Отчёт по лабораторной работе №6

Настройка пропускной способности глобальной сети с помощью Token Bucket Filter

Ким Реачна

Содержание

Список иллюстраций

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.

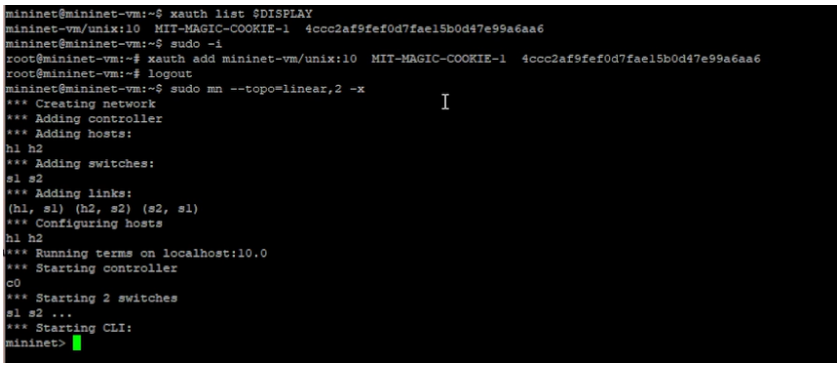

Рис. 2.1: Права запуска X-соединения

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

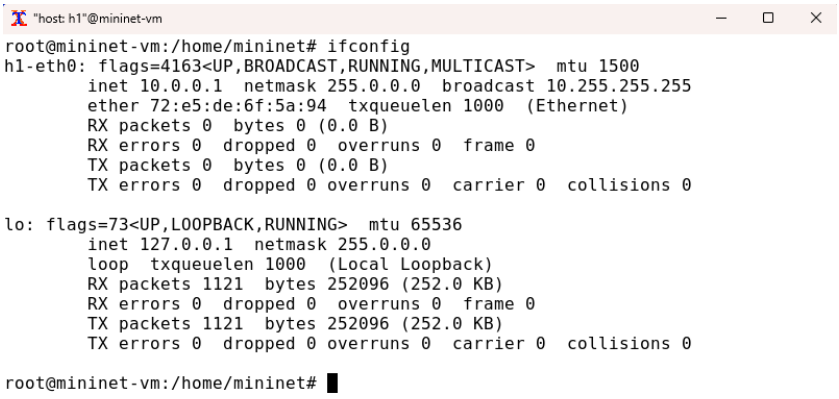

Рис. 2.2: Информация о сетевом интерфейсе и IP-адресе h1

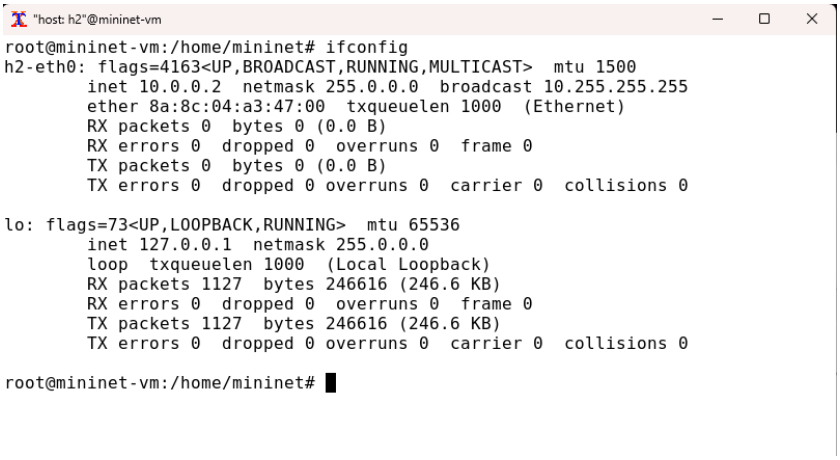

Рис. 2.3: Информация о сетевом интерфейсе и IP-адресе h2

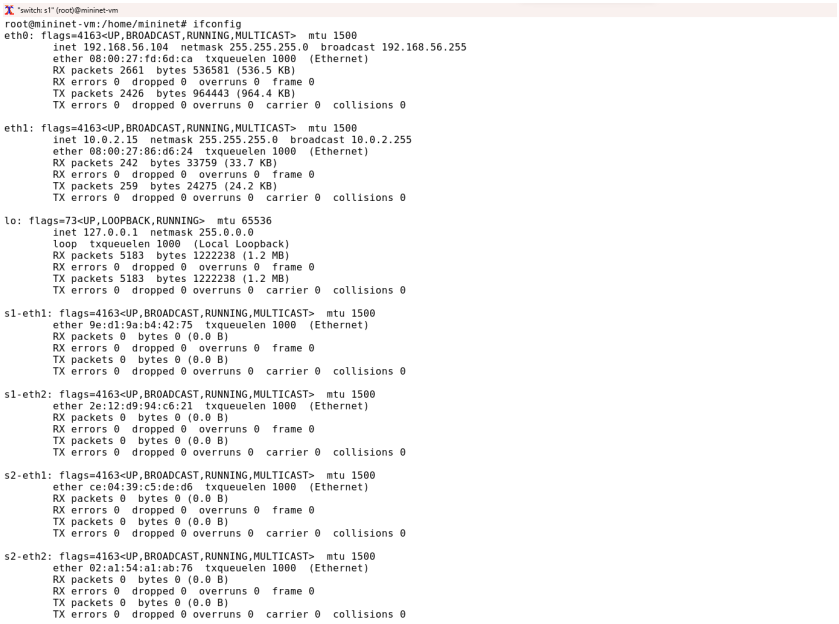

Рис. 2.4: Информация о сетевом интерфейсе и IP-адресе s1

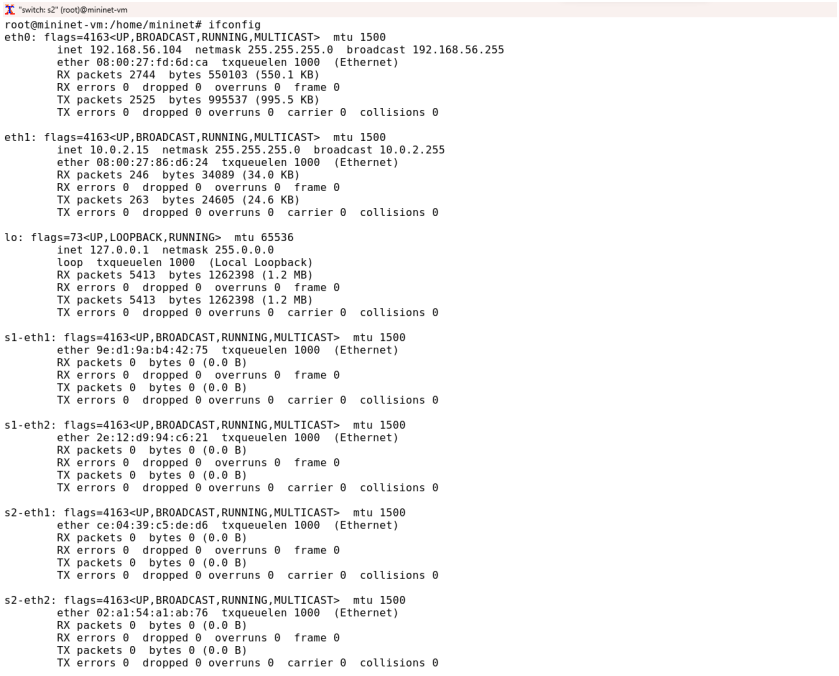

Рис. 2.5: Информация о сетевом интерфейсе и IP-адресе s2

```
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=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-vm:/home/mininet#
```

Рис. 2.6: Проверка подключение от h2 к h1

3. Запустила iPerf3

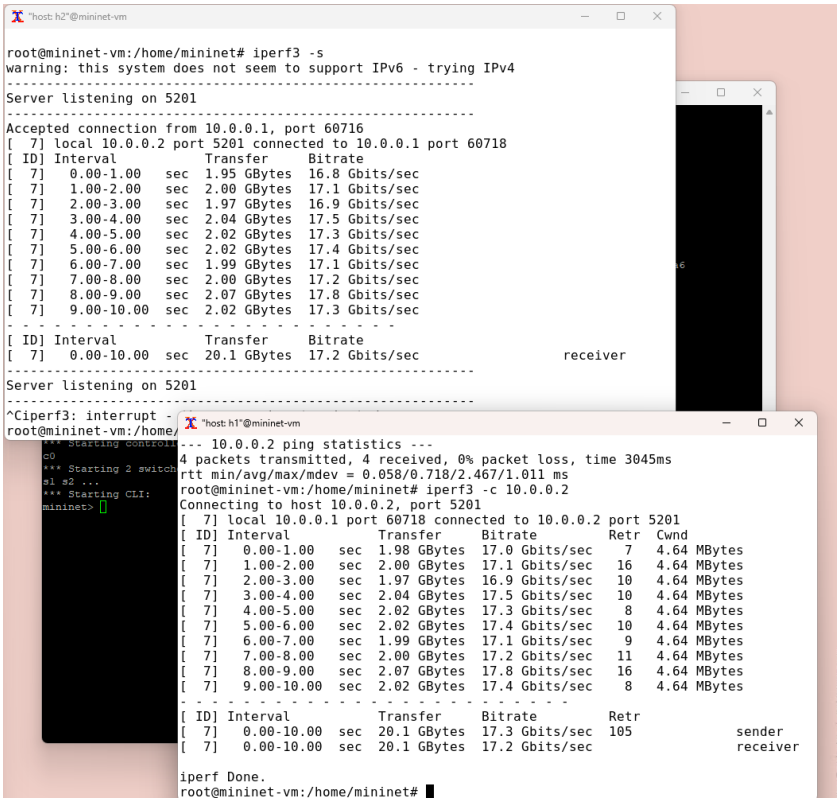

Рис. 2.7: Запуск iperf3

4. Измените ограничение скорости на конечных хостах и запуск iperf3 для проверки

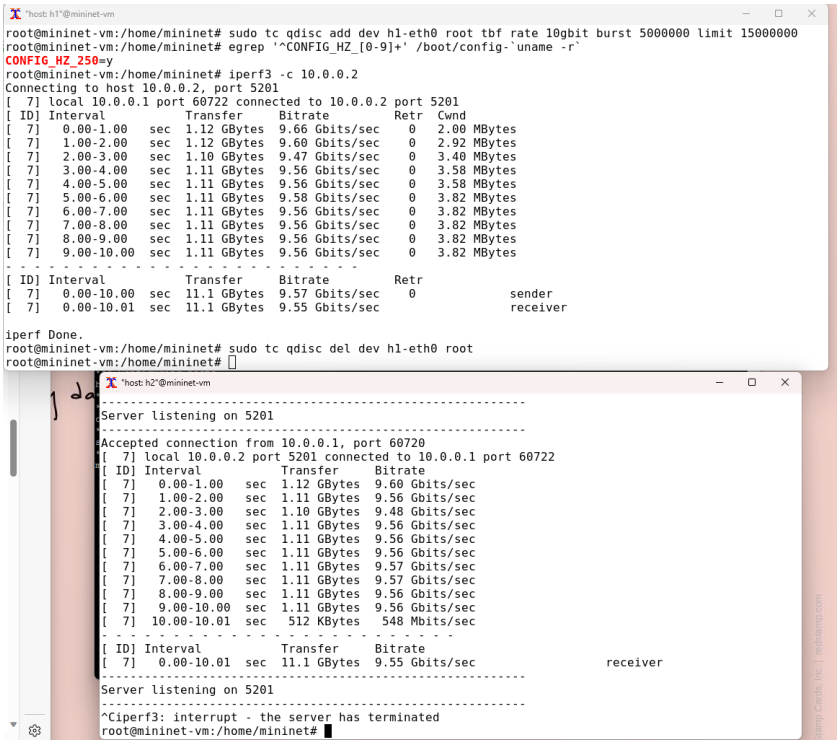

Рис. 2.8: Настройка tbf на конечных хостах и проверки

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

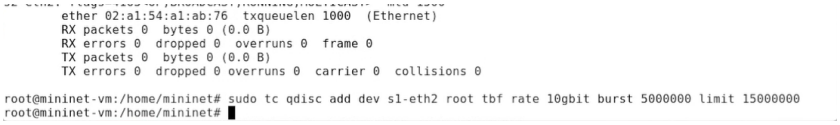

Рис. 2.9: Настройка tbf на коммутаторах

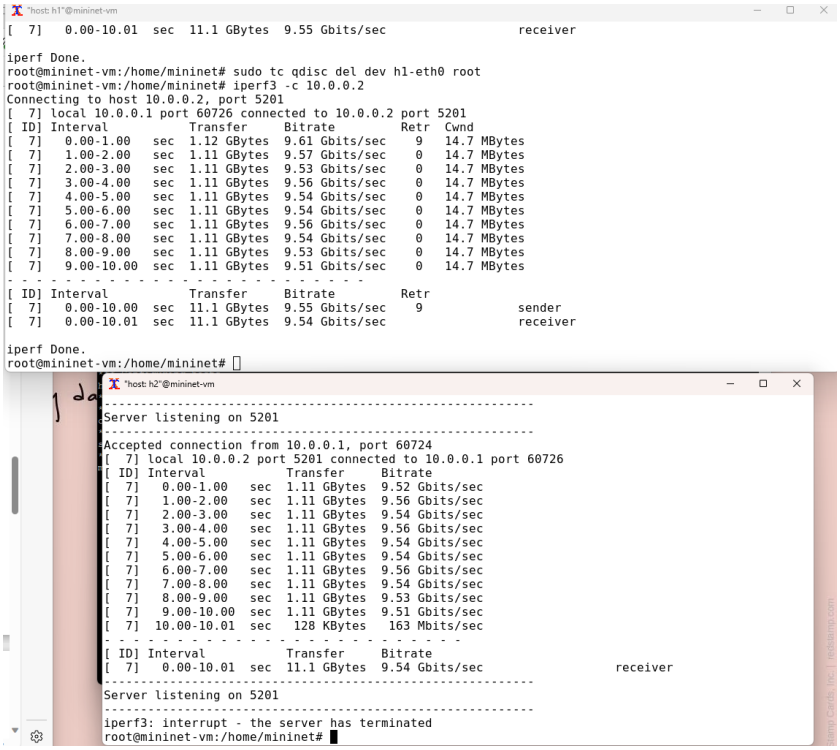

Рис. 2.10: Запуск iperf3 для проверки

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

Рис. 2.11: Объединение NETEM и TBF на коммутаторе s1

7. Добавление второе правило на коммутаторе s1 и запуск iperf3 для проверки:

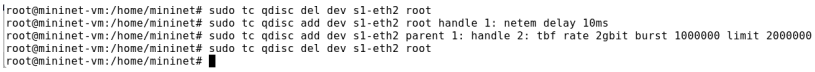

Рис. 2.12: Добавление второе правило на коммутаторе s1

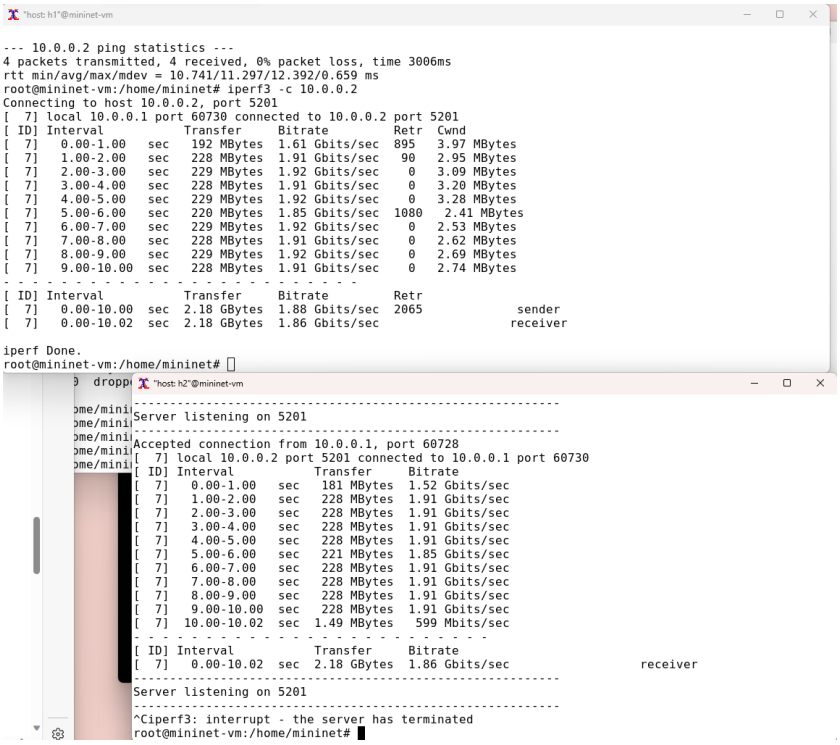

Рис. 2.13: Запуск iperf3 для проверки

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

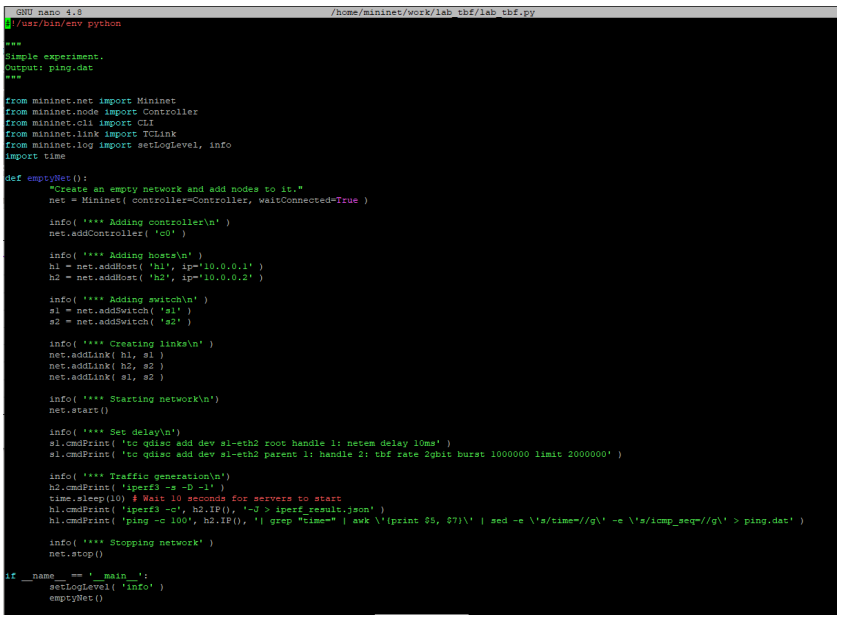

Рис. 2.14: Скрипт lab_tbf.py

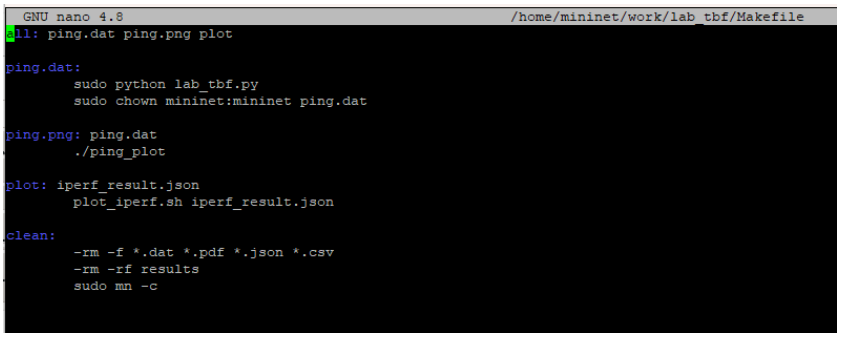

Рис. 2.15: Makefile

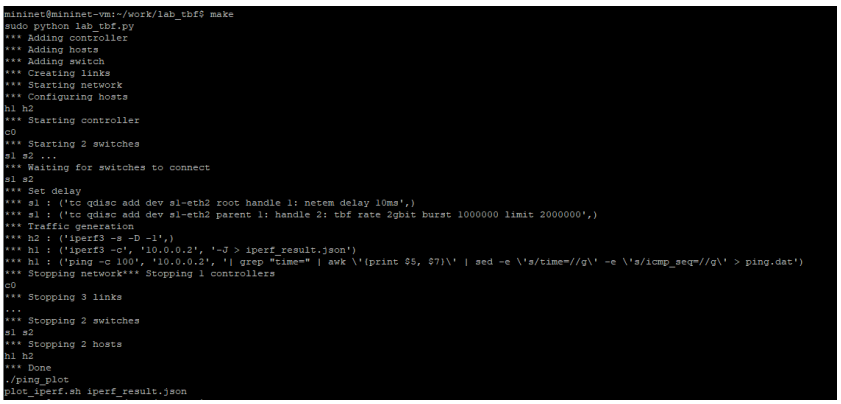

Рис. 2.16: Выполнение эксперимент

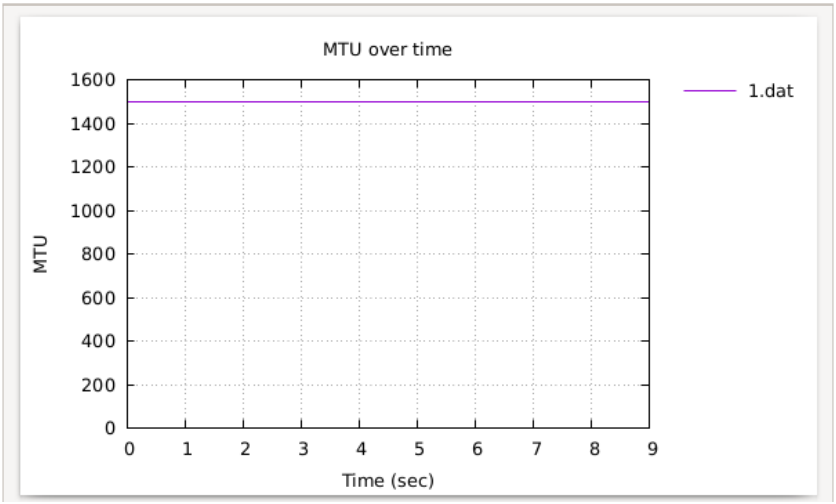

Рис. 2.17: Максимальная единица передачи

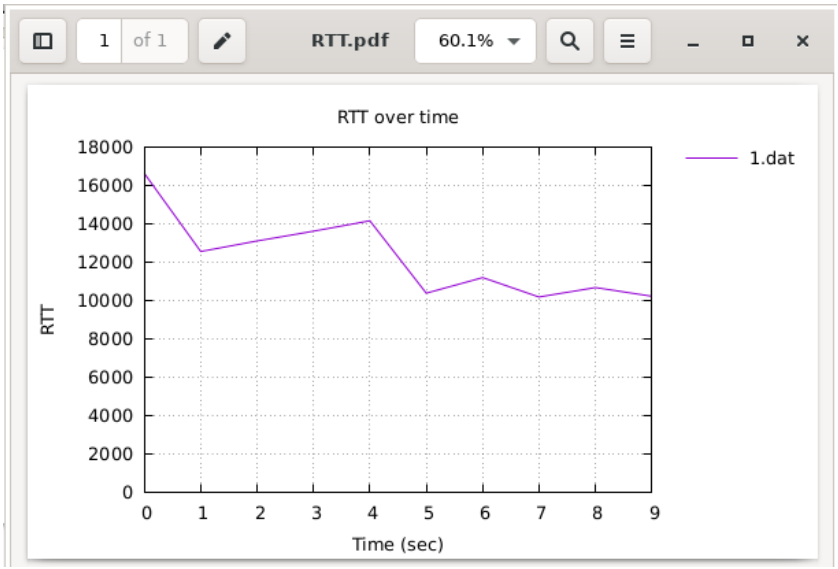

Рис. 2.18: Время приема-передачи

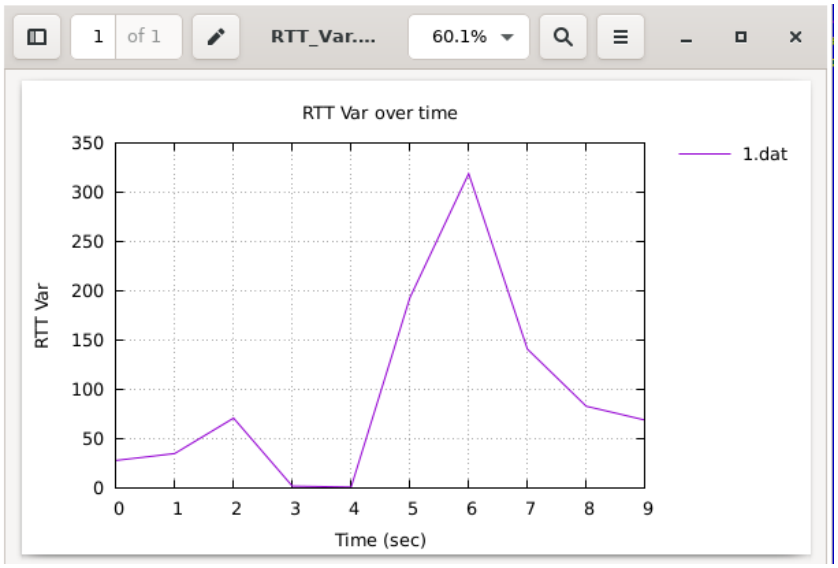

Рис. 2.19: Отклонение времени приема-передачи

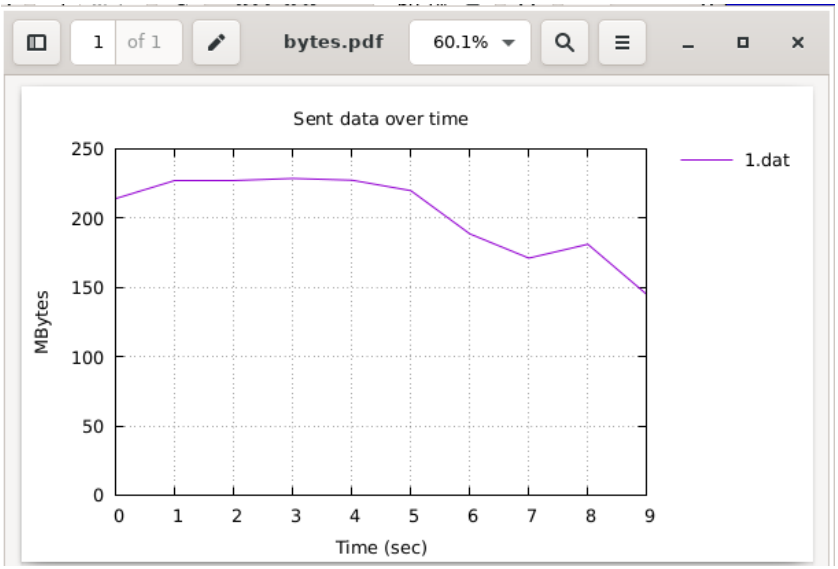

Рис. 2.20: Количество переданных байтов

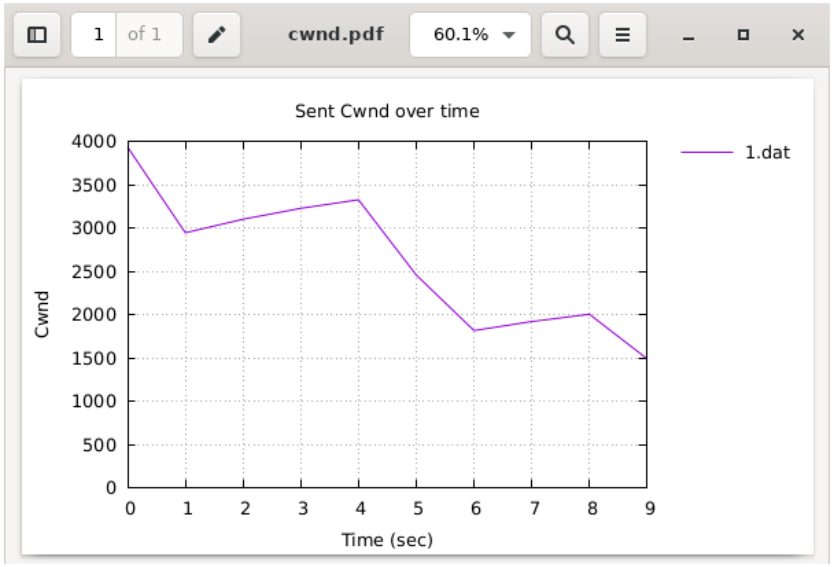

Рис. 2.21: Окно перегрузки

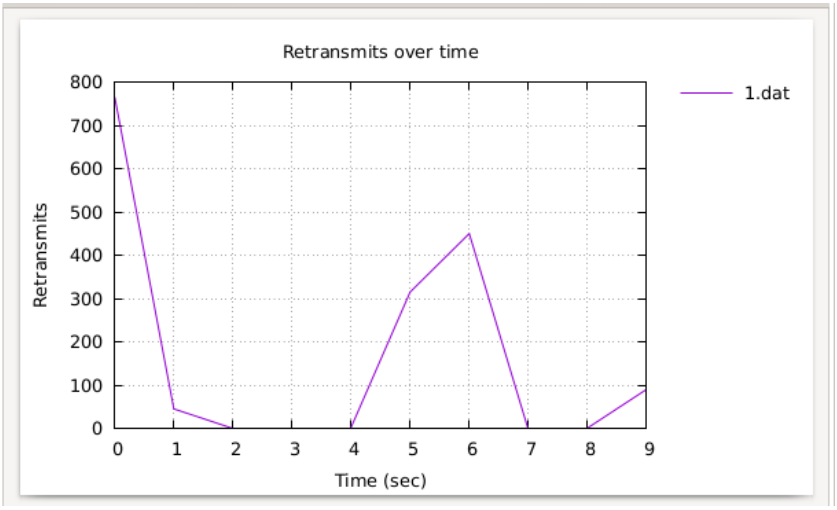

Рис. 2.22: Повторная передача

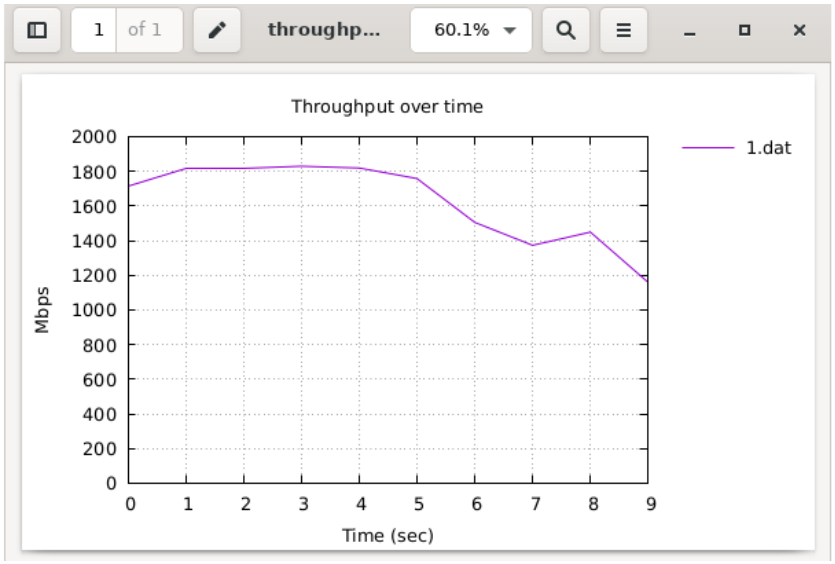

Рис. 2.23: Пропускная способность

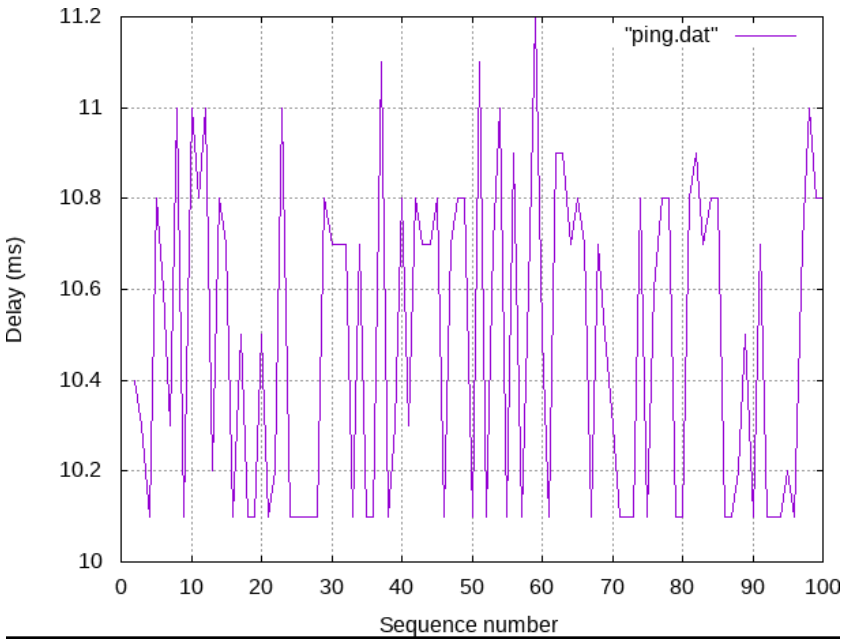

Рис. 2.24: График

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:
tbf 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.