TCP (TRANSMISSION CONTROL PROTOCOL):

1. It is highly reliable and connection oriented communication protocol (since it does handshaking and reliability control)
2. It ensures reliability in terms of congestion control ,error control ,flow control and connection control techniques.
3. It is one of the transport layer protocols which works on port addressing (port is the logical endpoint of communication and has three classes (well known, reserved and dynamic ports) with 16 bits for 65536 unique port numbers.) , segmentation and reassembly of packets (PDU) , reliability and congestion control techniques.
4. TCP header has mandatory fields for 20 bytes and optional content for 40 bytes. Some of the important fields in TCP segment are source and destination port , sequence number, acknowledgement number, header length, reserved, flags, window size, checksum, urgent pointer, options, padding, data.
5. Sequence number is used for re-ordering the packets at receiver side due to packet switching network (initial SN is chosen during tcp handshake)
6. Acknowledgment confirms the successful reception of packets upto the sequence number sent.
7. Flags are some special options like ECN (Explicit Congestion Notification) , CWR (Congestion Window Reduced) , Urgent pointer valid, Ack received, push to application without waiting for buffer to complete, reset , syn, fin
8. Window size determines the maximum amount of data that can be sent before expecting first acknowledgement.

TCP 3 Way handshaking :

1. Client sends SYN enabled packet to server port with ISN = x and ack = 0
2. Server responds with SYN-ACK with seq=y and ack=x+1 (acknowledging client’s syn)
3. Client acknowledges by ACK to server to synchronize by sending packet with seq=x+1 and ack=y+1

TCP 4 Way handshaking :

1. Client initiates termination by sending FIN and ACK with seq=x and ack=y
2. Server responds by sending ACK with seq=y and ack = x+1
3. After completing, server sends FIN and ACK with seq=y and ack=x+1
4. Then, client responds by sending FIN with seq=x+1 and ack=y+1 (now, client waits for 2\* maximum segment lifetime before termination)

TCP Congestion Control:

1. Slow start stage – where, congestion window is doubled for each RTT.
2. Congestion Avoidance – where, congestion window is incremented by 1 after one packet loss. (linear growth) (slow start threshold point)
3. If severe congestion continues, then, in Slow recovery mode, it reduces SST to be half of congestion window and starts with 1 MSS as new Congestion window from slow start stage
4. In Fast recovery mode, SST = half of congestion window but congestion window = 3 + MSS and starts again from congestion avoidance stage.

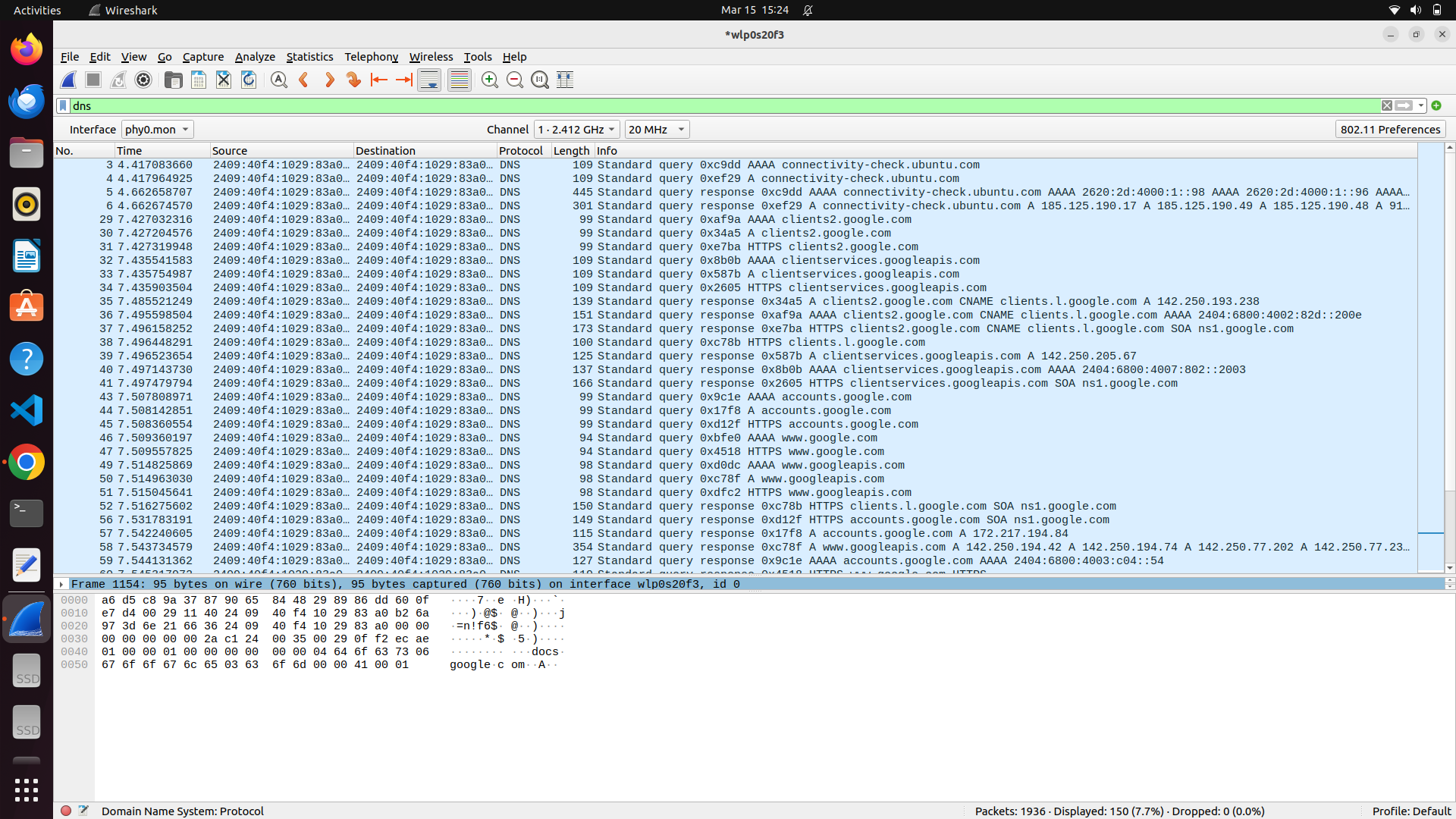
DNS (DOMAIN NAME SYSTEM):

1. The Domain Name System (DNS) is a hierarchical and decentralized system used to resolve human-readable domain names into IP addresses so that computers can communicate over networks like the internet.
2. Whenever, client wants to have communication using hostname, if first checks its local cache for mapping, if no mapping found, it proceeds next.
3. It requests recursive configured DNS resolver (ISP provided or Google DNS) for help. If it has records, it provides mapping. If not so, proceeds next.
4. DNS resolver queries DNS Root Server which could redirect to the top level domain server (.com .edu etc)
5. Top level name server returns authoritative domain name server (google.com ) so as to proceed next.
6. Authoritative name server will deliver actual IP address needed in reverse path.
7. Client caches this mapping for future use.
8. It uses UDP since it is lightweight, avoids handshaking, and faster.
9. DNS packet header contains transaction ID , flags (query/response , status code), options, question asked, answers and records from authoritative server etc.

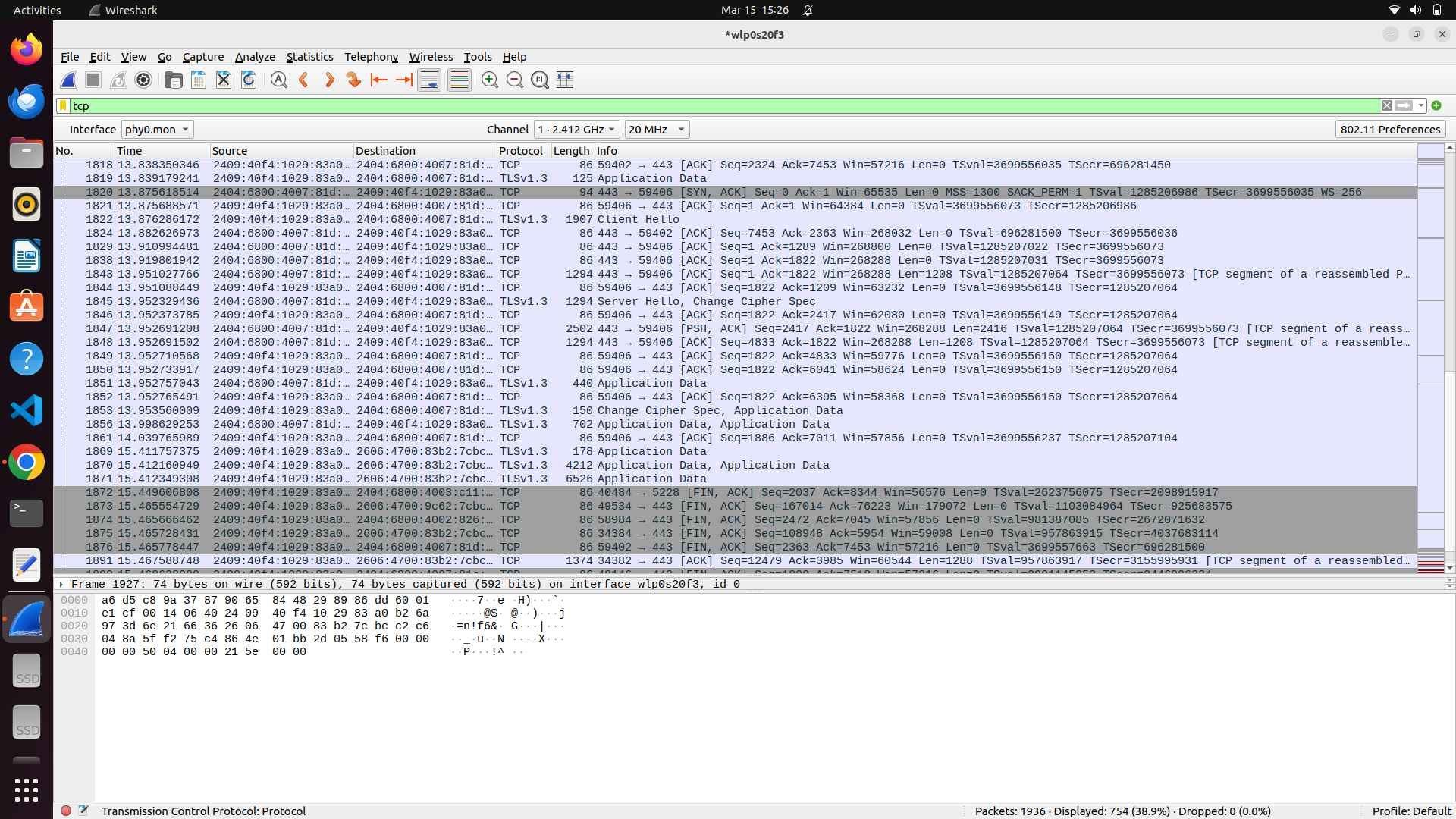
UDP (USER DATAGRAM PROTOCOL):

1. The User Datagram Protocol (UDP) is a connectionless, lightweight transport-layer protocol .
2. It is widely used for real-time applications, DNS, VoIP, streaming, and other latency-sensitive communication. Eg. DNS , DHCP, TFTP, VOIP etc.
3. It doesn’t provide reliability and congestion control as like TCP.
4. Offers simple multiplexing and demultiplexing scheme.
5. Contains only four fields as header -> source, dest port, length and checksum
6. Modern day networks uses QUIC (Quick UDP Internet Connection) which uses TLS based encryption with 0 or 1 RTT handshake and connection migration etc.

DNS PACKET CAPTURE IN WIRESHARK :



TCP PACKET CAPTURE IN WIRESHARK



UDP PACKET CAPTURE IN WIRESHARK

