Лабораторная работа № 5

Моделирование сетей передачи данных

Доберштейн Алина Сергеевна

Содержание

Список иллюстраций

# 1 Цель работы

Основной целью работы является получение навыков проведения интерактивных экспериментов в среде Mininet по исследованию параметров сети, связанных с потерей, дублированием, изменением порядка и повреждением пакетов при передаче данных. Эти параметры влияют на производительность протоколов и сетей.

# 2 Задание

1. Задайте простейшую топологию, состоящую из двух хостов и коммутатора с назначенной по умолчанию mininet сетью 10.0.0.0/8.
2. Проведите интерактивные эксперименты по по исследованию параметров сети, связанных с потерей, дублированием, изменением порядка и повреждением пакетов при передаче данных.
3. Реализуйте воспроизводимый эксперимент по добавлению правила отбрасывания пакетов в эмулируемой глобальной сети. На экран выведите сводную информацию о потерянных пакетах.
4. Самостоятельно реализуйте воспроизводимые эксперименты по исследованию параметров сети, связанных с потерей, изменением порядка и повреждением пакетов при передаче данных. На экран выведите своднуюинформацию о потерянных пакетах.

# 3 Теоретическое введение

Mininet – это эмулятор компьютерной сети. Под компьютерной сетью подразумеваются простые компьютеры — хосты, коммутаторы, а так же OpenFlow-контроллеры. С помощью простейшего синтаксиса в примитивном интерпретаторе команд можно разворачивать сети из произвольного количества хостов, коммутаторов в различных топологиях и все это в рамках одной виртуальной машины(ВМ). На всех хостах можно изменять сетевую конфигурацию, пользоваться стандартными утилитами(ifconfig, ping) и даже получать доступ к терминалу. На коммутаторы можно добавлять различные правила и маршрутизировать трафик.

# 4 Выполнение лабораторной работы

Запустила виртуальную среду с mininet. Подключилась из ОС. Проверила права запуска Х-соединения. Задала простейшую топологию, состоящую из двух хостов и коммутатора с назначенной по умолчанию mininet сетью 10.0.0.0/8. На хостах h1 и h2 ввела 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. 1).

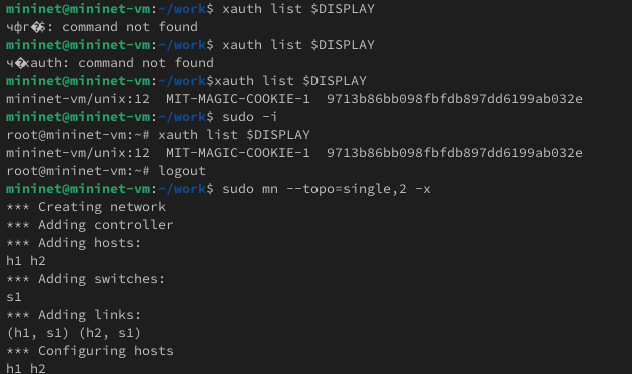


Рис. 1: Запуск простой топологии

Проверила подключение между хостами h1 и h2 с помощью команды ping с параметром -c 6 (рис. 2).

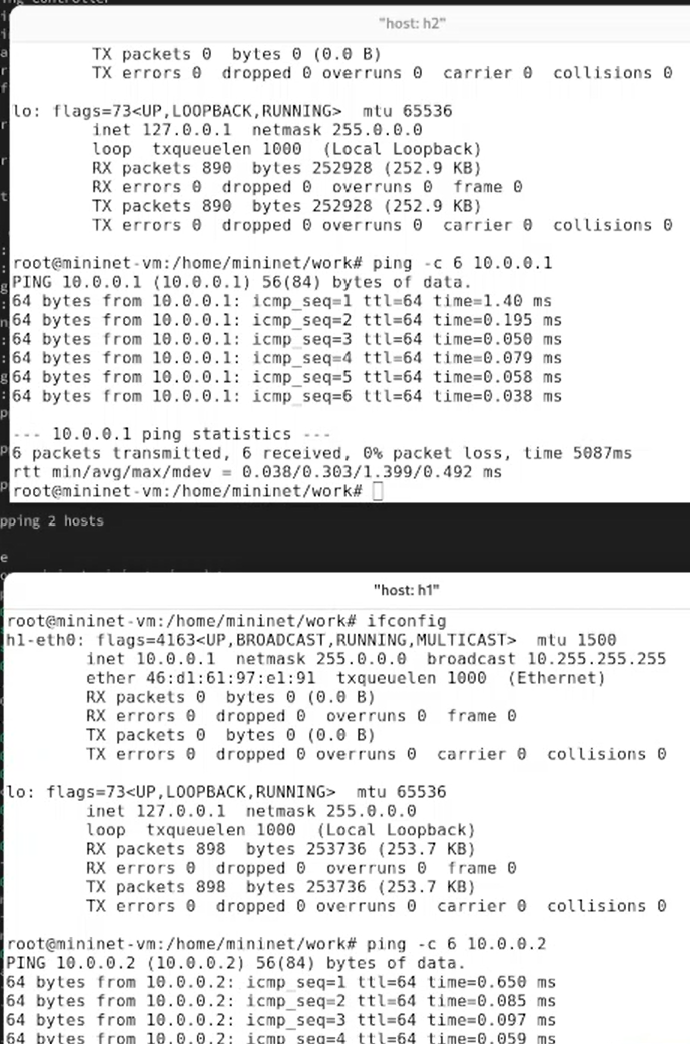


Рис. 2: Проверка соединения между хостами

# 5 Интерактивные эксперименты

## 5.1 Добавление потери пакетов на интерфейс,подключённый к эмулируемой глобальной сети

Пакеты могут быть потеряны в процессе передачи из-за таких факторов, как битовые ошибки и перегрузка сети. Скорость потери данных часто измеряется как процентная доля потерянных пакетов по отношению к количеству отправленных пакетов. На хосте h1 добавила 10% потерь пакетов к интерфейсу h1-eth0 (рис. 3-4).

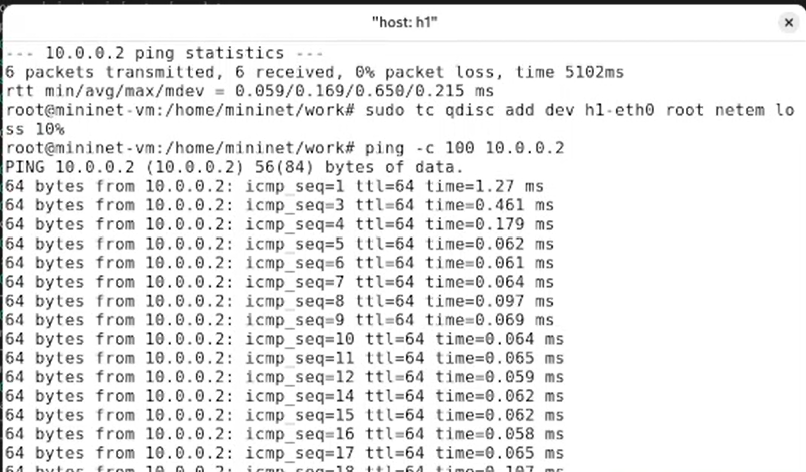


Рис. 3: Добавление потери пакетов

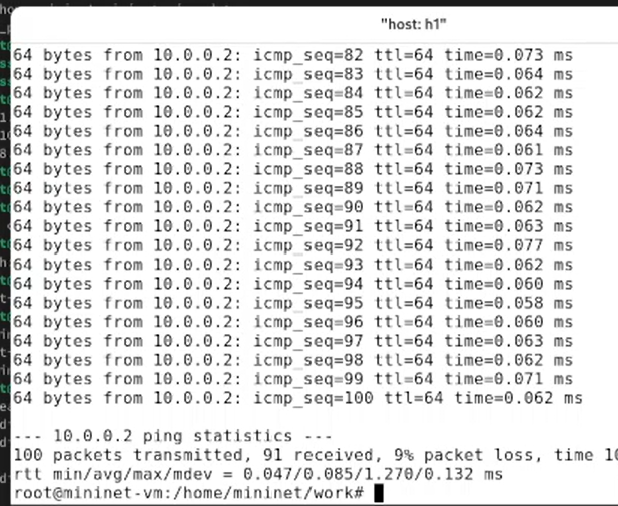


Рис. 4: Добавление потери пакетов

Здесь:

* sudo: выполнить команду с более высокими привилегиями;
* tc: вызвать управление трафиком Linux;
* qdisc: изменить дисциплину очередей сетевого планировщика;
* add: создать новое правило;
* dev h1-eth0: указать интерфейс, на котором будет применяться правило;
* netem: использовать эмулятор сети;
* loss 10%: 10% потерь пакетов.

Для эмуляции глобальной сети с потерей пакетов в обоих направлениях необходимо к соответствующему интерфейсу на хосте h2 также добавить 10% потерь пакетов. Проверила, что соединение между хостом h1 и хостом h2 имеет больший процент потерянных данных (10% от хоста h1 к хосту h2 и 10% от хоста h2 к хосту h1), повторив команду ping с параметром -c 100 на терминале хоста h1 (рис. 5).

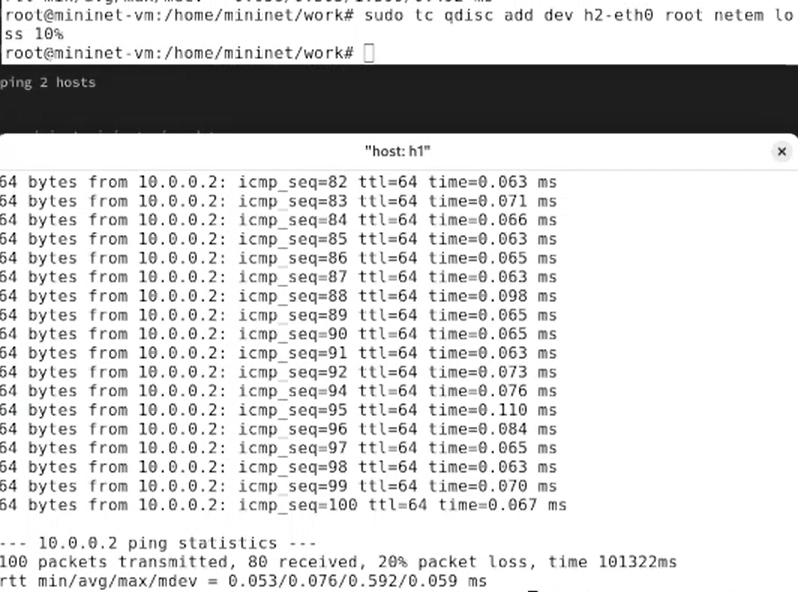


Рис. 5: Добавление потери пакетов

Восстановила конфигурацию по умолчанию. (рис. 6).

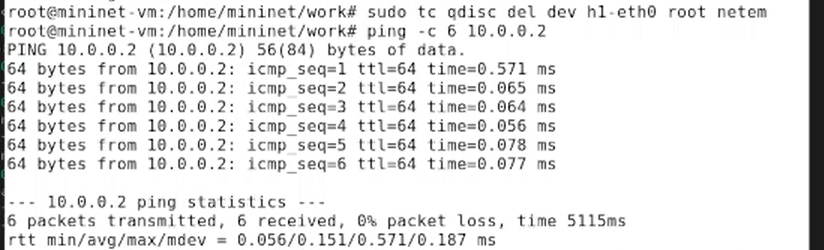


Рис. 6: Конфигурация по умолчанию

## 5.2 Добавление значения корреляции для потери пакетов в эмулируемой глобальной сети

Добавила на интерфейсе узла h1 коэффициент потери пакетов 50% (такой высокий уровень потери пакетов маловероятен), и каждая последующая вероятность зависит на 50% от последней: Проверила, что на соединении от хоста h1 к хосту h2 имеются потери пакетов, используя команду ping с параметром -c 50 с хоста h1 (рис. 7-8).

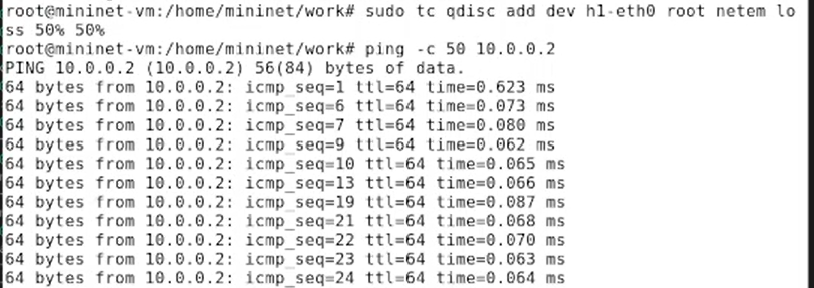


Рис. 7: Добавление значения корреляции для потери пакетов

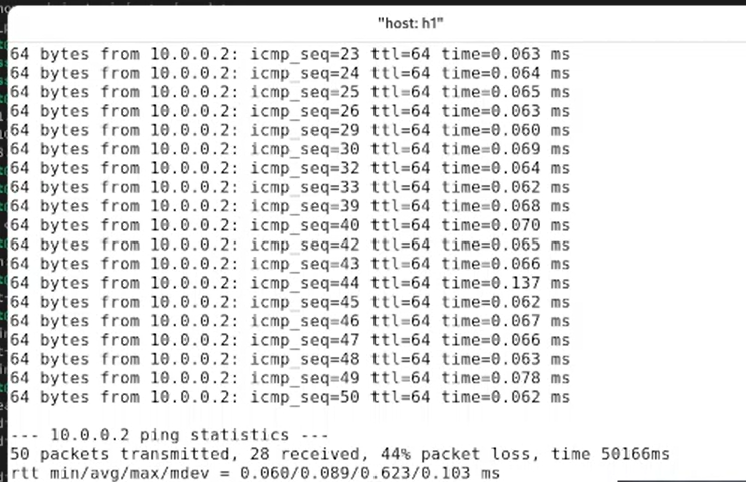


Рис. 8: Добавление значения корреляции для потери пакетов

## 5.3 Добавление повреждения пакетов в эмулируемой глобальной сети

Добавила на интерфейсе узла h1 0,01% повреждения пакетов. Проверила конфигурацию с помощью инструмента iPerf3 для проверки повторных передач (рис. 9).

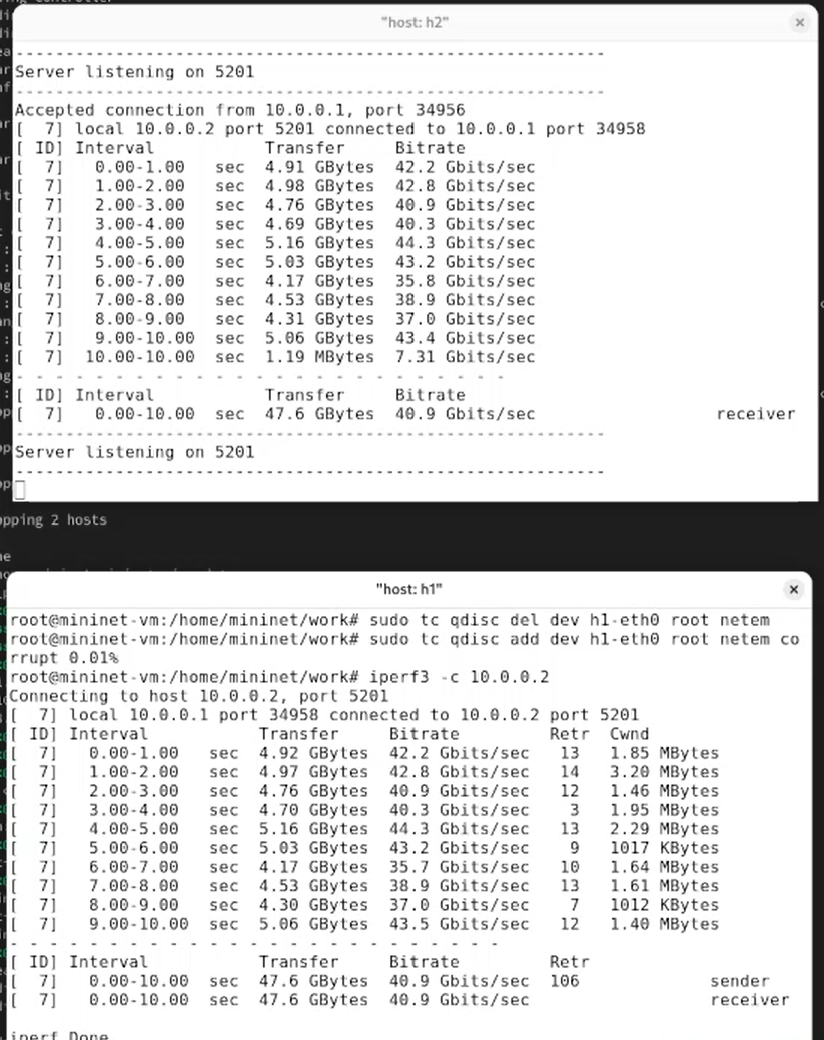


Рис. 9: Добавление повреждения пакетов

Значения повторной передачи на каждом временном интервале можно посмотреть в столбце Retr. Например, на 0.00-1.00 интервале retr=13, на следующем 14 и т.д. Всего 106 повторно отправленных пакетов.

## 5.4 Добавление переупорядочивания пакетов в интерфейс подключения к эмулируемой глобальной сети

Добавим на интерфейсе узла h1 следующее правило: 25% пакетов (со значением корреляции 50%) будут отправлены немедленно, а остальные 75% будут задержаны на 10 мс. Проверим, что на соединении от хоста h1 к хосту h2 имеются потери пакетов,используя команду ping с параметром -c 20 с хоста h1. Убедимся, что часть пакетов не будут иметь задержки (один из четырех, или 25%), а последующие несколько пакетов будут иметь задержку около 10 миллисекунд (три из четырех, или 75%) (рис. 10).

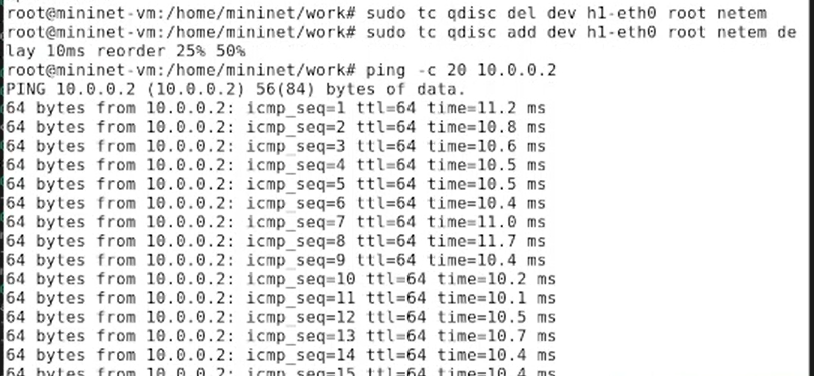


Рис. 10: Добавление переупорядочивания пакетов

Изменила значение корреляции для более вероятного результата. (рис. 11).



Рис. 11: Добавление переупорядочивания пакетов

## 5.5 Добавление дублирования пакетов в интерфейс подключения к эмулируемой глобальной сети

Для интерфейса узла h1 зададим правило c дублированием 50% пакетов (т.е. 50% пакетов должны быть получены дважды): Проверим, что на соединении от хоста h1 к хосту h2 имеются дублированные пакеты, используя команду ping с параметром -c 20 с хоста h1. Дубликаты пакетов помечаются как DUP! (рис. 12).

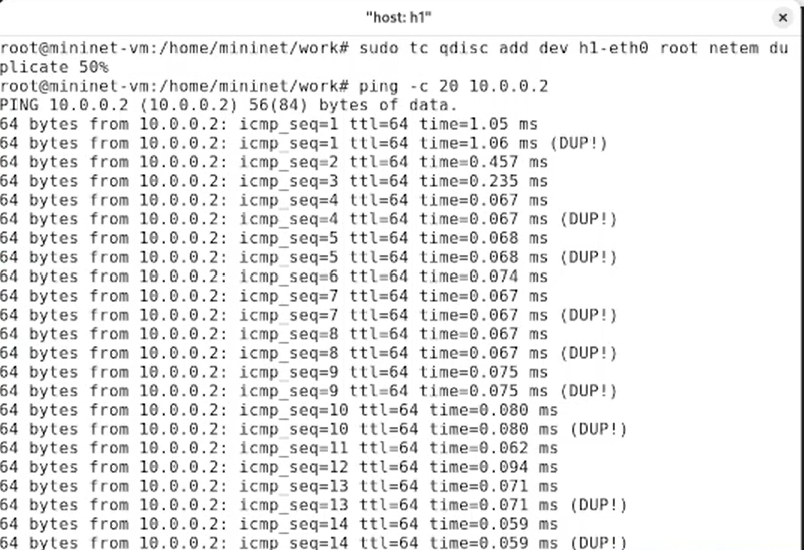


Рис. 12: Добавление дублирования пакетов

# 6 Воспроизводимые эксперименты

## 6.1 Добавление потери пакетов

В виртуальной среде mininet в своём рабочем каталоге с проектами создадим каталог simple-drop и перейдем в него. Создадим скрипт для эксперимента lab\_netem\_ii.py (рис. 13).

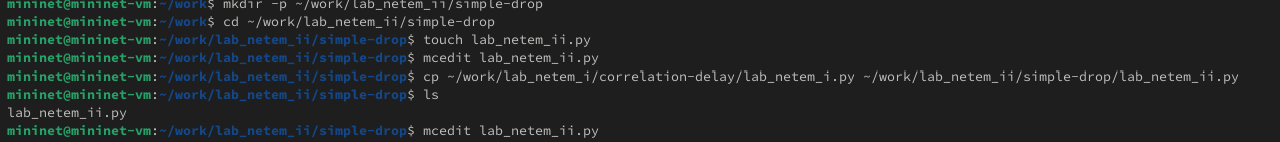


Рис. 13: Создание скрипта для эксперимента lab\_netem\_ii.py

Скорректируем скрипт так, чтобы на экран или в отдельный файл выводилась информация о потерях пакетов (рис. 14).

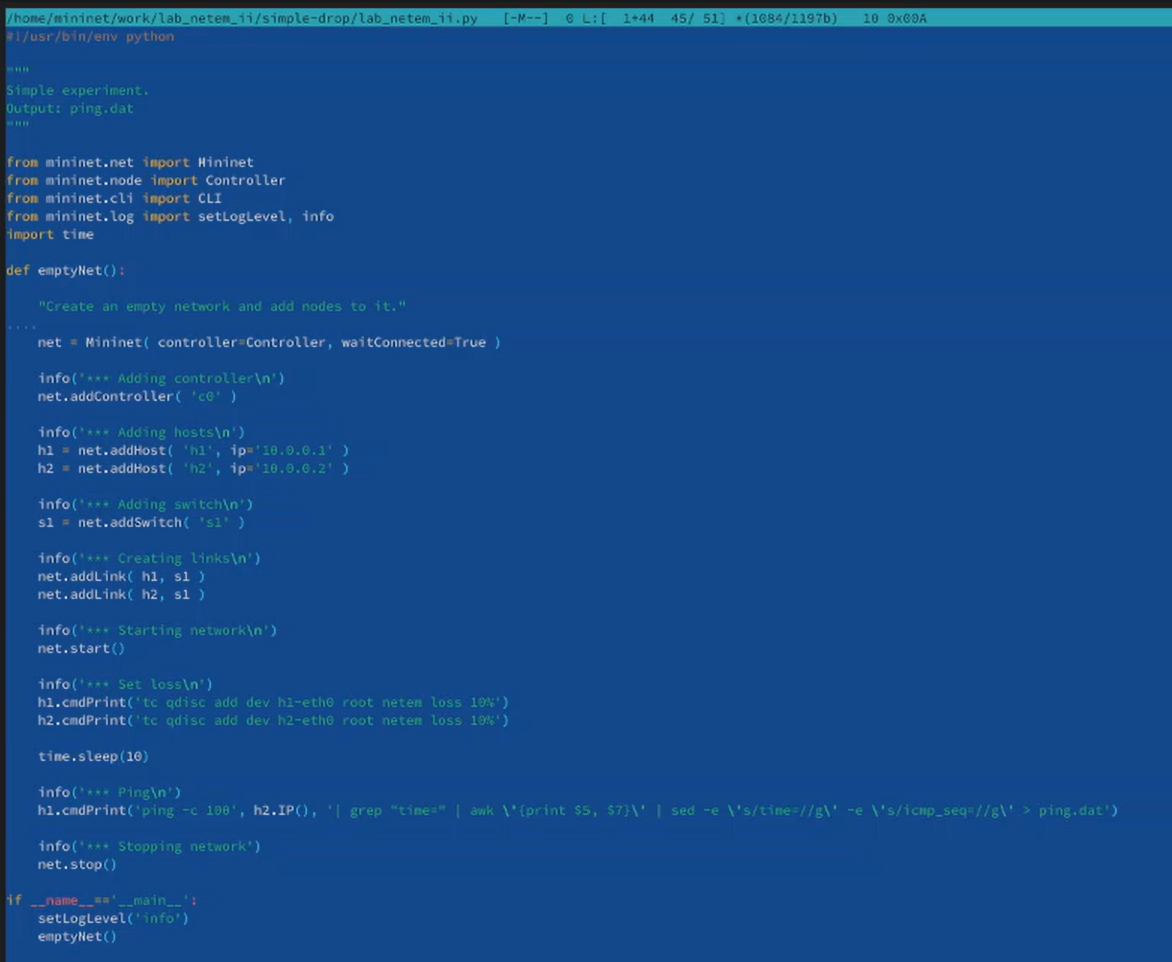


Рис. 14: Редактирование сркипта

Создадим Makefile для управления процессом проведения эксперимента (рис. 15).

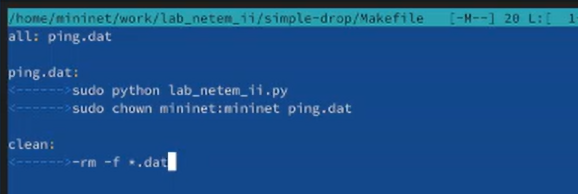


Рис. 15: Makefile

Выполним эксперимент (рис. 16).

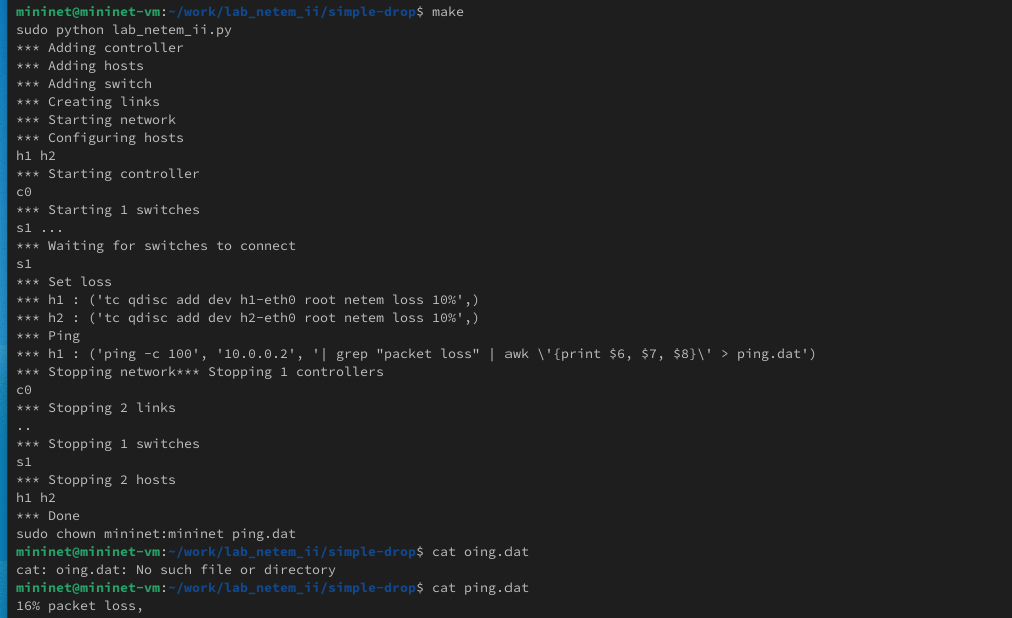


Рис. 16: Проведение эксперимента

## 6.2 Добавление значения корреляции

В виртуальной среде mininet в своём рабочем каталоге с проектами создадим каталог correaltion-drop и перейдем в него. Создадим скрипт для эксперимента lab\_netem\_ii.py (рис. 17).

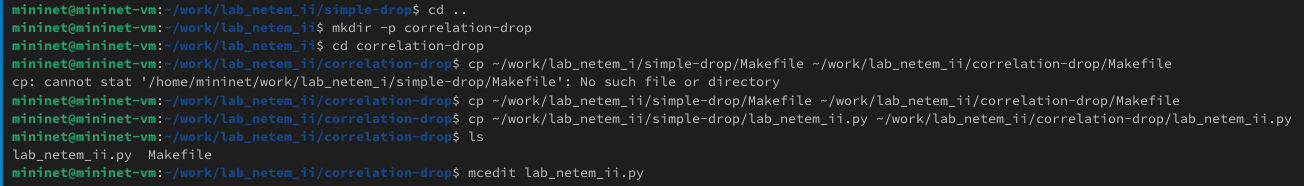


Рис. 17: Создание скрипта для эксперимента lab\_netem\_ii.py

Скорректируем скрипт так, чтобы на экран или в отдельный файл выводилась информация о потерях пакетов (рис. 18).

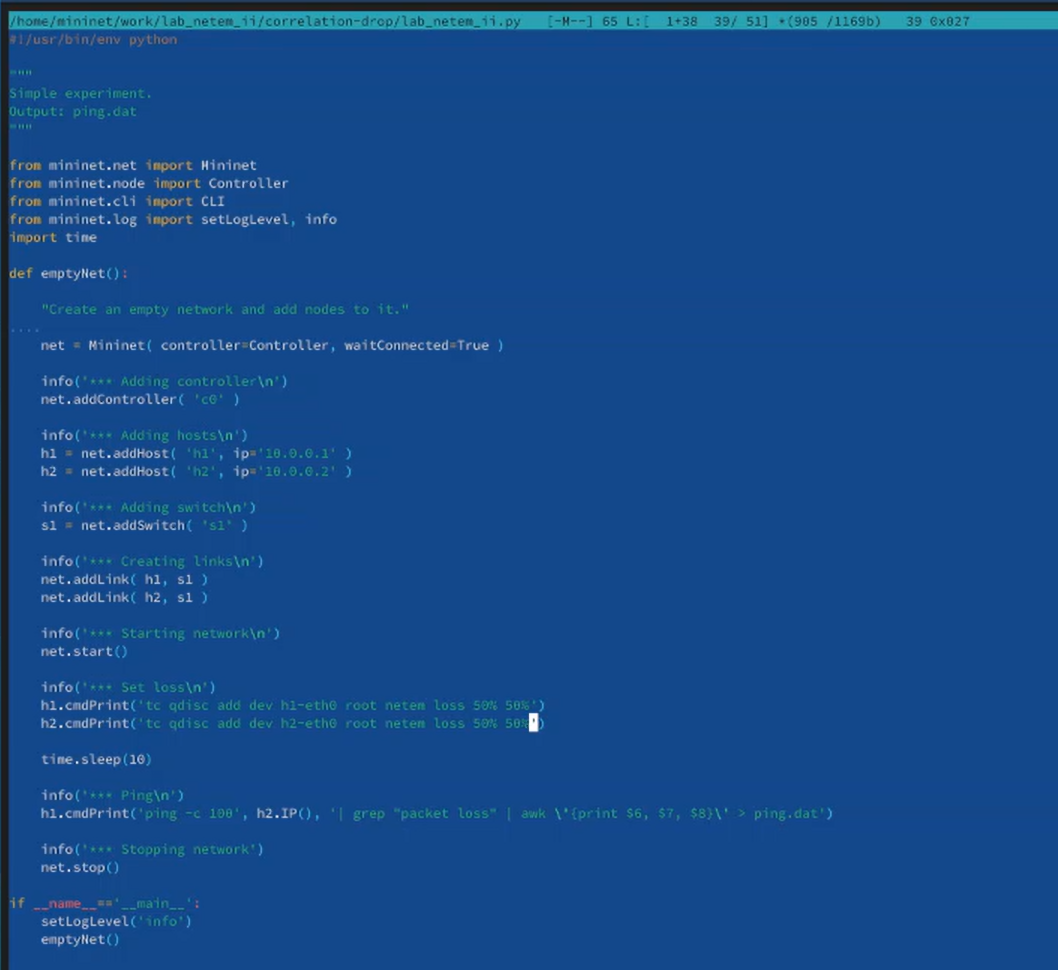


Рис. 18: Редактирование сркипта

Выполним эксперимент (рис. 19).

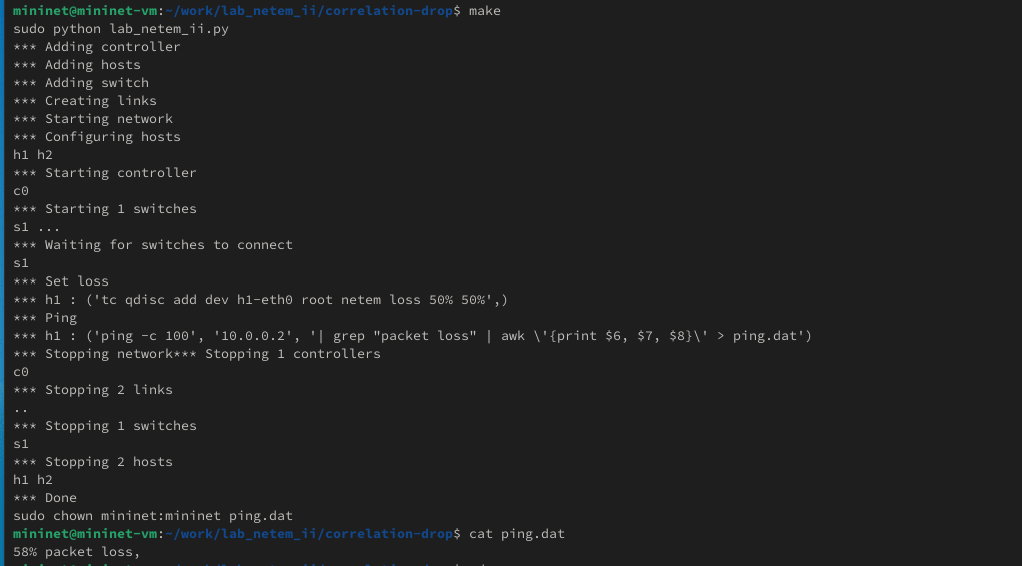


Рис. 19: Проведение эксперимента

## 6.3 Добавление повреждения пакетов

В виртуальной среде mininet в своём рабочем каталоге с проектами создадим каталог simple-corrupt и перейдем в него. Создадим скрипт для эксперимента lab\_netem\_ii.py (рис. 20).

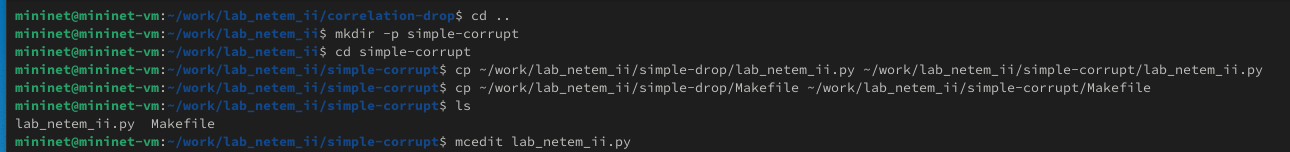


Рис. 20: Создание скрипта для эксперимента lab\_netem\_ii.py

Скорректируем скрипт так, чтобы на экран или в отдельный файл выводилась информация о потерях пакетов (рис. 21).

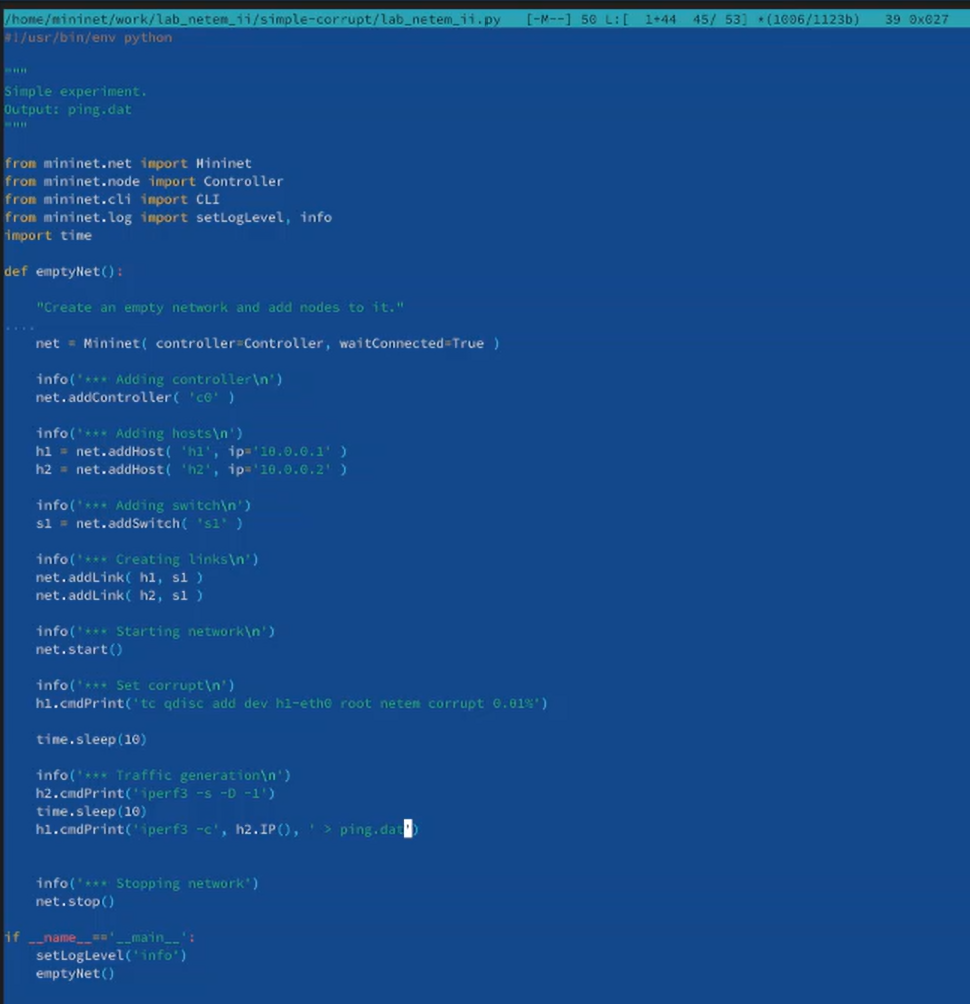


Рис. 21: Редактирование сркипта

Выполним эксперимент (рис. 22).



Рис. 22: Проведение эксперимента

## 6.4 Добавление переупорядочивания пакетов

В виртуальной среде mininet в своём рабочем каталоге с проектами создадим каталог simple-reorder и перейдем в него. Создадим скрипт для эксперимента lab\_netem\_ii.py (рис. 23).

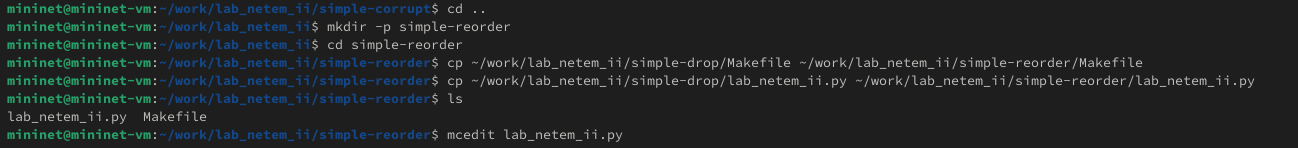


Рис. 23: Создание скрипта для эксперимента lab\_netem\_ii.py

Скорректируем скрипт так, чтобы на экран или в отдельный файл выводилась информация о потерях пакетов (рис. 24).

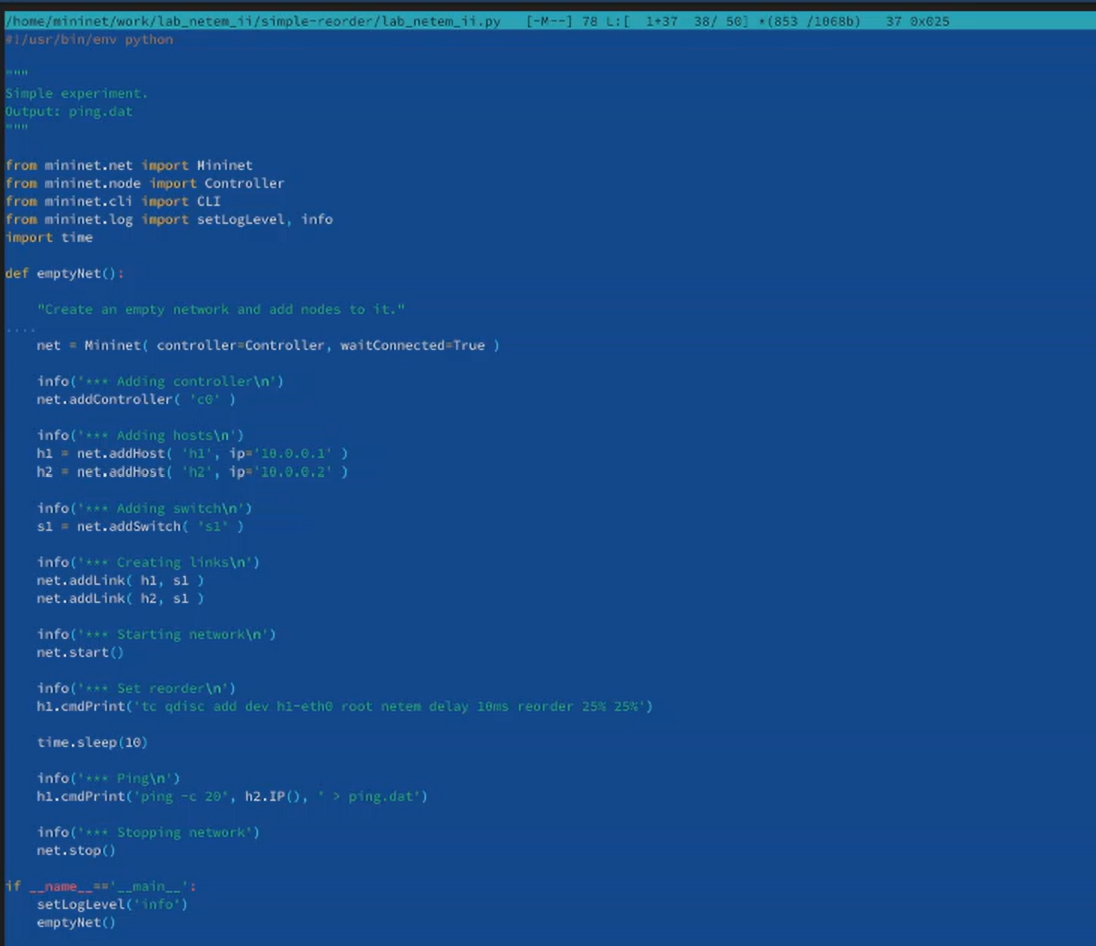


Рис. 24: Редактирование сркипта

Выполним эксперимент (рис. 25).

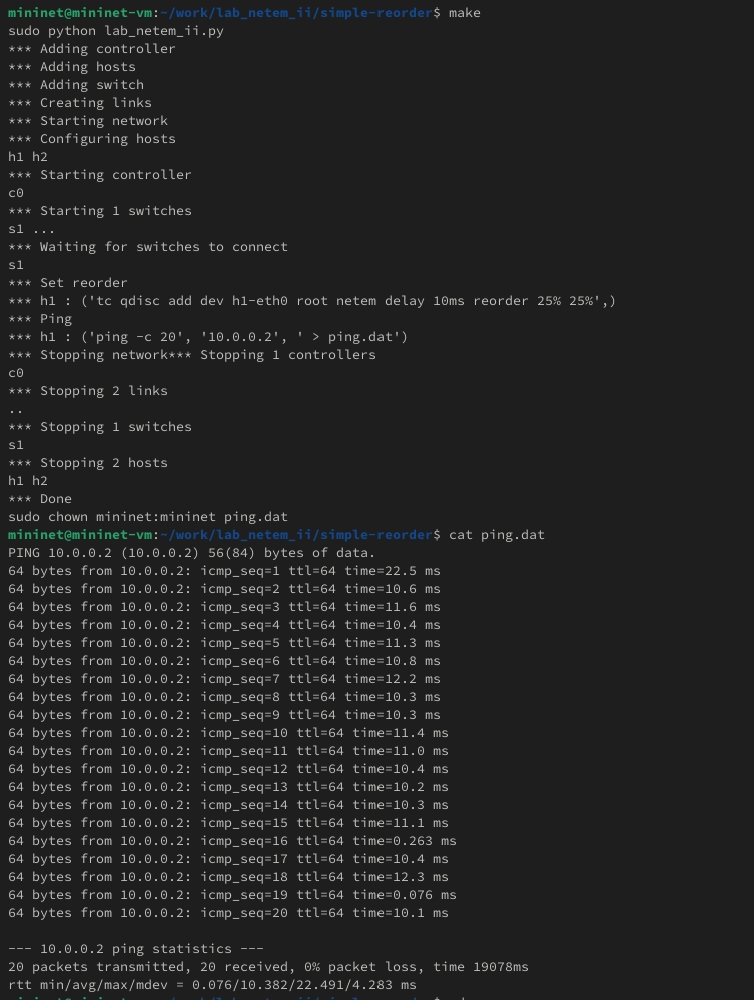


Рис. 25: Проведение эксперимента

## 6.5 Добавление дублирования пакетов

В виртуальной среде mininet в своём рабочем каталоге с проектами создадим каталог simple-duplicate и перейдем в него. Создадим скрипт для эксперимента lab\_netem\_ii.py (рис. 26).

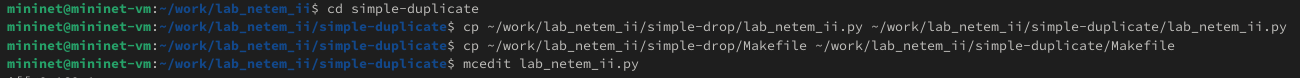


Рис. 26: Создание скрипта для эксперимента lab\_netem\_ii.py

Скорректируем скрипт так, чтобы на экран или в отдельный файл выводилась информация о потерях пакетов (рис. 27).

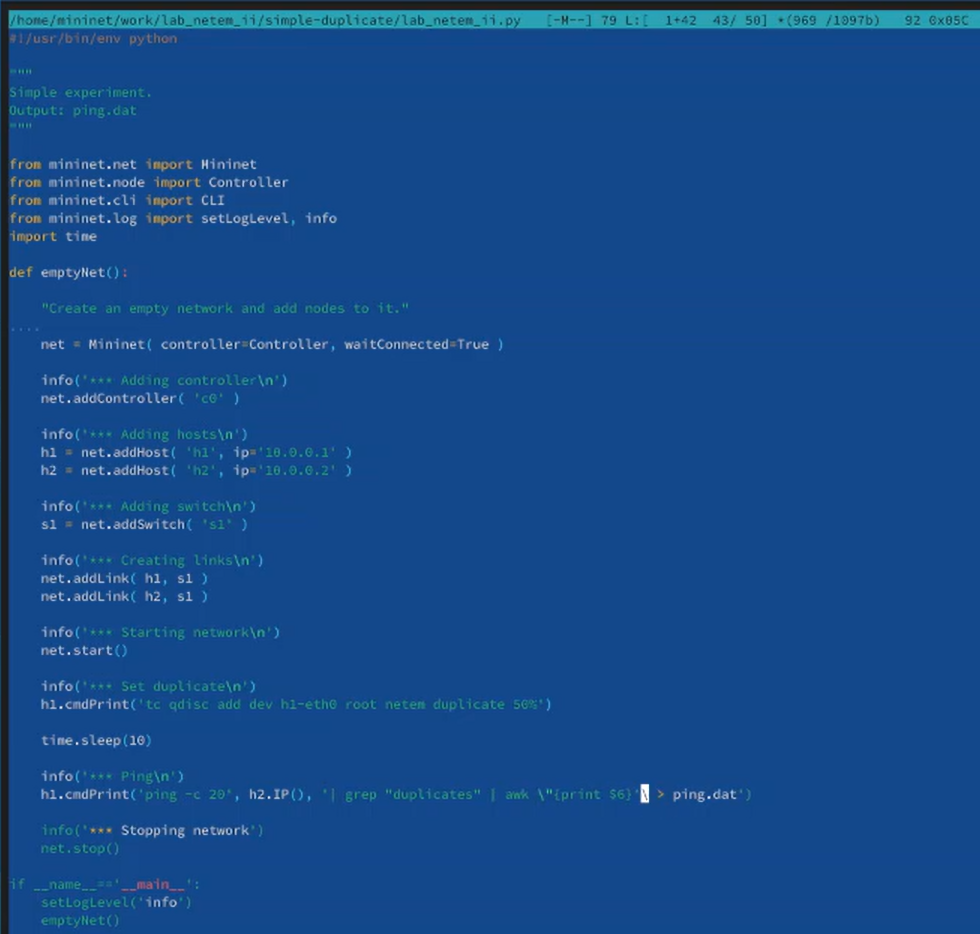


Рис. 27: Редактирование сркипта

Выполним эксперимент (рис. 28).

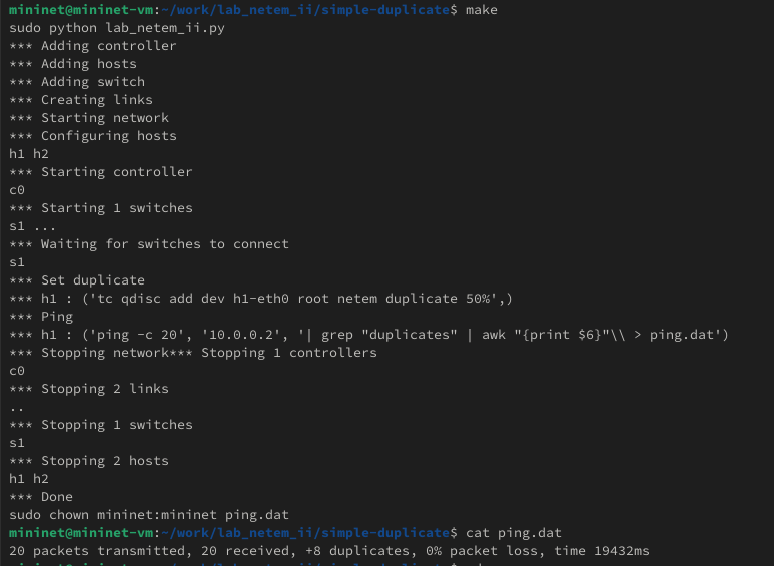


Рис. 28: Проведение эксперимента

# 7 Выводы

В процессе выполнения лабораторной работы я получила навыки проведения интерактивных экспериментов в среде Mininet по исследованию параметров сети, связанных с потерей, дублированием, изменением порядка и повреждением пакетов при передаче данных. Эти параметры влияют на производительность протоколов и сетей.

# Список литературы