**РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ**

**Факультет физико-математических и естественных наук**

**Кафедра прикладной информатики и теории вероятностей**

**ОТЧЕТ**

**по лабораторной работе № 4**

*дисциплина: Моделирование сетей передачи данных*

Студент: Абрамян А. А.

Группа: НПИбд-01-20

**МОСКВА**

2023 г.

**Цель работы**

Целью данной работы является знакомство с NETEM — инструментом для тестирования производительности приложений в виртуальной сети, а также получение навыков проведения интерактивного и воспроизводимого экспериментов по измерению задержки и её дрожания (jitter) в моделируемой сети в среде Mininet.

**Описание процесса выполнения работы**

**Постановка задачи**

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 добавлению/изменению задержки, джиттера, значения корреляции для джиттера и задержки, распределения времени задержки в эмулируемой глобальной сети.
3. Реализуйте воспроизводимый эксперимент по заданию значения задержки в эмулируемой глобальной сети. Постройте график.
4. Самостоятельно реализуйте воспроизводимые эксперименты по изменению задержки, джиттера, значения корреляции для джиттера и задержки, распределения времени задержки в эмулируемой глобальной сети. Постройте графики.

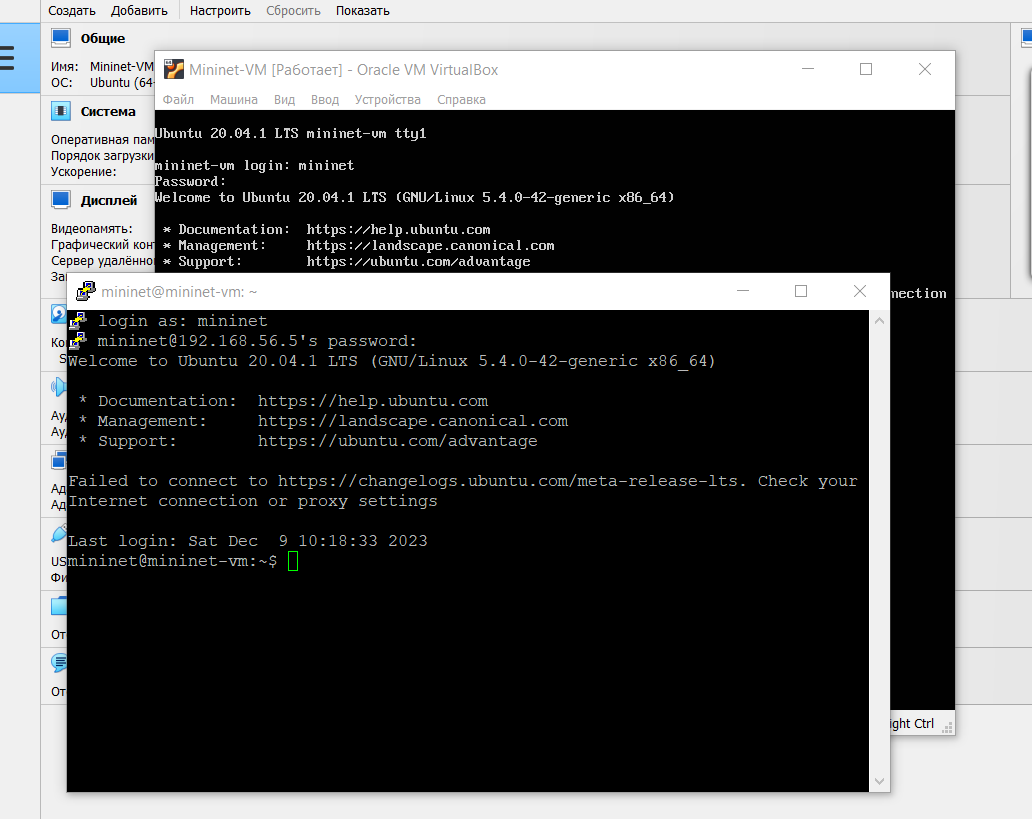
**Порядок выполнения работы**

**Запуск лабораторной топологии**

Запустите виртуальную среду с mininet.

Из основной ОС подключитесь к виртуальной машине:

ssh -Y mininet@192.168.x.y



В виртуальной машине mininet при необходимости исправьте права запуска X-соединения. Скопируйте значение куки (MIT magic cookie)1 своего пользователя mininet в файл для пользователя root:

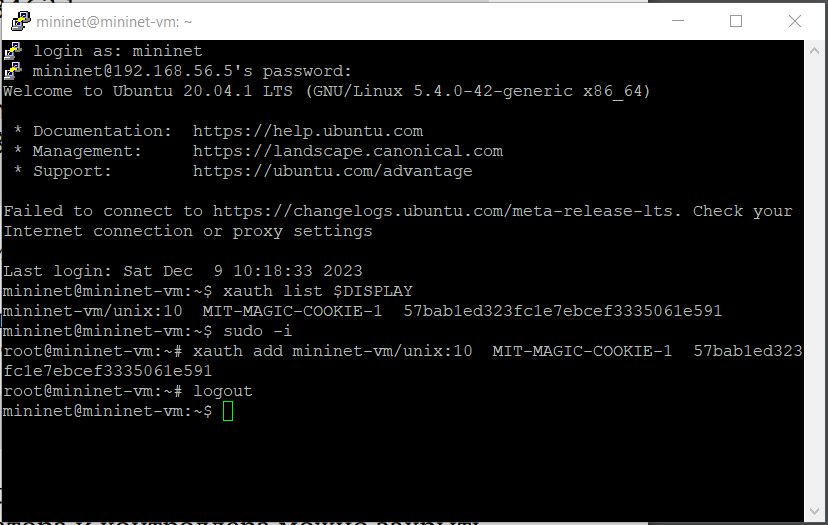
mininet@mininet-vm:~$ xauth list $DISPLAY

mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 295acad8e35d17636924c5ab80e8462d

mininet@mininet-vm:~$ sudo -i

root@mininet-vm:~# xauth add mininet-vm/unix:10 MIT-MAGIC-COOKIE-1 295acad8e35d17636924c5ab80e8462d

root@mininet-vm:~# logout

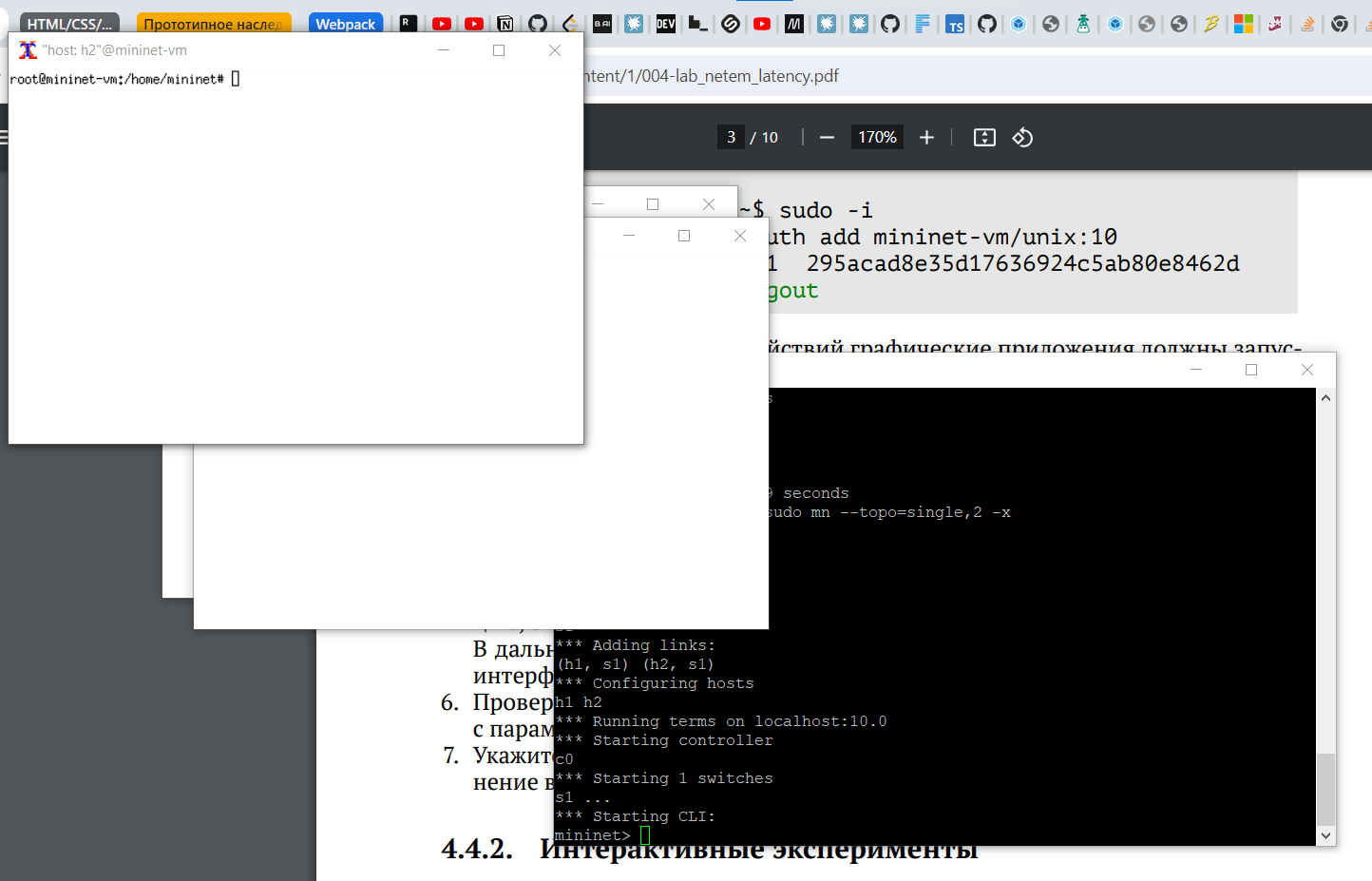


После выполнения этих действий графические приложения должны запускаться под пользователем mininet.

Задайте простейшую топологию, состоящую из двух хостов и коммутатора

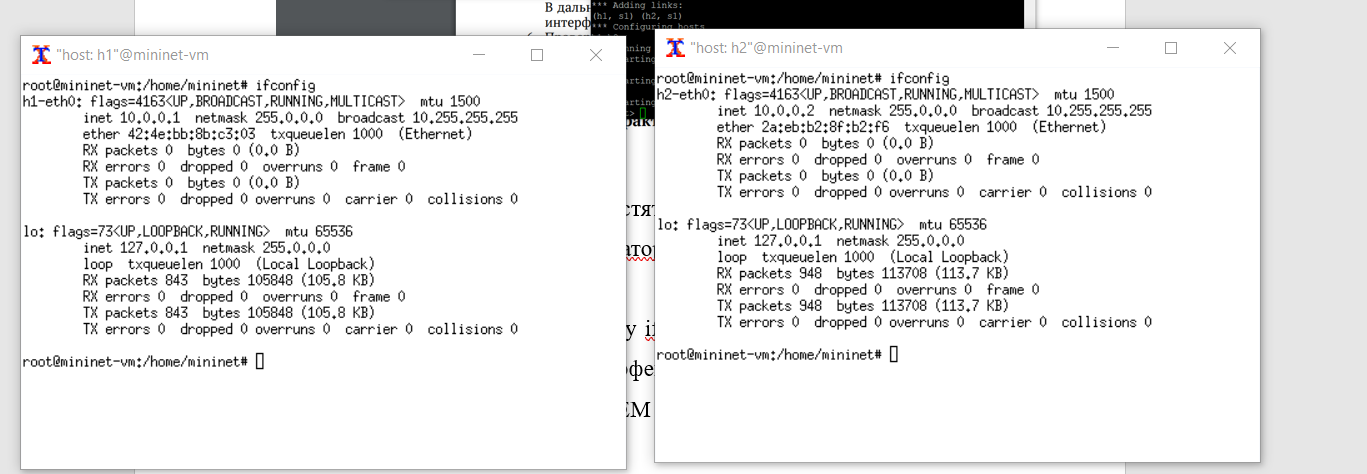
с назначенной по умолчанию mininet сетью 10.0.0.0/8:

sudo mn --topo=single,2 -x

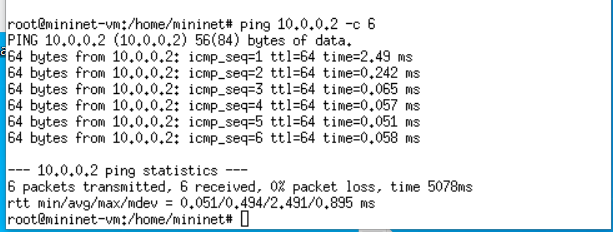


После введения этой команды запустятся терминалы двух хостов, коммутатора и контроллера. Терминалы коммутатора и контроллера можно закрыть.

На хостах h1 и h2 введите 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.



Проверьте подключение между хостами h1 и h2 с помощью команды ping с параметром -c 6.



Укажите в отчёте минимальное, среднее, максимальное и стандартное отклонение времени приёма-передачи (RTT).

Min = 0.051 ms

Avg = 0.494 ms

Max = 2.494 ms

Mdev = 0.895 ms

**Интерактивные эксперименты**

**Добавление/изменение задержки в эмулируемой глобальной сети**

Сетевые эмуляторы задают задержки на интерфейсе. Например, задержка, вносимая в интерфейс коммутатора A, который подключён к интерфейсу коммутатора B, может представлять собой задержку распространения WAN, соединяющей оба коммутатора.

На хосте h1 добавьте задержку в 100 мс к выходному интерфейсу:

sudo tc qdisc add dev h1-eth0 root netem delay 100ms



Здесь:

– sudo: выполнить команду с более высокими привилегиями;

– tc: вызвать управление трафиком Linux;

– qdisc: изменить дисциплину очередей сетевого планировщика;

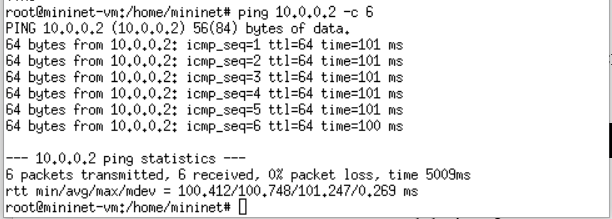
– add: создать новое правило;

– dev h1-eth0: указать интерфейс, на котором будет применяться правило;

– netem: использовать эмулятор сети;

– delay 100ms: задержка ввода 100 мс.

Проверьте, что соединение от хоста h1 к хосту h2 имеет задержку 100 мс, используя команду ping с параметром -c 6 с хоста h1. Укажите в отчёте минимальное, среднее, максимальное и стандартное отклонение времени приёма-передачи (RTT).



Min = 100.412 ms

Avg = 100.748 ms

Max = 101.247 ms

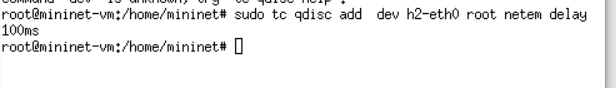
Mdev = 0.269 ms

Для эмуляции глобальной сети с двунаправленной задержкой необходимо

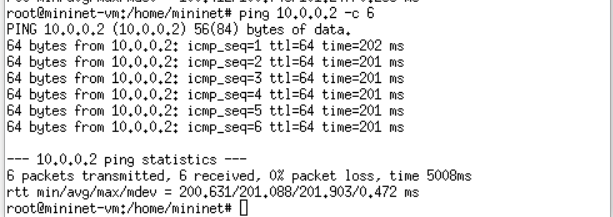
к соответствующему интерфейсу на хосте h2 также добавить задержку в 100

миллисекунд:

sudo tc qdisc add dev h2-eth0 root netem delay 100ms



Проверьте, что соединение между хостом h1 и хостом h2 имеет RTT в 200 мс (100 мс от хоста h1 к хосту h2 и 100 мс от хоста h2 к хосту h1), повторив команду ping с параметром -c 6 на терминале хоста h1. Укажите в отчёте минимальное, среднее, максимальное и стандартное отклонение времени приёма-передачи (RTT).



Min = 200.631 ms

Avg = 201.088 ms

Max = 201.903 ms

Mdev = 0.472 ms

**Изменение задержки в эмулируемой глобальной сети**

Измените задержку со 100 мс до 50 мс для отправителя h1:

sudo tc qdisc change dev h1-eth0 root netem delay 50ms

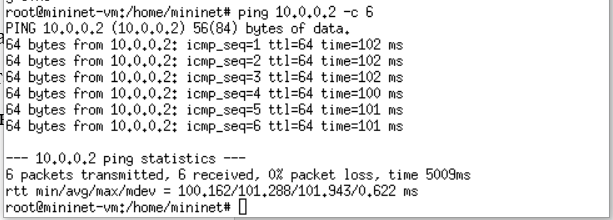


и для получателя h2:

sudo tc qdisc change dev h2-eth0 root netem delay 50ms



Проверьте, что соединение от хоста h1 к хосту h2 имеет задержку 100 мс, используя команду ping с параметром -c 6 с терминала хоста h1. Укажите в отчёте минимальное, среднее, максимальное и стандартное отклонение времени приёма-передачи (RTT).



Min = 100.162 ms

Avg = 101.288 ms

Max = 101.943 ms

Mdev = 0.622 ms

**Восстановление исходных значений (удаление правил) задержки в эмулируемой глобальной сети**

Восстановите конфигурацию по умолчанию, удалив все правила, применённые к сетевому планировщику соответствующего интерфейса. Для

отправителя h1:

sudo tc qdisc del dev h1-eth0 root netem



Для получателя h2:

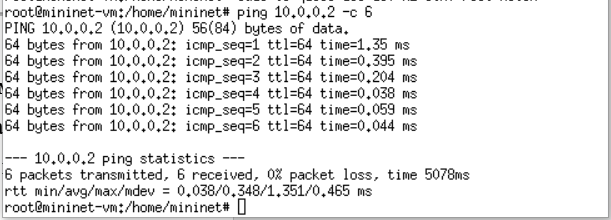
sudo tc qdisc del dev h2-eth0 root netem



Проверьте, что соединение между хостом h1 и хостом h2 не имеет явно

установленной задержки, используя команду ping с параметром -c 6 с терминала хоста h1. Укажите в отчёте минимальное, среднее, максимальное

и стандартное отклонение времени приёма-передачи (RTT).



Min = 0.038 ms

Avg = 0.348 ms

Max = 1.351 ms

Mdev = 0.465 ms

**Добавление значения дрожания задержки в интерфейс подключения к эмулируемой глобальной сети**

В сетях нет постоянной задержки. Она может варьироваться в зависимости

от других потоков трафика, конкурирующих за тот же путь. Джиттер (jitter) —

это изменение времени задержки. Параметры задержки описываются в терминах теории вероятностей средним значением 𝜇, стандартным отклонением

𝜎 и корреляцией. По умолчанию NETEM использует равномерное распределение,

так что задержка находится в пределах 𝜇 ± 𝜎. Параметр корреляции управляет

отношением между последовательными псевдослучайными значениями.

При необходимости восстановите конфигурацию интерфейсов по умолчанию

на узлах h1 и h2:

sudo tc qdisc del dev h1-eth0 root netem

sudo tc qdisc del dev h2-eth0 root netem

Добавьте на узле h1 задержку в 100 мс со случайным отклонением 10 мс:

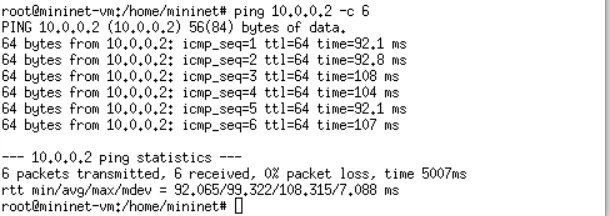
sudo tc qdisc add dev h1-eth0 root netem delay 100ms 10ms



Проверьте, что соединение от хоста h1 к хосту h2 имеет задержку 100 мс со

случайным отклонением ±10 мс, используя в терминале хоста h1 команду

ping с параметром -c 6. Укажите в отчёте минимальное, среднее, максимальное и стандартное отклонение времени приёма-передачи (RTT).



Min = 92.065 ms

Avg = 99.322 ms

Max = 108.315 ms

Mdev = 7.088 ms

Восстановите конфигурацию интерфейса по умолчанию на узле h1.



**Добавление значения корреляции для джиттера и задержки в интерфейс подключения к эмулируемой глобальной сети**

При необходимости восстановите конфигурацию интерфейсов по умолчанию

на узлах h1 и h2.

Добавьте на интерфейсе хоста h1 задержку в 100 мс с вариацией ±10 мс

и значением корреляции в 25%:

sudo tc qdisc add dev h1-eth0 root netem delay 100ms 10ms 25%

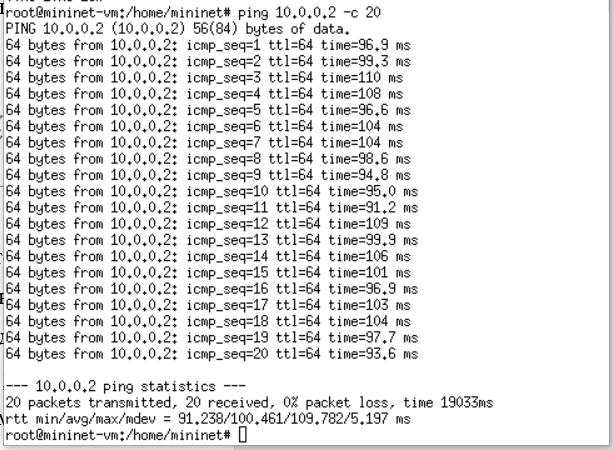
Убедитесь, что все пакеты, покидающие устройство h1 на интерфейсе h1-

eth0, будут иметь время задержки 100 мс со случайным отклонением ±10 мс,

при этом время передачи следующего пакета зависит от предыдущего значения на 25%. Используйте для этого в терминале хоста h1 команду ping

с параметром -c 20. Укажите в отчёте минимальное, среднее, максимальное

и стандартное отклонение времени приёма-передачи (RTT).



Min = 91.238 ms

Avg = 100.461 ms

Max = 109.782 ms

Mdev = 5.197 ms

Восстановите конфигурацию интерфейса по умолчанию на узле h1.



**Распределение задержки в интерфейсе подключения к эмулируемой глобальной сети**

NETEM позволяет пользователю указать распределение, которое описывает,

как задержки изменяются в сети. В реальных сетях передачи данных задержки

неравномерны, поэтому при моделировании может быть удобно использовать

некоторое случайное распределение, например, нормальное, парето или паретонормальное.

При необходимости восстановите конфигурацию интерфейсов по умолчанию

на узлах h1 и h2.

Задайте нормальное распределение задержки на узле h1 в эмулируемой сети:

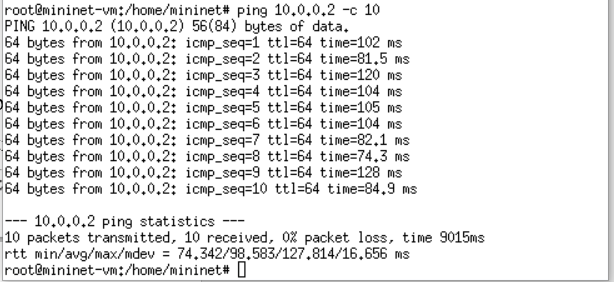
sudo tc qdisc add dev h1-eth0 root netem delay 100ms 20ms distribution normal



Убедитесь, что все пакеты, покидающие хост h1 на интерфейсе h1-eth0, будут иметь время задержки, которое распределено в диапазоне 100 мс ±20 мс.

Используйте для этого команду ping на терминале хоста h1 с параметром

-c 10.



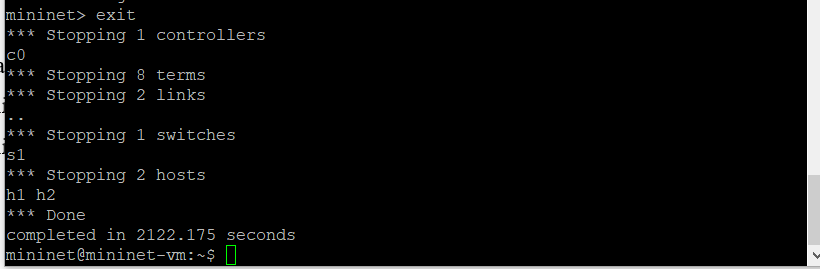
Восстановите конфигурацию интерфейса по умолчанию на узле h1



Завершите работу mininet в интерактивном режиме, введя в интерфейсе

mininet:

mininet> exit

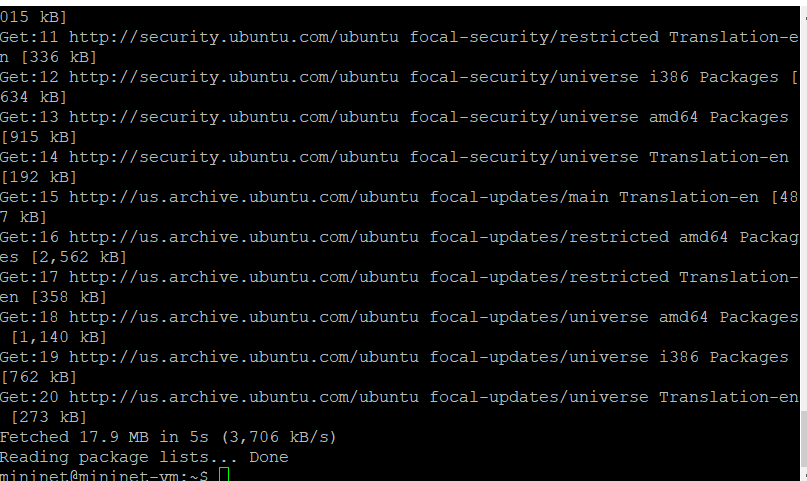


**Воспроизведение экспериментов**

**Предварительная подготовка**

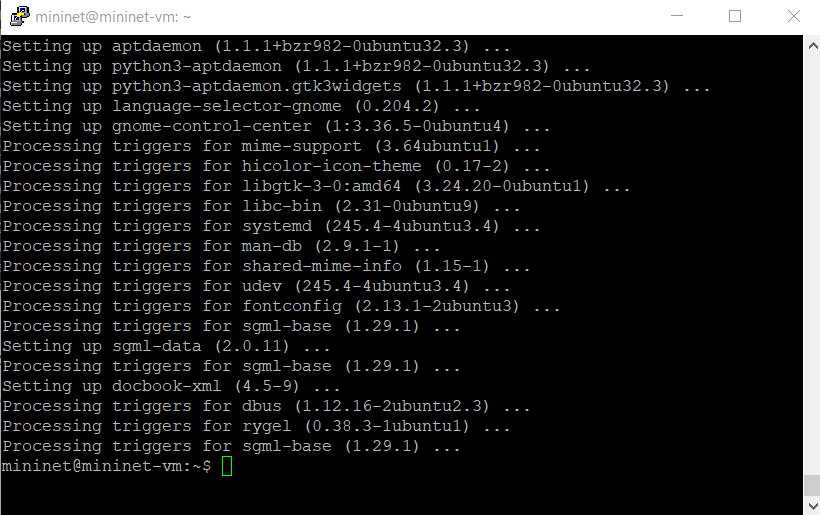
Обновите репозитории программного обеспечения на виртуальной машине:

sudo apt-get update



Установите пакет geeqie — понадобится для просмотра файлов png:

sudo apt install geeqie

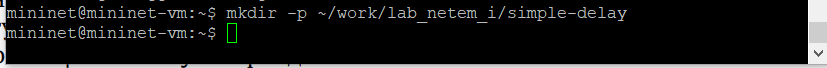


Для каждого воспроизводимого эксперимента expname создайте свой каталог, в котором будут размещаться файлы эксперимента:

mkdir -p ~/work/lab\_netem\_i/expname

Здесь expname может принимать значения simple-delay, change-delay,

jitter-delay, correlation-delay и т.п



Для каждого случая создайте скрипт для проведения эксперимента

lab\_netem\_i.py и скрипт для визуализации результатов ping\_plot.

**Добавление задержки для интерфейса, подключающегося к эмулируемой глобальной сети**

С помощью API Mininet воспроизведите эксперимент по добавлению задержки для интерфейса хоста, подключающегося к эмулируемой глобальной сети.

1. В виртуальной среде mininet в своём рабочем каталоге с проектами создайте

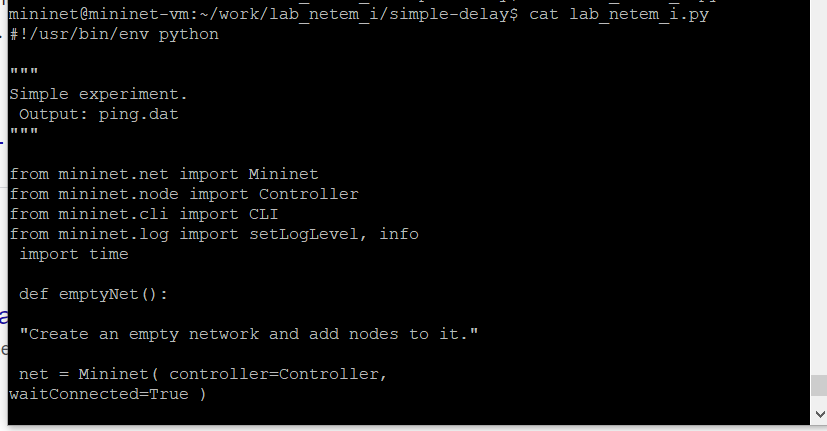
каталог simple-delay и перейдите в него:

mkdir -p ~/work/lab\_netem\_i/simple-delay

cd ~/work/lab\_netem\_i/simple-delay

****

Создаёте скрипт для эксперимента lab\_netem\_i.py:



В отчёте поясните содержание скрипта lab\_netem\_i.py.

В каких строках скрипта задается значение задержки для интерфейса хоста?

info( '\*\*\* Set delay\n')

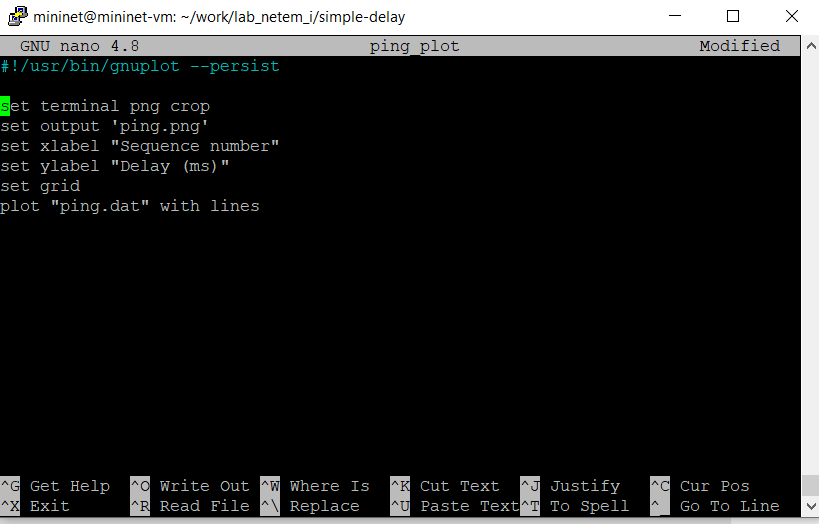
h1.cmdPrint( 'tc qdisc add dev h1-eth0 root netem delay 100ms' )

h2.cmdPrint( 'tc qdisc add dev h2-eth0 root netem delay 100ms' )

Каким образом формируется файл с результатами эксперимента для последующего построения графиков, какие значения в нём размещены?

---------------

Создаёте скрипт для визуализации ping\_plot результатов эксперимента:

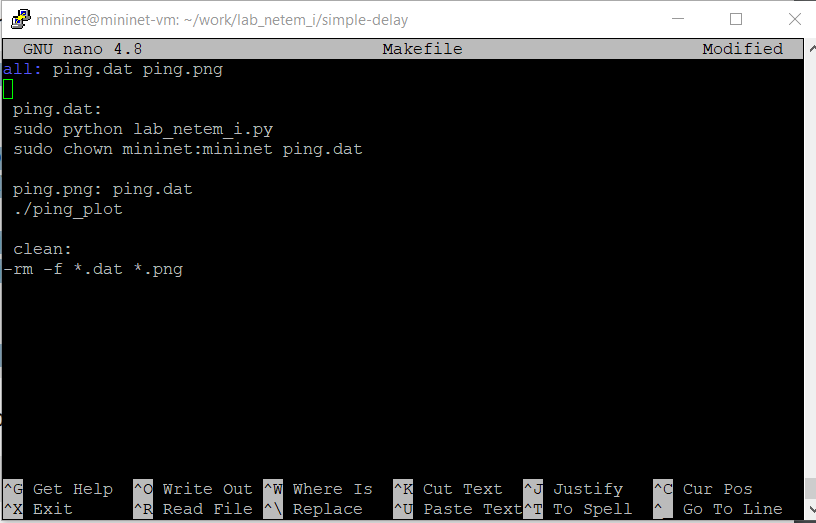


Задайте права доступа к файлу скрипта:

chmod +x ping\_plot

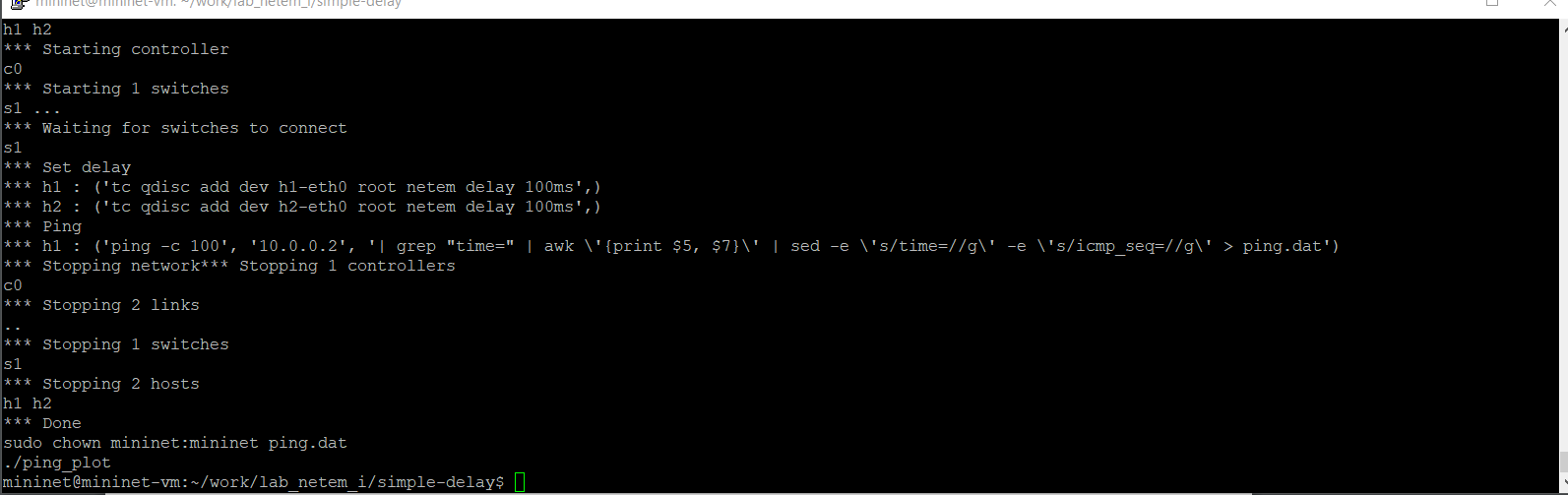


Создайте Makefile для управления процессом проведения эксперимента:

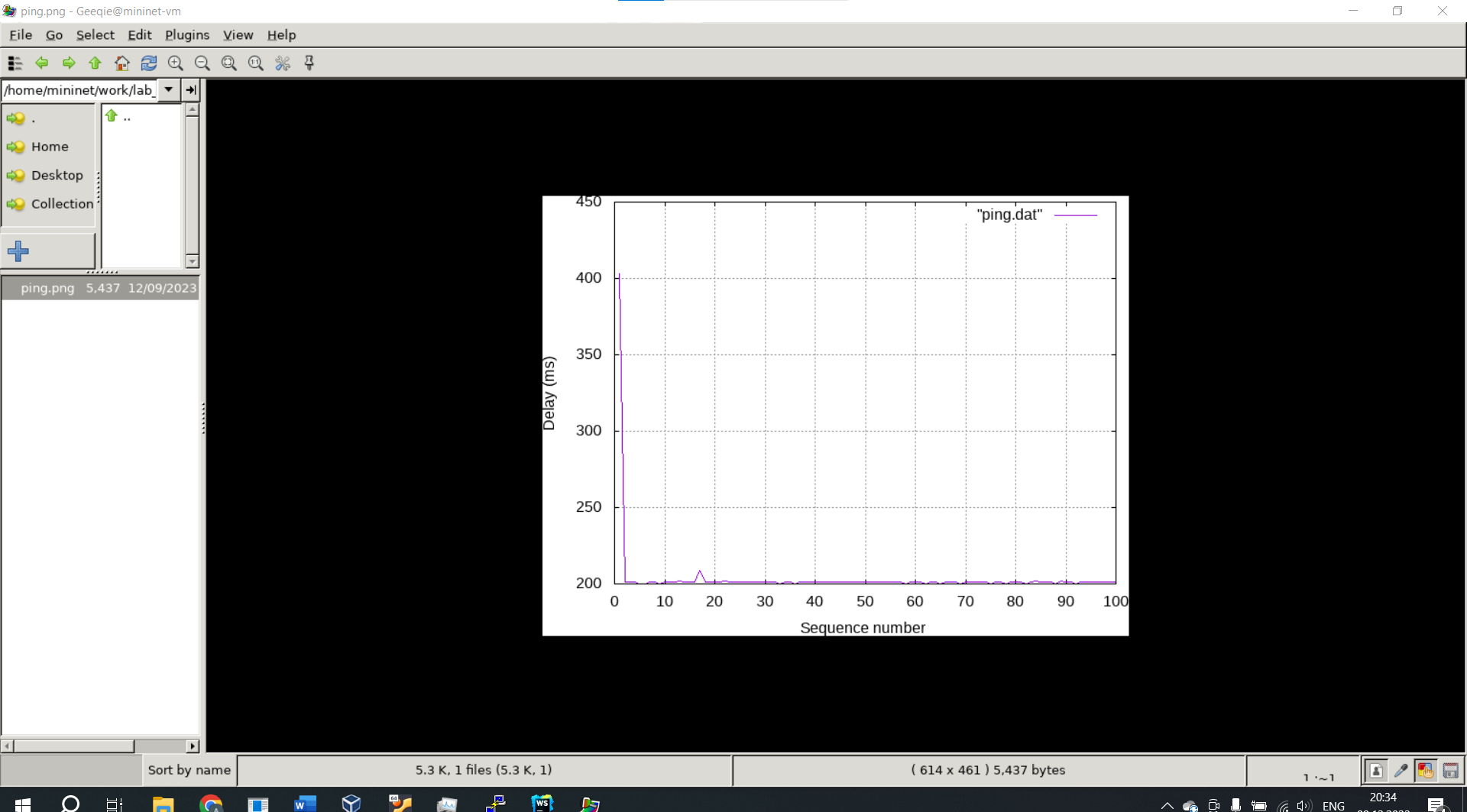


Выполните эксперимент:

Make



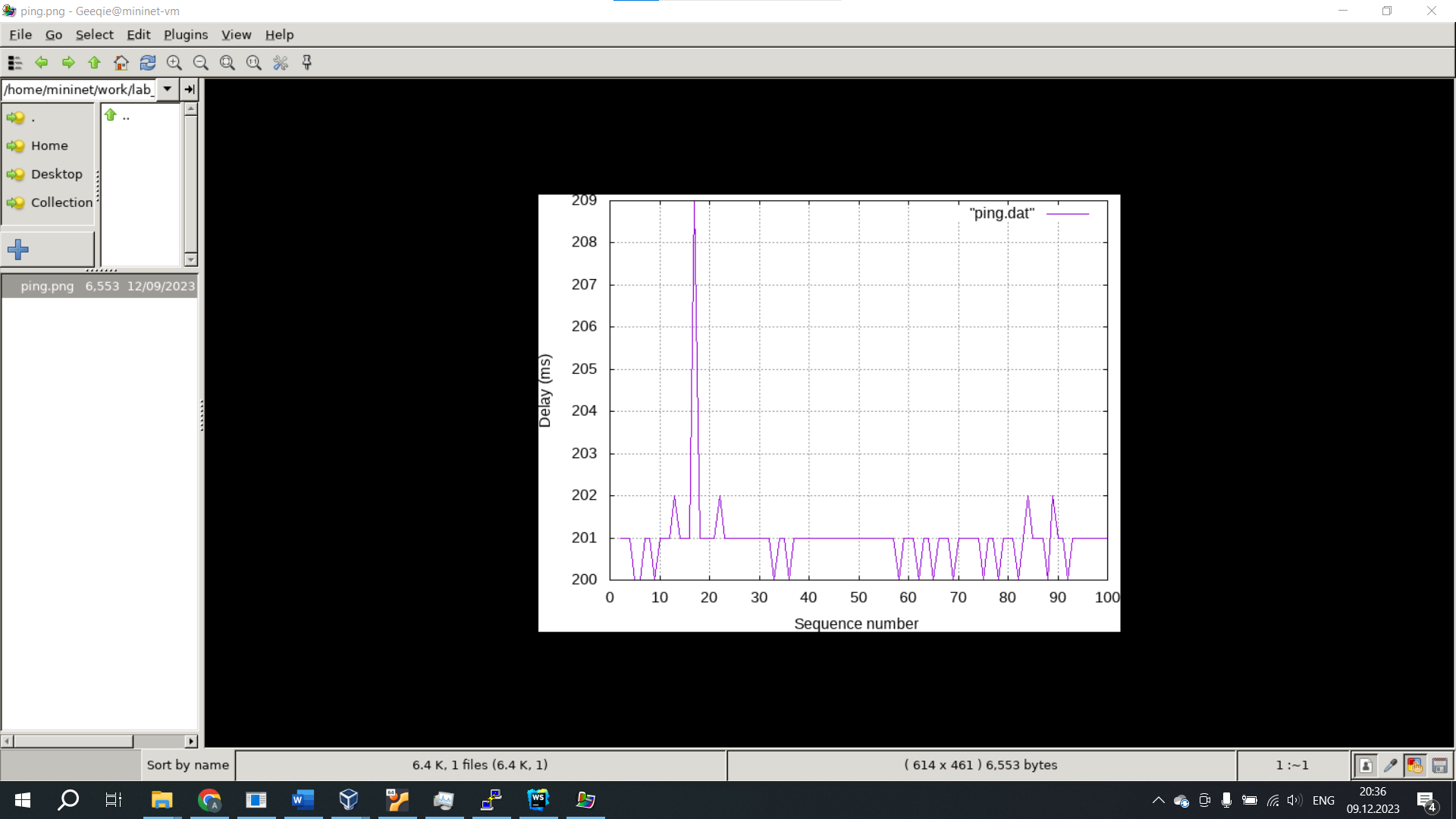
Продемонстрируйте построенный в результате выполнения скриптов график.



Из файла ping.dat удалите первую строку и заново постройте график:

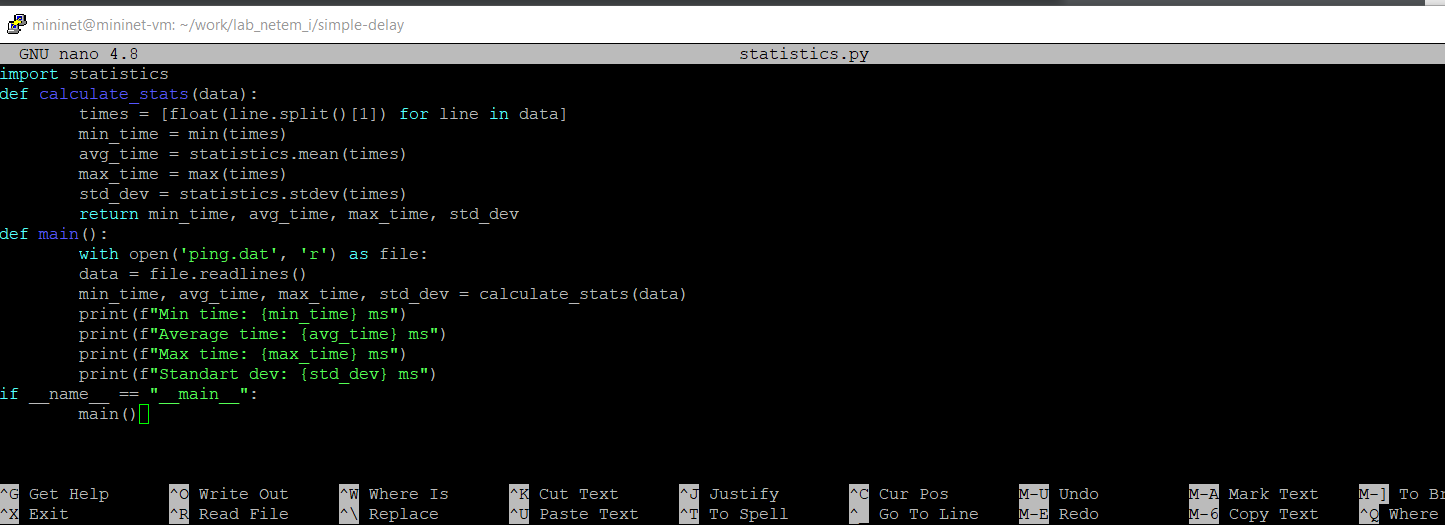
make ping.png

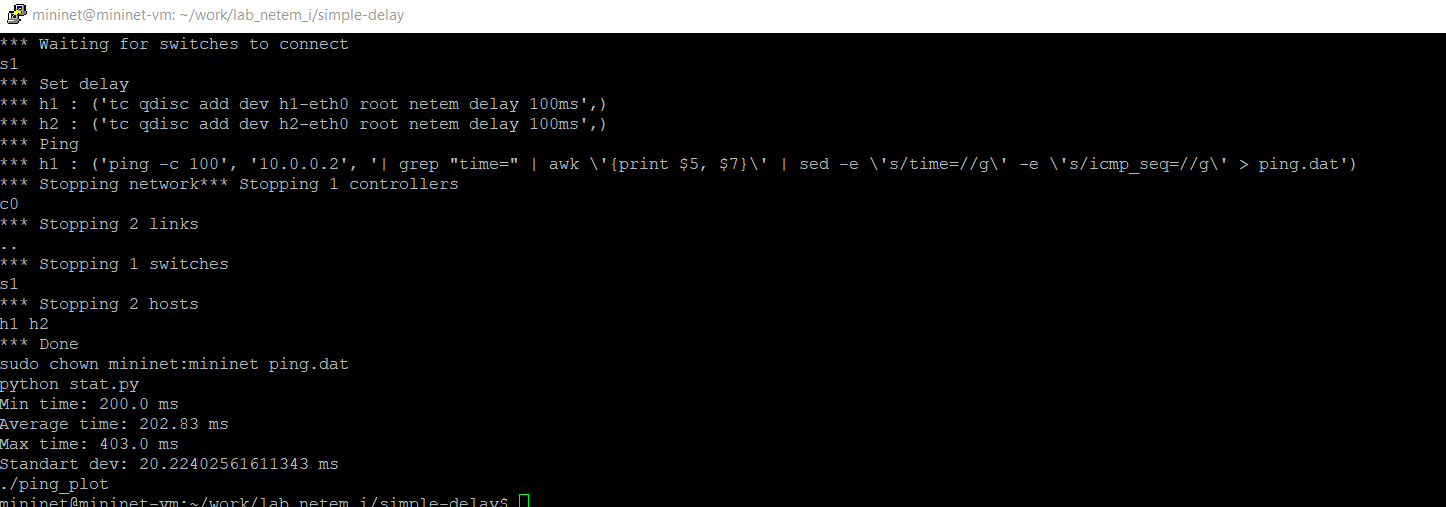
Продемонстрируйте построенный в результате график.



Разработайте скрипт для вычисления на основе данных файла ping.dat минимального, среднего, максимального и стандартного отклонения времени

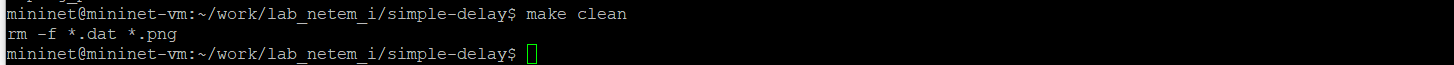
приёма-передачи. Добавьте правило запуска скрипта в Makefile. Продемонстрируйте работу скрипта с выводом значений на экран или в отдельный файл.





Очистите каталог от результатов проведения экспериментов:

make clean



**Задание для самостоятельной работы**

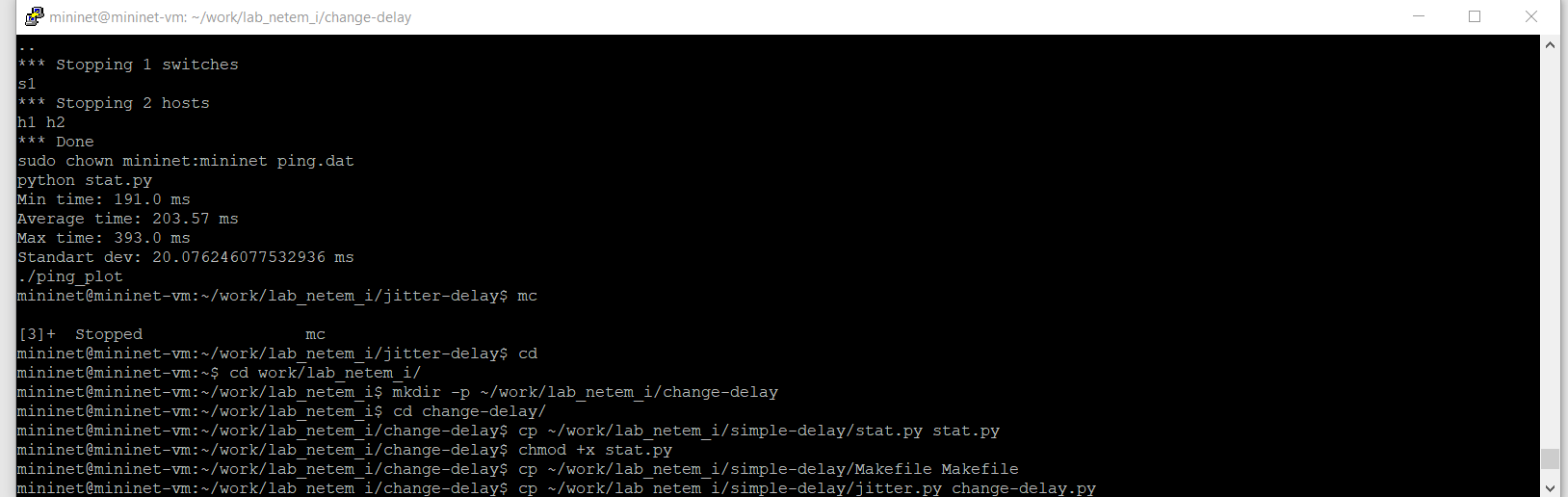
Самостоятельно реализуйте воспроизводимые эксперименты по изменению

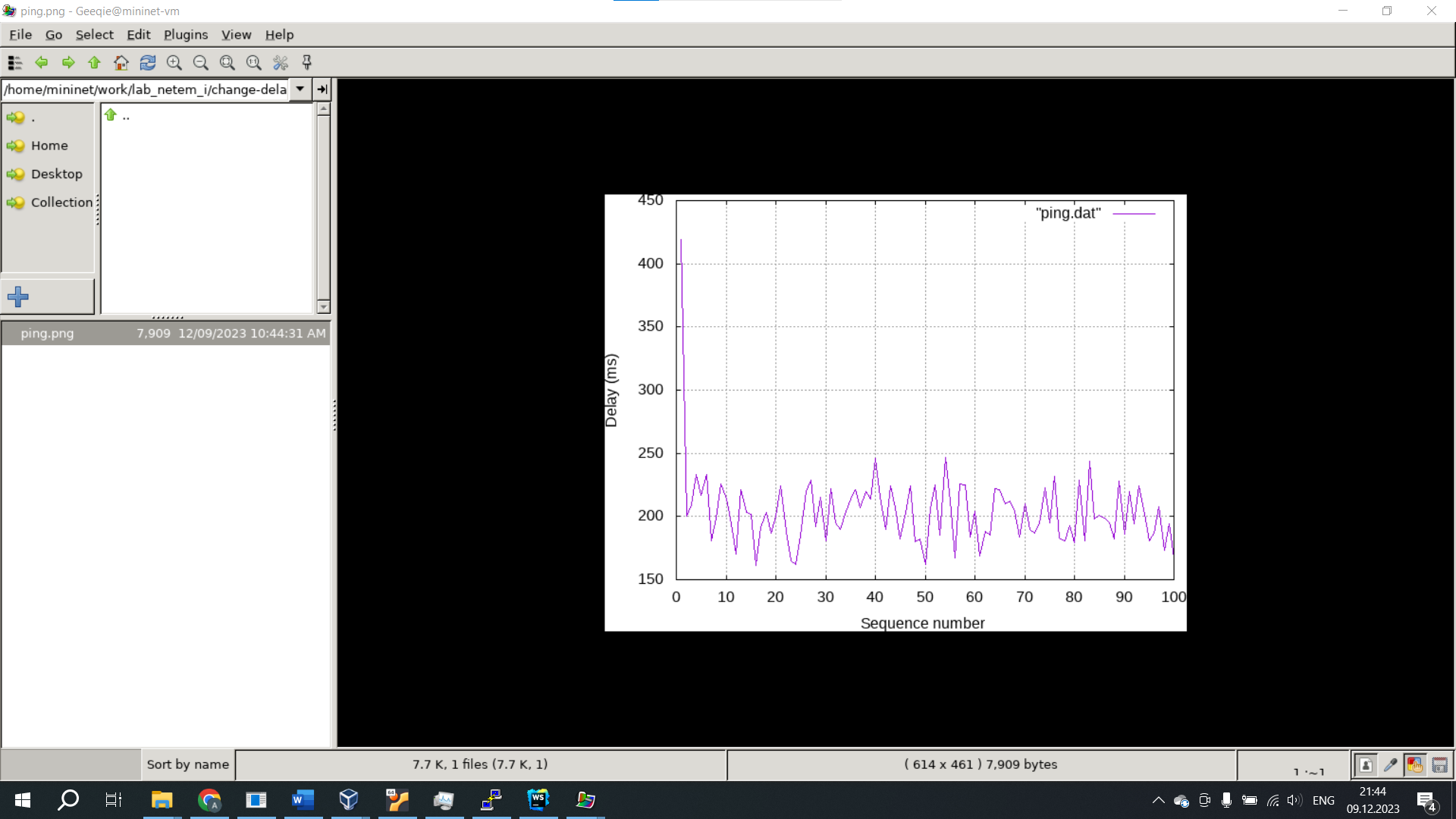
задержки, джиттера, значения корреляции для джиттера и задержки, распределения времени задержки в эмулируемой глобальной сети. Постройте графики.

Вычислите минимальное, среднее, максимальное и стандартное отклонение

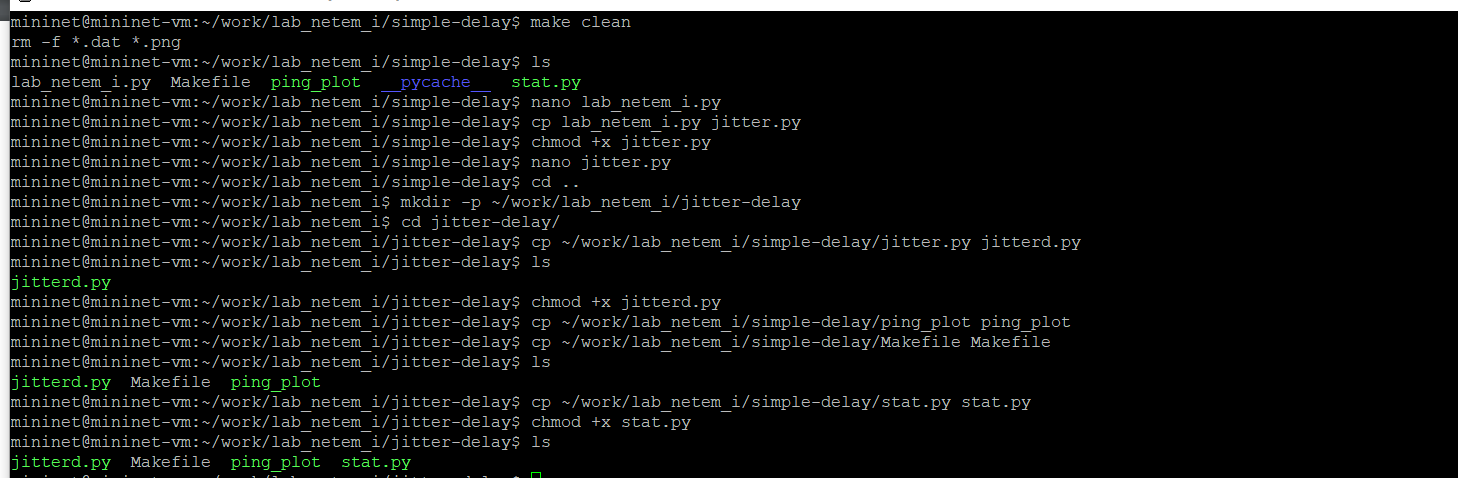
времени приёма-передачи для каждого случая.

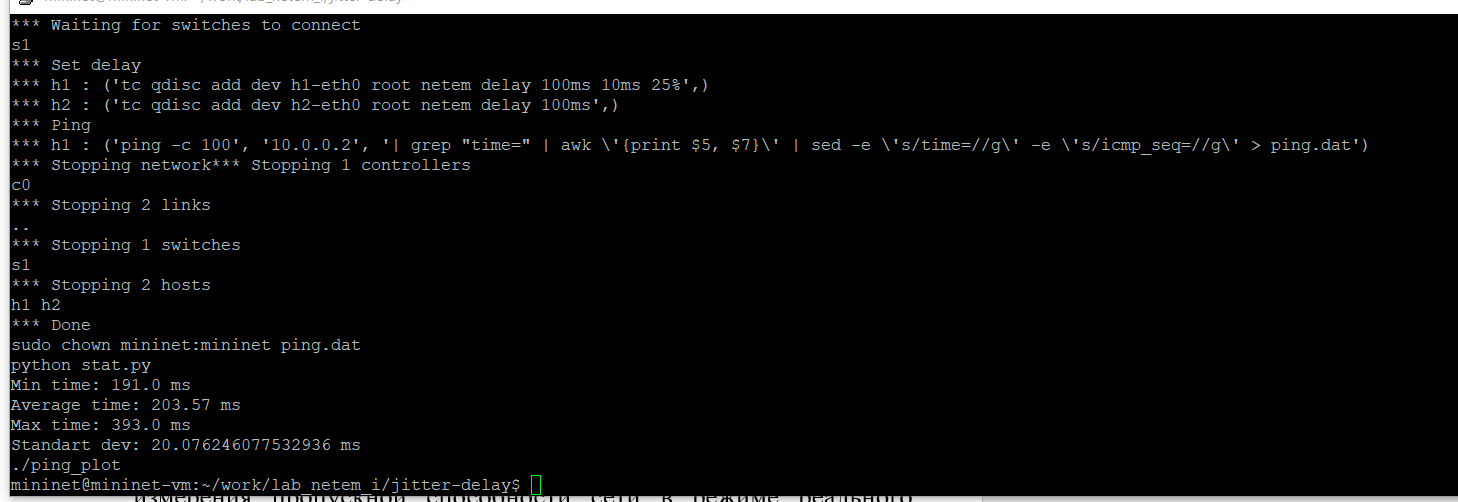
**Change-delay**

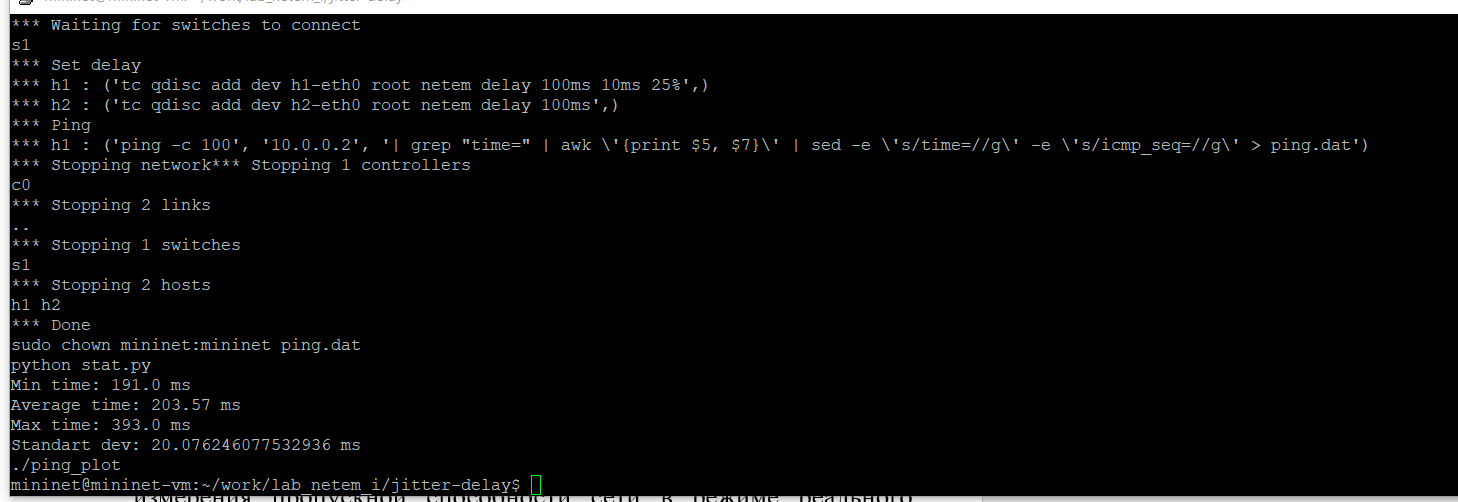
****

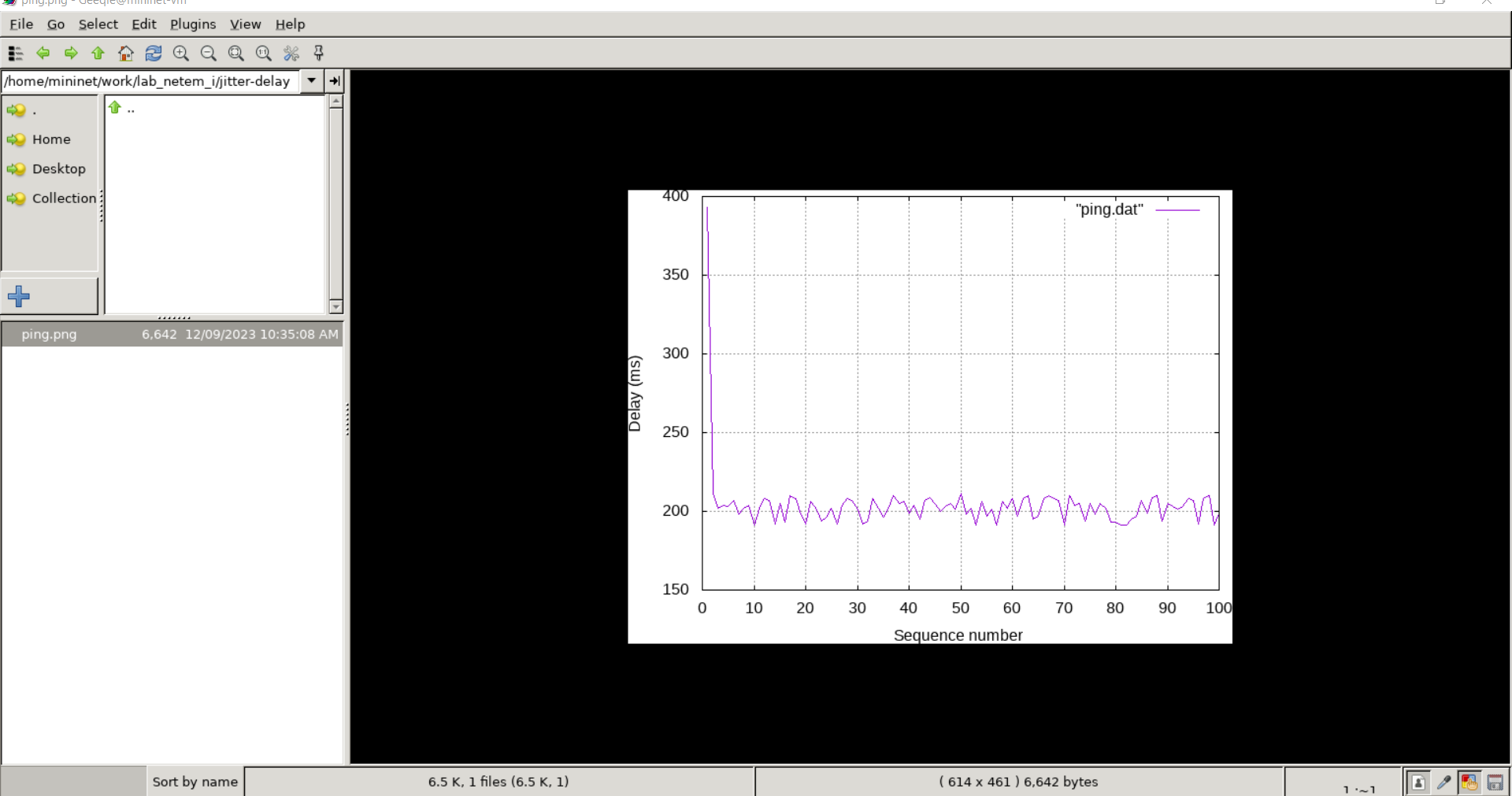
****

**Jitter**

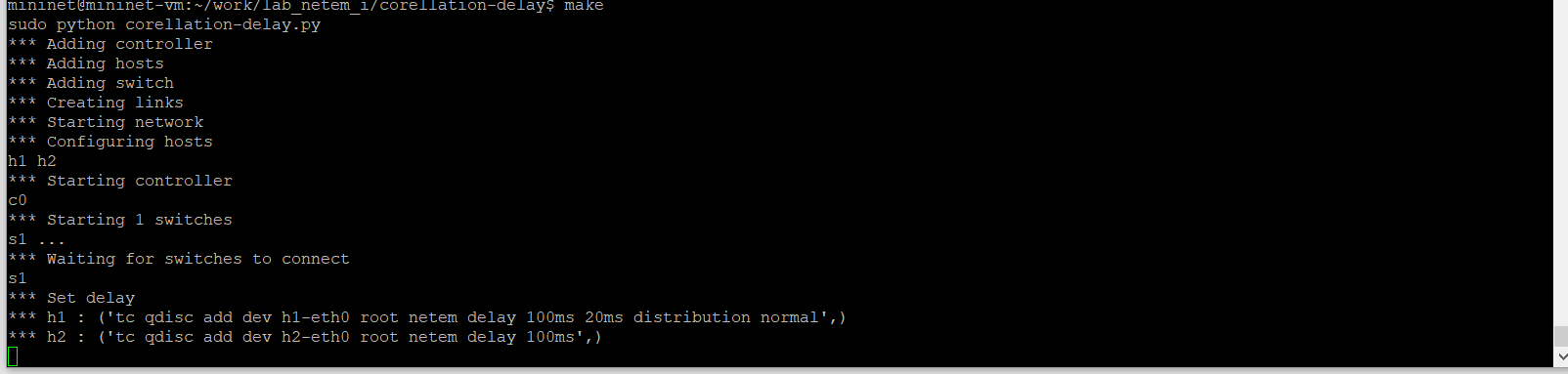


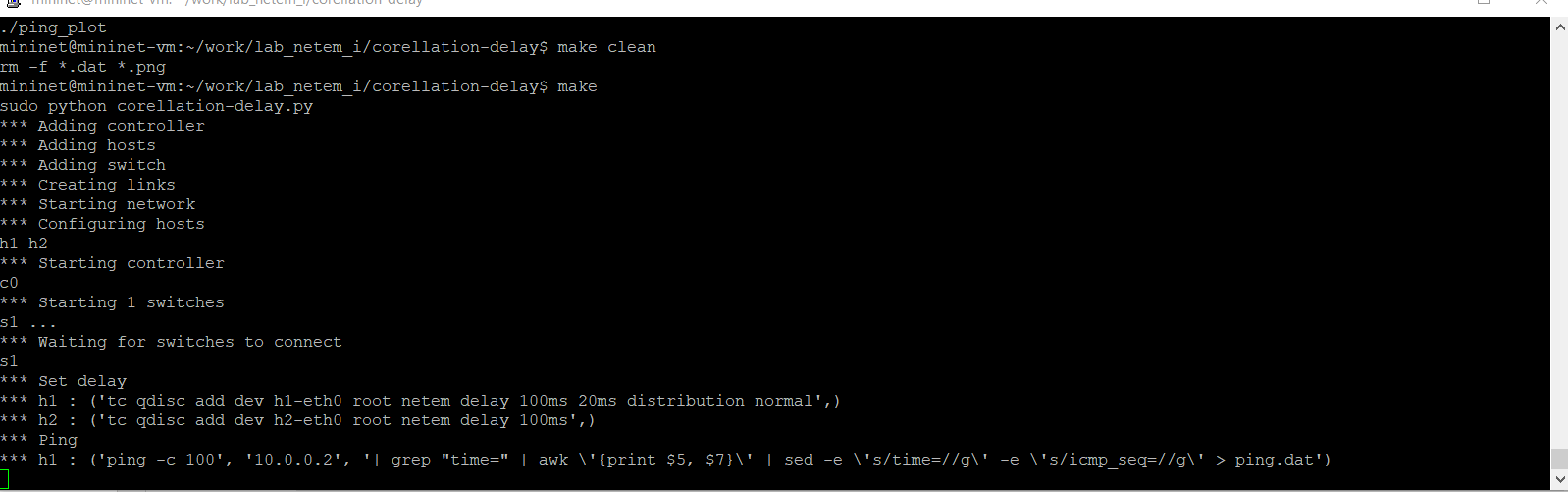
****

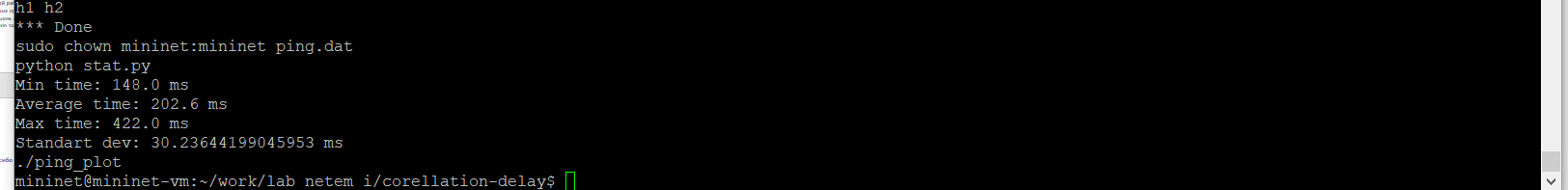


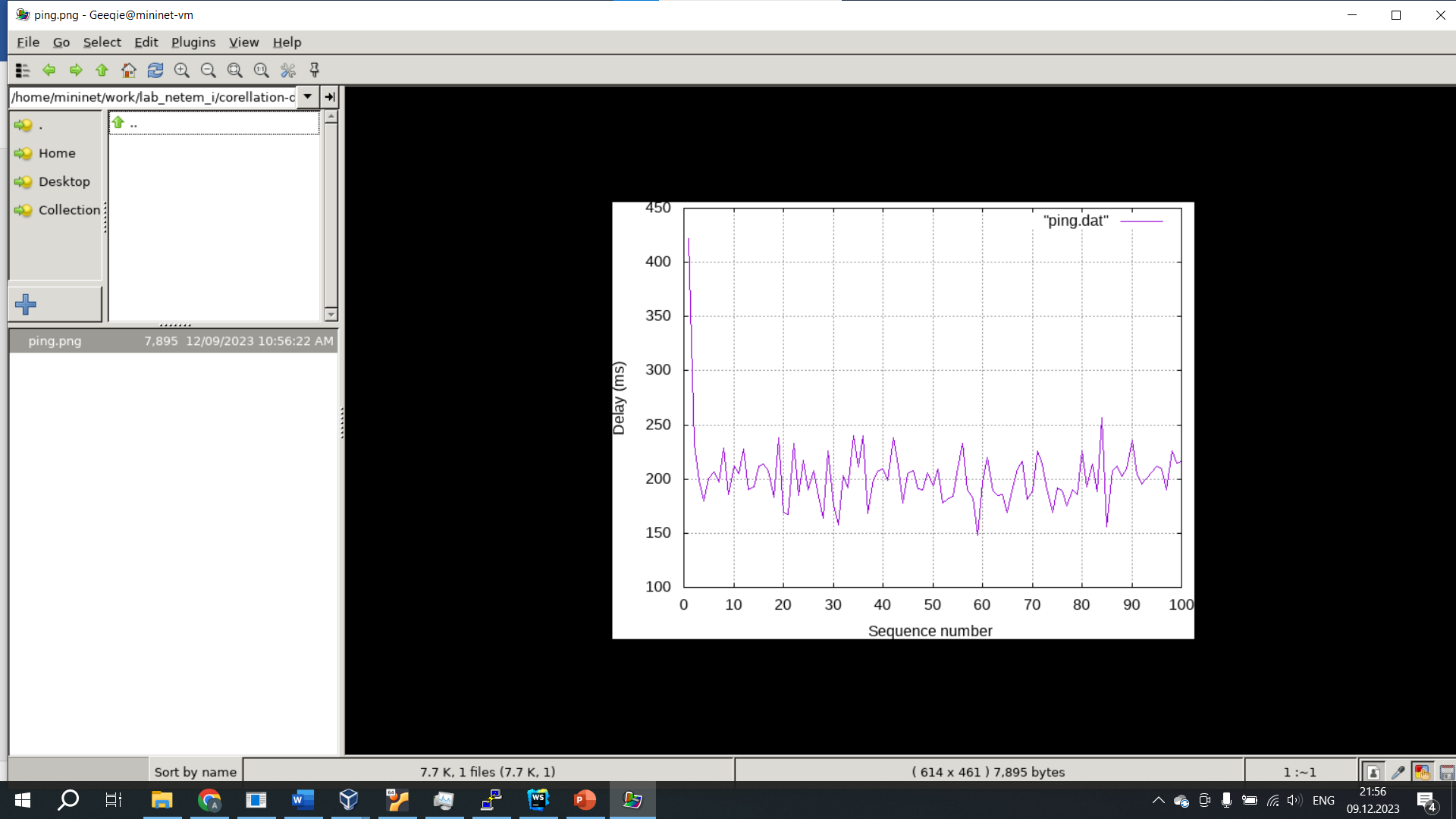


**Correlation-delay**

****

****

****

****

**Вывод**

Итогом лабораторной работы стало знакомство с NETEM — инструментом для тестирования производительности приложений в виртуальной сети, а также получение навыков проведения интерактивного и воспроизводимого экспериментов по измерению задержки и её дрожания (jitter) в моделируемой сети в среде Mininet.