Лабораторная работа №2

Измерение и тестирование пропускной способности сети. Интерактивный эксперимент

Тазаева Анастасия Анатольевна

Содержание

Список иллюстраций

Список таблиц

# 1 Цель работы

Основной целью работы является знакомство с инструментом для измерения пропускной способности сети в режиме реального времени — iPerf3, а также получение навыков проведения интерактивного эксперимента по измерению пропускной способности моделируемой сети в среде Mininet.

# 2 Задание

1. Установить на виртуальную машину mininet iPerf3 и дополнительное программное обеспечение для визуализации и обработки данных.
2. Провести ряд интерактивных экспериментов по измерению пропускной способности с помощью iPerf3 с построением графиков.

# 3 Выполнение лабораторной работы

## 3.1 Установка необходимого программного обеспечения

1. Запустила виртуальную среду с mininet.
2. Из основной ОС подключилась к виртуальной машине (рис. 1) с помощью команды ssh -Y mininet@192.168.56.101.

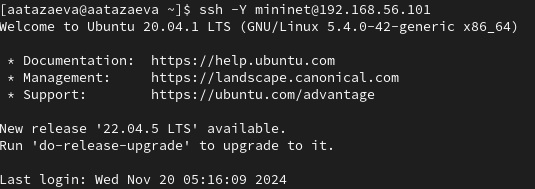


Рис. 1: Подключение к mininet через хостовой терминал

1. Проверила IP-адреса машины ifconfig (рис. 2), адрес NAT активен.

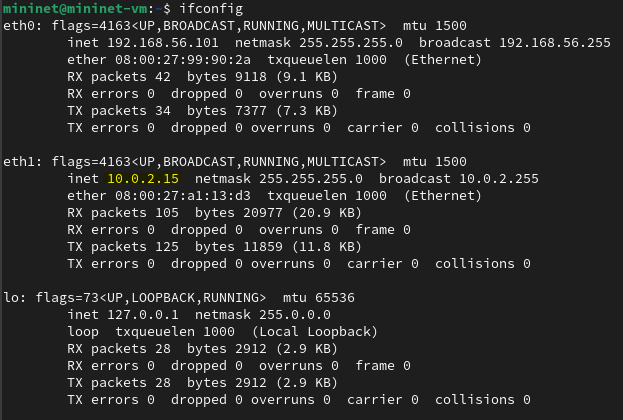


Рис. 2: ifconfig

1. Обновила репозитории ПО на виртуальной машине (рис. 3): sudo apt-get update.

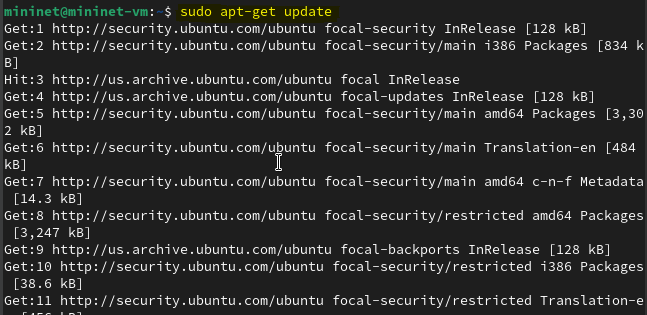


Рис. 3: Обновление репозиториев ПО

1. Установила iperf3 (рис. 4): sudo apt-get install iperf3.

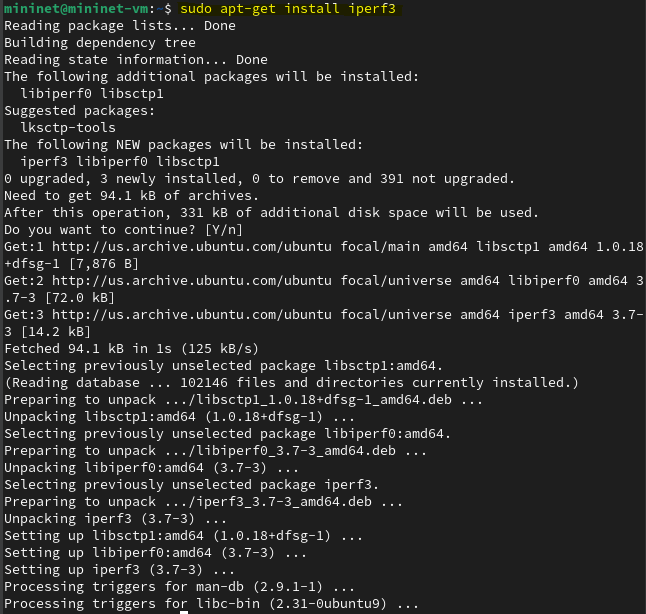


Рис. 4: Установка iperf3

1. Установила необходимое дополнительное ПО на виртуальную машину (рис. 5): sudo apt-get git jq gnuplot-nox evince.

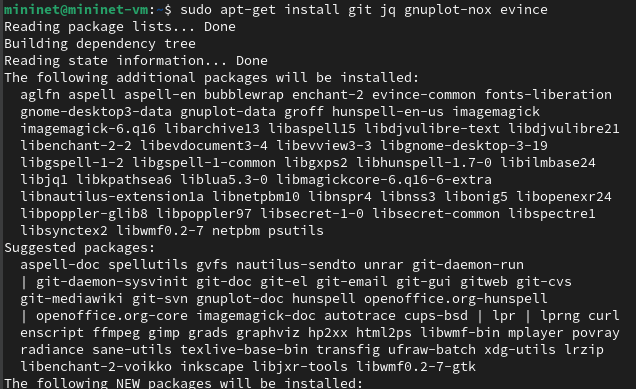


Рис. 5: Установка доп. ПО

1. Установила iperf3\_plotter, сначала скачав репозиторий во временный каталог(рис. 6), а потом перенесла исполняемые файлы в папку /usr/bin (рис. 7).



Рис. 6: Установка iperf3\_plotter. git clone

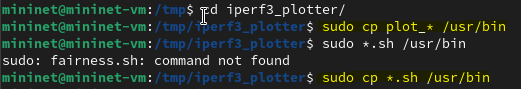


Рис. 7: Установка iperf3\_plotter. перенос файлов

## 3.2 Интерактивные эксперименты

1. Попыталась задать простейшую топологию из двух хостов и коммутатора. Появилась ошибка X11 connection rejected because of wrong authentification (рис. 8). Исправила её (рис. 9) и повторно задала простейшую топологию (рис. 10): sudo mn --topo=single,2 -x. Открылись 4 терминала (рис. 11): два хоста, коммутатор и контроллера. Терминалы коммутатора и контроллера нам не понадобятся, потому их просто закрыла.

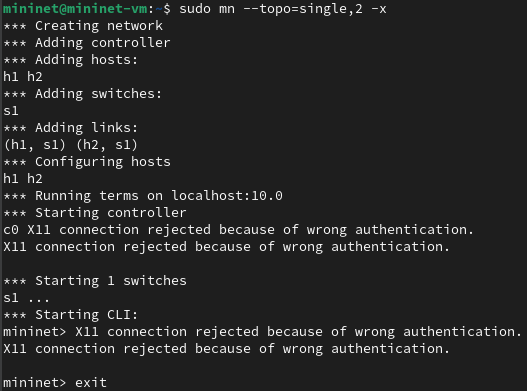


Рис. 8: Попытка задать топологию. Ошибка

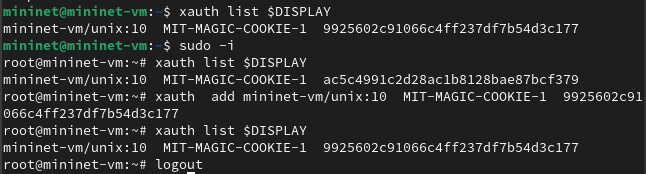


Рис. 9: Исправление ошибки

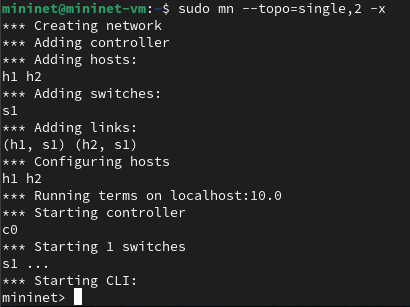


Рис. 10: Создание топологии

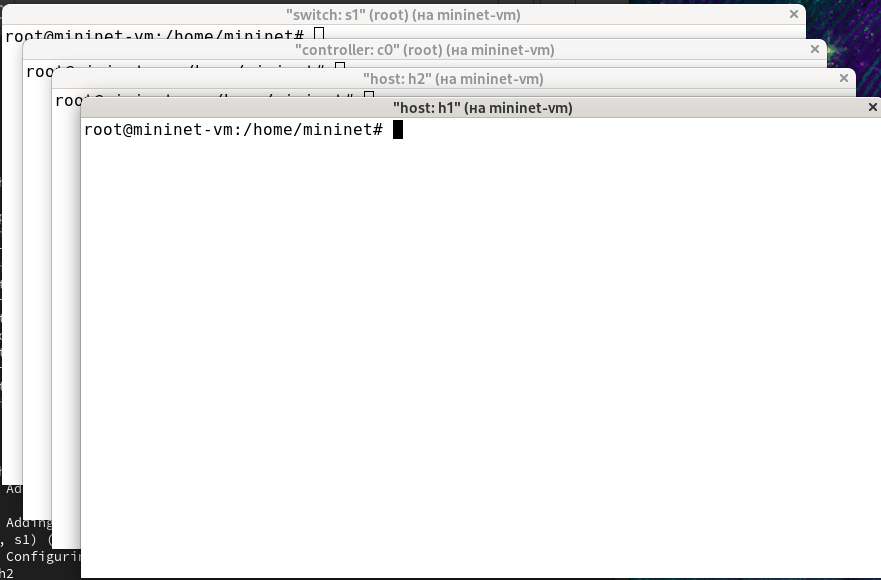


Рис. 11: Терминалы

1. Посмотрела параметры запущенной в интерактивном режиме топологии (рис. 12).

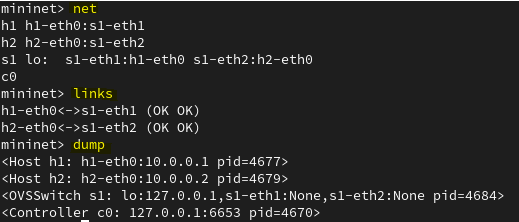


Рис. 12: Параметры топологии

1. Провела простейший интерактивный эксперимент по измерению пропускной способности с помощью iperf3. Для этого сначала в терминале хоста 2 h2 запустила *сервер* iperf3 (рис. 13): iperf3 -s. После запуска этой команды хост перешел в режим прослушивания 5201-го порта в ожидании входящих подключений. Далее после запуска сервера в терминале хоста 1 h1 запустила *клиент* iperf3 (рис. 14): iperf3 -c 10.0.0.2.

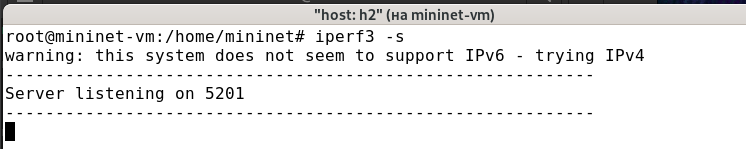


Рис. 13: Запуск сервера iperf3

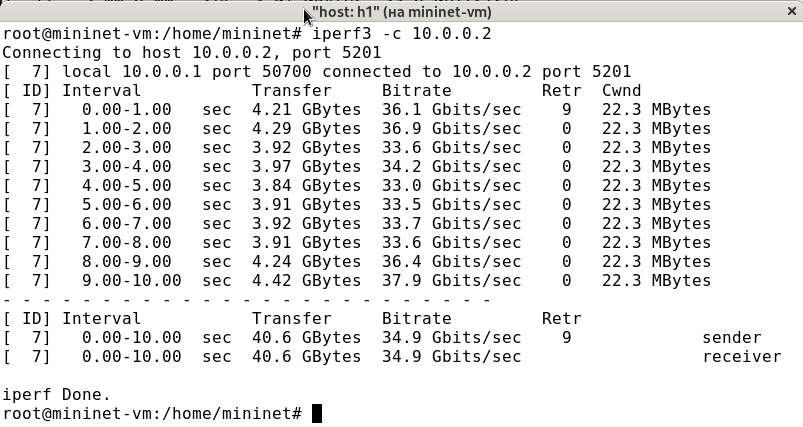


Рис. 14: Запуск клиента iperf3

Начался тест с измерением пропускной способности, который длился 10 секунд по умолчанию. В итоге мы получили следующие данные: - ID: идентификационный номер соединения. - интервал (Interval): временной интервал для периодических отчетов о пропускной способности (по умолчанию временной интервал равен 1 секунде); - передача (Transfer): сколько данных было передано за каждый интервал времени; - пропускная способность (Bitrate): измеренная пропускная способность в каждом временном интервале; - Retr: количество повторно переданных TCP-сегментов за каждый временной интервал (это поле увеличивается, когда TCP-сегменты теряются в сети из-за перегрузки или повреждения); - Cwnd: указывает размер окна перегрузки в каждом временном интервале (TCP использует эту переменную для ограничения объёма данных, которые TCP-клиент может отправить до получения подтверждения отправленных данных). Суммарные данные на сервере аналогичны данным на стороне клиента iPerf3 и должны интерпретироваться таким же образом.

1. Провела аналогичный эксперимент только в интерфейсе mininet (рис. 15 и 16).

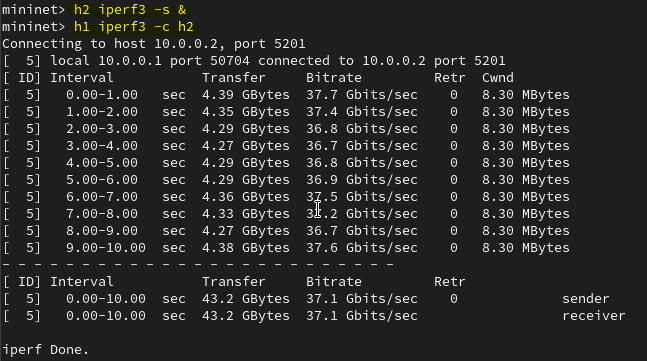


Рис. 15: Запуск сервера и клиента iperf3 в интерфейсе mininet. Измерение пропускной способности

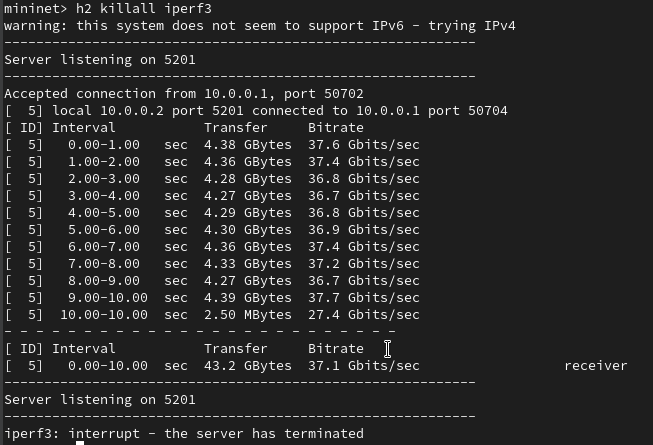


Рис. 16: Остановка серверного процесса в интерфейсе mininet

1. Для указания iPerf3 периода времени для передачи можно использовать ключ -t (или –time) — время в секундах для передачи (по умолчанию 10 секунд). Сначала запустила сервер. Далее запустила клиент iPerf3 с параметром -t, за которым следует количество секунд (рис. 17): iperf3 -c 10.0.0.2 -t 5

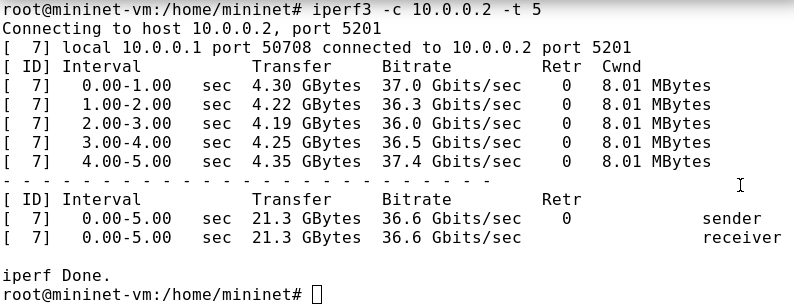


Рис. 17: Измерение пропускной способности. Опция -t

1. Настроила клиент iPerf3 для выполнения теста пропускной способности с 2-секундным интервалом времени отсчёта как на клиенте, так и на сервере. Использовала опцию -i для установки интервала между отсчётами, измеряемого в секундах:

* В терминале h2 запустила сервер iPerf3 (рис. 18): iperf3 -s -i 2
* В терминале h1 запустила клиент iPerf3 (рис. 19): iperf3 -c 10.0.0.2 -i 2

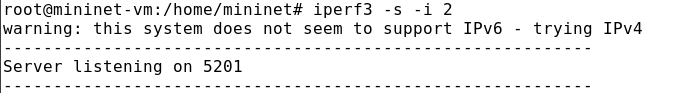


Рис. 18: Измерение пропускной способности. Опция -i. Запуск сервера

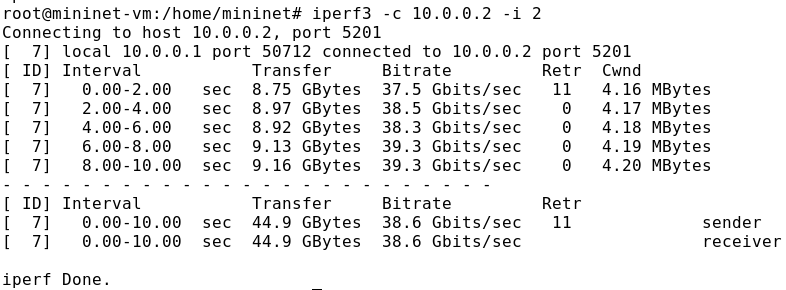


Рис. 19: Измерение пропускной способности. Опция -i. Запуск клиента

Как мы видим ничего, кроме интервала не поменялось. Общее кол-во переданных данных и пропускная способность не поменялись.

1. Задала на клиенте iPerf3 отправку определённого объёма данных. Использовала опцию -n для установки количества байт для передачи:

* В терминале h2 запустила сервер iPerf3: iperf3 -s
* В терминале h1 запустила клиент iPerf3, задав объём данных 16 Гбайт (рис. 20): iperf3 -c 10.0.0.2 -n 16G

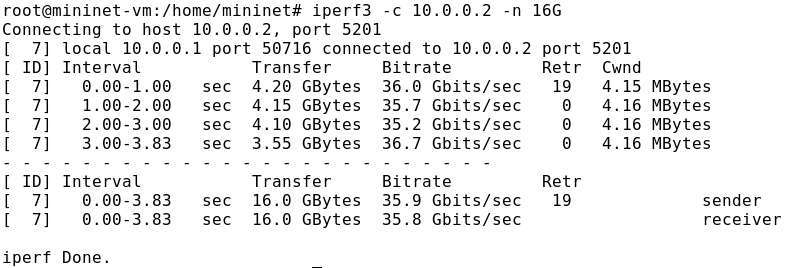


Рис. 20: Измерение пропускной способности. Опция -n. Запуск клиента

По умолчанию iPerf3 выполняет измерение пропускной способности в течение 10 секунд, но при задании количества данных для передачи клиент iPerf3 будет продолжать отправлять пакеты до тех пор, пока не будет отправлен весь объем данных, указанный пользователем.

1. Изменила в тесте измерения пропускной способности iPerf3 протокол передачи данных с TCP (установлен по умолчанию) на UDP. iPerf3 автоматически определяет протокол транспортного уровня на стороне сервера. Для изменения протокола использовала опцию -u на стороне клиента iPerf3:

* В терминале h2 запустила сервер iPerf3: iperf3 -s
* В терминале h1 запустила клиент iPerf3, задав протокол UDP (рис. 21): iperf3 -c 10.0.0.2 -u

После завершения теста отобразились следующие сводные данные: - ID, интервал, передача, битрейт: то же, что и у TCP. - Jitter: разница в задержке пакетов. - Lost/Total: указывает количество потерянных дейтаграмм по сравнению с общим количеством отправленных на сервер (и процентное соотношение).

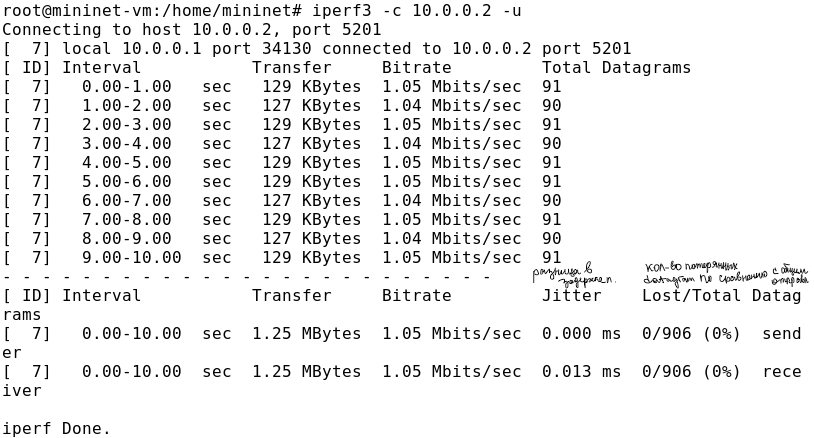


Рис. 21: Измерение пропускной способности. Опция -u. Запуск клиента

1. В тесте измерения пропускной способности iPerf3 изменила номер порта для отправки/получения пакетов или датаграмм через указанный порт. Использовала для этого опцию -p:

* В терминале h2 запустила сервер iPerf3, используя параметр -p, чтобы указать порт прослушивания (рис. 22): iperf3 -s -p 3250
* В терминале h1 запустила клиент iPerf3, указав порт (рис. 23): iperf3 -c 10.0.0.2 -p 3250

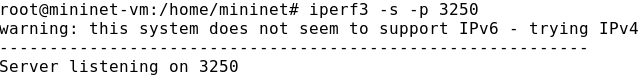


Рис. 22: Измерение пропускной способности. Опция -p. Запуск сервера

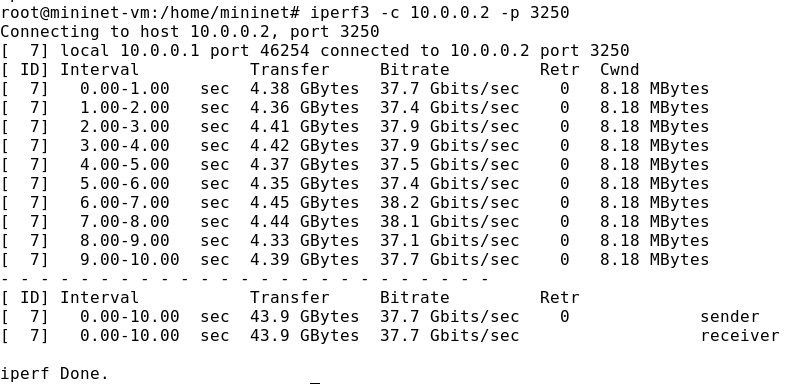


Рис. 23: Измерение пропускной способности. Опция -p. Запуск клиента

1. По умолчанию после запуска сервер iPerf3 постоянно прослушивает входящие соединения. В тесте измерения пропускной способности iPerf3 задала для сервера параметр обработки данных только от одного клиента с остановкой сервера по завершении теста. Для этого использовала опцию -1 на сервере iPerf3:

* В терминале h2 запустила сервер iPerf3, используя параметр -1, чтобы принять только одного клиента (рис. 24): iperf3 -s -1
* В терминале h1 запустила клиент iPerf3: iperf3 -c 10.0.0.2

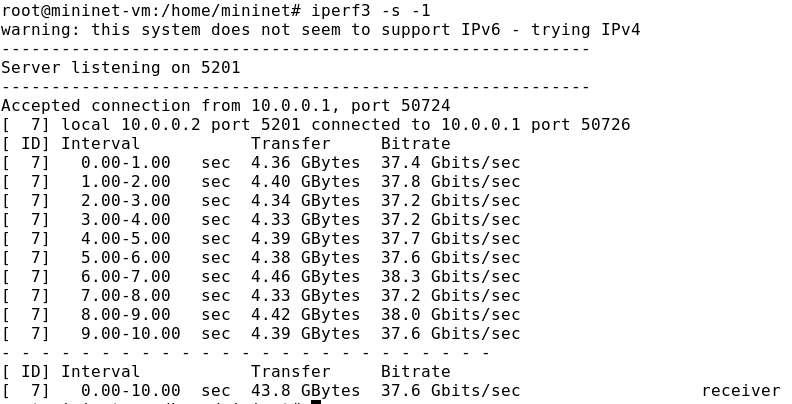


Рис. 24: Измерение пропускной способности. Опция -1. Запуск сервера

После завершения этого теста сервер iPerf3 немедленно останавливается.

1. Экспортировала результаты теста измерения пропускной способности iPerf3 в файл JSON. Для этого в виртуальной машине mininet создала каталог для работы над проектом: mkdir -p ~/work/lab\_iperf3. В терминале h2 запустила сервер iPerf3: iperf3 -s. В терминале h1 запустила клиент iPerf3, указав параметр -J для отображения вывода результатов в формате JSON (рис. 25): iperf3 -c 10.0.0.2 -J.

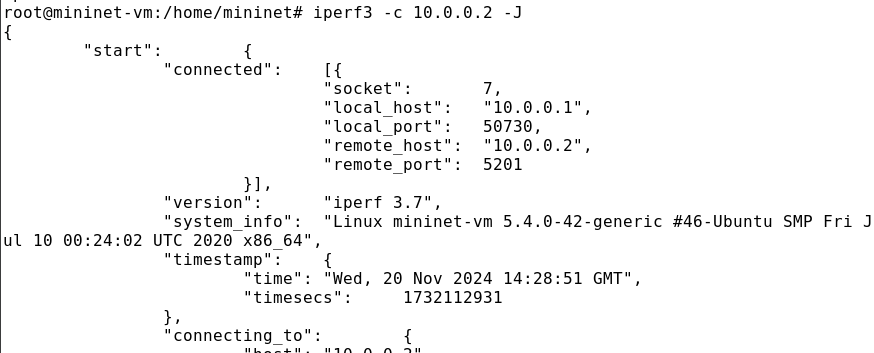


Рис. 25: Измерение пропускной способности. Опция -J. Запуск клиента.

В данном случае параметр -J выведет текст JSON на экран через стандартный вывод (stdout) после завершения теста. Экспортировала вывод результатов теста в файл, перенаправив стандартный вывод в файл (рис. 26): iperf3 -c 10.0.0.2 -J > /home/mininet/work/lab\_iperf3/iperf\_results.json Убедилась, что файл *iperf\_results.json* создан в указанном каталоге. Для этого в терминале хоста h1 ввела следующие команды (рис. 27): cd/home/mininet/work/lab\_iperf3 и ls -l.

Рис. 26: Перенаправление вывода в файл

Рис. 26: Перенаправление вывода в файл

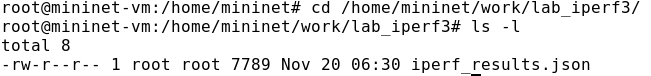


Рис. 27: Проверка создания файла с выводом информации

1. Так как мы уже исправили права запуска X-соединения выше, то этот пункт пропустила.
2. В виртуальной машине mininet перешла в каталог для работы над проектом, проверила права, скорректировала их (рис. 28).

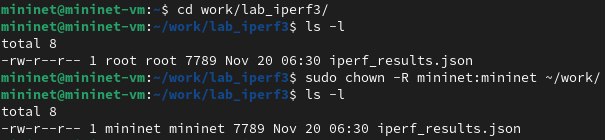


Рис. 28: Изменение прав доступа к файлу JSON

1. Сгенерировала выходные данные для файла JSON Iperf3 (рис. 29): plot\_iperf.sh iperf\_results.json

Рис. 29: Генерация выходных данных для файла JSON

Рис. 29: Генерация выходных данных для файла JSON

1. Сценарий построения создал файл CSV (1.dat) (рис. 30), который может использоваться другими приложениями. В подкаталоге results каталога, в котором был выполнен скрипт, сценарий создал графики для следующих полей файла JSON:

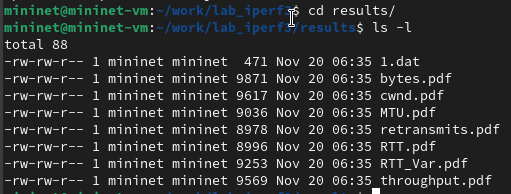


Рис. 30: Файлы с графиками

* окно перегрузки (cwnd.pdf) (рис. 31);

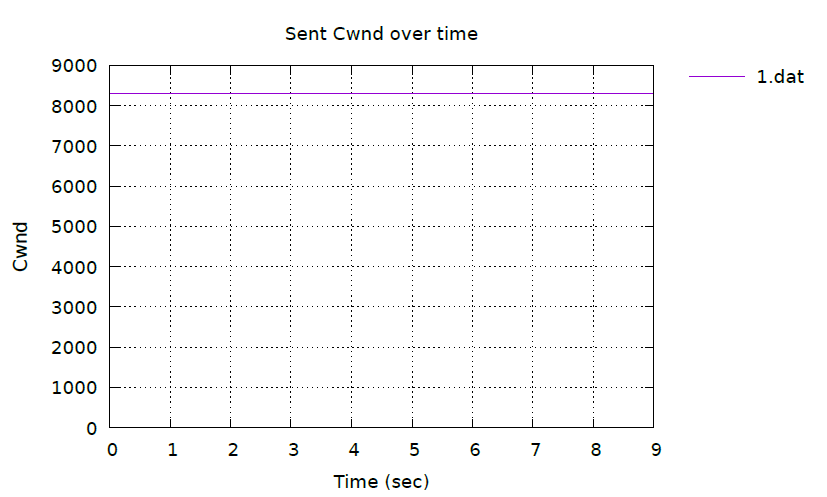


Рис. 31: cwnd.pdf

* повторная передача (retransmits.pdf) (рис. 32);

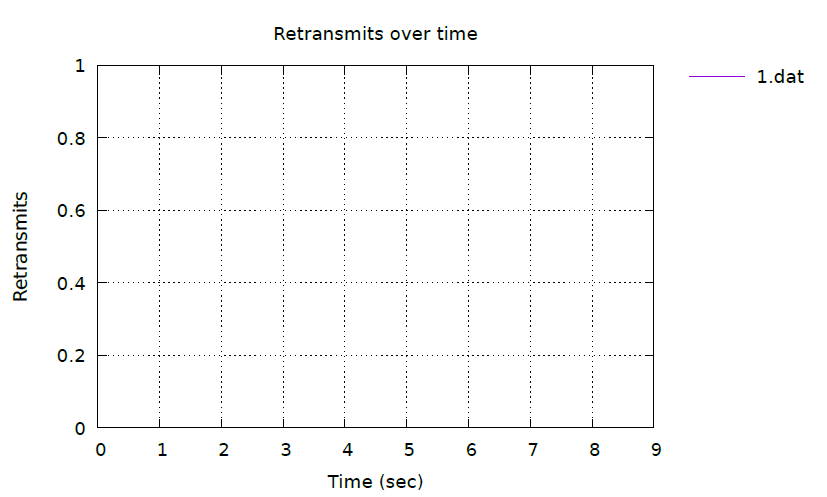


Рис. 32: retransmits.pdf

* время приема-передачи (RTT.pdf) (рис. 33);

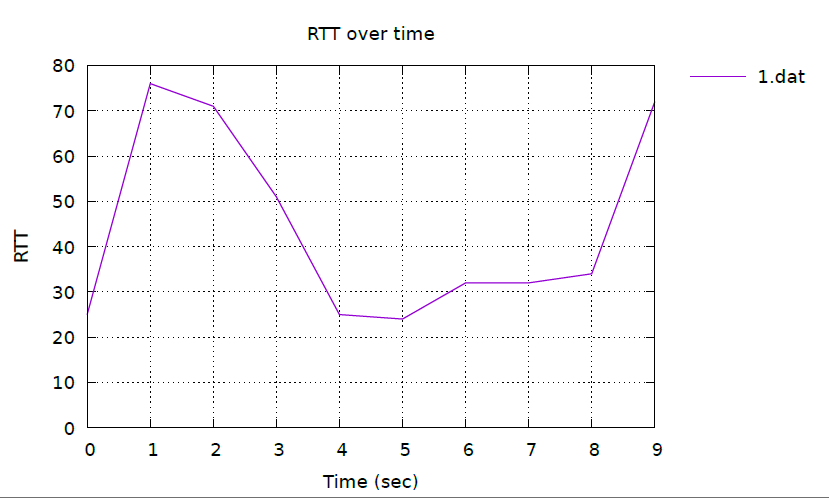


Рис. 33: RTT.pdf

* отклонение времени приема-передачи (RTT\_Var.pdf) (рис. 34);

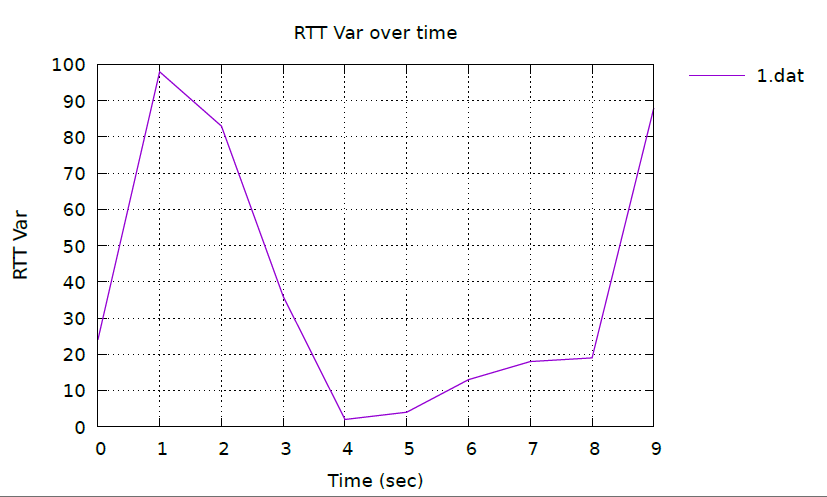


Рис. 34: RTT\_Var.pdf

* пропускная способность (throughput.pdf) (рис. 35);

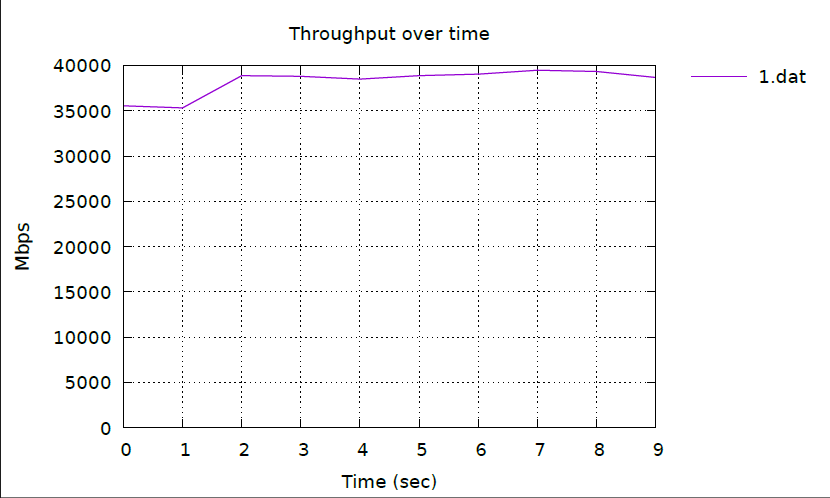


Рис. 35: throughput.pdf

* максимальная единица передачи (MTU.pdf) (рис. 36);

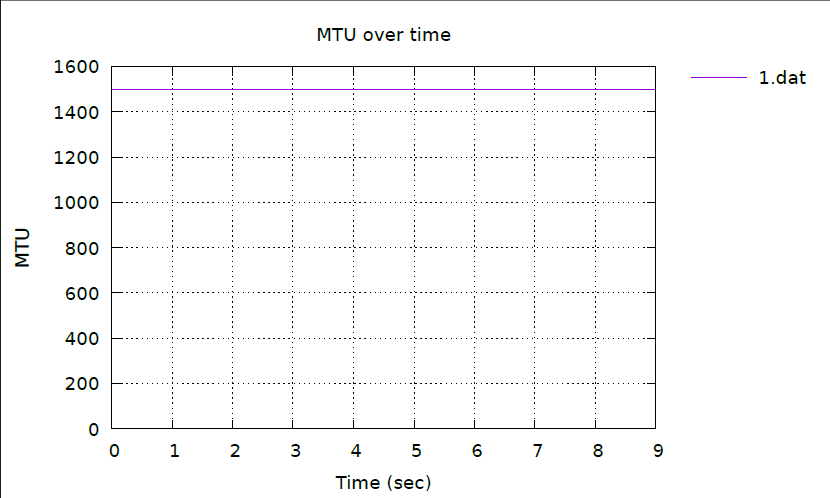


Рис. 36: MTU.pdf

* количество переданных байтов (bytes.pdf)(рис. 37).



Рис. 37: bytes.pdf

# 4 Выводы

В ходе лабораторной работы я познакомилась с инструментом для измерения пропускной способности сети в режиме реального времени — iPerf3, а также получила навыки проведения интерактивного эксперимента по измерению пропускной способности моделируемой сети в среде Mininet.