Отчёт по лабораторной работе №2

Моделирование сетей передачи данных

Ищенко Ирина Олеговна

Содержание

# 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 обеспечения для визуализации и обработки данных.
2. Провести ряд интерактивных экспериментов по измерению пропускной способности с помощью iPerf3 с построением графиков.

# 3. Выполнение лабораторной работы

Запустим виртуальную среду с mininet. Из основной ОС подключимся к виртуальной машине по SSH и активируем второй интерфейс для выхода в сеть (рис. [Рисунок 1](#fig-001)).

|  |
| --- |
| Рисунок 1: Подключение к mininet по ssh |

Обновим репозитории программного обеспечения на виртуальной машине. Установим iperf3 и необходимое дополнительное программное обеспечение на виртуальную машину. Развернем iperf3\_plotter (рис. [Рисунок 2](#fig-002)).

|  |
| --- |
| Рисунок 2: Развертывание iperf3\_plotter |

Зададим простейшую топологию, состоящую из двух хостов и коммутатора с назначенной по умолчанию mininet сетью 10.0.0.0/8. После введения этой команды запустятся терминалы двух хостов, коммутатора и контроллера. Терминалы коммутатора и контроллера закроем. В терминале виртуальной машины посмотрим параметры запущенной в интерактивном режиме топологии (рис. [Рисунок 3](#fig-003)).

|  |
| --- |
| Рисунок 3: Задание простейшей топологии. Параметры |

Проведем простейший интерактивный эксперимент по измерению пропускной способности с помощью iPerf3 (рис. [Рисунок 4](#fig-004)).

|  |
| --- |
| Рисунок 4: Тестовое соединение между хостами |

Проанализируем полученный в результате выполнения теста сводный отчёт, отобразившийся как на клиенте, так и на сервере iPerf3. Он содержет следующие данные:

* ID: идентификационный номер соединения – 7.
* 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 – было пепредано от 1.89 до 2.20 GB в секунду;
* пропускная способность (Bitrate): измеренная пропускная способность в каждом временном интервале – от 16.3 до 18.9 Gbit/sec;
* Retr: количество повторно переданных TCP-сегментов за каждый временной интервал (это поле увеличивается, когда TCP-сегменты теряются в сети из-за перегрузки или повреждения) – чем больше пропускная способность, тем больше число повторно переданных TCP-сегментов. Равен 0;
* 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) – от 461 KB до 6.27 MB.

Проведем аналогичный эксперимент в интерфейсе mininet. Сравним результаты. Увидим, что во втором случае было передано на 1,1 GB меньше; пропускная способность увеличилась на 4,1; потери пакетов все также нет.

После убьем процесс на сервере (рис. [Рисунок 5](#fig-005)).

|  |
| --- |
| Рисунок 5: Эксперимент в интерфейсе mininet |

Для указания iPerf3 периода времени для передачи можно использовать ключ -t (или –time) — время в секундах для передачи (по умолчанию 10 секунд) (рис. [Рисунок 6](#fig-006)).

|  |
| --- |
| Рисунок 6: Указание периода времени передачи |

Настроем клиент iPerf3 для выполнения теста пропускной способности с 2-секундным интервалом времени отсчёта как на клиенте, так и на сервере. Используем опцию -i для установки интервала между отсчётами, измеряемого в секундах (рис. [Рисунок 7](#fig-007)).

|  |
| --- |
| Рисунок 7: Выполнения теста пропускной способности с 2-секундным интервалом |

Можно увидеть, что действительно интервал увеличился в два раза, в результате чего в два раза увеличился также вес переданный за один интервал времени, но пропускная способность и суммарные величины очевидно практически не изменились.

Зададим на клиенте iPerf3 отправку определённого объёма данных. Используем опцию -n для установки количества байт для передачи (рис. [Рисунок 8](#fig-008)).

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.

|  |
| --- |
| Рисунок 8: Задание в тесте определённого объёма данных |

Изменим 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 используем опцию -u на стороне клиента iPerf3 (рис. [Рисунок 9](#fig-009)).

|  |
| --- |
| Рисунок 9: Изменение протокола передачи данных |

В тесте измерения пропускной способности iPerf3 изменим номер порта для отправки/получения пакетов или датаграмм через указанный порт (рис. [Рисунок 10](#fig-010)).

|  |
| --- |
| Рисунок 10: Изменение номера порта для отправки/получения пакетов |

По умолчанию после запуска сервер iPerf3 постоянно прослушивает входящие соединения. В тесте измерения пропускной способности iPerf3 зададим для сервера параметр обработки данных только от одного клиента с остановкой сервера по завершении теста. Для этого используем опцию -1 на сервере iPerf3 (рис. [Рисунок 11](#fig-011)).

После завершения этого теста сервер iPerf3 немедленно останавливается.

|  |
| --- |
| Рисунок 11: Параметр обработки данных только от одного клиента с остановкой сервера по завершении теста |

Экспортируем результаты теста измерения пропускной способности iPerf3 в файл JSON (рис. [Рисунок 12](#fig-012)) и (рис. [Рисунок 13](#fig-013)).

|  |
| --- |
| Рисунок 12: iPerf3 в JSON |

|  |
| --- |
| Рисунок 13: Экспорт результатов теста измерения пропускной способности iPerf3 в файл JSON |

Убедимся, что файл iperf\_results.json создан в указанном каталоге. Для этого в терминале ВМ введем следующие команды (рис. [Рисунок 14](#fig-014)).

|  |
| --- |
| Рисунок 14: Проверка создания файла iperf\_results.json |

Конец эмуляции (рис. [Рисунок 15](#fig-015)).

|  |
| --- |
| Рисунок 15: Конец эмуляции |

В виртуальной машине mininet исправим права запуска X-соединения. Скопируем значение куки (MIT magic cookie) своего пользователя mininet в файл для пользователя root. Визуализируем результаты эксперимента. В виртуальной машине mininet перейдем в каталог для работы над проектом, проверим права доступа к файлу JSON. Сгенерируем выходные данные для файла JSON iPerf3. Убедимся, что файлы с данными и графиками сформировались (рис. [Рисунок 16](#fig-016)).

|  |
| --- |
| Рисунок 16: Визуализация результатов эксперимента |

# 4. Выводы

В результате выполнения 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.