Лабораторная работа №6

Настройка пропускной способности глобальной сети с помощью Token Bucket Filter

Ланцова Яна Игоревна

Содержание

Список иллюстраций

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 Задачи

1. Задайте топологию, состоящую из двух хостов и двух коммутаторов с назначенной по умолчанию mininet сетью 10.0.0.0/8.
2. Проведите интерактивные эксперименты по ограничению пропускной способности сети с помощью TBF в эмулируемой глобальной сети.
3. Самостоятельно реализуйте воспроизводимые эксперимент по применению TBF для ограничения пропускной способности. Постройте соответствующие графики.

3 Выполнение лабораторной работы

3.1 Запуск лабораторной топологии

Из основной ОС подключимся к виртуальной машине и в исправим права запуска X-соединения. Скопируем значение куки (MIT magic cookie) своего пользователя mininet в файл для пользователя root (рис. 3.1).

Рис. 3.1: Исправление MIT magic cookie

Зададим простейшую топологию, состоящую из двух хостов и коммутатора с назначенной по умолчанию mininet сетью 10.0.0.0/8 (рис. 3.2). После введения этой команды запустятся терминалы двух хостов, коммутатора и контроллера. Терминалы коммутатора и контроллера можно закрыть.

Рис. 3.2: Простейшая топология

На комммутаторах s1 и s2 введем команду ifconfig, чтобы отобразить информацию, относящуюся к их сетевым интерфейсам и назначенным им IP-адресам. В дальнейшем при работе с NETEM и командой tc будут использоваться интерфейсы s1-eth2 (рис. 3.3).

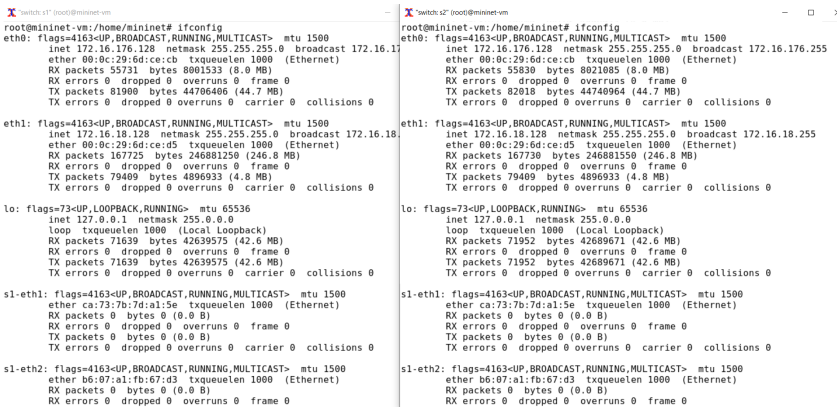

Рис. 3.3: ifconfig на комммутаторах s1 и s2

На хостах h1 и h2 введем команду ifconfig, чтобы отобразить информацию, относящуюся к их сетевым интерфейсам и назначенным им IP-адресам. В дальнейшем при работе с NETEM и командой tc будут использоваться интерфейсы h1-eth0 и h2-eth0 (рис. 3.4).

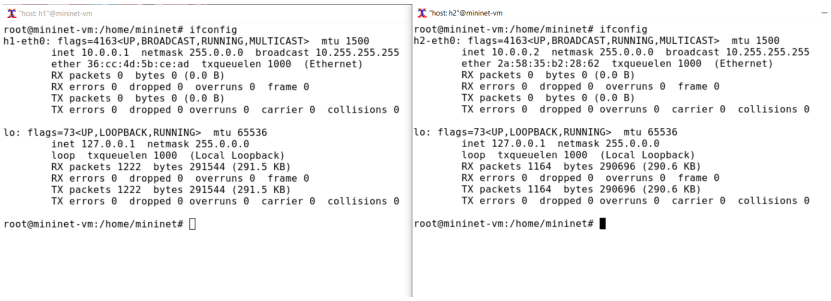

Рис. 3.4: ifconfig на хостат h1 и h2

Проверим подключение между хостами h1 и h2 с помощью команды ping с параметром -c 4 (рис. 3.5).

```
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=26.2 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=1.08 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.204 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.245 ms

--- 10.0.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3022ms
rtt min/avg/max/mdev = 0.204/6.927/26.183/11.122 ms
root@mininet-vm:/home/mininet#
```

Рис. 3.5: Проверка соединения между хостами

Запустим iPerf3 на хостах и посмотрим результат отработки на данном этапе (рис. 3.6).

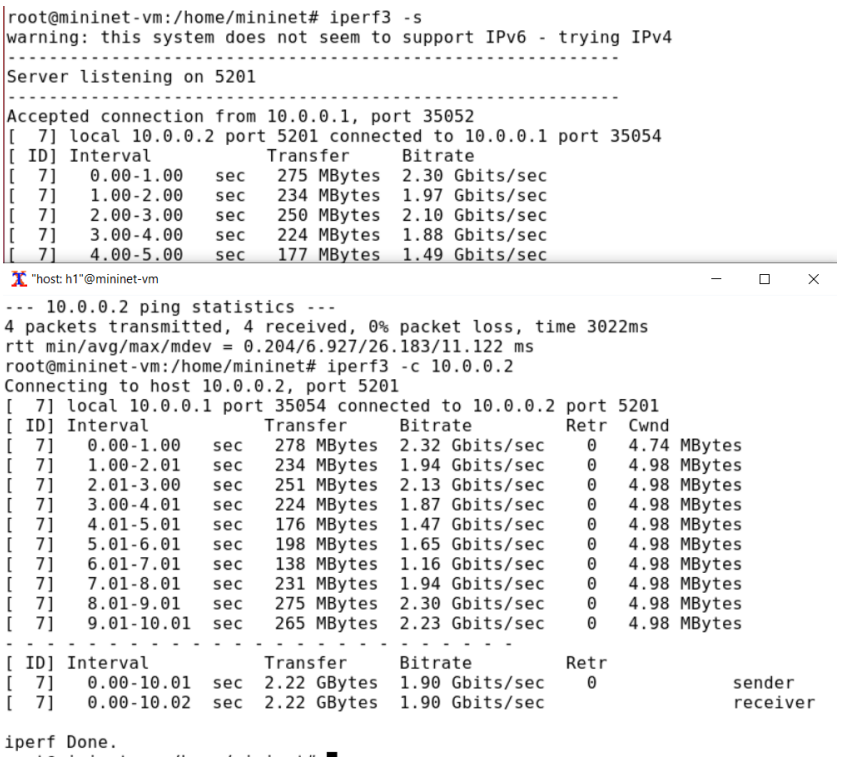

Рис. 3.6: Запуск iperf3 на хостах

3.2 Интерактивные эксперименты

Изменим пропускную способность хоста h1, установив пропускную способность на 10 Гбит/с на интерфейсе h1-eth0 и параметры TBF-фильтра (рис. 3.7).

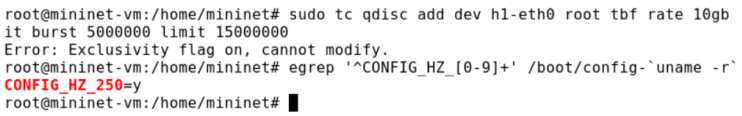

Рис. 3.7: Ограничение скорости на конечных хостах

Убедимся, что пропускная способность (Bitrate) увеличится (рис. 3.8)

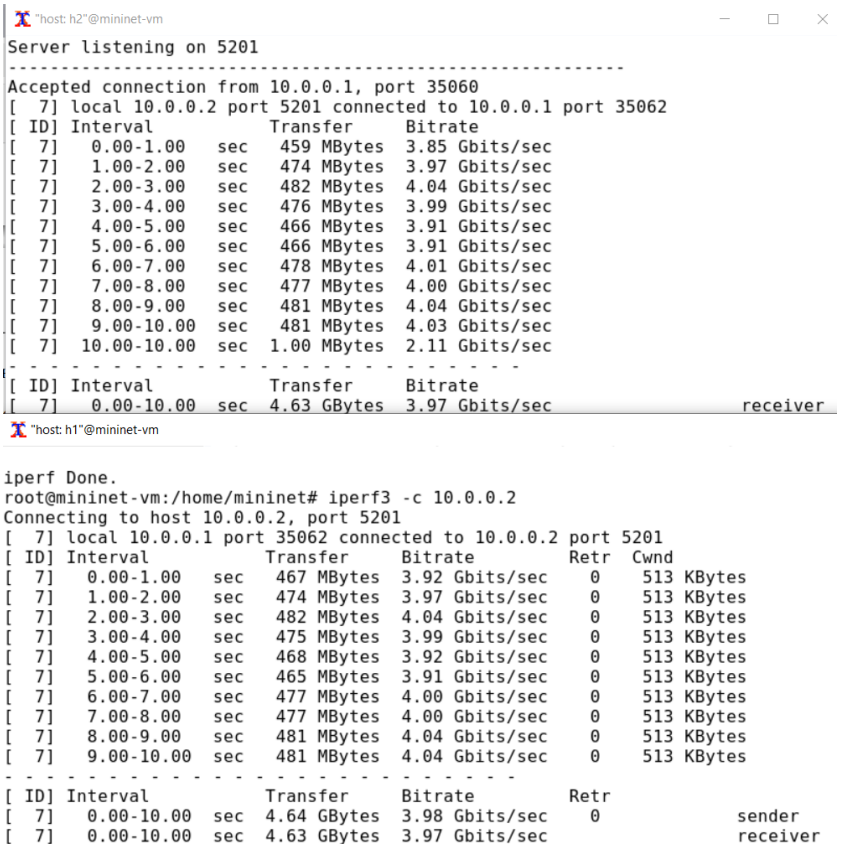

Рис. 3.8: Проверка пропускной способности

Удалим модифицированную конфигурацию 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. 3.9). Проверим примененные ограничения

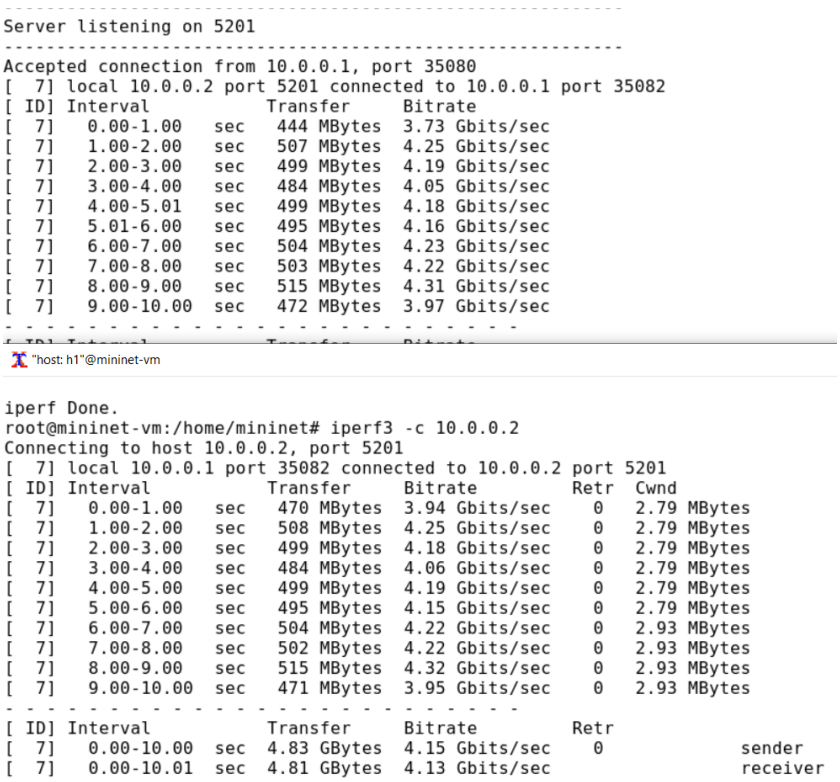

Рис. 3.9: Ограничение скорости на коммутаторах

Удалим модифицированную конфигурацию на коммутаторе s1.

Объединим NETEM и TBF, введя на интерфейсе s1-eth2 коммутатора s1 задержку, джиттер, повреждение пакетов и указав скорость. 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. 3.10).

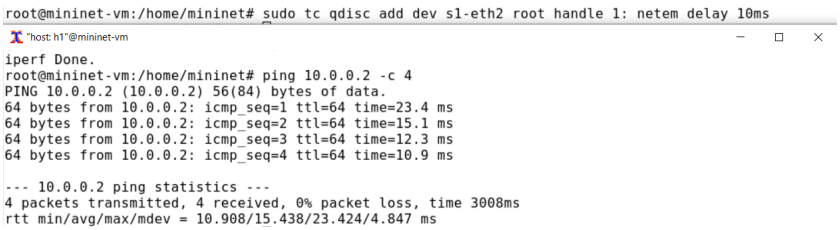

Рис. 3.10: Объединение NETEM и TBF

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: и проверим внесенные изменения с помощью iPerf3 (рис. 3.11):

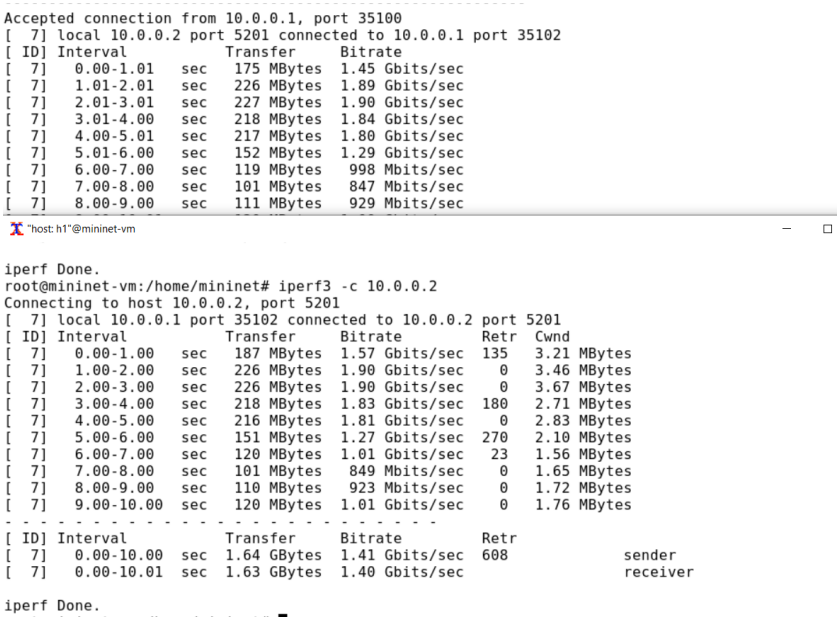

Рис. 3.11: Объединение NETEM и TBF

Удалим модифицированную конфигурацию на коммутаторе s1.

3.3 Воспроизводимые эксперименты

Самостоятельно реализуем воспроизводимые эксперименты по использованию TBF для ограничения пропускной способности. Для этого создадим рабочий каталог и файл со скриптом lab_netem_iii.py (рис. 3.12).

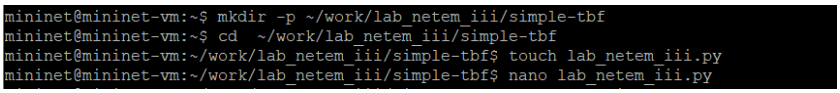

Рис. 3.12: Создание рабочего каталога

Создадим скрипт для эксперимента lab_netem_iii.py. В этом скрипте создается простейшая топология сети, затем с помощью комнанд, использованных нами ранее изменяется пропускная способность для первого хоста, после чего пингуется второй хост (100 сообщений отправляется), при этом из сообщений при пинге вытаскиваются номер сообщения и значение времени, которые записываются в файл с данными. (рис. 3.13).

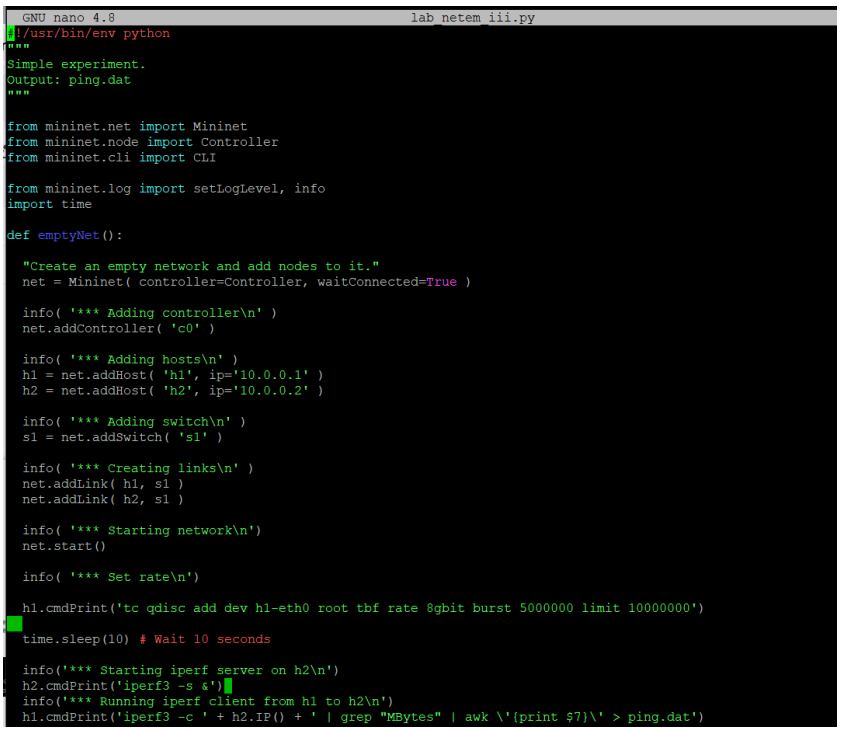

Рис. 3.13: Скрипт на Python для эксперемента

Создадим Makefile для управления процессом проведения эксперимента и записи данных в файл(рис. 3.14).

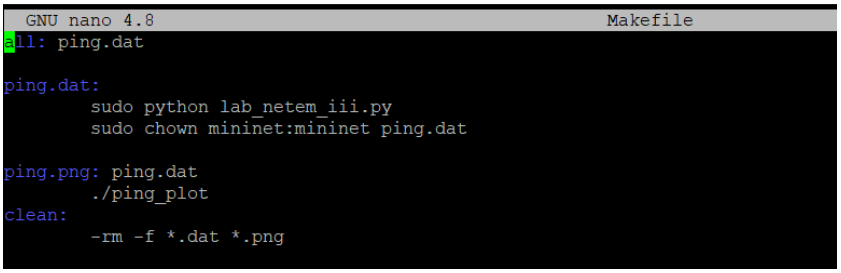

Рис. 3.14: Makefile для управления процессом проведения эксперимента

Создадим файл ping_plot для отображения данных (рис. 3.15).

Рис. 3.15: Создание файла для изображения графика

Выполним эксперимент, написав команду make (рис. 3.16).

```
mininet@mininet-vm:~/work/lab_netem_iii/simple-tbf$ make
sudo python lab_netem_iii.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 rate
*** h1 : ('tc qdisc add dev h1-eth0 root tbf rate 8gbit burst 5000000 limit 10000000',)
*** Starting iperf server on h2
*** h2 : ('iperf3 -s &',)
*** Running iperf client from h1 to h2
*** h1 : ('iperf3 -c 10.0.0.2 | grep "MBytes" | awk \'{print $7}\' > 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
```

Рис. 3.16: Запуск эксперимента

Изучим созданный график (рис. 3.17).

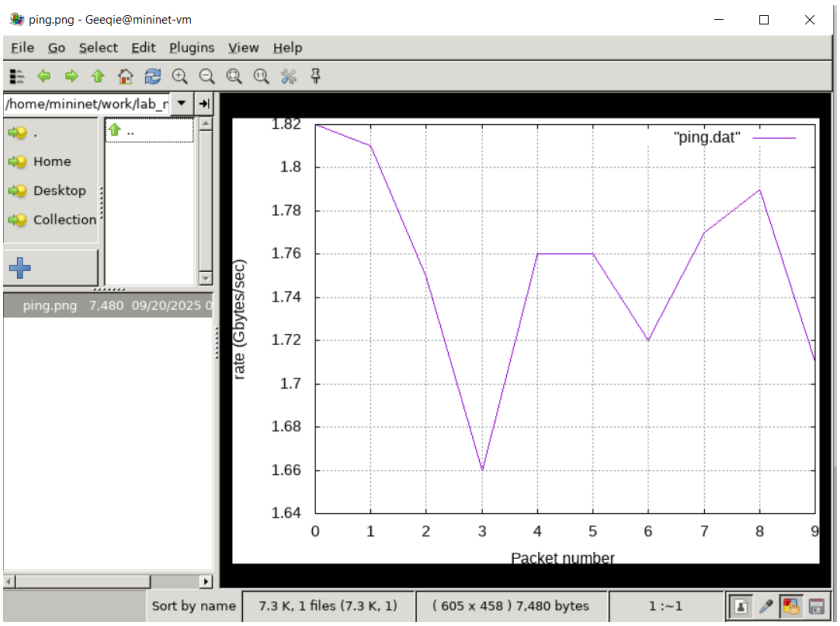

Рис. 3.17: График изменения скорости передачи

4 Выводы

В результате выполнения данной лабораторной работы 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.