

# **Отчёт по лабораторной работе №6**

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

Ким Реачна

# Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
3	Листинги программы	18
4	Вывод	21

## Список иллюстраций

2.1	Права запуска X-соединения . . . . .	5
2.2	Информация о сетевом интерфейсе и IP-адресе h1 . . . . .	6
2.3	Информация о сетевом интерфейсе и IP-адресе h2 . . . . .	6
2.4	Информация о сетевом интерфейсе и IP-адресе s1 . . . . .	7
2.5	Информация о сетевом интерфейсе и IP-адресе s2 . . . . .	7
2.6	Проверка подключение от h2 к h1 . . . . .	8
2.7	Запуск iperf3 . . . . .	8
2.8	Настройка tbf на конечных хостах и проверки . . . . .	9
2.9	Настройка tbf на коммутаторах . . . . .	9
2.10	Запуск iperf3 для проверки . . . . .	10
2.11	Объединение NETEM и TBF на коммутаторе s1 . . . . .	10
2.12	Добавление второе правило на коммутаторе s1 . . . . .	10
2.13	Запуск iperf3 для проверки . . . . .	11
2.14	Скрипт lab_tbf.py . . . . .	12
2.15	Makefile . . . . .	12
2.16	Выполнение эксперимент . . . . .	13
2.17	Максимальная единица передачи . . . . .	13
2.18	Время приема-передачи . . . . .	14
2.19	Отклонение времени приема-передачи . . . . .	14
2.20	Количество переданных байтов . . . . .	15
2.21	Окно перегрузки . . . . .	15
2.22	Повторная передача . . . . .	16
2.23	Пропускная способность . . . . .	16
2.24	График . . . . .	17

# 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

1. Запустила виртуальную среду с mininet и исправила права запуска X-соединения.

```
mininet@mininet-vm:~$ xauth list $DISPLAY
mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 4ccc2af9fef0d7fae15b0d47e99a6aa6
mininet@mininet-vm:~$ sudo -i
root@mininet-vm:~$ xauth add mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 4ccc2af9fef0d7fae15b0d47e99a6aa6
root@mininet-vm:~$ logout
mininet@mininet-vm:~$ sudo mn --topo=linear,2 -x
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2
*** Adding switches:
s1 s2
*** Adding links:
(h1, s1) (h2, s2) (s2, s1)
*** Configuring hosts
h1 h2
*** Running terms on localhost:10.0
*** Starting controller
c0
*** Starting 2 switches
s1 s2 ...
*** Starting CLI:
mininet>
```

Рис. 2.1: Права запуска X-соединения

2. Задала топологию сети, отображать информацию с помощью ifconfig на хостах h1, h2 и коммутаторах s1 и s2 и проверка соединения между хостами

```
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 72:e5:de:6f:5a:94 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 1121 bytes 252096 (252.0 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1121 bytes 252096 (252.0 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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

Рис. 2.2: Информация о сетевом интерфейсе и IP-адресе h1

```
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 8a:8c:04:a3:47:00 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 1127 bytes 246616 (246.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1127 bytes 246616 (246.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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

Рис. 2.3: Информация о сетевом интерфейсе и IP-адресе h2

```

X switch:s1? (root@mininet-vm)
root@mininet-vm:/home/mininet# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.104 netmask 255.255.255.0 broadcast 192.168.56.255
    ether 08:00:27:fd:bd:ca txqueuelen 1000 (Ethernet)
    RX packets 2661 bytes 536581 (536.5 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2426 bytes 964443 (964.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    ether 08:00:27:86:d6:24 txqueuelen 1000 (Ethernet)
    RX packets 242 bytes 33759 (33.7 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 259 bytes 24275 (24.2 KB)
    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 5183 bytes 1222238 (1.2 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 5183 bytes 1222238 (1.2 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

s1-eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    ether 9e:d1:9a:b4:42:75 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

s1-eth2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    ether 2e:12:d9:94:c6:21 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 ce:04:39:c5:de:d6 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 02:a1:54:a1:ab:76 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

```

Рис. 2.4: Информация о сетевом интерфейсе и IP-адресе s1

```

X switch:s2? (root@mininet-vm)
root@mininet-vm:/home/mininet# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.104 netmask 255.255.255.0 broadcast 192.168.56.255
    ether 08:00:27:fd:bd:ca txqueuelen 1000 (Ethernet)
    RX packets 2744 bytes 550103 (550.1 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2525 bytes 995537 (995.5 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    ether 08:00:27:86:d6:24 txqueuelen 1000 (Ethernet)
    RX packets 246 bytes 34089 (34.0 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 263 bytes 24605 (24.6 KB)
    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 5413 bytes 1262398 (1.2 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 5413 bytes 1262398 (1.2 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

s1-eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    ether 9e:d1:9a:b4:42:75 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

s1-eth2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    ether 2e:12:d9:94:c6:21 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 ce:04:39:c5:de:d6 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 02:a1:54:a1:ab:76 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

```

Рис. 2.5: Информация о сетевом интерфейсе и IP-адресе s2

```

root@mininet-virtual-machine:/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=2.47 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.221 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.058 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.127 ms

--- 10.0.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3045ms
rtt min/avg/max/mdev = 0.058/0.718/2.467/1.011 ms
root@mininet-virtual-machine:/home/mininet#

```

Рис. 2.6: Проверка подключения от h2 к h1

### 3. Запустила iPerf3

```

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

Accepted connection from 10.0.0.1, port 60716
[ 7] local 10.0.0.2 port 5201 connected to 10.0.0.1 port 60718
[ ID] Interval      Transfer      Bitrate
[ 7] 0.00-1.00 sec  1.95 GBytes  16.8 Gbits/sec
[ 7] 1.00-2.00 sec  2.00 GBytes  17.1 Gbits/sec
[ 7] 2.00-3.00 sec  1.97 GBytes  16.9 Gbits/sec
[ 7] 3.00-4.00 sec  2.04 GBytes  17.5 Gbits/sec
[ 7] 4.00-5.00 sec  2.02 GBytes  17.3 Gbits/sec
[ 7] 5.00-6.00 sec  2.02 GBytes  17.4 Gbits/sec
[ 7] 6.00-7.00 sec  1.99 GBytes  17.1 Gbits/sec
[ 7] 7.00-8.00 sec  2.00 GBytes  17.2 Gbits/sec
[ 7] 8.00-9.00 sec  2.07 GBytes  17.8 Gbits/sec
[ 7] 9.00-10.00 sec 2.02 GBytes  17.3 Gbits/sec
[ ID] Interval      Transfer      Bitrate
[ 7] 0.00-10.00 sec 20.1 GBytes  17.2 Gbits/sec
Server listening on 5201

^Ciperf3: interrupt
root@mininet-virtual-machine:/home/mininet#
root@mininet-virtual-machine:/home/mininet# ping 10.0.0.2 -c 4
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
4 packets transmitted, 4 received, 0% packet loss, time 3045ms
rtt min/avg/max/mdev = 0.058/0.718/2.467/1.011 ms
root@mininet-virtual-machine:/home/mininet# iperf3 -c 10.0.0.2
Connecting to host 10.0.0.2, port 5201
[ 7] local 10.0.0.1 port 60718 connected to 10.0.0.2 port 5201
[ ID] Interval      Transfer      Bitrate      Retr      Cwnd
[ 7] 0.00-1.00 sec  1.98 GBytes  17.0 Gbits/sec    7    4.64 MBytes
[ 7] 1.00-2.00 sec  2.00 GBytes  17.1 Gbits/sec   16    4.64 MBytes
[ 7] 2.00-3.00 sec  1.97 GBytes  16.9 Gbits/sec   10    4.64 MBytes
[ 7] 3.00-4.00 sec  2.04 GBytes  17.5 Gbits/sec   10    4.64 MBytes
[ 7] 4.00-5.00 sec  2.02 GBytes  17.3 Gbits/sec    8    4.64 MBytes
[ 7] 5.00-6.00 sec  2.02 GBytes  17.4 Gbits/sec   10    4.64 MBytes
[ 7] 6.00-7.00 sec  1.99 GBytes  17.1 Gbits/sec    9    4.64 MBytes
[ 7] 7.00-8.00 sec  2.00 GBytes  17.2 Gbits/sec   11    4.64 MBytes
[ 7] 8.00-9.00 sec  2.07 GBytes  17.8 Gbits/sec   16    4.64 MBytes
[ 7] 9.00-10.00 sec 2.02 GBytes  17.4 Gbits/sec    8    4.64 MBytes
[ ID] Interval      Transfer      Bitrate      Retr      Cwnd
[ 7] 0.00-10.00 sec 20.1 GBytes  17.3 Gbits/sec  105    105
[ 7] 0.00-10.00 sec 20.1 GBytes  17.2 Gbits/sec
iperf Done.
root@mininet-virtual-machine:/home/mininet#

```

Рис. 2.7: Запуск iPerf3

### 4. Измените ограничение скорости на конечных хостах и запуск iPerf3 для проверки



```
root@mininet-virtual-machine:~# sudo tc qdisc add dev h1-eth0 root tbf rate 10gbit burst 5000000 limit 15000000
root@mininet-virtual-machine:~# egrep '^CONFIG_HZ_[0-9]+' /boot/config-`uname -r`
CONFIG_HZ_250=y
root@mininet-virtual-machine:~# iperf3 -c 10.0.0.2
Connecting to host 10.0.0.2, port 5201
[ 7] local 10.0.0.1 port 60722 connected to 10.0.0.2 port 5201
[ ID] Interval           Transfer             Bitrate          Retr  Cwnd
[ 7] 0.00-1.00 sec      1.12 GBytes        9.66 Gbits/sec    0     2.00 MBytes
[ 7] 1.00-2.00 sec      1.12 GBytes        9.60 Gbits/sec    0     2.92 MBytes
[ 7] 2.00-3.00 sec      1.10 GBytes        9.47 Gbits/sec    0     3.40 MBytes
[ 7] 3.00-4.00 sec      1.11 GBytes        9.56 Gbits/sec    0     3.58 MBytes
[ 7] 4.00-5.00 sec      1.11 GBytes        9.56 Gbits/sec    0     3.58 MBytes
[ 7] 5.00-6.00 sec      1.11 GBytes        9.58 Gbits/sec    0     3.82 MBytes
[ 7] 6.00-7.00 sec      1.11 GBytes        9.56 Gbits/sec    0     3.82 MBytes
[ 7] 7.00-8.00 sec      1.11 GBytes        9.56 Gbits/sec    0     3.82 MBytes
[ 7] 8.00-9.00 sec      1.11 GBytes        9.56 Gbits/sec    0     3.82 MBytes
[ 7] 9.00-10.00 sec     1.11 GBytes        9.56 Gbits/sec    0     3.82 MBytes
-----
[ ID] Interval           Transfer             Bitrate          Retr
[ 7] 0.00-10.00 sec     11.1 GBytes        9.57 Gbits/sec    0
[ 7] 0.00-10.01 sec     11.1 GBytes        9.55 Gbits/sec    0
sender
receiver

iperf Done.
root@mininet-virtual-machine:~# sudo tc qdisc del dev h1-eth0 root
root@mininet-virtual-machine:~#
```

```
Server listening on 5201
-----
Accepted connection from 10.0.0.1, port 60720
[ 7] local 10.0.0.2 port 5201 connected to 10.0.0.1 port 60722
[ ID] Interval           Transfer             Bitrate          Retr
[ 7] 0.00-1.00 sec      1.12 GBytes        9.60 Gbits/sec    0
[ 7] 1.00-2.00 sec      1.11 GBytes        9.56 Gbits/sec    0
[ 7] 2.00-3.00 sec      1.10 GBytes        9.48 Gbits/sec    0
[ 7] 3.00-4.00 sec      1.11 GBytes        9.56 Gbits/sec    0
[ 7] 4.00-5.00 sec      1.11 GBytes        9.56 Gbits/sec    0
[ 7] 5.00-6.00 sec      1.11 GBytes        9.56 Gbits/sec    0
[ 7] 6.00-7.00 sec      1.11 GBytes        9.57 Gbits/sec    0
[ 7] 7.00-8.00 sec      1.11 GBytes        9.57 Gbits/sec    0
[ 7] 8.00-9.00 sec      1.11 GBytes        9.56 Gbits/sec    0
[ 7] 9.00-10.00 sec     1.11 GBytes        9.56 Gbits/sec    0
[ 7] 10.00-10.01 sec     512 KBytes        548 Mbits/sec    0
-----
[ ID] Interval           Transfer             Bitrate
[ 7] 0.00-10.01 sec     11.1 GBytes        9.55 Gbits/sec
receiver

Server listening on 5201
^Ciperf3: interrupt - the server has terminated
root@mininet-virtual-machine:~#
```

Рис. 2.8: Настройка tbf на конечных хостах и проверки

5. Измените ограничение скорости на коммутаторах и запуск iperf3 для проверки:

```
root@mininet-virtual-machine:~# sudo tc qdisc add dev s1-eth2 root tbf rate 10gbit burst 5000000 limit 15000000
root@mininet-virtual-machine:~#
```

Рис. 2.9: Настройка tbf на коммутаторах

```

[ 7] 0.00-10.01 sec 11.1 GBytes 9.55 Gbits/sec receiver

iperf Done.
root@mininet-virtual-machine:/home/mininet# sudo tc qdisc del dev h1-eth0 root
root@mininet-virtual-machine:/home/mininet# iperf3 -c 10.0.0.2
Connecting to host 10.0.0.2, port 5201
[ 7] local 10.0.0.1 port 60726 connected to 10.0.0.2 port 5201
[ ID] Interval      Transfer     Bitrate      Retr  Cwnd
[ 7] 0.00-1.00 sec 1.12 GBytes 9.61 Gbits/sec 9    14.7 MBytes
[ 7] 1.00-2.00 sec 1.11 GBytes 9.57 Gbits/sec 0    14.7 MBytes
[ 7] 2.00-3.00 sec 1.11 GBytes 9.53 Gbits/sec 0    14.7 MBytes
[ 7] 3.00-4.00 sec 1.11 GBytes 9.56 Gbits/sec 0    14.7 MBytes
[ 7] 4.00-5.00 sec 1.11 GBytes 9.54 Gbits/sec 0    14.7 MBytes
[ 7] 5.00-6.00 sec 1.11 GBytes 9.54 Gbits/sec 0    14.7 MBytes
[ 7] 6.00-7.00 sec 1.11 GBytes 9.56 Gbits/sec 0    14.7 MBytes
[ 7] 7.00-8.00 sec 1.11 GBytes 9.54 Gbits/sec 0    14.7 MBytes
[ 7] 8.00-9.00 sec 1.11 GBytes 9.53 Gbits/sec 0    14.7 MBytes
[ 7] 9.00-10.00 sec 1.11 GBytes 9.51 Gbits/sec 0    14.7 MBytes
[ ID] Interval      Transfer     Bitrate      Retr  Cwnd
[ 7] 0.00-10.00 sec 11.1 GBytes 9.55 Gbits/sec 9
[ 7] 0.00-10.01 sec 11.1 GBytes 9.54 Gbits/sec 9
sender
receiver

iperf Done.
root@mininet-virtual-machine:/home/mininet#

```

```

Server listening on 5201
Accepted connection from 10.0.0.1, port 60724
[ 7] local 10.0.0.2 port 5201 connected to 10.0.0.1 port 60726
[ ID] Interval      Transfer     Bitrate      Retr  Cwnd
[ 7] 0.00-1.00 sec 1.11 GBytes 9.52 Gbits/sec 9    14.7 MBytes
[ 7] 1.00-2.00 sec 1.11 GBytes 9.56 Gbits/sec 0    14.7 MBytes
[ 7] 2.00-3.00 sec 1.11 GBytes 9.54 Gbits/sec 0    14.7 MBytes
[ 7] 3.00-4.00 sec 1.11 GBytes 9.56 Gbits/sec 0    14.7 MBytes
[ 7] 4.00-5.00 sec 1.11 GBytes 9.54 Gbits/sec 0    14.7 MBytes
[ 7] 5.00-6.00 sec 1.11 GBytes 9.54 Gbits/sec 0    14.7 MBytes
[ 7] 6.00-7.00 sec 1.11 GBytes 9.56 Gbits/sec 0    14.7 MBytes
[ 7] 7.00-8.00 sec 1.11 GBytes 9.54 Gbits/sec 0    14.7 MBytes
[ 7] 8.00-9.00 sec 1.11 GBytes 9.53 Gbits/sec 0    14.7 MBytes
[ 7] 9.00-10.00 sec 1.11 GBytes 9.51 Gbits/sec 0    14.7 MBytes
[ 7] 10.00-10.01 sec 128 KBytes 163 Mbits/sec
[ ID] Interval      Transfer     Bitrate      Retr  Cwnd
[ 7] 0.00-10.01 sec 11.1 GBytes 9.54 Gbits/sec 9
receiver

Server listening on 5201
iperf3: interrupt - the server has terminated
root@mininet-virtual-machine:/home/mininet#

```

Рис. 2.10: Запуск iperf3 для проверки

- Объединение NETEM и TBF на коммутаторе s1 и проверки соединения от хоста h1 к хосту h2 с помощью команды ping с параметром -c 4:

```

root@mininet-virtual-machine:/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=12.4 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=11.2 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=10.8 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=10.7 ms

--- 10.0.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 10.741/11.297/12.392/0.659 ms
root@mininet-virtual-machine:/home/mininet#
root@mininet-virtual-machine:/home/mininet# sudo tc qdisc del dev s1-eth2 root
root@mininet-virtual-machine:/home/mininet# sudo tc qdisc add dev s1-eth2 root handle 1: netem delay 10ms
root@mininet-virtual-machine:/home/mininet#

```

Рис. 2.11: Объединение NETEM и TBF на коммутаторе s1

- Добавление второго правила на коммутаторе s1 и запуск iperf3 для проверки:

```

root@mininet-virtual-machine:/home/mininet# sudo tc qdisc del dev s1-eth2 root
root@mininet-virtual-machine:/home/mininet# sudo tc qdisc add dev s1-eth2 root handle 1: netem delay 10ms
root@mininet-virtual-machine:/home/mininet# sudo tc qdisc add dev s1-eth2 parent 1: handle 2: tbf rate 2gbit burst 1000000 limit 2000000
root@mininet-virtual-machine:/home/mininet# sudo tc qdisc del dev s1-eth2 root
root@mininet-virtual-machine:/home/mininet#

```

Рис. 2.12: Добавление второе правило на коммутаторе s1

```
host: h1@mininet-vm
--- 10.0.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 10.741/11.297/12.392/0.659 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 60730 connected to 10.0.0.2 port 5201
[ ID] Interval      Transfer    Bitrate    Retr    Cwnd
[ 7] 0.00-1.00 sec  192 MBytes  1.61 Gbits/sec  895    3.97 MBytes
[ 7] 1.00-2.00 sec  228 MBytes  1.91 Gbits/sec   90    2.95 MBytes
[ 7] 2.00-3.00 sec  229 MBytes  1.92 Gbits/sec    0    3.09 MBytes
[ 7] 3.00-4.00 sec  228 MBytes  1.91 Gbits/sec    0    3.20 MBytes
[ 7] 4.00-5.00 sec  229 MBytes  1.92 Gbits/sec    0    3.28 MBytes
[ 7] 5.00-6.00 sec  220 MBytes  1.85 Gbits/sec 1080    2.41 MBytes
[ 7] 6.00-7.00 sec  229 MBytes  1.92 Gbits/sec    0    2.53 MBytes
[ 7] 7.00-8.00 sec  228 MBytes  1.91 Gbits/sec    0    2.62 MBytes
[ 7] 8.00-9.00 sec  229 MBytes  1.92 Gbits/sec    0    2.69 MBytes
[ 7] 9.00-10.00 sec 228 MBytes  1.91 Gbits/sec    0    2.74 MBytes
[ ID] Interval      Transfer    Bitrate    Retr
[ 7] 0.00-10.00 sec 2.18 GBytes 1.88 Gbits/sec 2065
[ 7] 0.00-10.02 sec 2.18 GBytes 1.86 Gbits/sec
sender
receiver

iperf Done.
root@mininet-vm:/home/mininet#
host: h2@mininet-vm
-----
dme/mininet Server listening on 5201
dme/mininet
dme/mininet Accepted connection from 10.0.0.1, port 60728
dme/mininet [ 7] local 10.0.0.2 port 5201 connected to 10.0.0.1 port 60730
dme/mininet [ ID] Interval      Transfer    Bitrate
[ 7] 0.00-1.00 sec  181 MBytes  1.52 Gbits/sec
[ 7] 1.00-2.00 sec  228 MBytes  1.91 Gbits/sec
[ 7] 2.00-3.00 sec  228 MBytes  1.91 Gbits/sec
[ 7] 3.00-4.00 sec  228 MBytes  1.91 Gbits/sec
[ 7] 4.00-5.00 sec  228 MBytes  1.91 Gbits/sec
[ 7] 5.00-6.00 sec  221 MBytes  1.85 Gbits/sec
[ 7] 6.00-7.00 sec  228 MBytes  1.91 Gbits/sec
[ 7] 7.00-8.00 sec  228 MBytes  1.91 Gbits/sec
[ 7] 8.00-9.00 sec  228 MBytes  1.91 Gbits/sec
[ 7] 9.00-10.00 sec 228 MBytes  1.91 Gbits/sec
[ 7] 10.00-10.02 sec 1.49 MBytes  599 Mbits/sec
[ ID] Interval      Transfer    Bitrate
[ 7] 0.00-10.02 sec 2.18 GBytes 1.86 Gbits/sec
receiver
-----
Server listening on 5201
^Ciperf3: interrupt - the server has terminated
root@mininet-vm:/home/mininet#
```

Рис. 2.13: Запуск iperf3 для проверки

8. Реализации воспроизводимых экспериментов по использованию TBF для ограничения пропускной способности:

```

GNU nano 4.8 /home/mininet/work/lab_tbf/lab_tbf.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.link import TCLink
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 switches\n' )
    s1 = net.addSwitch( 's1' )
    s2 = net.addSwitch( 's2' )

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

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

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

    info( '*** Traffic generation\n' )
    h2.cmdPrint( 'iperf3 -s -D -l' )
    time.sleep(10) # Wait 10 seconds for servers to start
    h1.cmdPrint( 'iperf3 -c -s, h2.IP(), -D > iperf_result.json' )
    h1.cmdPrint( 'ping -c 100, h2.IP(), -i 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. 2.14: Скрипт lab\_tbf.py

```

GNU nano 4.8 /home/mininet/work/lab_tbf/Makefile
all: ping.dat ping.png plot

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

ping.png: ping.dat
    ./ping_plot

plot: iperf_result.json
    plot_iperf.sh iperf_result.json

clean:
    -rm -f *.dat *.pdf *.json *.csv
    -rm -rf results
    sudo mn -c

```

Рис. 2.15: Makefile

```

mininet@mininet-vmt:~/work/lab_tbf$ make
sudo python lab_tbf.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',)
*** s1 : ('tc qdisc add dev s1-eth2 parent 1: handle 2: tbf rate 2gbit burst 1000000 limit 2000000',)
*** Traffic generation
h2 : ('iperf3 -s -D -i',)
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
~/ping_plot
plot_iperf.sh iperf_result.json

```

Рис. 2.16: Выполнение эксперимент

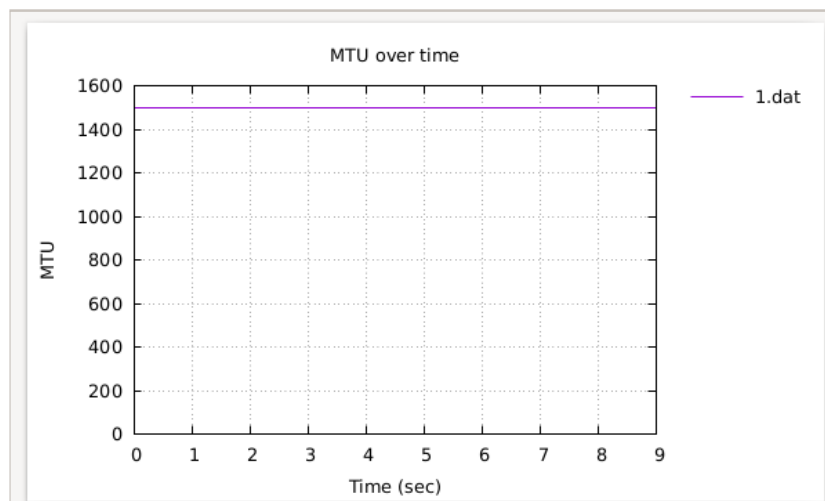


Рис. 2.17: Максимальная единица передачи

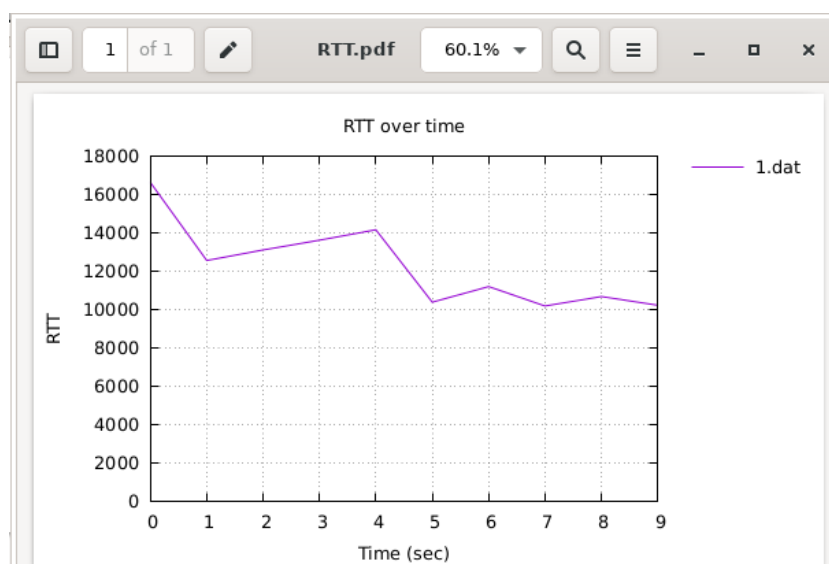


Рис. 2.18: Время приема-передачи

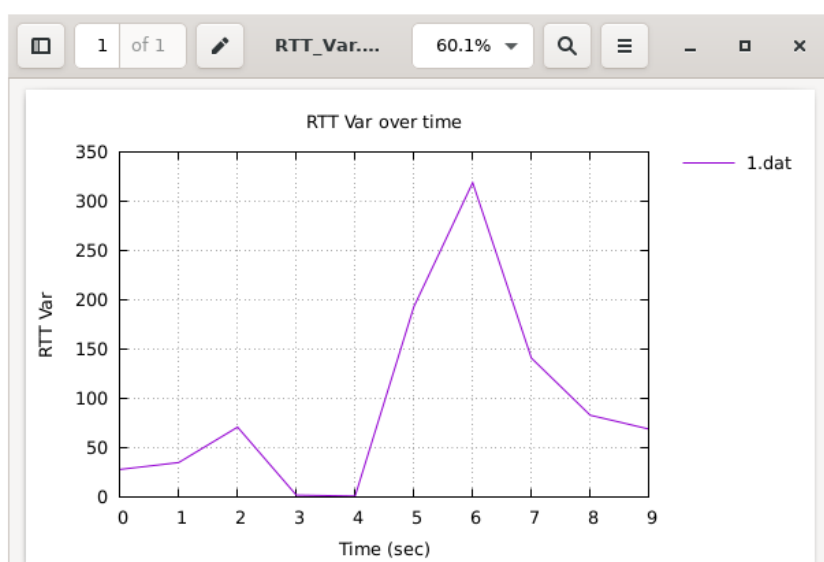


Рис. 2.19: Отклонение времени приема-передачи

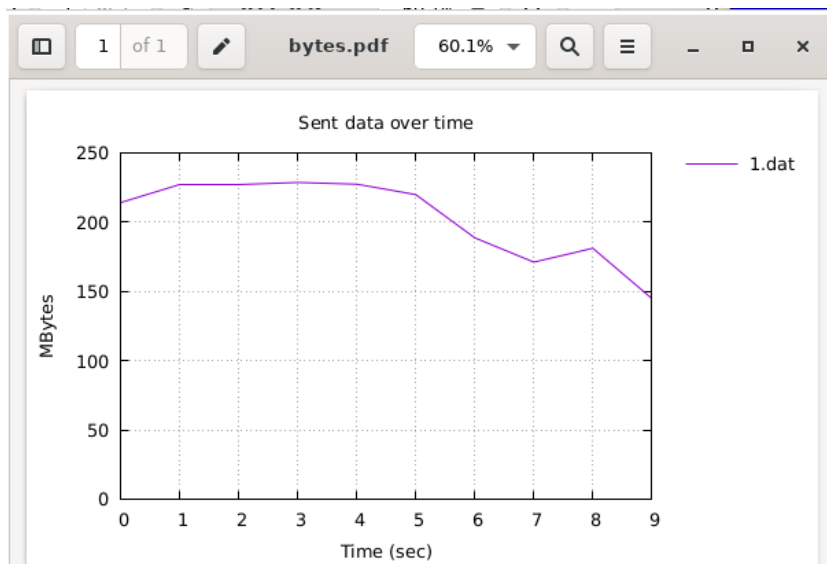


Рис. 2.20: Количество переданных байтов

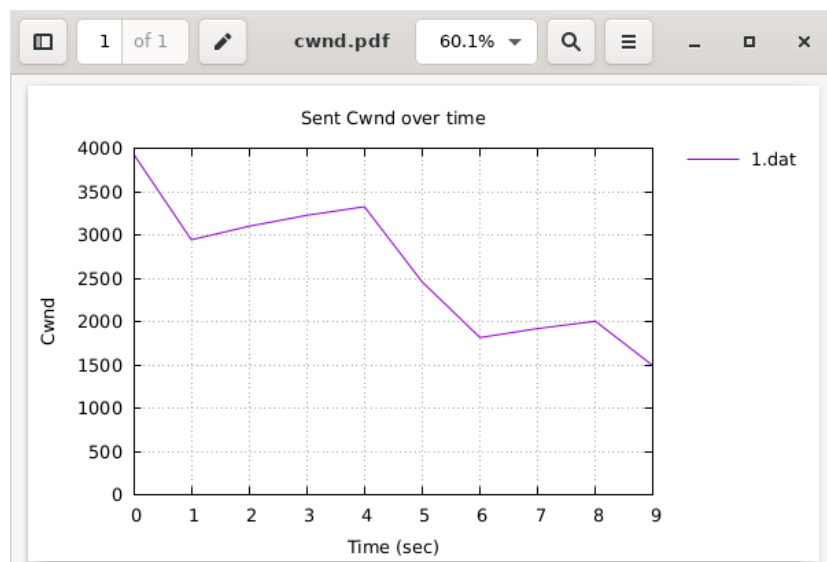


Рис. 2.21: Окно перегрузки

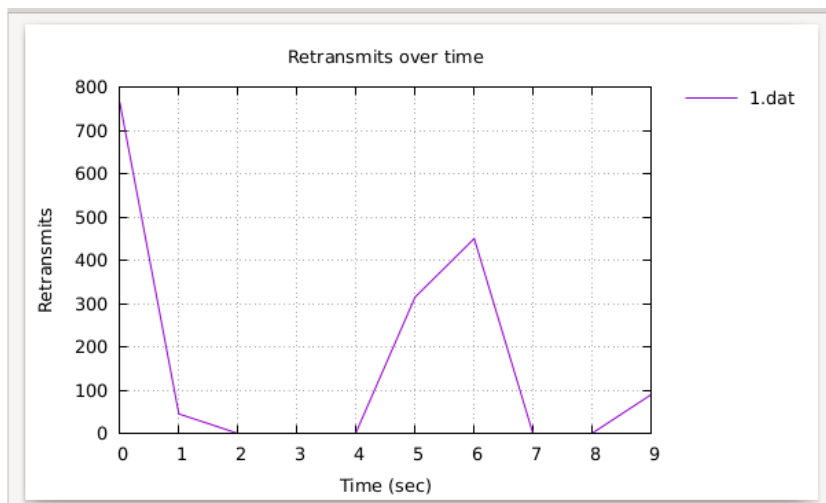


Рис. 2.22: Повторная передача

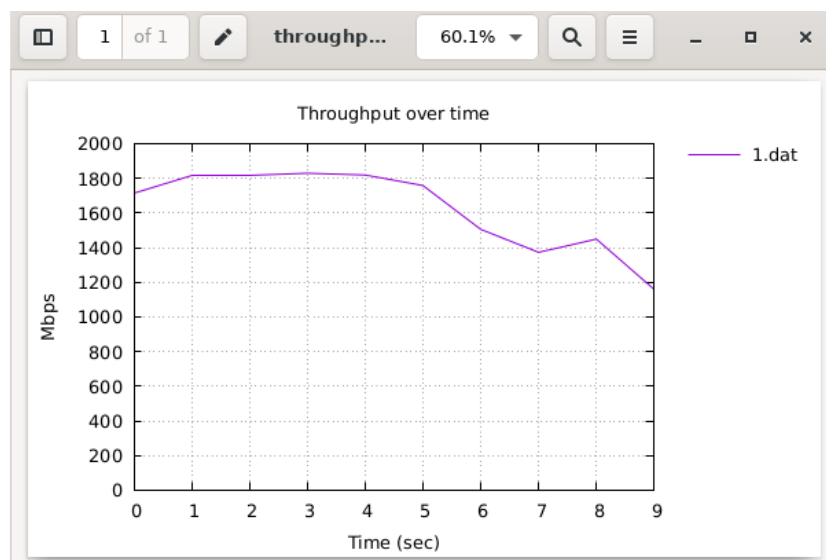


Рис. 2.23: Пропускная способность



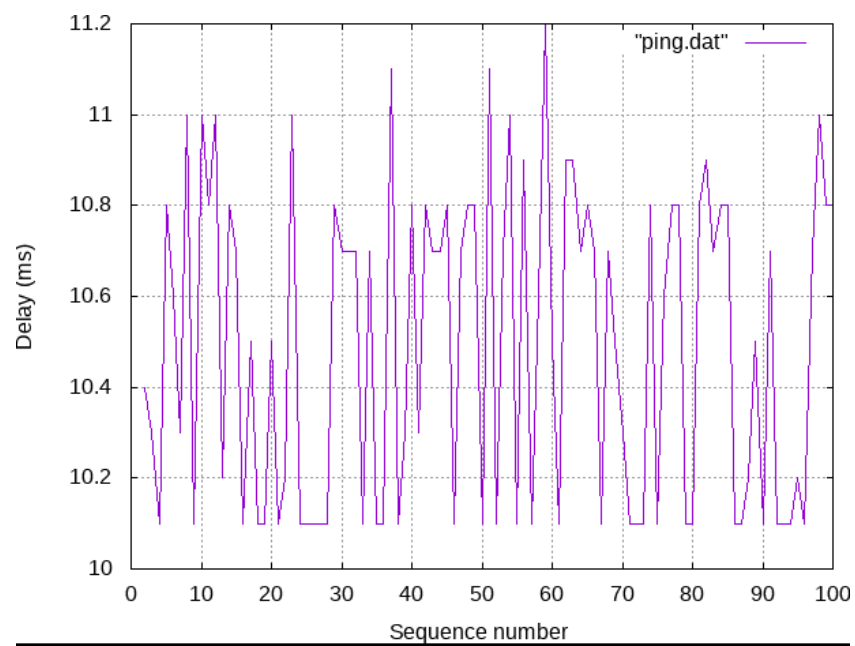


Рис. 2.24: График

## 3 Листинги программы

- Скрипт lab\_tbf.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.link import TCLink
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' )

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

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

info( '*** Set delay\n' )
s1.cmdPrint( 'tc qdisc add dev s1-eth2 root handle 1: netem delay 10ms' )
s1.cmdPrint( 'tc qdisc add dev s1-eth2 parent 1: handle 2:
tbft rate 2gbit burst 1000000 limit 2000000' )

info( '*** Traffic generation\n' )
h2.cmdPrint( 'iperf3 -s -D -1' )
time.sleep(10) # Wait 10 seconds for servers to start
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\' > pi

info( '*** Stopping network' )

```

```
net.stop()
```

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

- Скрипт Makefile

```
all: ping.dat ping.png plot
```

```
ping.dat:
```

```
    sudo python lab_tbf.py  
    sudo chown mininet:mininet ping.dat
```

```
ping.png: ping.dat
```

```
    ./ping_plot
```

```
plot: iperf_result.json
```

```
    plot_iperf.sh iperf_result.json
```

```
clean:
```

```
    -rm -f *.dat *.pdf *.json *.csv  
    -rm -rf results  
    sudo mn -c
```

## 4 Вывод

Я познакомилась 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.