 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',)
*** s1 : ('tc qdisc add dev s1-eth2 parent 1: handle 2: tbf rate 2gbit burst 1000000 limit 2
000000',)
*** 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, $7}\'} | sed -e \'s/t
ime=//g\' -e \'s/icmp_seq=//g\' > ping.dat')
*** Stopping network
*** Stopping 1 controllers
c0
*** Stopping 3 links
...
*** Stopping 2 switches
s1 s2
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
./ping_plot
mininet@mininet-vm:~/work/exp1$ █

```

Рис. 2.6: Объединение NETEM и TBF

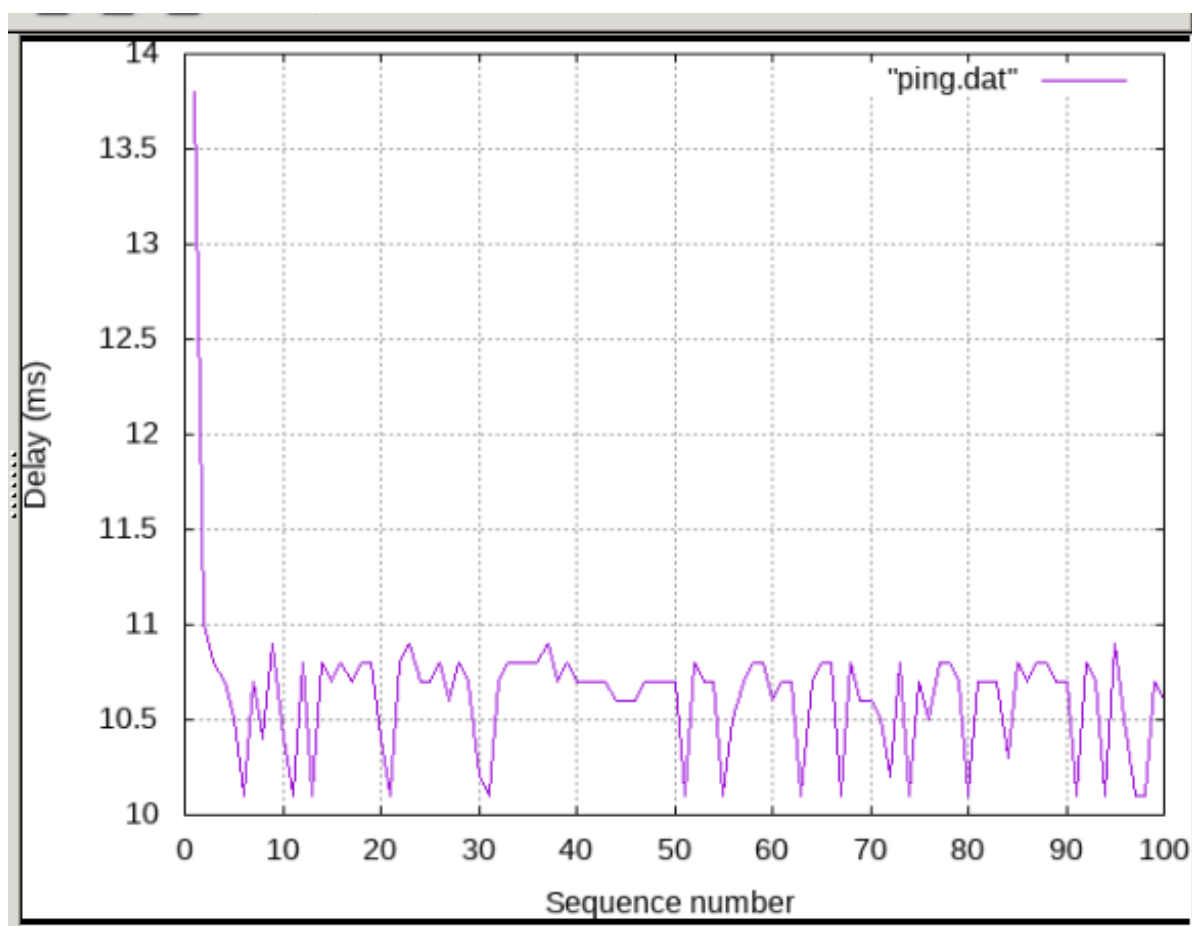


Рис. 2.7: Объединение NETEM и TBF

Ограничение скорости на коммутаторах

```

*** Starting network
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 2 switches
s1 s2 ...
*** Waiting for switches to connect
s1 s2
*** Set delay
*** s1 : ('sudo 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, $7}\'' | sed -e \'/s/time=//g\' -e \'/s/icmp_seq=//g\' > ping.dat')
*** Stopping network
*** Stopping 1 controllers
c0
*** Stopping 3 links
...
*** Stopping 2 switches
s1 s2
*** Stopping 2 hosts
h1 h2
*** Done
sudo chown mininet:mininet ping.dat
./ping_plot
mininet@mininet-vm:~/work/exp2$ █

```

Рис. 2.8: Ограничение скорости на коммутаторах

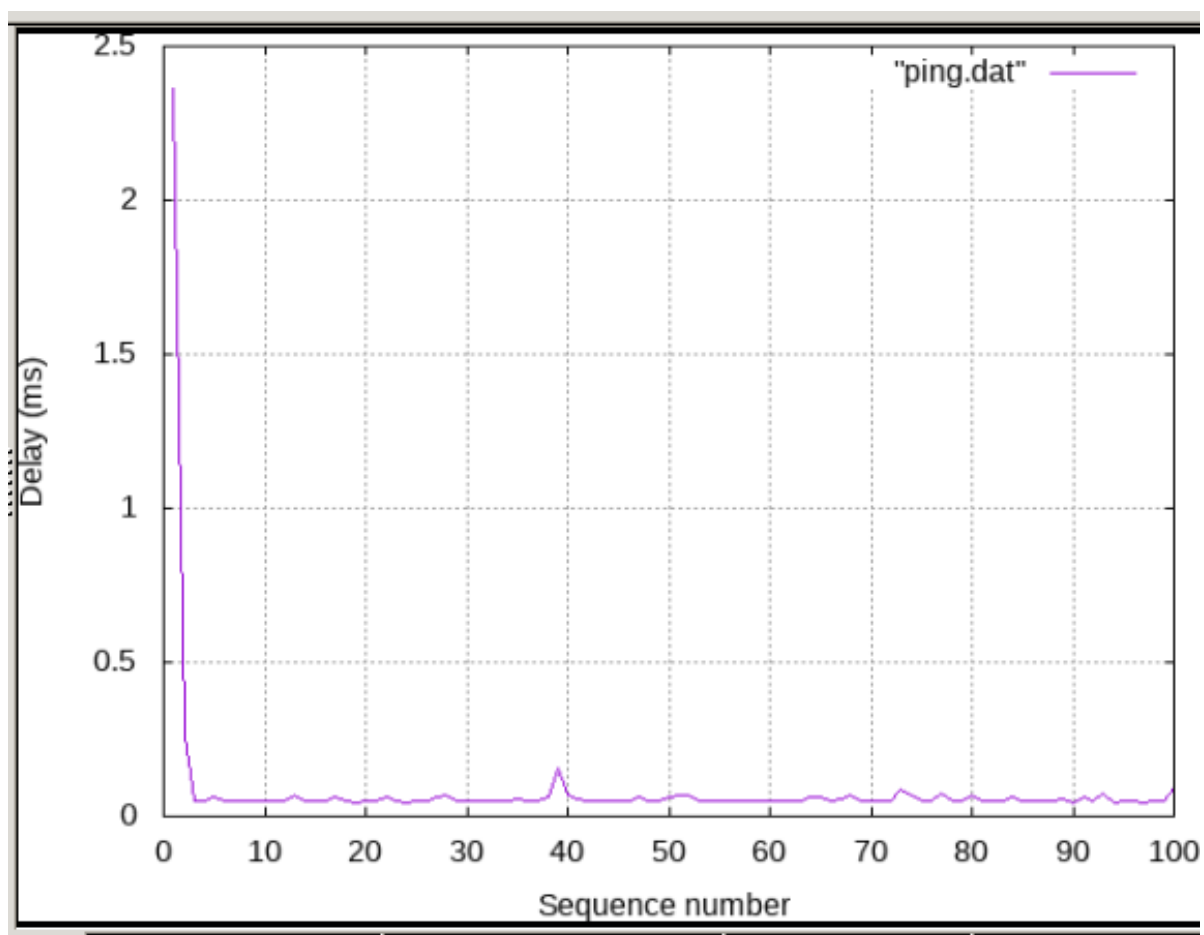


Рис. 2.9: Ограничение скорости на коммутаторах

## 3 Выводы

Я познакомилась с принципами работы дисциплины очереди Token Bucket Filter, которая формирует входящий/исходящий трафик для ограничения пропускной способности, а также получила навыки моделирования и исследования поведения трафика посредством проведения интерактивного и воспроизводимого экспериментов в Mininet.



## 4 Список литературы

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