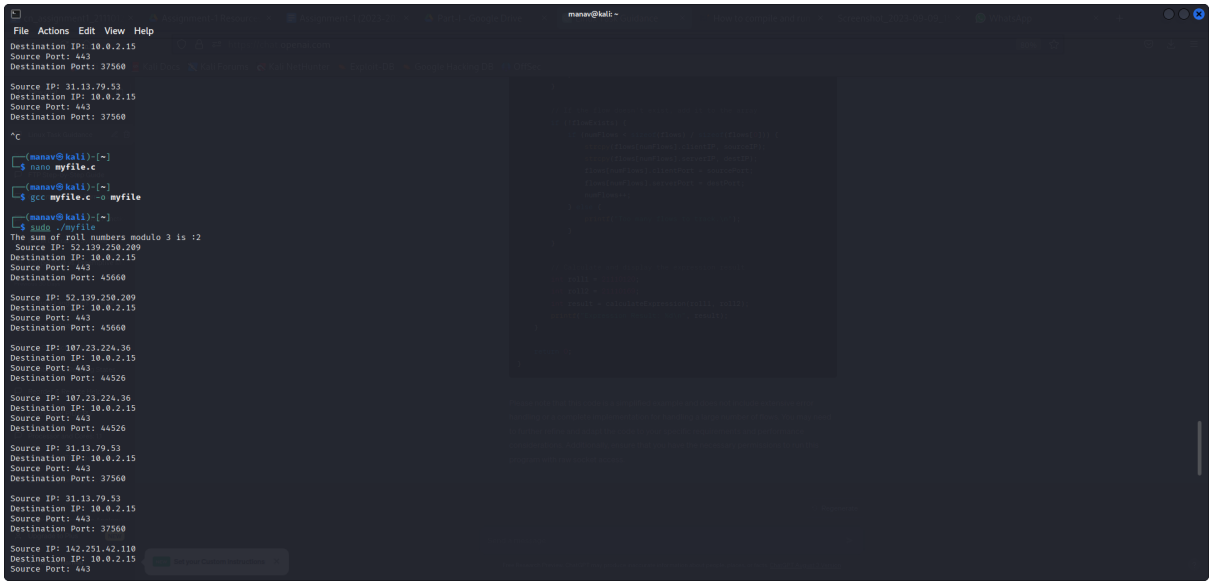
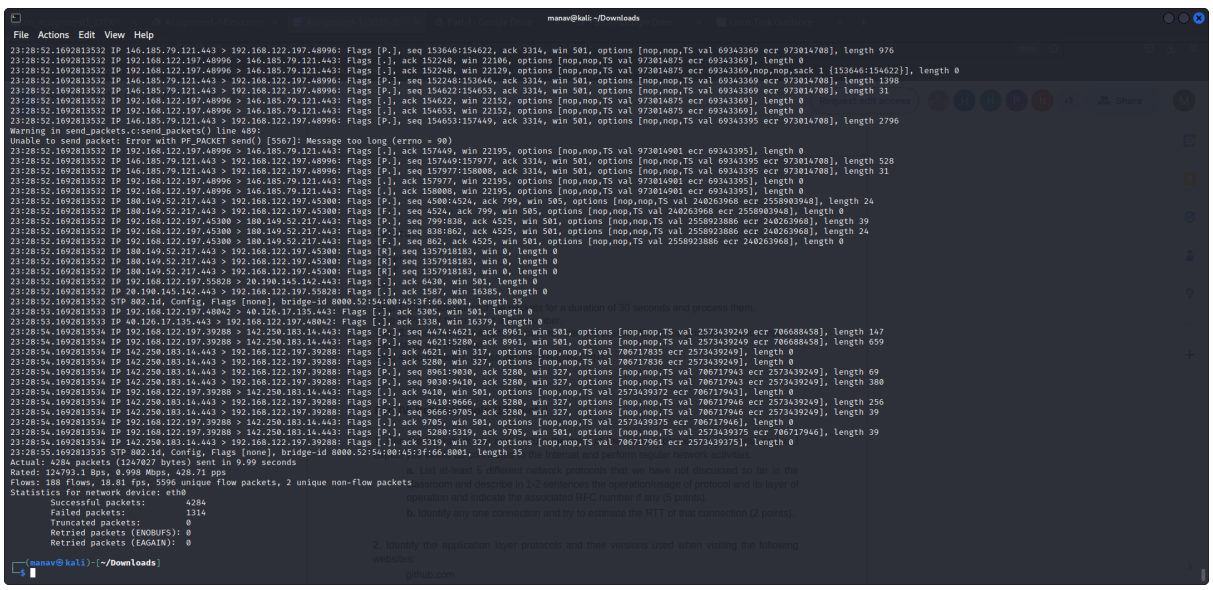
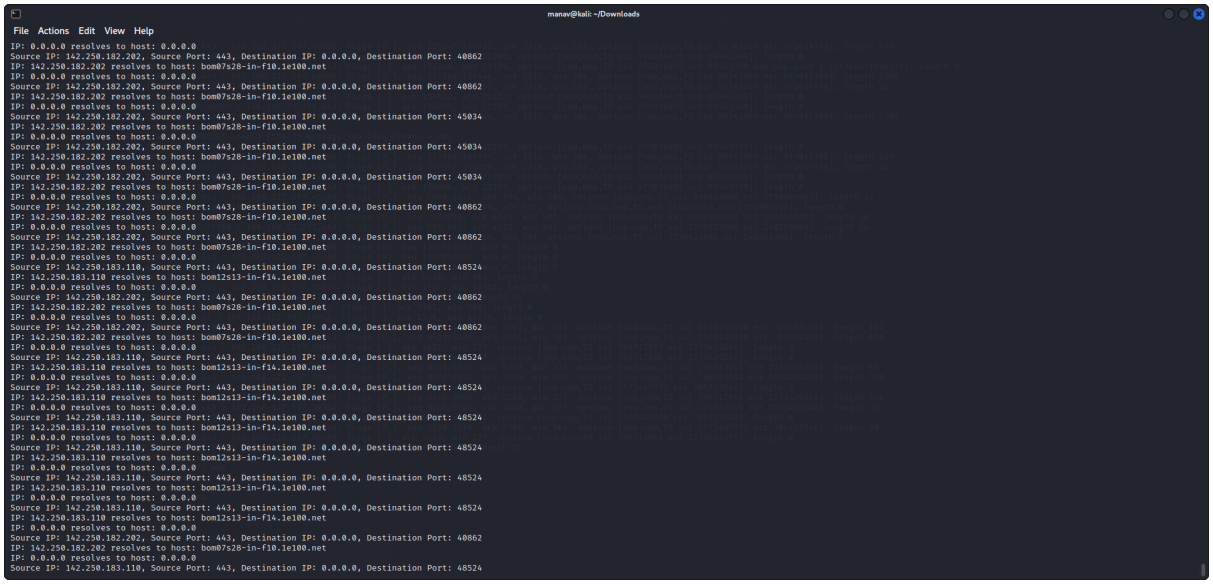
Q1.

Output of question 1:



After running the code (as our proc file is 2.cap), the result obtained is as follows:
tcpreplay -i eth0 --mbps 1 -v 2.pcap





```
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.182.202, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 40862
IP: 142.250.182.202 resolves to host: bom07s28-in-f10.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.182.202, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 40862
IP: 142.250.182.202 resolves to host: bom07s28-in-f10.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.182.202, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 45034
IP: 142.250.182.202 resolves to host: bom07s28-in-f10.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.182.202, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 45034
IP: 142.250.182.202 resolves to host: bom07s28-in-f10.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.182.202, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 45034
IP: 142.250.182.202 resolves to host: bom07s28-in-f10.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.182.202, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 40862
IP: 142.250.182.202 resolves to host: bom07s28-in-f10.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.182.202, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 40862
IP: 142.250.182.202 resolves to host: bom07s28-in-f10.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.183.110, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 48524
IP: 142.250.183.110 resolves to host: bom12s13-in-f14.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.182.202, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 40862
IP: 142.250.182.202 resolves to host: bom07s28-in-f10.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.182.202, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 40862
IP: 142.250.182.202 resolves to host: bom07s28-in-f10.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.183.110, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 48524
IP: 142.250.183.110 resolves to host: bom12s13-in-f14.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.183.110, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 48524
IP: 142.250.183.110 resolves to host: bom12s13-in-f14.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.183.110, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 48524
IP: 142.250.183.110 resolves to host: bom12s13-in-f14.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.183.110, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 48524
IP: 142.250.183.110 resolves to host: bom12s13-in-f14.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.183.110, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 48524
IP: 142.250.183.110 resolves to host: bom12s13-in-f14.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.183.110, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 48524
IP: 142.250.183.110 resolves to host: bom12s13-in-f14.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.182.202, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 40862
IP: 142.250.182.202 resolves to host: bom07s28-in-f10.1e100.net
IP: 0.0.0.0 resolves to host: 0.0.0.0
Source IP: 142.250.183.110, Source Port: 443, Destination IP: 0.0.0.0, Destination Port: 48524
```

Explanation about the implementation

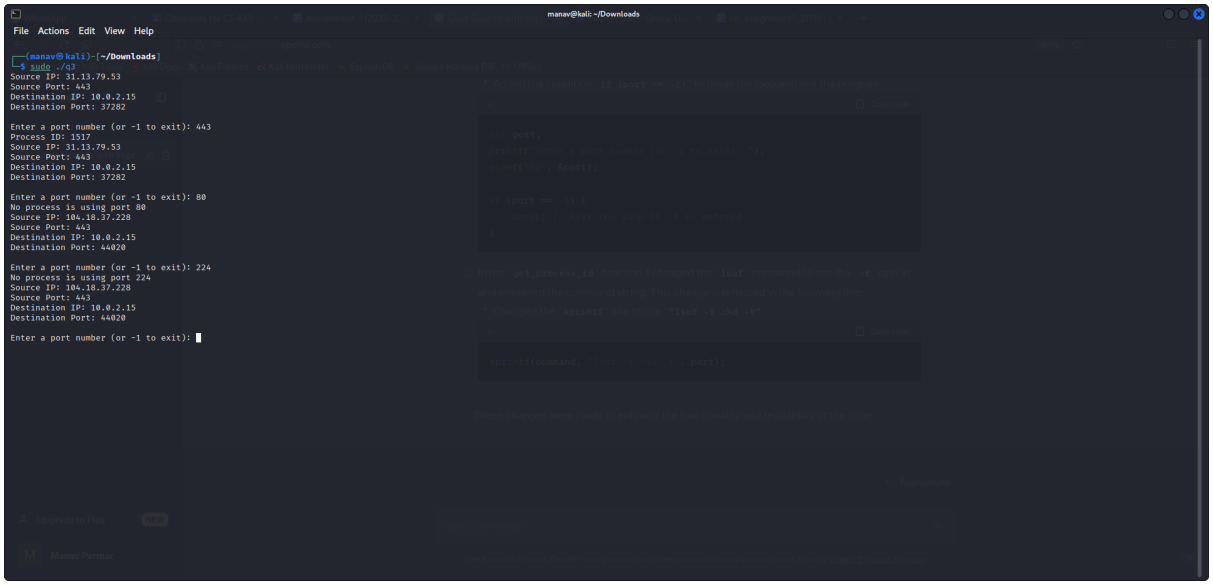
1a.

1. The program defines two functions:
    - process_packet: This function takes two arguments pointer to the package and the data size. It extracts the IP header and the TCP header from the data packet. It extracts the source and destination IP addresses using inet_ntop and stores them in sourceIP and destIP, respectively. It extracts the source and destination port numbers from the TCP header and stores them in sourcePort and destPort.Finally, it prints the source IP, destination IP, source port, and destination port to the console.

2. In the main function:
    - It creates a raw socket to capture all TCP packets. Then, it enters an infinite loop to capture the process packets. It receives the package in the packet buffer. It captures the package and sends it to the process_packet. This loop continues indefinitely, capturing and processing the incoming packets.
3. When the user exits, the program closes the socket and terminates.

Q3.

Here is the output after running:



References:

Portions of code are written with the help of OpenAI's ChatGPT 3.5