Networks Lab Assignment 1

Q.1

- **UDP**
  1. **Steps Followed :** *iperf3 -c 10.5.18.163 -u -b 28000*
  2. **Different Protocols Observed :** *UDP, TCP, IPV4*
  3. **Justification :** IPv4 is the Internet Protocol over which TCP and UDP is executed. In this link TCP and UDP are both used to run a test.

- **TCP**
  1. **Steps Followed :** *wget --no -proxy http://10.5.18.163:8000/1.jpg*
  2. **Different Protocols Observed :** *HTTP, TCP*
  3. **Justification :** HTTP is the application layer protocol through which the GET request for the image is sent to the server. The image data is sent to the client from the server via the TCP.

Q.2

a.

1. **Steps Followed :**

```
wget --no -proxy http://10.5.18.163:8000/1.jpg
wget --no -proxy http://10.5.18.163:8000/2.jpg
wget --no -proxy http://10.5.18.163:8000/3.jpg
wget --no -proxy http://10.5.18.163:8000/4.jpg
wget --no -proxy http://10.5.18.163:8000/5.jpg
```

2. **Observation:**

Image	Number Of TCP Packets	Different Sizes of Packets Observed
1.jpg	919	**66** (ACK), **1514** (Image packets), **74** (SYN-ACK), **83** (PSH, ACK)
2.jpg	3175	**66** (ACK), **1514** (Image packets), **74** (SYN-ACK), **78** (DUP ACK), **83** (PSH, ACK), **303** (TCP Previous segment not captured)
3.jpg	6714	**66** (ACK), **1514** (Image packets), **74**(SYN-ACK), **78** (DUP ACK), **416** (TCP Previous segment not captured), **482** (TCP Previous segment not captured), **83** (PSH, ACK)
4.jpg	3567	**66** (ACK), **1514** (Image packets), **74** (SYN-ACK), **78** (DUP ACK), **83** (PSH, ACK), **516** (TCP Previous segment not captured)
5.jpg	3713	**66** (ACK), **1514** (Image packets), **74** (SYN-ACK), **194** (Window Full), **78** (DUP ACK), **83** (PSH, ACK)

iii. **Justification :** The primary packets sent through the connection are the Image Data Packets and the Acknowledgement Packets (ACK). The other packets are:-

- SYN, SYN-ACK, ACK - This is the TLS three way handshake that happens during the establishment of the connection between the client and the server.
- DUP ACK - This packet is primarily sent when packets arrive out of order or when there is a missing packet
- PSH, ACK - Indicates that the sender is acknowledging receipt of some previous data and also transmitting some more data.
- TCP Window Full - This packet indicates that the receiving window (buffer) is full.
- TCP Preveious Segment not Captured - This indicates there is a gap in the sequence transmitted.

b.

i. **Steps Followed :** *iperf3 -c 10.5.18.163 -u -b 28000*

ii. **Observation :** Two sizes were observed i.e. 834, 46

iii. **Justification :** Most of the packets are of length 834 Bytes whereas the first 2 UDP packets are of 46 Bytes

c.

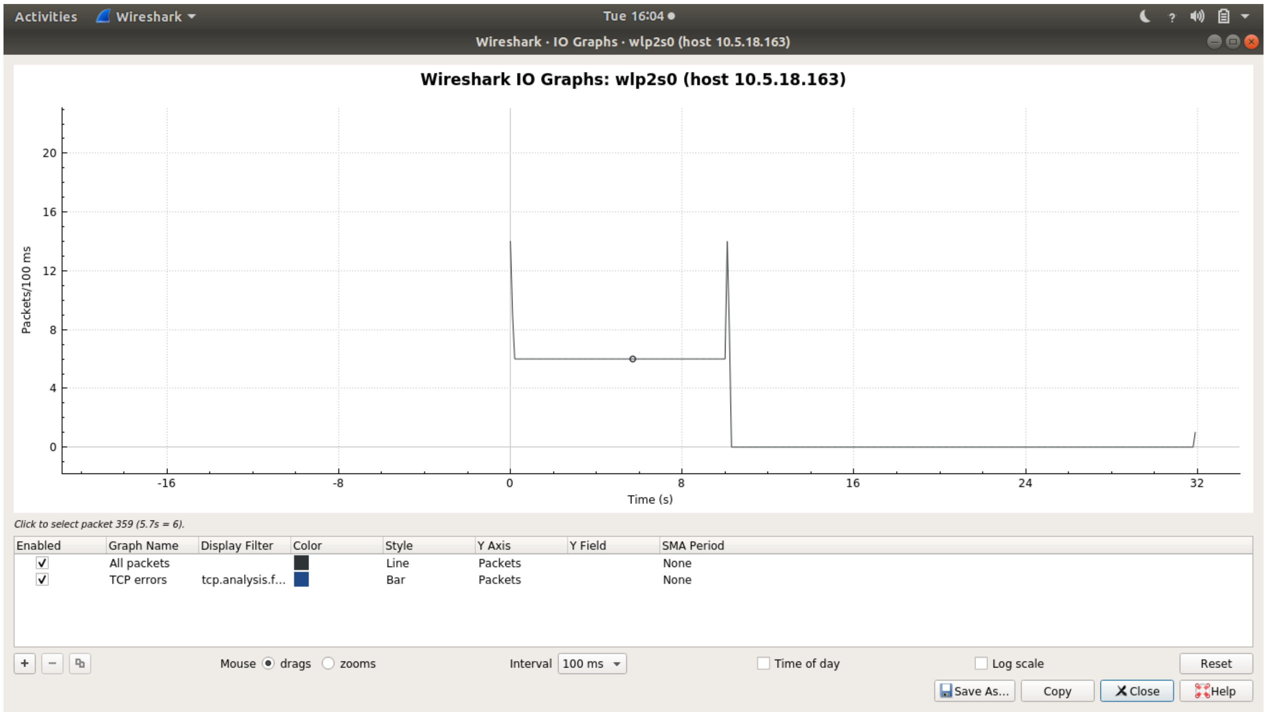
i. **Steps Followed :**

```
iperf3 -c 10.5.18.163 -u -b 28000

wget --no -proxy http://10.5.18.163:8000/1.jpg
wget --no -proxy http://10.5.18.163:8000/2.jpg
wget --no -proxy http://10.5.18.163:8000/3.jpg
wget --no -proxy http://10.5.18.163:8000/4.jpg
wget --no -proxy http://10.5.18.163:8000/5.jpg
```

ii. **Observation (Graphs):**

UDP



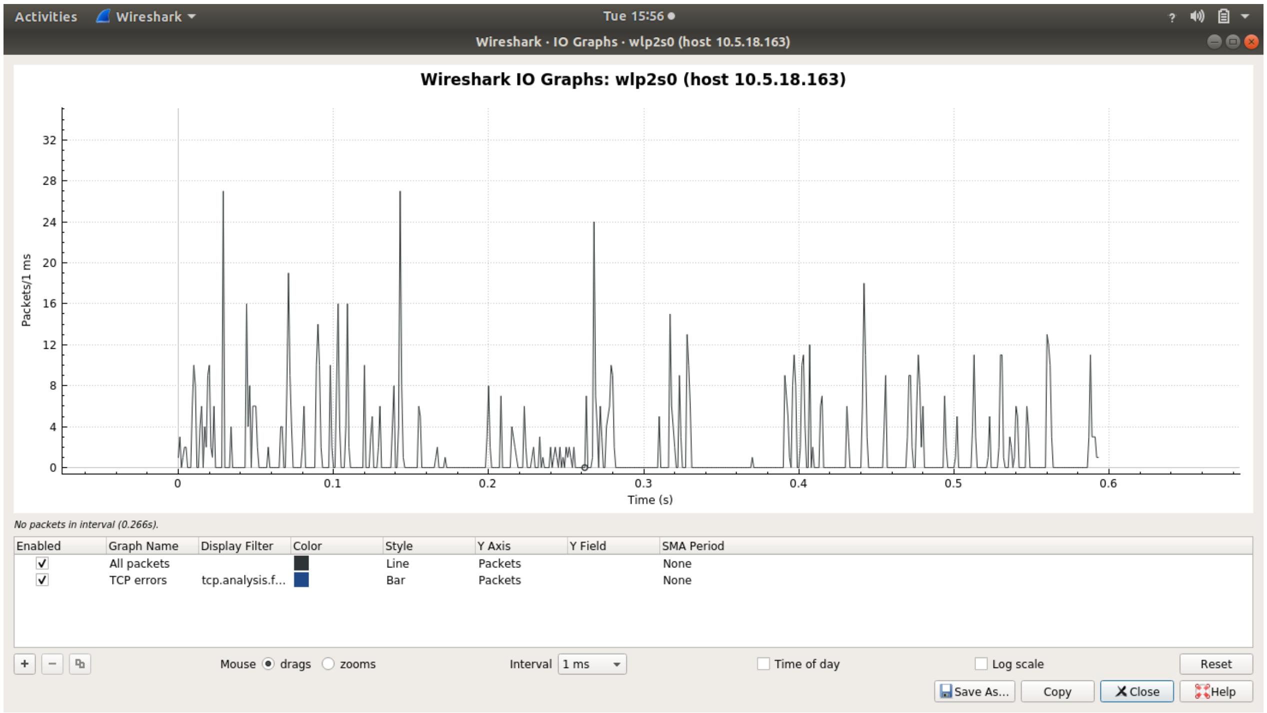
Click to select packet 359 (5.7s = 6).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period
☑	All packets		■	Line	Packets		None
☑	TCP errors	tcp.analysis.f...	■	Bar	Packets		None

Mouse ● drags ○ zooms Interval 100 ms □ Time of day □ Log scale

TCP

Image 1

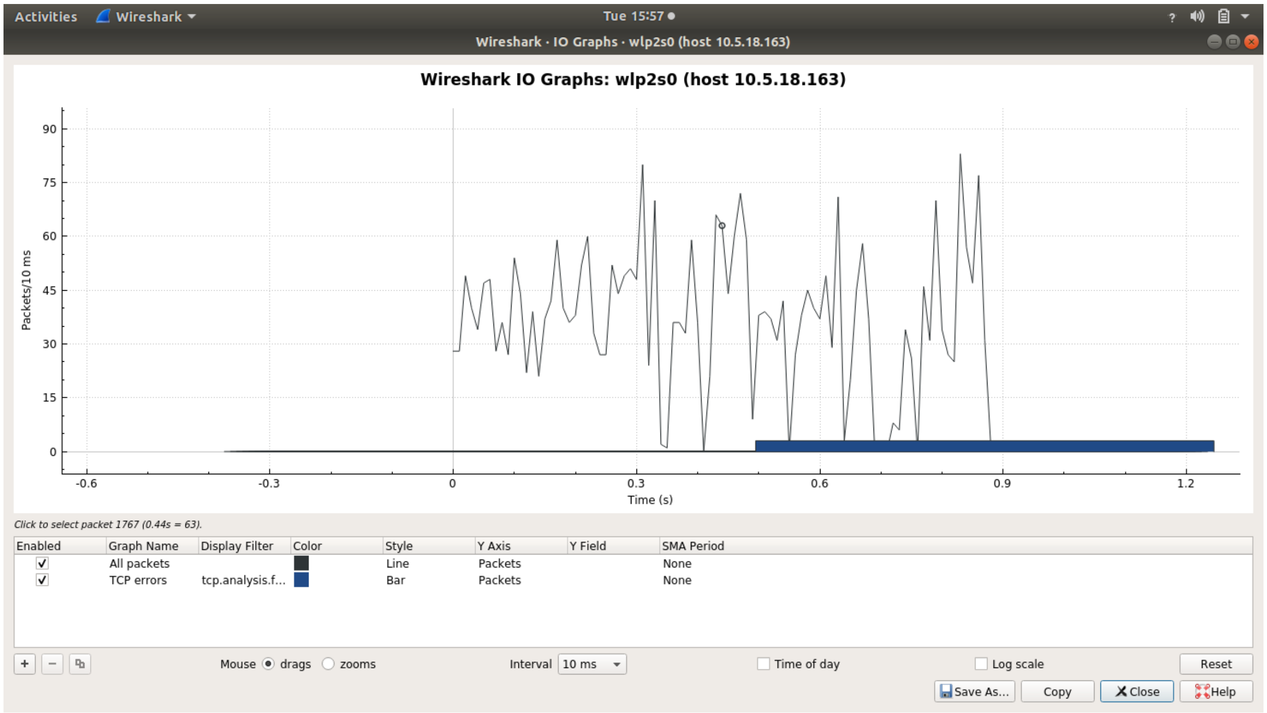


No packets in interval (0.266s).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period
☑	All packets		■	Line	Packets		None
☑	TCP errors	tcp.analysis.f...	🟦	Bar	Packets		None

Mouse ● drags ○ zooms Interval 1 ms □ Time of day □ Log scale

Image 2

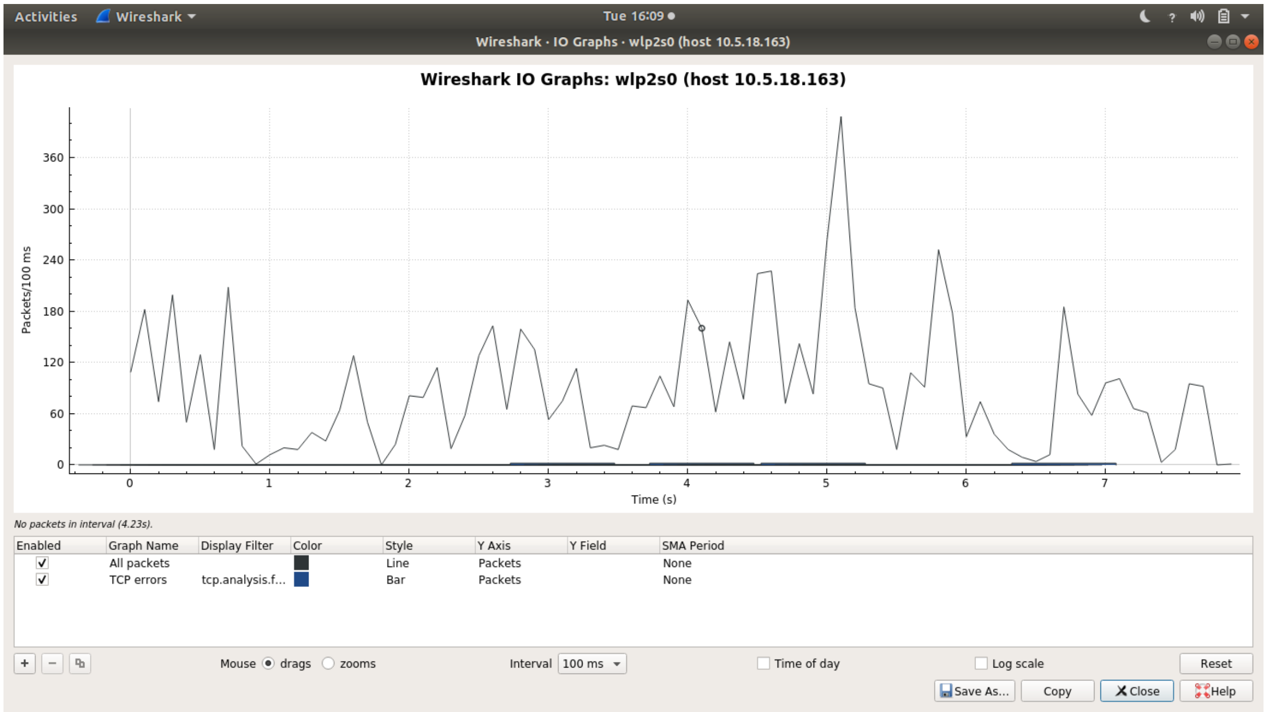


Click to select packet 1767 (0.44s = 63).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period
☑	All packets		■	Line	Packets		None
☑	TCP errors	tcp.analysis.f...	🟦	Bar	Packets		None

Mouse ● drags ○ zooms Interval 10 ms □ Time of day □ Log scale

Image 3

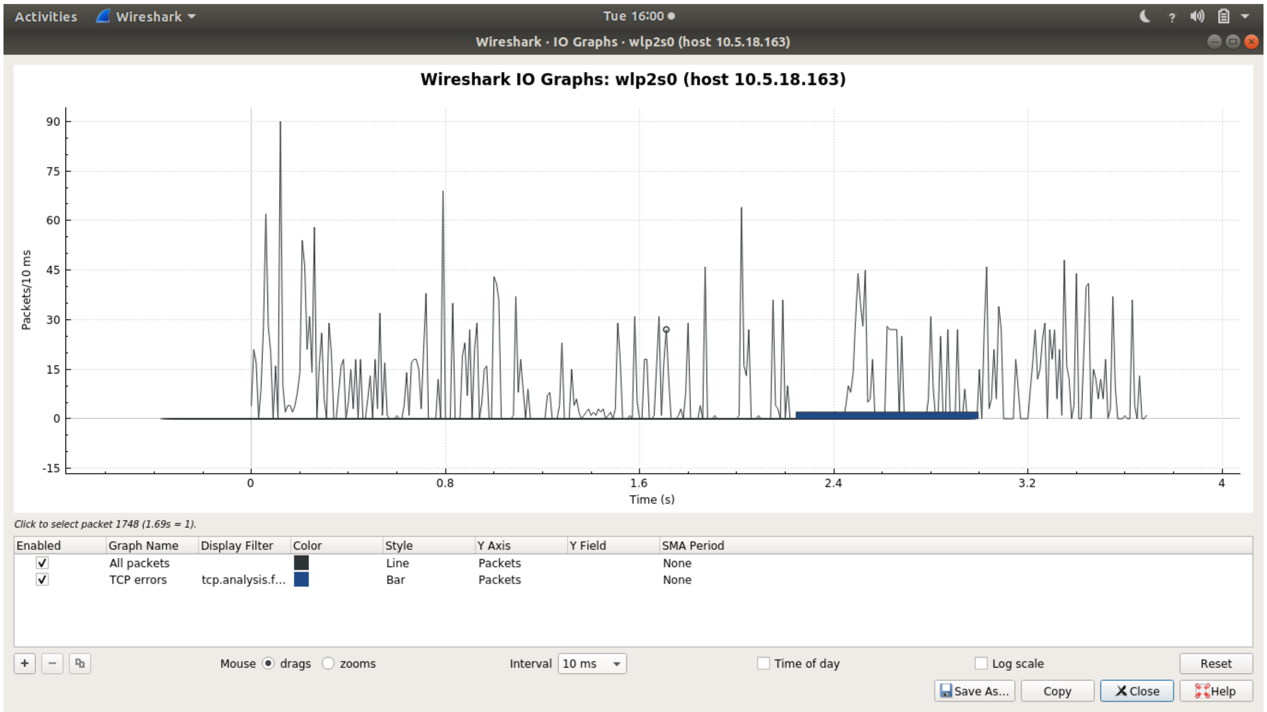


No packets in interval (4.23s).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period
☑	All packets		⬛	Line	Packets		None
☑	TCP errors	tcp.analysis.f...	🟦	Bar	Packets		None

Mouse ● drags ○ zooms   Interval 100 ms   □ Time of day   □ Log scale

Image 4

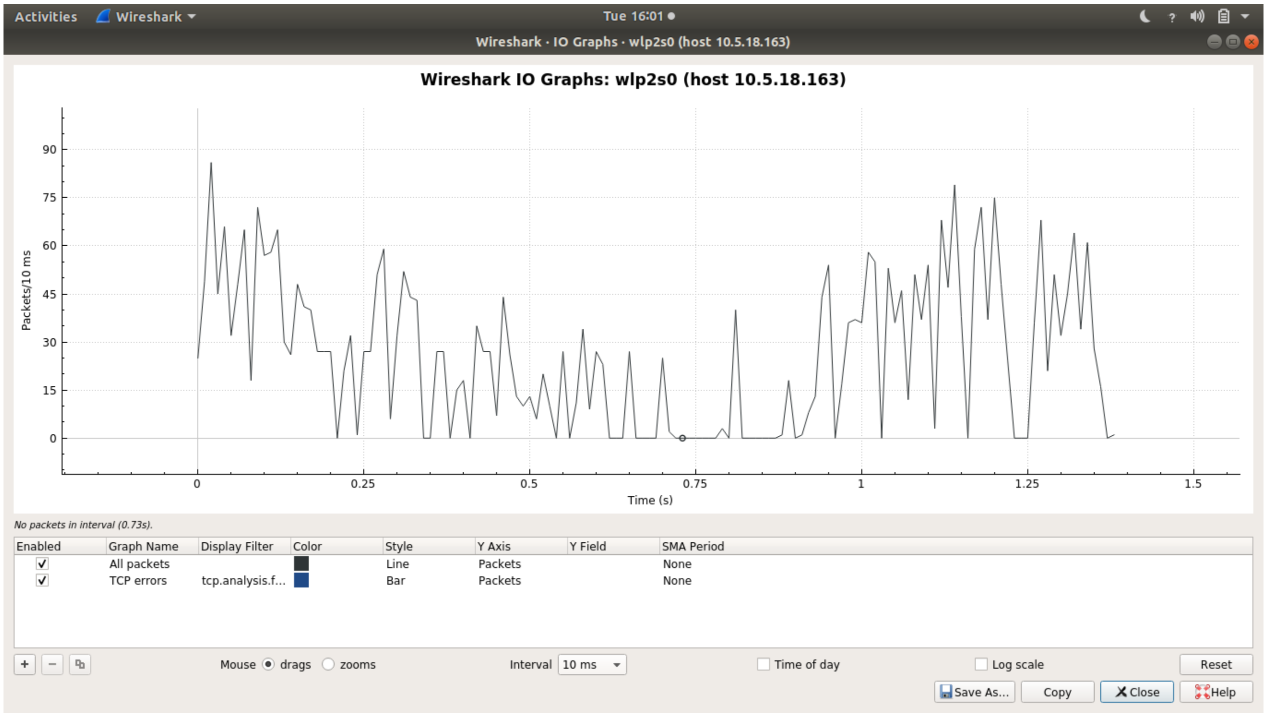


Click to select packet 1748 (1.69s = 1).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period
☑	All packets		⬛	Line	Packets		None
☑	TCP errors	tcp.analysis.f...	🟦	Bar	Packets		None

Mouse ● drags ○ zooms   Interval 10 ms   □ Time of day   □ Log scale

Image 5



No packets in interval (0.73s).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period
☑	All packets		■	Line	Packets		None
☑	TCP errors	tcp.analysis.f...	🟦	Bar	Packets		None

Mouse ● drags ○ zooms

Interval 10 ms

□ Time of day

□ Log scale

iii. **Justification:** When packets are sent using UDP, the throughput remains almost constant because every time packets are sent independently of each other and is not restricted by the error recovery process and just discards erroneous packets.

But in TCP, the throughput has a huge variance which depends on the server and the link established between the client and the server. This can be seen from the statistics of the TCP trace for various images.

d.

i. **Steps Taken:**

```
iperf3 -c 10.5.18.163 -u -b 28000
iperf3 -c 10.5.18.163 -u -b 64000
iperf3 -c 10.5.18.163 -u -b 128000
iperf3 -c 10.5.18.163 -u -b 256000
iperf3 -c 10.5.18.163 -u -b 512000
iperf3 -c 10.5.18.163 -u -b 1024000
iperf3 -c 10.5.18.163 -u -b 2048000
```

ii. **Observation**

Bandwidth (Kbps)	Throughput (Bytes / sec)
28	4393
64	8528
128	16000
256	31200
512	62400
1024	121000
2048	242000

iii. **Justification:** As we increase the bandwidth, the amount of data transmitted per second increases almost linearly. This is because, the amount of data tries to reach the maximum possible value restricted by the bandwidth as UDP doesnt use ACK packets.

Q.3

i. **Steps Followed :**

```
wget --no -proxy http://10.5.18.163:8000/1.jpg
wget --no -proxy http://10.5.18.163:8000/2.jpg
wget --no -proxy http://10.5.18.163:8000/3.jpg
wget --no -proxy http://10.5.18.163:8000/4.jpg
wget --no -proxy http://10.5.18.163:8000/5.jpg
```

ii. **Observation**

Image	Packets Retransmitted
1	0
2	1
3	2
4	0

5	1

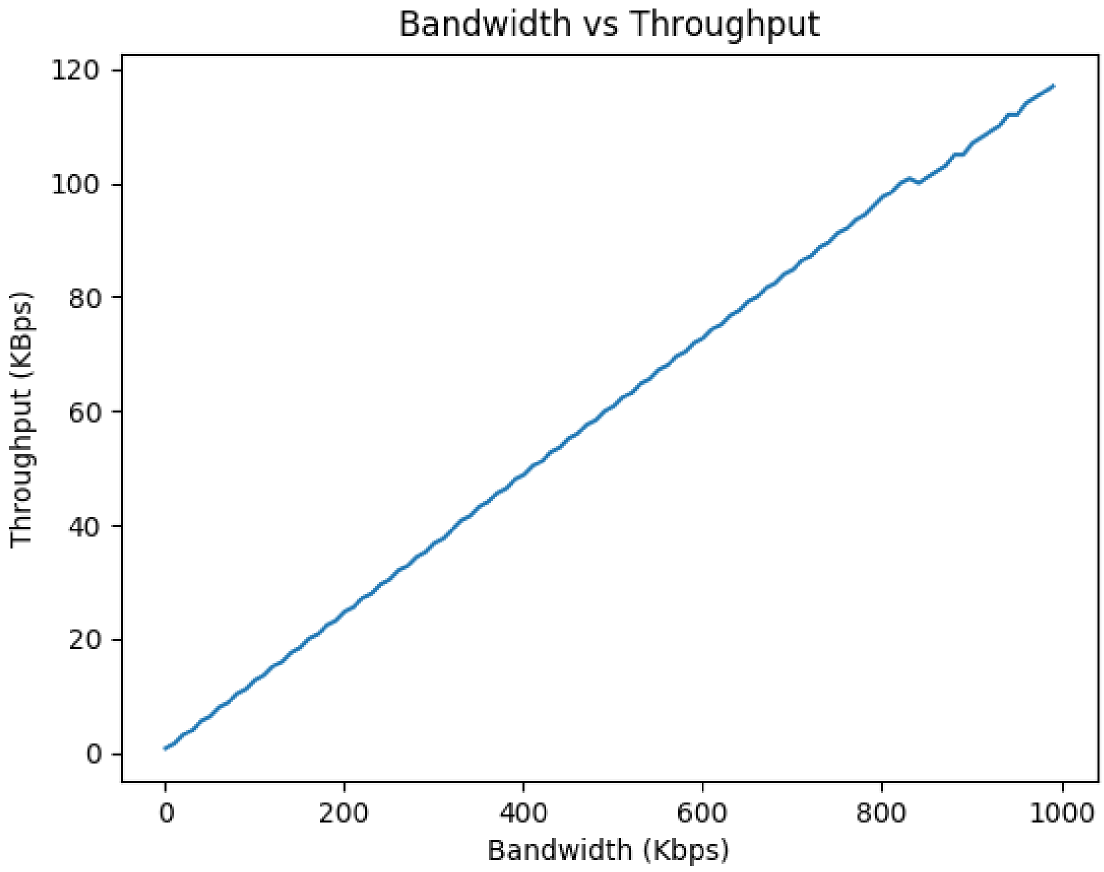
iii. **Justification:** Packets are re-transmitted when a particular packet is lost in the link when it is sent from the server to the client or vice-versa. Usually the percentage of packets that is lost amounts to a very small percentage of the total number of packets sent, and hence the number of packets that are re-transmitted are extremely low. In these images, the percentage of packets re-transmitted varies from 0.01 - 0.05 percent.

Q.4

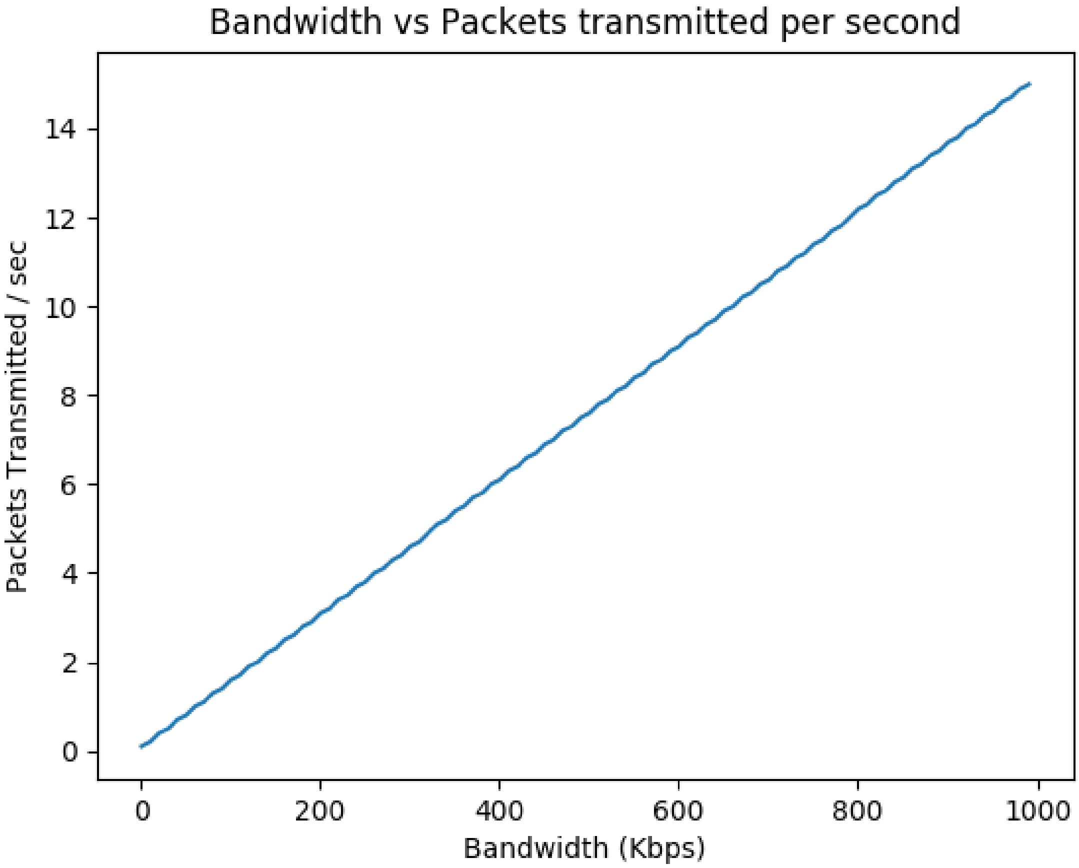
i. **Steps Taken :** Retrieved throughput and packet count for 100 different bandwidths and plotted using matplotlib library in Python.

ii. **Observation :**

a.



b.



iii. **Justification :** We can see that the number of packets transmitted per second and the throughput varies almost linearly with the bandwidth. This can be understood intuitively from the fact that bandwidth restricts the maximum data that can be sent at a time. The amount of data transmitted will always try to approach the theoretical maximum because the UDP protocol sends a continuous stream of packets with a lack of ACK. Hence, we can see a linear increase in the packets transmitted and the throughput as the bandwidth increases.