МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра автоматики та управління в технічних системах

**Лабораторна робота № 4**

з дисципліни «Тестування та контроль якості (QA) вбудованих систем»

|  |
| --- |
| Виконала:  студентка групи ІА-92  Артеменко К.С |

Київ – 2022

**Мета лабараторної роботи:**

Ознайомитись з транспортним рівнем. Протоколи TCP та UDP. Виконати

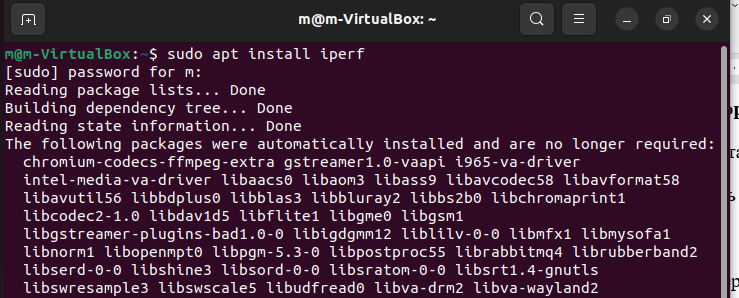
тестування навантаження.

**Порядок виконання завдання лабораторної роботи**

1) Завантажити iperf. Iperf потрібно встановити на двох комп’ютерах. Оскільки

одна система повинна виконувати роль сервера, а інша - роль клієнта.

sudo apt install iperf



2) Запустити Wireshark на клієнті та сервері.

**TCP протокол**

3) За замовчуванням клієнт Iperf підключається до сервера Iperf через TCP-порт

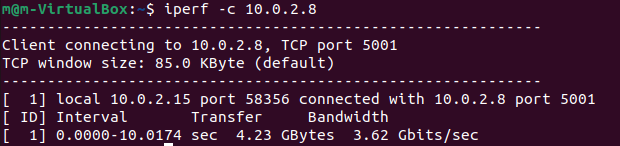
5001. На сервері запустіть iperf-server командою iperf -s. Ви зможете побачити

підключення та результати на своєму сервері iPerf.



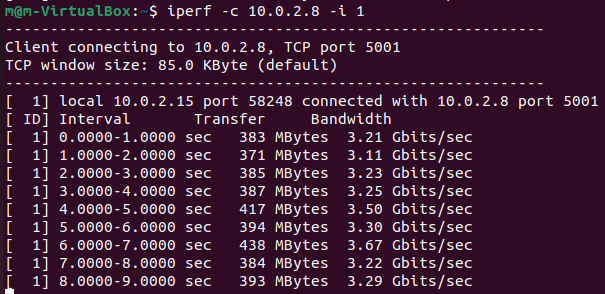
Після цього на клієнті введіть наступну команду:

iperf -c 10.0.2.8

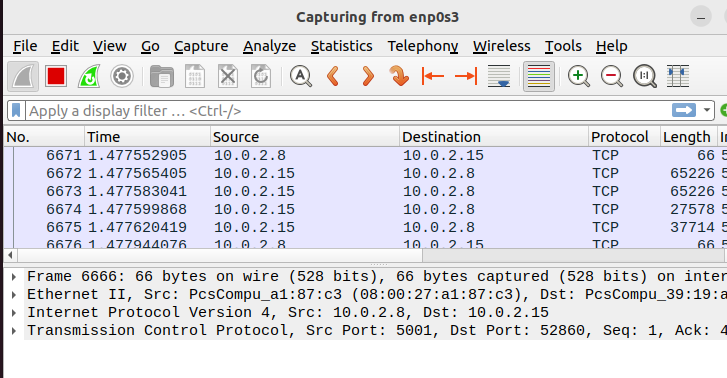


Також, можна використати команду. Параметр -і – інтервал часу між виводами

результату тестування

iperf -c 10.0.0.1 -i 1

4) Відкрийте Wireshark і подивіться на протокол TCP



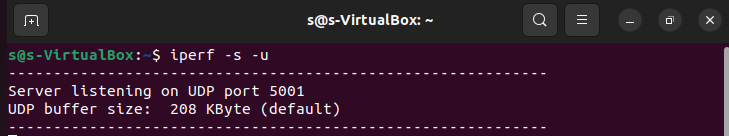
**UDP протокол**

Використовуючи iPerf, ви також можете перевірити максимальну пропускну

здатність, досягнуту через UDP-з'єднання.

5) Запустіть UDP iperf-сервер командою

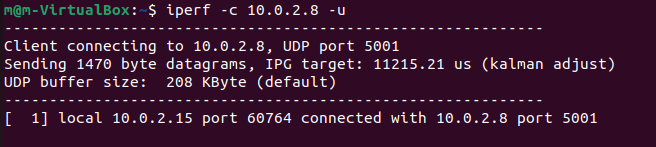
iperf -s -u



6) Підключіть клієнт до UDP iperf-сервера. Замість 10.0.0.1 використайте ІР-

адресу вашого iperf-сервера.

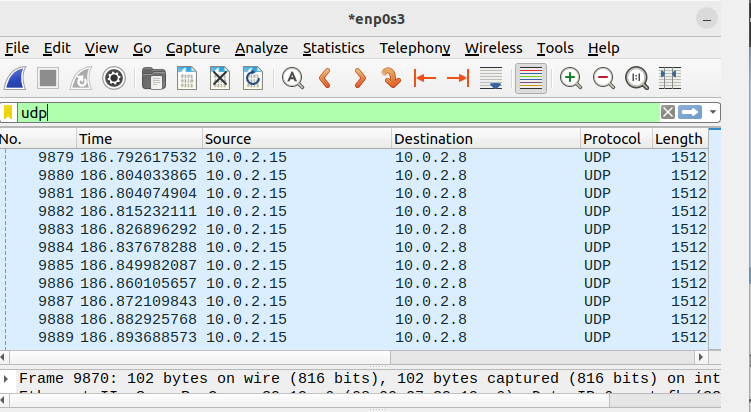
iperf -c 10.0.0.1 -u



Важливо використовувати саме опцію -u. Таким чином ми передаємо iperf те,

що бажаємо підключитися по UDP.

7) Відкрийте Wireshark і подивіться на протокол UDP.



**Тестування навантаження**

1) MTBF test ( Mean time between failures) – середній наробіток між відмовами.

За допомогою TCP-трафіку перевіримо стабільність системи протягом якогось

часу. Для цього нам знадобиться bash скрипт та gnuplot скрипт, завдяки якому

можна проаналізувати трафік у вигляді графіку.

Ось приклад bash скрипту:



В останньому рядку використовується команда для підключення до серверу

iperf. Параметр -t 30 – час який буде виводитися трафік у секундах. Ви можете

збільшити цей час, наприклад, до декількох годин. Краще проводити цей тест

під час ночі, оскільки в цей час найменше навантаження на систему. Також, не

забуваємо про параметр -i 1, який відповідає за інтервал виводу інформації про

трафік у секундах. Тож, якщо тестувати трафік протягом декількох годин, то

цей параметр краще збільшити, щоб не сильно навантажувати систему (

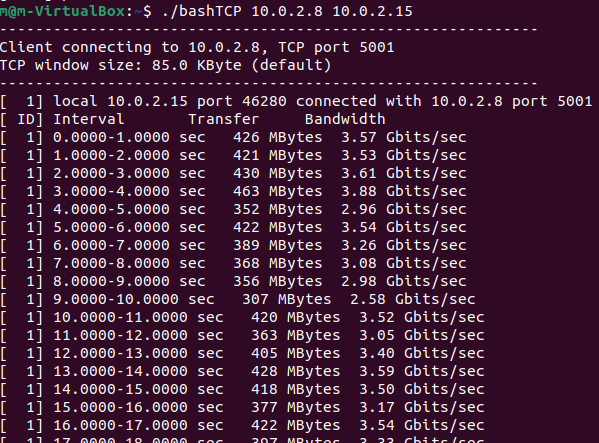
наприклад, зробити параметр -i 60).

Після створення bash скрипту не забуваємо зробити з нього виконуваний файл:



Тепер можна запустити скрипт командою:

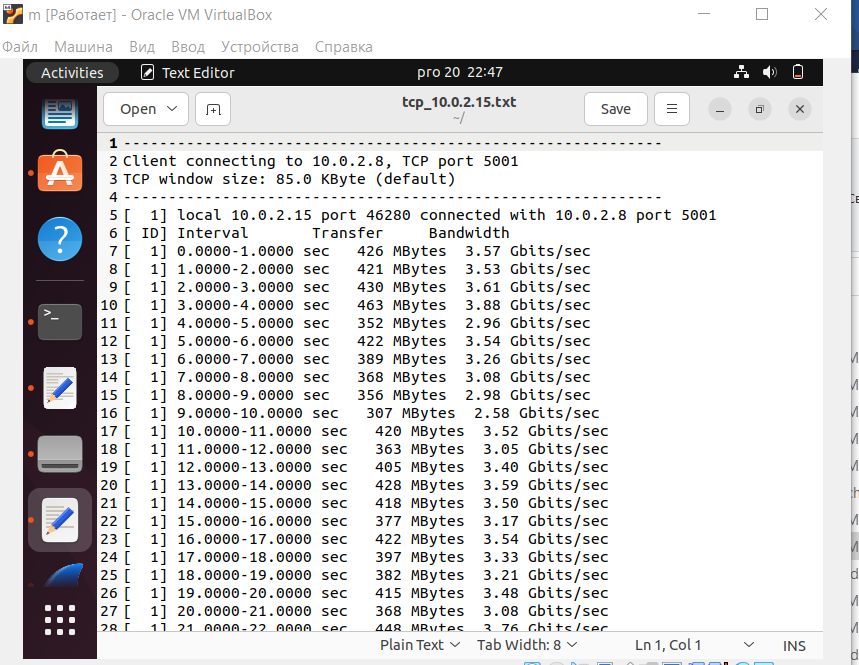
./bashTCP <IP1> <IP2> , де IP1 – адреса сервера, а IP2 – адреса клієнта.



Перед виконанням скрипта, запустіть iperf сервер. В результаті, в папці із

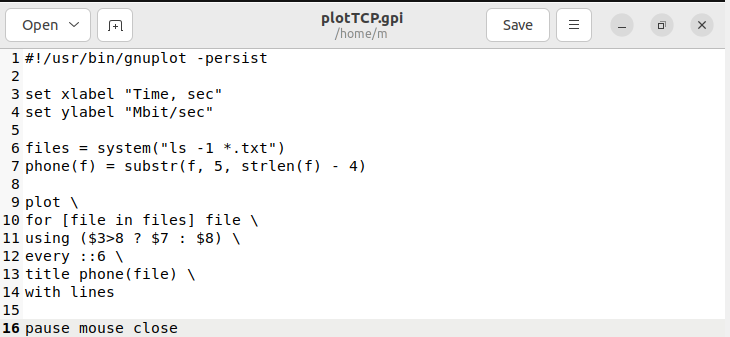
скриптом створиться \*.txt файл, який буде містити трафік, що буде

використаний для графіку.



Тепер нам потрібно створити gnuplot скрипт, який виведе графік. Ось його

приклад:



Після його створення, також зробимо цей файл виконуваним:

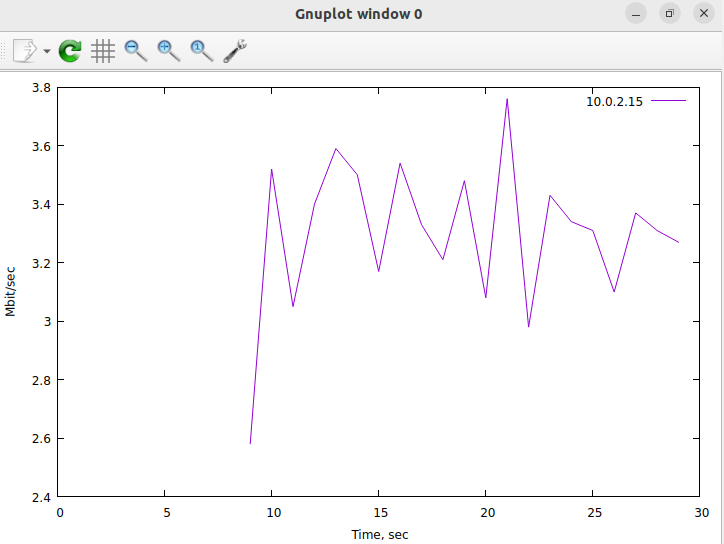


Та викличимо його:

./plotTCP.gpi

Для виклику цього скрипта не потрібні параметри, якщо цей скрипт та \*.txt

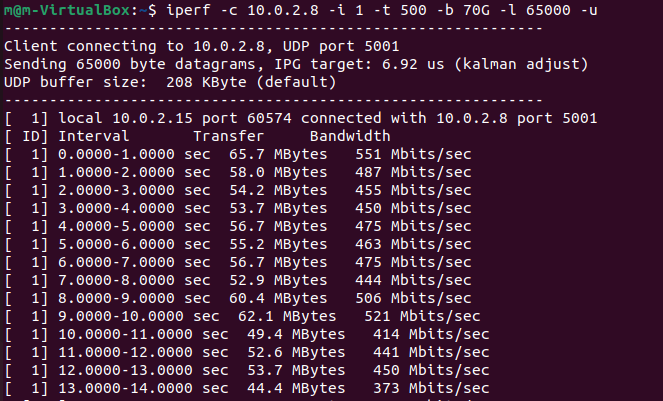
файл із трафіком знаходяться в одній директорії.

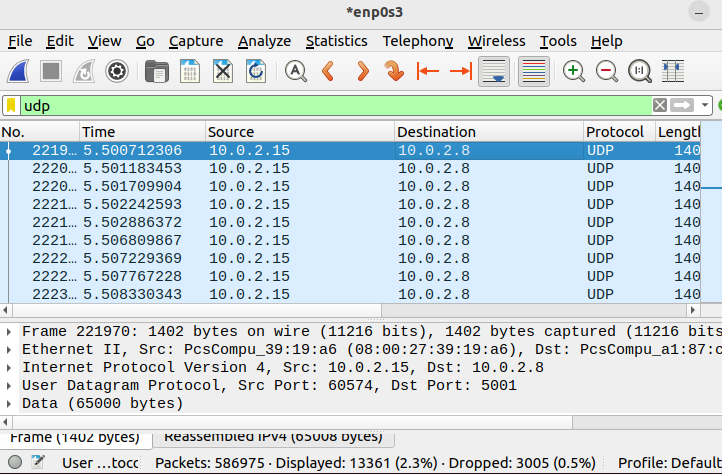


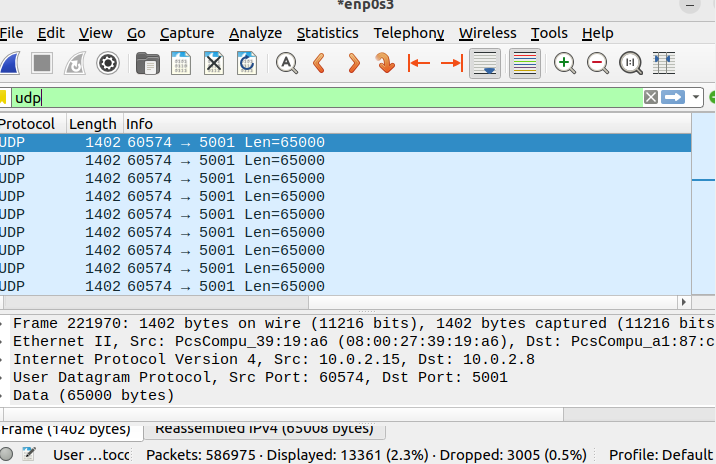
2) Виконаємо тестування за допомогою UDP – трафіку. Будемо надсилати

трафік великими пакетами, і при цьому буде навантажуватися оперативна

пам’ять iperf -c 192.168.1.105 -i 1 -t 500 -b 70G -l 65000 -u





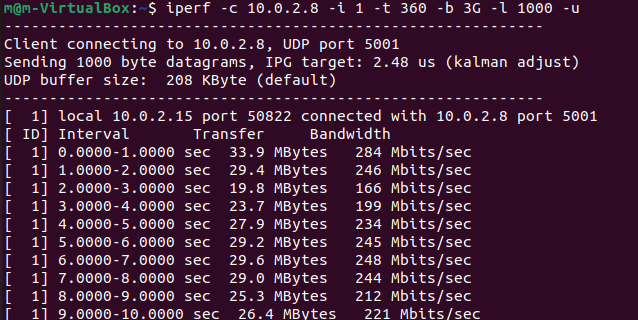
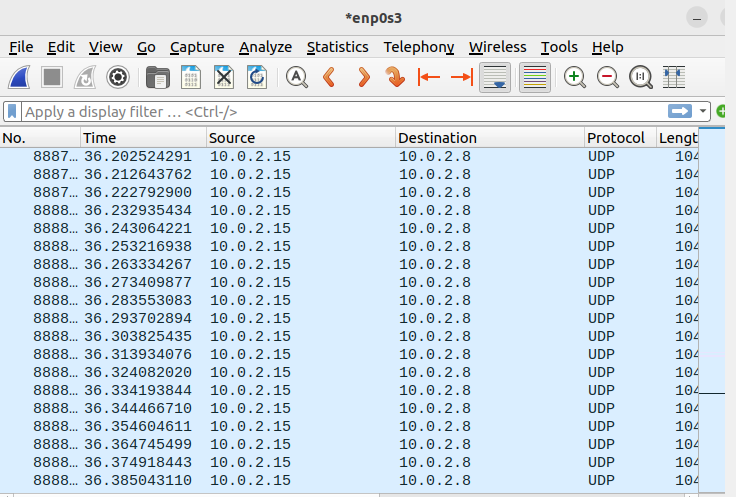


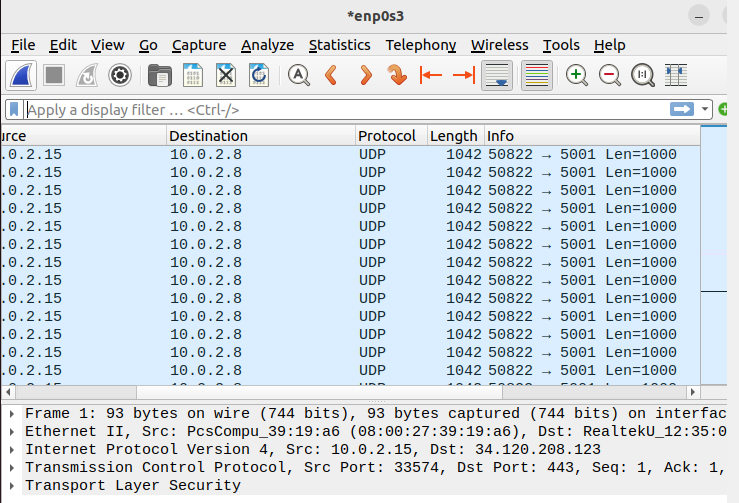
3) Також проведемо тестування за допомогою UDP – трафіку. Але теперь

будемо надсилати трафік маленькими пакетами. Таким чином, буде

навантажуватися CPU.

iperf -c 192.168.1.105 -i 1 -t 360 -b 3G -l 1000 -u



При такому тестуванні (як в 2 та 3 пунктах) ми перевіряємо, чи відповідає

консоль, веб – камера, або інші основні функції. Тому що, при навантаженні

процесора та оперативної пам’яті система може не впоратися. Також, можна

перевірити, чи впоряється роутер з таким навантаженням ( під час тестування

спробувати вийти в інтернет з іншого пристрою)

Посилання на репозиторій: https://github.com/kkotuha/LR4

**Висновок:** Під час виконання даної лабораторної роботи я ознайомилась з транспортним рівнем, протоколами TCP та UDP. Виконала тестування навантаження.