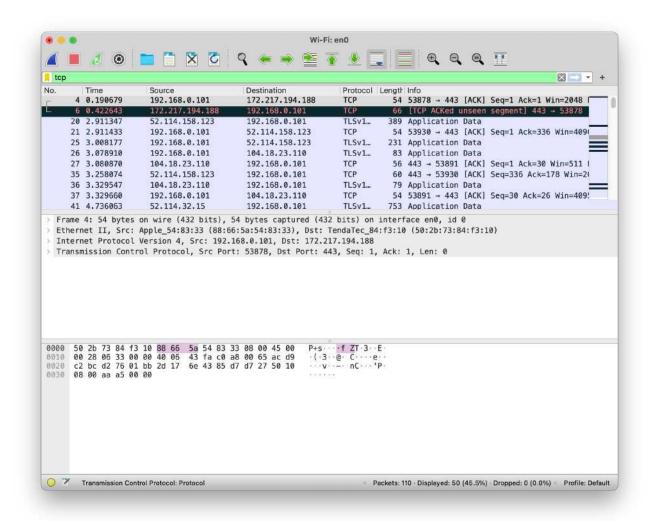


# **ISAA LAB ASSIGNMENT - 5**

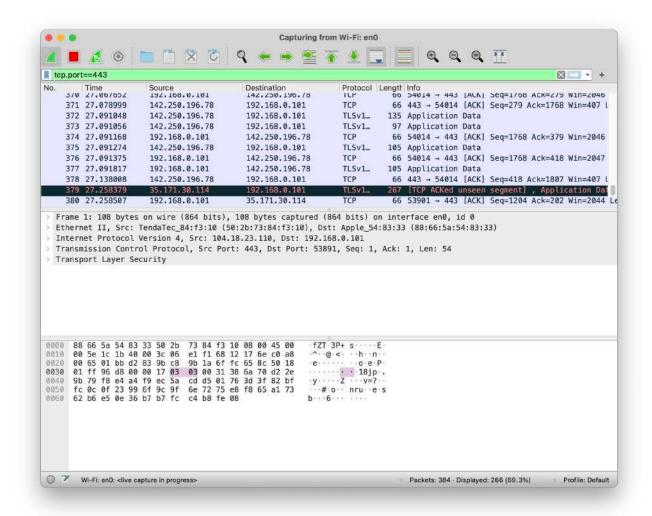
NAME	APOORVA REDDY BAGEPALLI
REGISTRATION NUMBER	19BCE2196

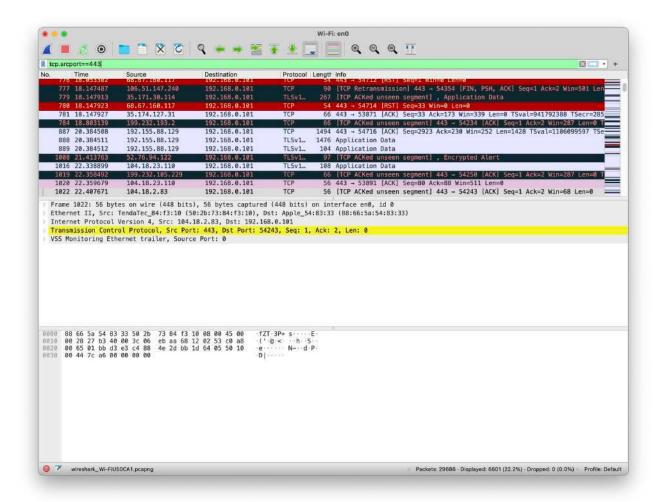
## **Network Packets Sniffing using WireShark**

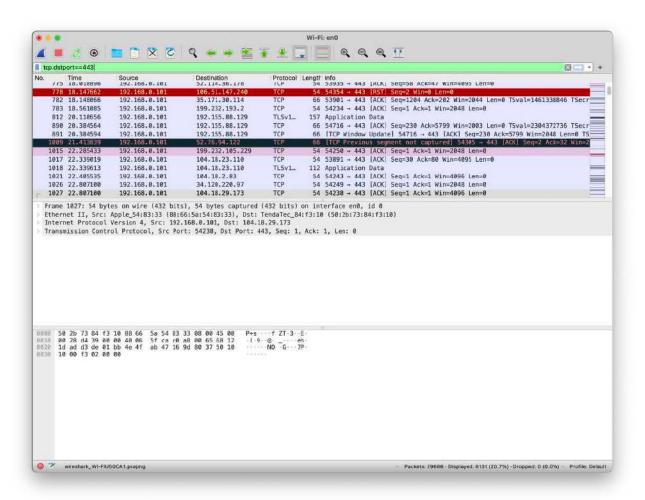
- 1. Filtering the packets by specifying a protocol
  - TLSv1: This protocol depends on TCP.



- 2. Filter the packets based on the port
  - tcp.port == 443
  - Tcp.dstport==443
  - tcp.srcport==443



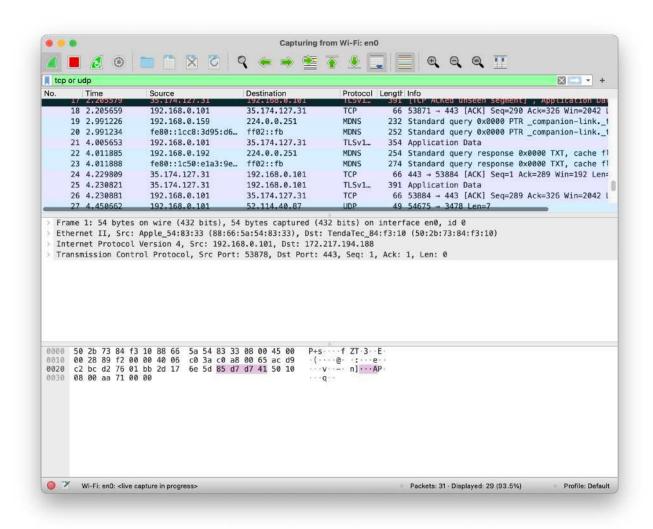


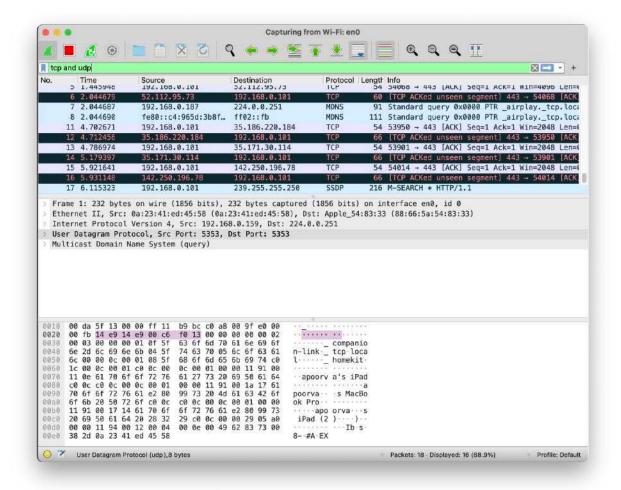


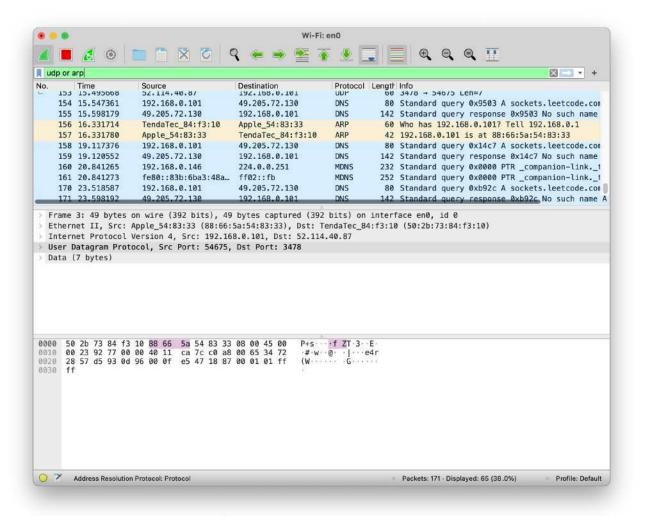
#### 3. Filter results based on or/and

Or : Independent And : Dependent

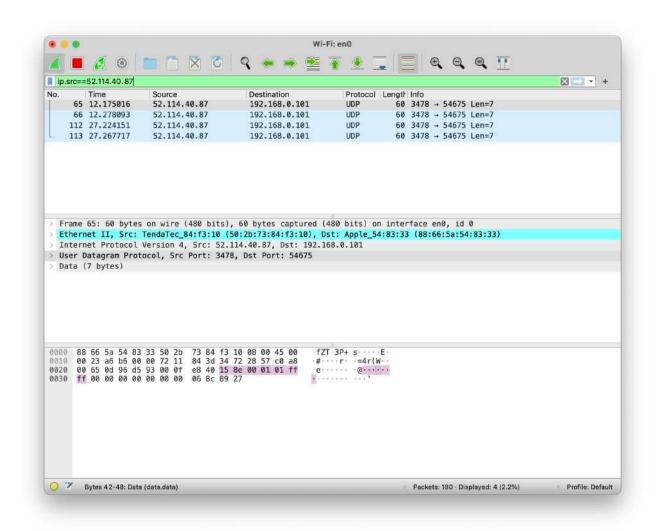
• Tep or udp / tep and udp / udp or arp

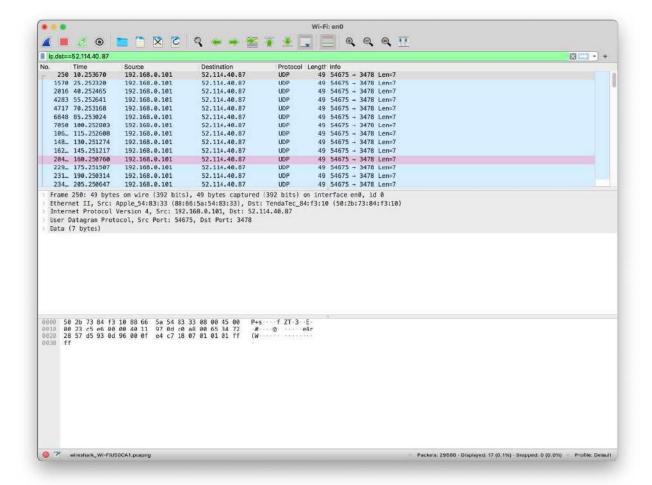




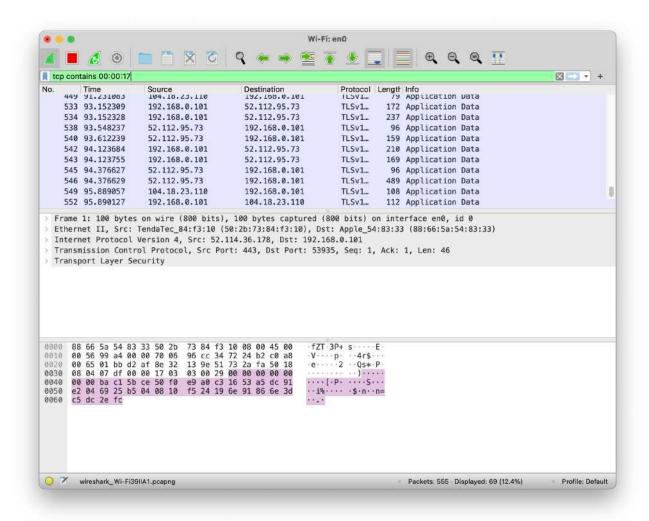


- 4. Filter results based on Ip addresses
  - ip.src==52.114.40.87
  - Ip.dst==52.114.40.87

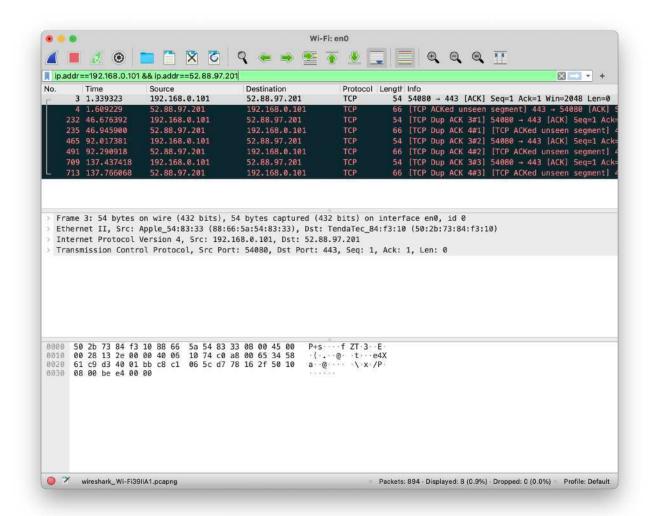


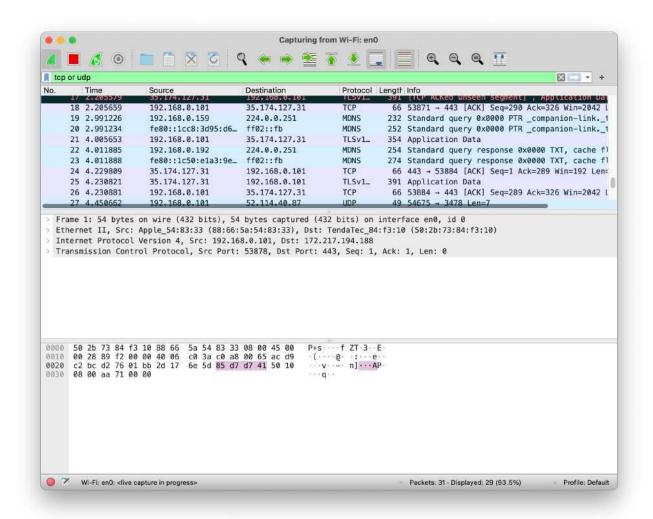


- 5. Filter results based on the byte sequence
  - Tcp contains 00:00:17

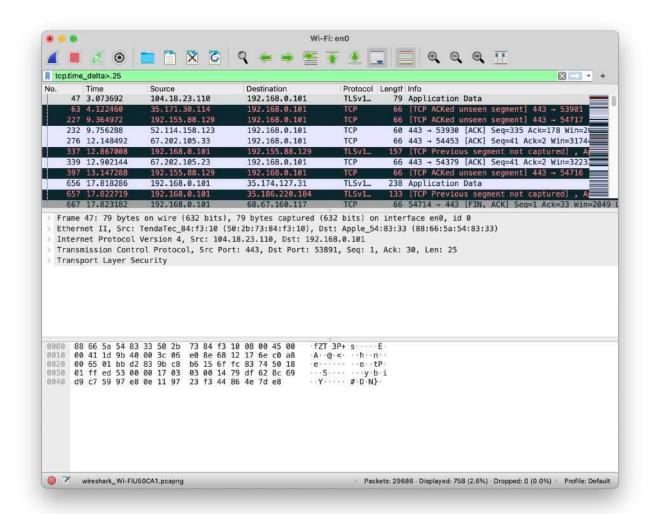


- 6. Filter result based on two IP addresses
  - ip.addr==192.168.0.101 && ip.addr==52.88.97.201
  - (ip.addr==192.168.0.101 or ip.addr==49.205.72.130) and(ip.addr==142.250.195.100 or ip.addr==142.250.77.110)

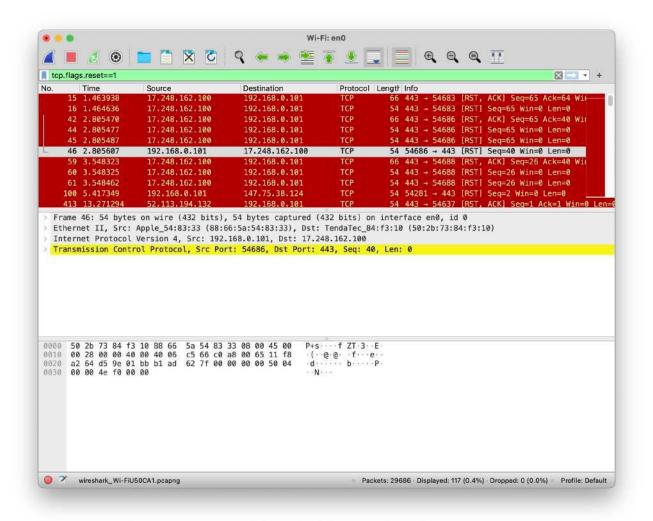


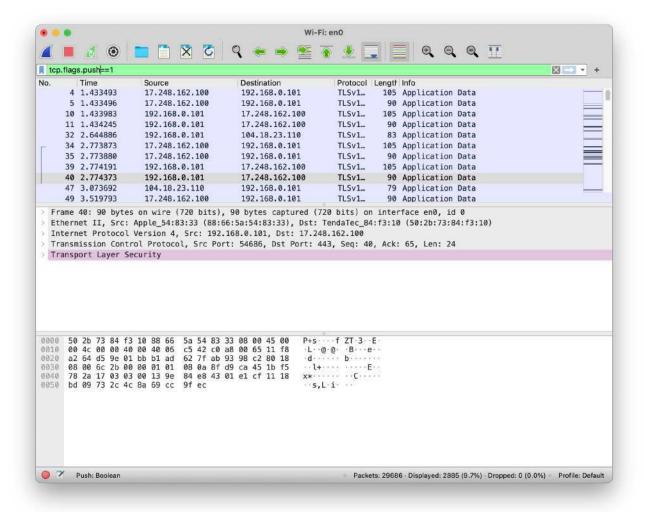


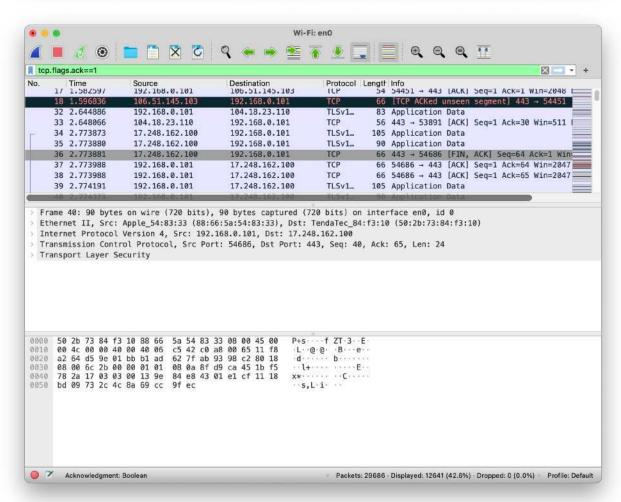
- 7. Filter the results based on timestamp
  - tcp.time\_delta>.25

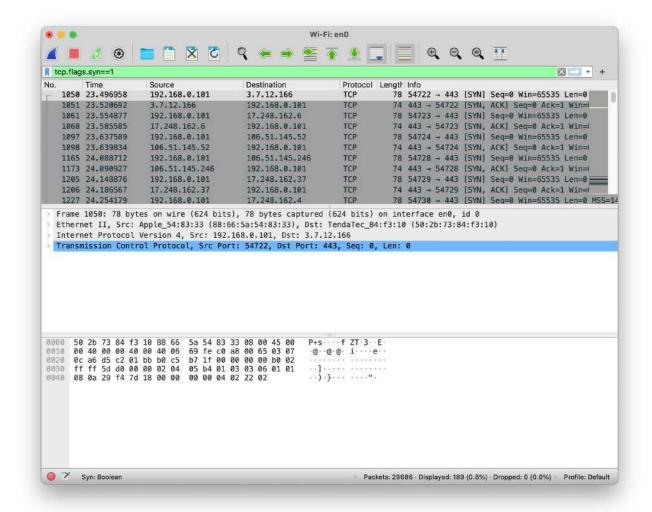


- 8. Filter the results based on the flags
  - tcp.flags.reset==1
  - tcp.flags.push==1
  - tcp.flags.ack==1
  - tcp.flags.syn==1



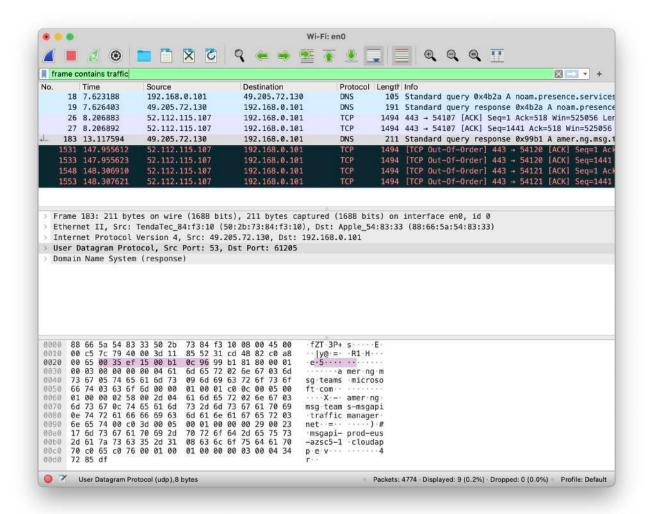






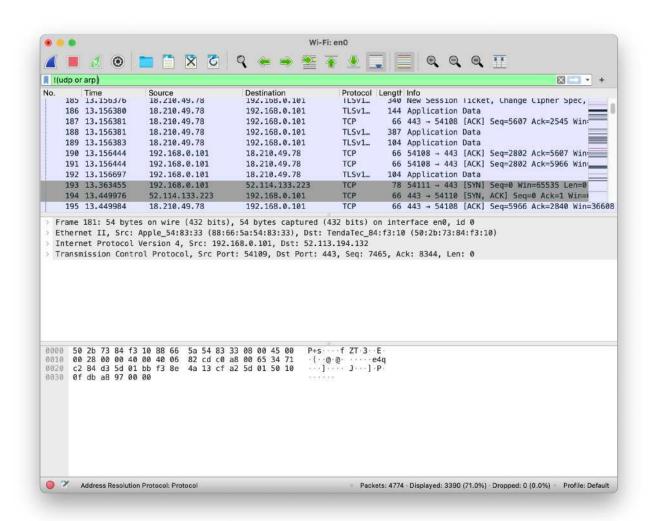
#### 9. Traffic

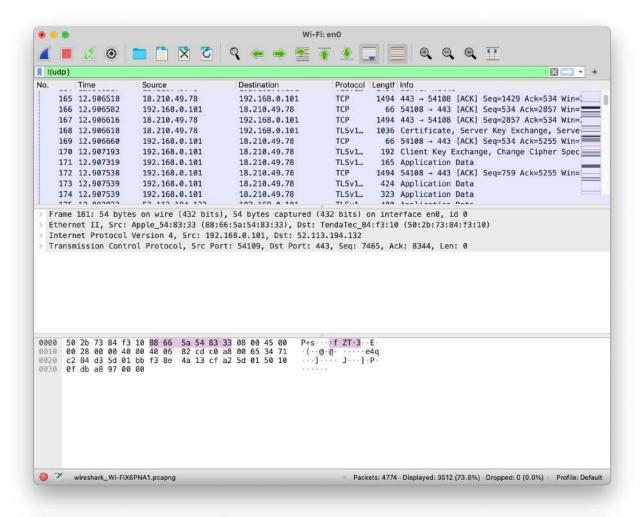
• frame contains traffic

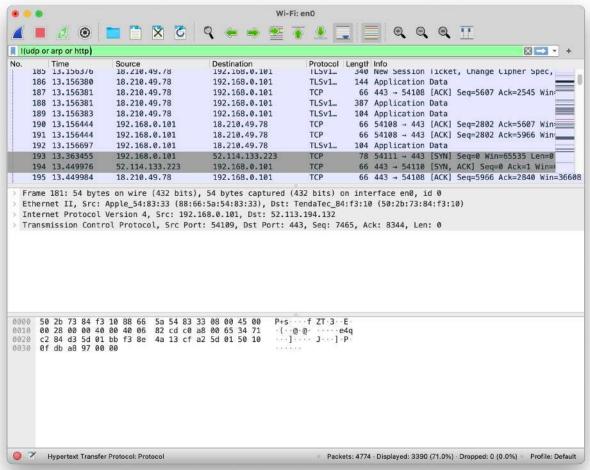


#### 10. Filter based on NOT

• !(udp or arp or http) / !(udp) / !(udp or arp)

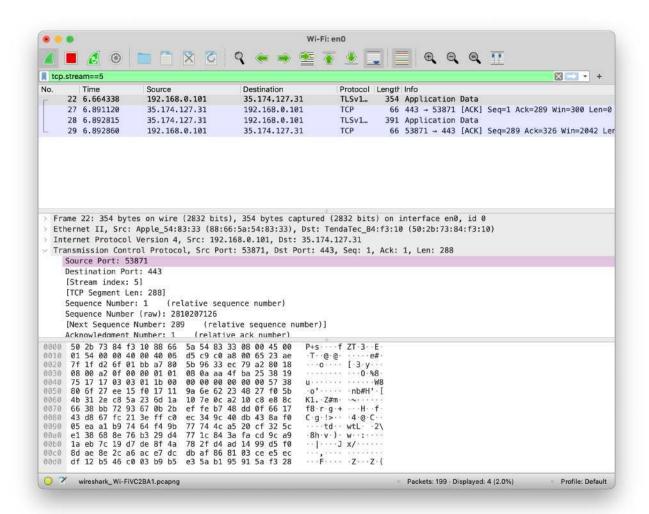




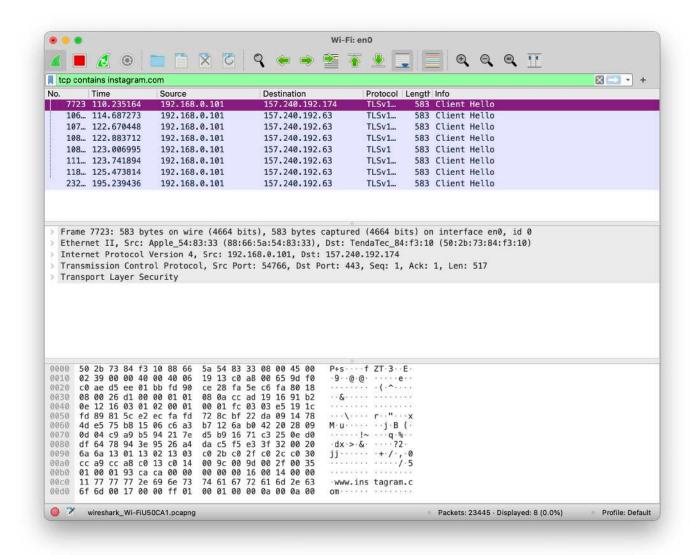


### 11. Filter based on the stream number

• tcp.stream==5



- 12. Filter based on the website
  - tcp contains instagram.com



13. Filter the packets based on retransmission and duplicate acks

• tcp analysis.flags

