==========

Assignment 3: TCP Sockets Submission:

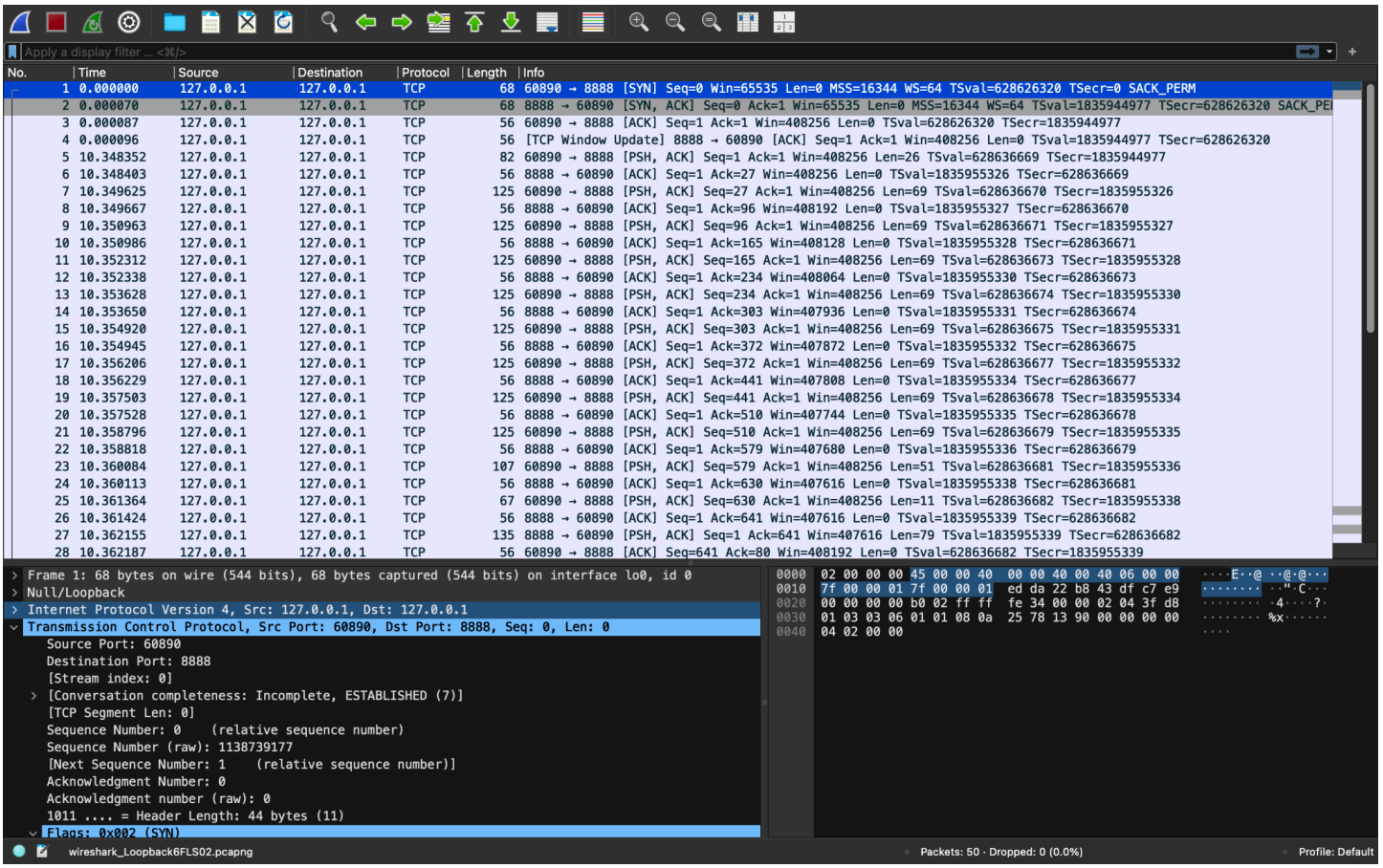
Name: Chandransh Singh

Roll number: 22CS30017

Link of the pcap file: TCP_packets

==========

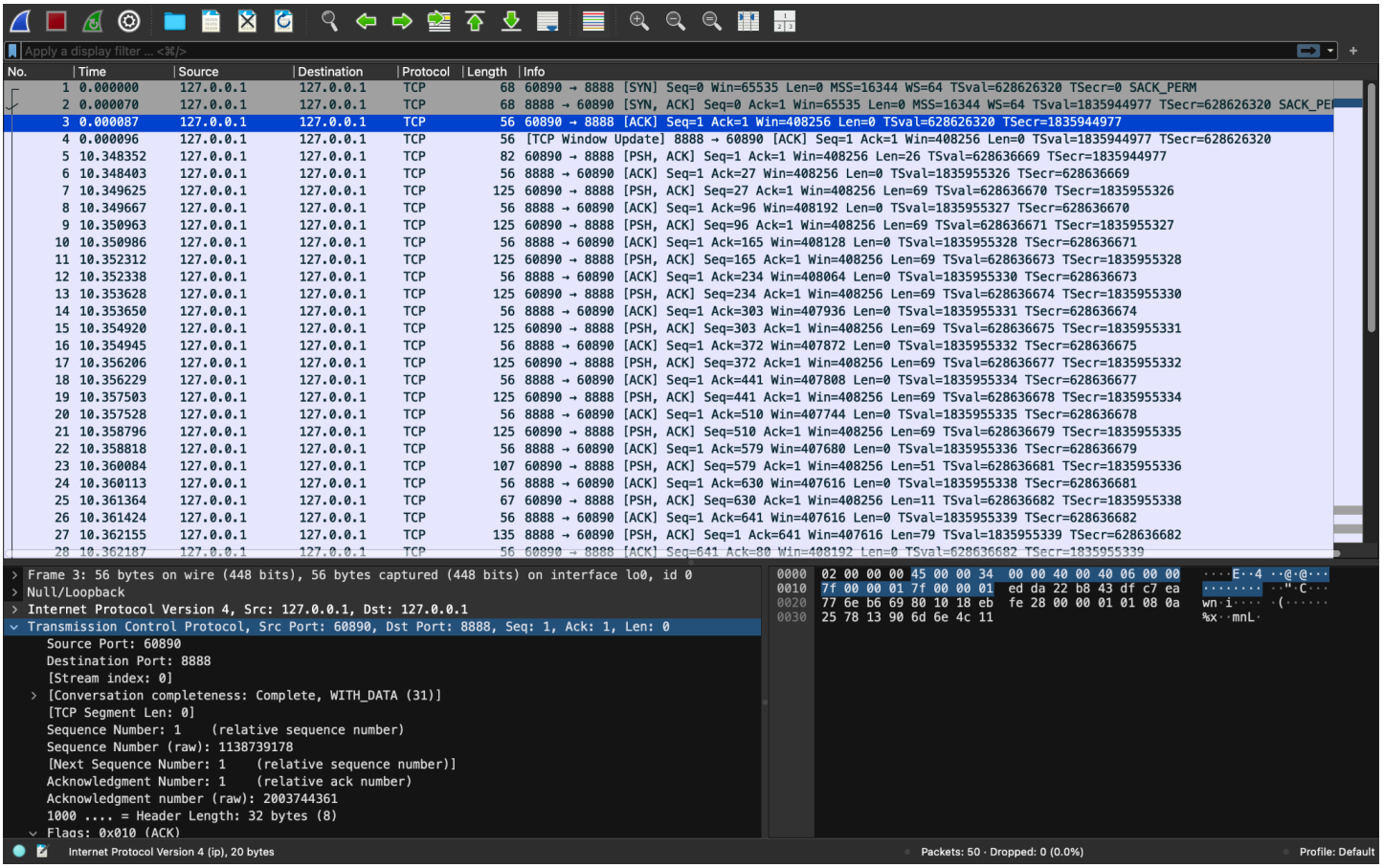
1. What are the source and destination IP addresses and ports? Share the screenshots to justify your answer.



From the above screenshot, we can see the first SYN packet where:

- Source (client):
  - Address: `127.0.0.1`
  - Port: `60890`
- Destination (server):
  - Address: `127.0.0.1`
  - Port: `8888`

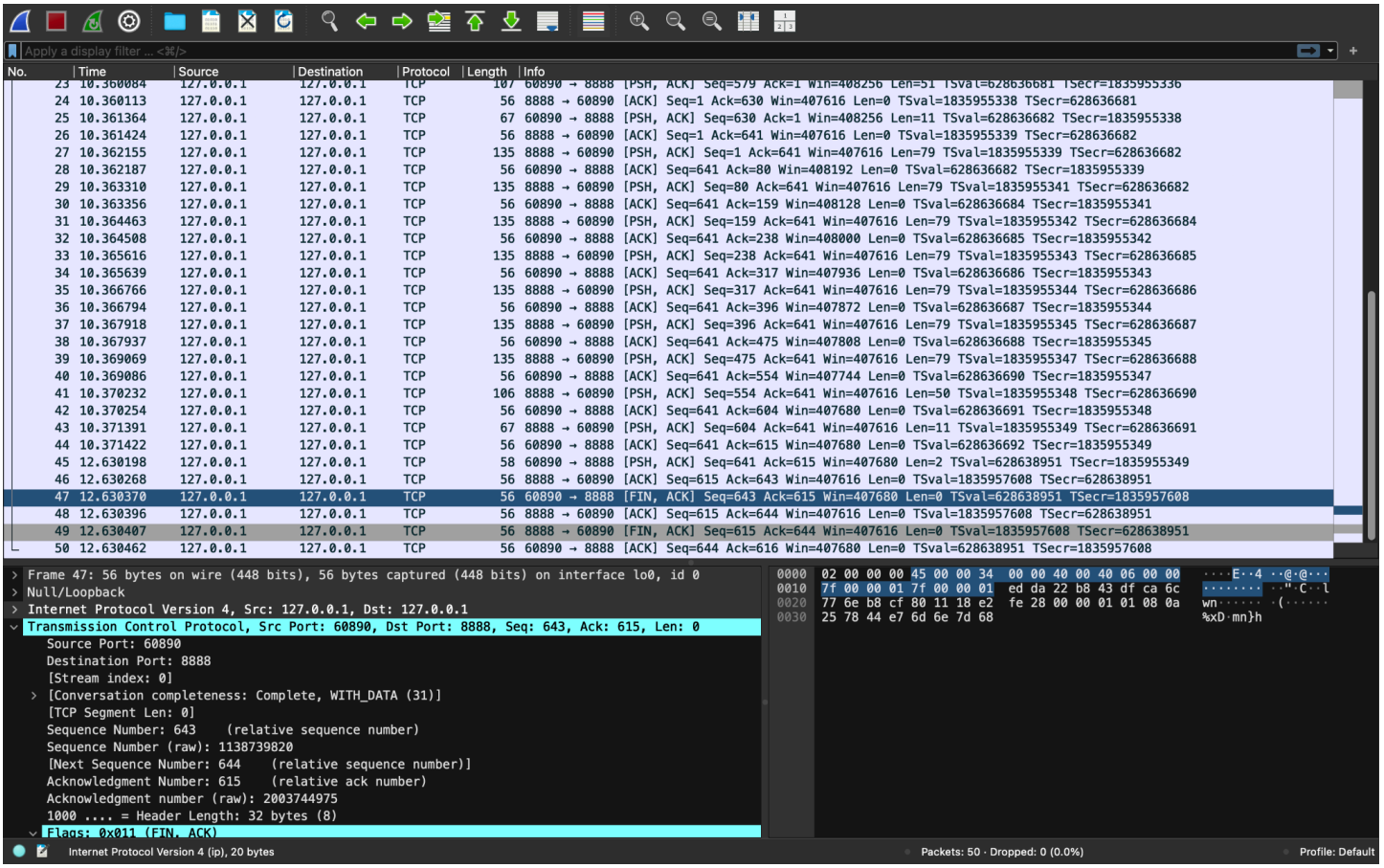
2. Inspect the Three-way handshaking procedure and capture all packets exchanged for it. Attach the necessary screenshots to demonstrate it



Three-way Handshake:

- Look for the **first three** packets in the above screenshot:
  - SYN (Client → Server)
  - SYN-ACK (Server → Client)
  - ACK (Client → Server)

3. Inspect the connection closure procedure and capture all packets exchanged for it. Attach the necessary screenshots to demonstrate it.



We can find the sequence **at the end**:

- FIN (Client → Server)
- ACK (Server → Client)
- FIN (Server → Client)
- ACK (Client → Server)

4. Inspect the traffic and count the number of packets exchanged for the transfer of a file(related to data only) between client and server. Plot a graph ‘file size vs the number of packets’ clearly based on your observation.

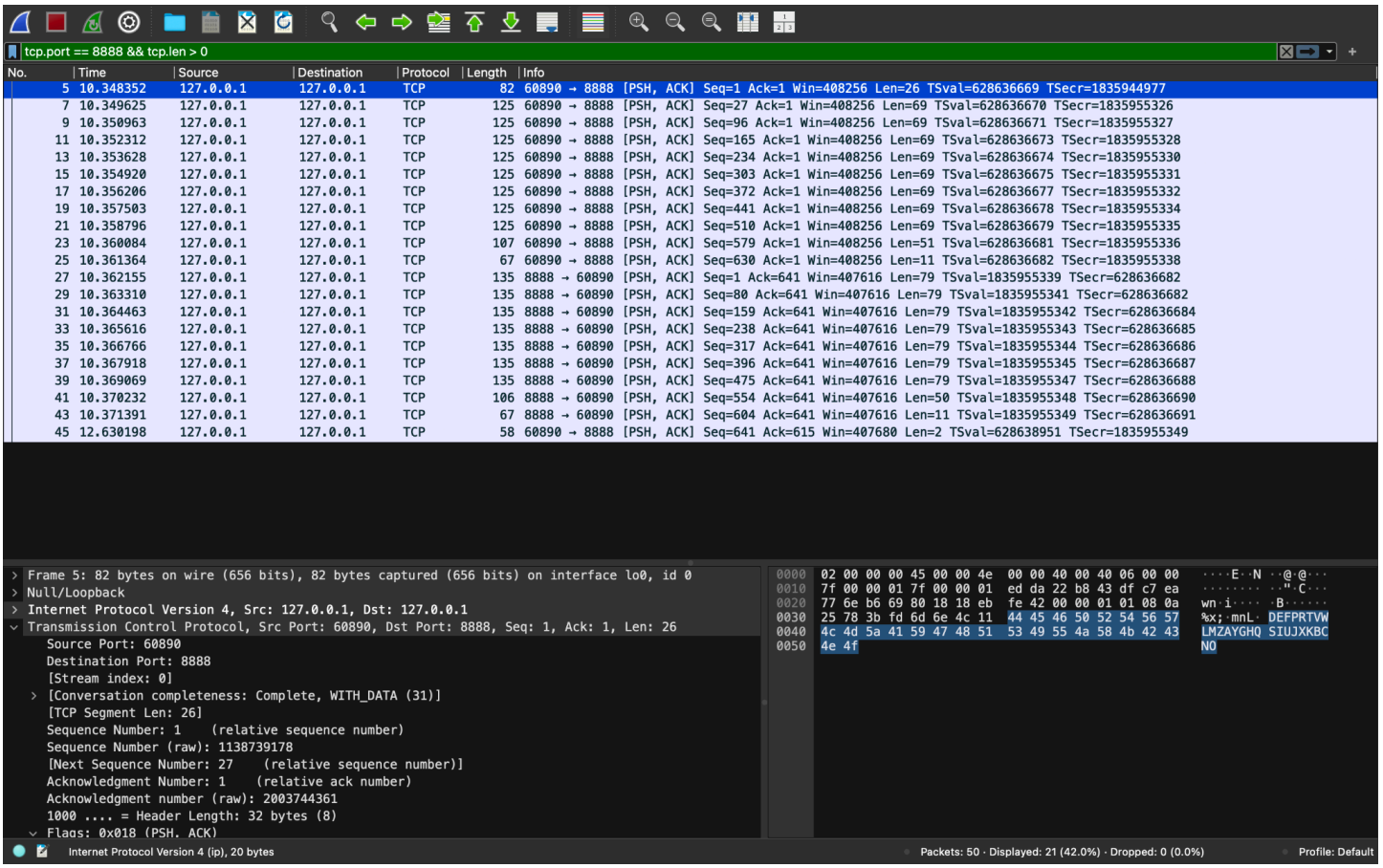
To count data packets, I used:

- Filter: `tcp.port == 8888 && tcp.len > 0`
- This shows only packets containing data

Note:

- Client sends file in chunks of max size = 70 bytes
- Server sends file in chunks of max size = 80 bytes

But for receiving a chuck, both use a buffer of size 100 bytes, because we were stated that chuck size will not be greater than 100 bytes. So, whatever be the size of the chunk used to send, the other server/client can receive it in a single packet.



In the above screenshot,

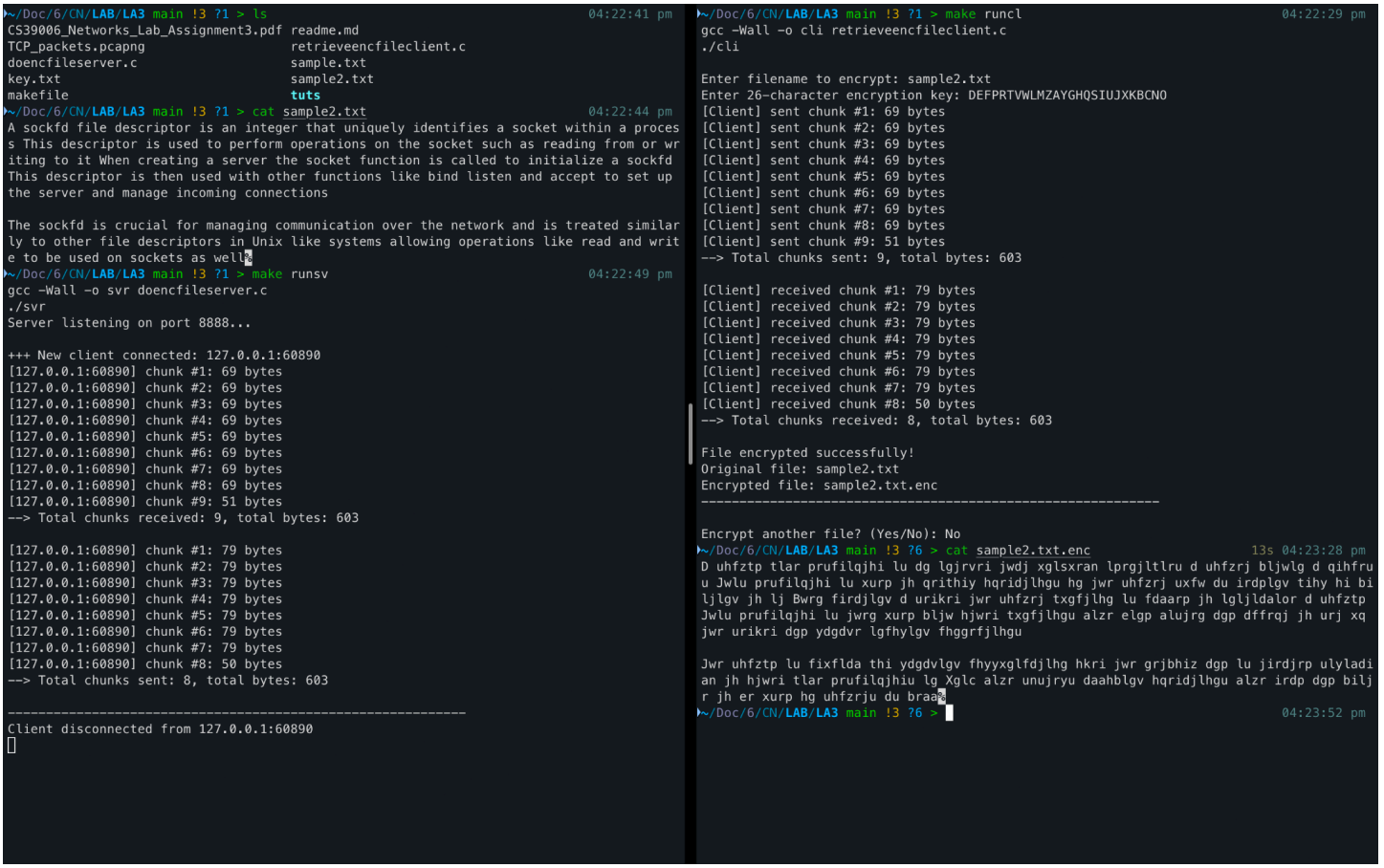
- No.5 is the KEY packet
- **From No.7 to No.23 are the packets from client to server of the original file**
- No.25 is the END_OF_FILE
- **From No.27 to No.41 are the packets from server to client of the encrypted file**
- No.43 is the END_OF_FILE
- No.45 is the “No” from the client to close the connection

Total number of packets exchanged for the transfer of a file(related to data only) between client and server:
9 (from client to server) + 8(from server to client) = 17 packets

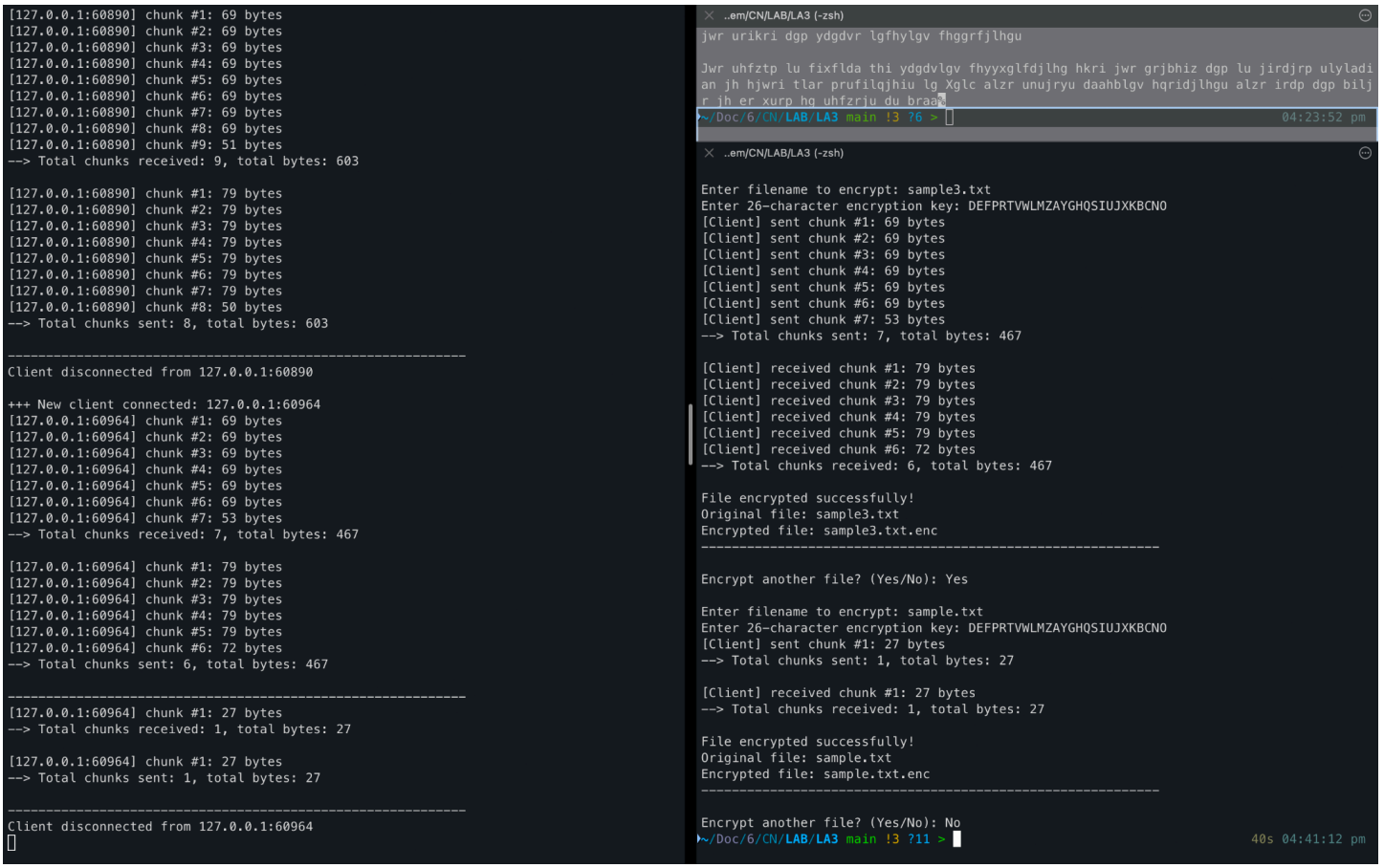
File size = sum of the data size of each packet sent by client = sum of data size of each packet sent by server
= 69 + 69 + 69 + 69 + 69 + 69 + 69 + 69 + 51 = 603 bytes
= 79 + 79 + 79 + 79 + 79 + 79 + 79 + 50 = 603 bytes

I have also implemented the function to print the chunk number and the size of each chunk:
So the overall statistics can also be seen from the terminal

Terminal 👩‍💻

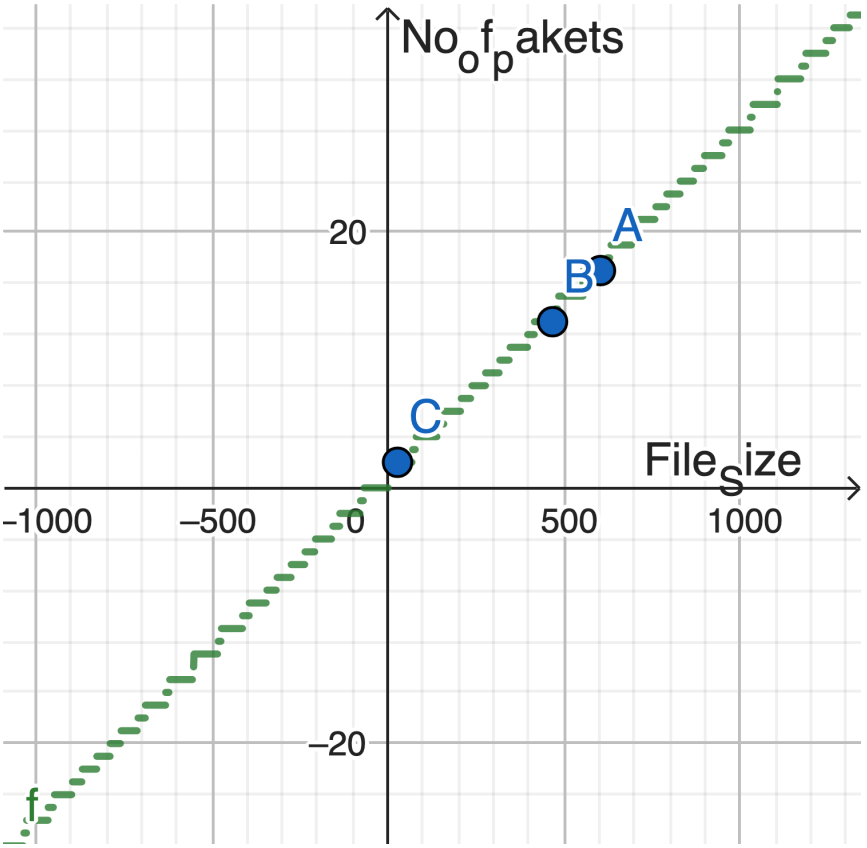


Plot a graph ‘file size vs the number of packets’ clearly based on your observation:

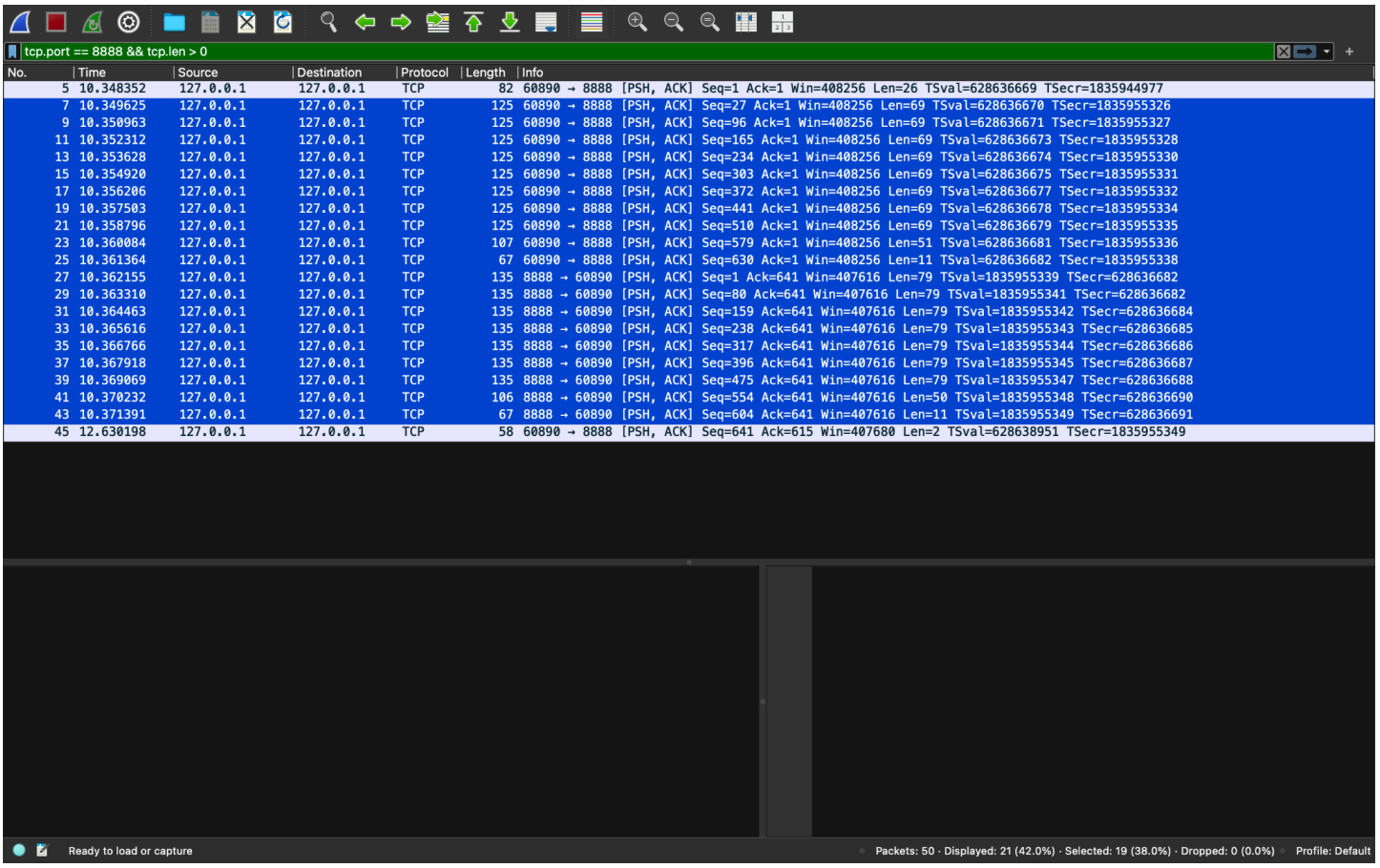


From the above three sample files:
1) 603 bytes -> 17 packets
2) 467 bytes -> 13 packets
3) 27 bytes -> 2 packets

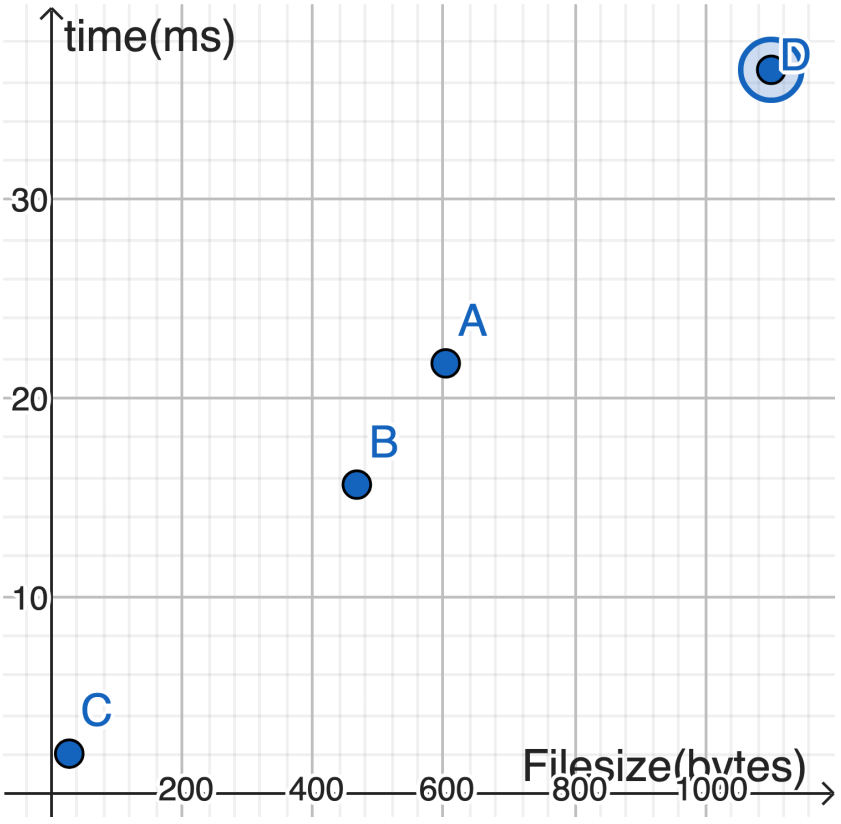
Mathematically: for a file of size B bytes
→ Client will send *ceil(B/69)* packets
→ Server will send *ceil(B/79)* packets



5. Measure the total time taken for the file transfer , its encryption and send it back from server to the client. Plot a graph ‘file size vs time’ clearly based on your observation and also attach the necessary screenshots.



- Total Time taken for the file transfer of size 603 bytes = 10.371391- 10.349626 = 0.0217652 seconds
- More observations
  - 467 bytes → 10.577517 - 10.561878 = 0.015639 seconds
  - 27 bytes → 0.003645 - 0.001589 = 0.002056 seconds
  - 1101 bytes → 0.037686 - 0.001100 = 0.036586 seconds



6. Calculate the average size packet exchanged during the data communication? Take reference of the plotted graph in the above question.

==========

Packet Lengths:

Topic / Item	Count	Average	Min Val	Max Val	Rate (ms)	Percent	Burst Rate	Burst Start
Packet Lengths	50	81.60	56	135	0.0040	100%	0.4000	10.348
0-19	0	-	-	-	0.0000	0.00%	-	-
20-39	0	-	-	-	0.0000	0.00%	-	-
40-79	32	57.50	56	68	0.0025	64.00%	0.2200	10.348
80-159	18	124.44	82	135	0.0014	36.00%	0.1800	10.348
160-319	0	-	-	-	0.0000	0.00%	-	-
320-639	0	-	-	-	0.0000	0.00%	-	-
640-1279	0	-	-	-	0.0000	0.00%	-	-
1280-2559	0	-	-	-	0.0000	0.00%	-	-
2560-5119	0	-	-	-	0.0000	0.00%	-	-
5120 and greater	0	-	-	-	0.0000	0.00%	-	-

Using the Statistics button in wireshark, the **average packet length** is **81.60 bytes** for the file size 603 bytes.