Topic: OSI Model, TCP/IP Model & Wireshark Packet Analysis

1. OSI Layer Explanations

Layer 1 – Physical Layer

This layer is the quiet workhorse of the network world. It deals with the raw signals—electrical pulses, light flashes, or radio waves—that travel through cables and air. No understanding of data happens here; it's just bits marching in a steady rhythm. Ethernet cables, fiber optics, Wi-Fi radio signals, voltage levels, connectors, and physical topologies belong here. You can imagine it like the actual road on which vehicles travel: without the asphalt or the rails, nothing else can move.

Layer 2 – Data Link Layer

Here the network gains awareness of neighbors. The Data Link Layer shapes raw bits into frames, attaches MAC addresses, and checks for errors using CRC. Switches, NICs, VLANs, ARP, and protocols like Ethernet and PPP live here. It's the traffic manager inside a city block: it ensures local deliveries reach the right door, even if the rest of the world remains a mystery. Collisions are handled, and corrupted frames are quietly filtered out so higher layers don't panic

Layer 3 – Network Layer

This is the layer where a device learns how to travel beyond its local area. The Network Layer handles logical addressing and routing—IP addresses, subnets, routers, and routing protocols like OSPF, RIP, and BGP. Packets may hop through multiple networks, each router choosing the next turn based on routing tables. It's like the postal service's long-distance system: envelopes move from city to city through a maze of interconnected routes until they find the right region.

Layer 4 – Transport Layer

Here the network learns to care about reliability and order. This layer decides whether data should arrive carefully (TCP) or quickly (UDP). It forms segments, tracks port numbers, handles re-transmissions, flow control, and ensures that scattered pieces of data get reassembled like puzzle tiles. It resembles a courier service that either delivers fragile items with great caution or drops off parcels rapidly with no signatures—depending on what the sender wants.

Layer 5 – Session Layer

The Session Layer is like the host of a long conversation. It creates, manages, and gracefully ends communication sessions between devices. It keeps track of who is speaking, when, and ensures that if a connection drops, it can be resumed. Protocols such as RPC and NetBIOS sit here. Think of it as a moderator in a meeting room who ensures that the discussion keeps flowing without two people talking over each other.

Layer 6 – Presentation Layer

This layer acts as the translator and stylist of the network. It converts data into formats that applications can understand—handling encryption, compression, and encoding. SSL/TLS (for encryption), JPEG, MP3, GIF, and text encoding schemes like ASCII and UTF-8 show up here. Picture a multilingual editor who reformats and translates a document so the final reader receives it in the exact style and language they expect.

Layer 7 – Application Layer

At the top sits the layer users actually interact with—even if they don't realize it. It provides interfaces for email, browsing, file transfers, and other network services. Protocols like HTTP, HTTPS, FTP, SMTP, DNS, and DHCP operate here. It's like the front counter of a service desk: people walk up, make requests, and receive what they need, while the deeper layers do the hidden heavy lifting behind the wall.

2. OSI Mnemonic

"People Do Not Trust Senior People Always."

People --- Layer 1 (Physical)

Do ---Layer2 (Data Link)

Not ---Layer3 (Network)

Trust ---Layer4 (Transport)

Senior ---Layer5 (Session)

People ---Layer6 (Presentation)

Always ---Layer7 (Application)

3. OSI vs TCP/IP Model Comparison

The OSI model is like a detailed blueprint that explains *every step* of how data is supposed to travel across a network. It breaks everything into seven layers, giving each job its own space — from how bits move through cables to how apps like browsers and email work. The TCP/IP model, on the other hand, is more practical. It's the model the internet actually uses. Instead of seven layers, it keeps things simpler with four layers, combining some of the OSI functions because real-world networking doesn't always separate them.

In short, OSI is great for learning and understanding the theory, while TCP/IP is the actual engine running behind websites, apps, and the internet. OSI is the "textbook model," and TCP/IP is the "real-life model." Both describe the same journey of data — just in different levels of detail.

OSI Layer	TCP/IP Layer	Explanation
Application (L7)	**Application Layer**	TCP/IP combines user-facing services like web, email, and DNS in this layer.
Presentation (L6)	**Application Layer**	Data formatting, encryption, and compression are handled inside the TCP/IP Application layer.
Session (L5)	**Application Layer**	TCP/IP places session control (start/maintain/end communication) within application protocols.
Transport (L4)	**Transport Layer**	Ensures end-to-end delivery using TCP/UDP, handling reliability and port numbers.
Network (L3)	**Internet Layer**	Handles IP addressing, routing, and packet forwarding across networks.
Data Link (L2)	**Network Access Layer**	Responsible for frames, MAC addresses, and communication with the physical network.
Physical (L1)	**Network Access Layer**	Deals with cables, electrical/optical signals, connectors, and actual bit transmission.

4. Protocol Data Units (PDUs)

OSI Layer	PDU Name	Human-Friendly Explanation
Layer 4 – Transport	**Segments (TCP) / Datagrams (UDP)**	TCP breaks data into *segments* (reliable). UDP breaks data into *datagrams* (fast, no guarantee).
Layer 3 – Network	**Packets**	The layer that adds IP addresses and decides the route; it works with *packets*.
Layer 2 – Data Link	**Frames**	Adds MAC addresses and prepares data for local delivery; uses *frames*.
Layer 1 – Physical	**Bits**	Pure 0s and 1s turned into signals—just raw *bits*.

5. Addressing Concepts

1. MAC Address – used at Layer 2 (Data Link)

A MAC address is a physical, built-in address burned into your network card. Devices inside the same LAN use MAC addresses to deliver data to the right machine. Think of it like a permanent *house number* inside a neighborhood. Layer 2 uses MAC addresses to create and deliver **frames** within a local network.

2. IP Address – used at Layer 3 (Network)

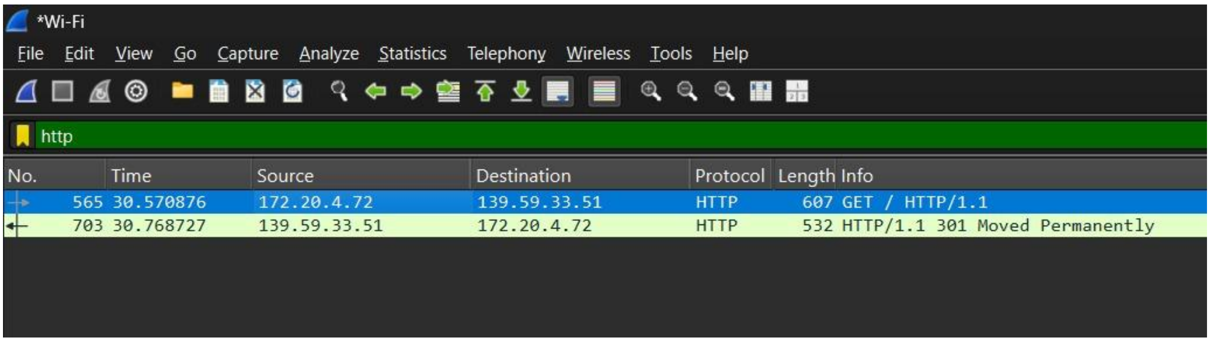
An IP address is a logical address that helps data travel between different networks. It's like your *city* + *street address* on the internet—routers use it to decide the best path for packets to reach another network. Layer 3 uses IP addresses to move **packets** across routers and different networks.

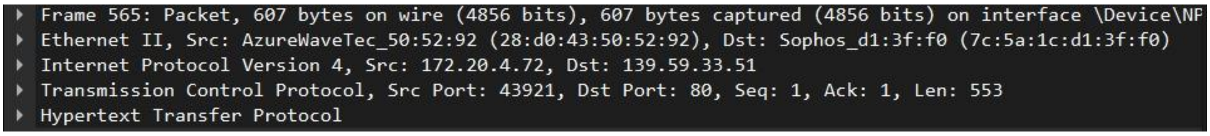
3. Port Number – used at Layer 4 (Transport)

A port number identifies a specific application or service running on a device. While the IP address finds the right device, the port number finds the right *app* inside that device. It's like reaching the correct room inside a building. Layer 4 uses port numbers along with TCP/UDP to deliver **segments/datagrams** to the right application.

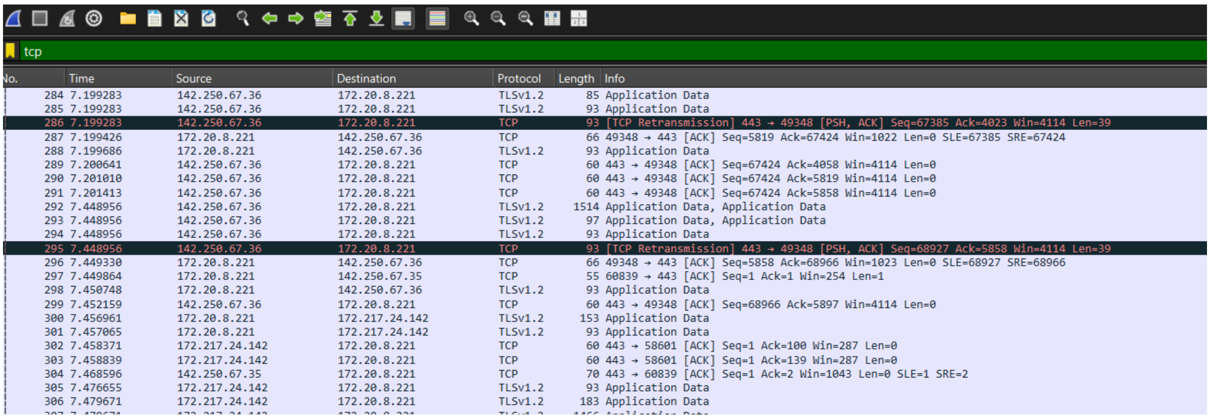
Part B – Wireshark Practical

1. HTTP Traffic (Application over TCP)





2. TCP Packets



```
▾ Transmission Control Protocol, Src Port: 64109, Dst Port: 443, Seq: 496, Ack: 1332, Len: 0
    Source Port: 64109
    Destination Port: 443
    [Stream index: 14]
    [Stream Packet Number: 15]
  ▸ [Conversation completeness: Incomplete (12)]
    [TCP Segment Len: 0]
    Sequence Number: 496    (relative sequence number)
    Sequence Number (raw): 2162120628
    [Next Sequence Number: 496    (relative sequence number)]
    Acknowledgment Number: 1332    (relative ack number)
    Acknowledgment number (raw): 1788069262
    0101 .... = Header Length: 20 bytes (5)
  ▸ Flags: 0x010 (ACK)
    Window: 253
    [Calculated window size: 253]
    [Window size scaling factor: -1 (unknown)]
    Checksum: 0xc51b [unverified]
    [Checksum Status: Unverified]
    Urgent Pointer: 0
  ▸ [Timestamps]
  ▸ [SEQ/ACK analysis]
Transmission Control Protocol: Protocol                                    Packets: 13287 ·
```

3. UDP Packets

udp

No.	Time	Source	Destination	Protocol	Length	Info
393	8.263961	172.20.8.221	142.251.43.234	QUIC	77	Protected Payload (KP0), DCID=fbc4dc8e3c75e92c
394	8.264132	172.20.8.221	142.251.43.234	QUIC	73	Protected Payload (KP0), DCID=fbc4dc8e3c75e92c
395	8.264202	172.20.8.221	142.251.43.234	QUIC	73	Protected Payload (KP0), DCID=fbc4dc8e3c75e92c
404	8.310223	142.251.43.234	172.20.8.221	QUIC	70	Protected Payload (KP0)
686	8.695618	172.20.8.221	8.8.8.8	DNS	70	Standard query 0x51ee HTTPS dns.google
690	8.696080	172.20.8.221	8.8.8.8	DNS	70	Standard query 0xdeaf A dns.google
692	8.697338	172.20.8.221	8.8.8.8	DNS	80	Standard query 0x09d5 A tunnel.googlezip.net
699	8.714501	8.8.8.8	172.20.8.221	DNS	150	Standard query response 0x51ee HTTPS dns.google SOA ns1.zdns.google
700	8.715098	8.8.8.8	172.20.8.221	DNS	106	Standard query response 0xdeaf A dns.google A 8.8.4.4 A 8.8.8.8
701	8.715785	8.8.8.8	172.20.8.221	DNS	100	Standard query response 0x09d5 A tunnel.googlezip.net A 216.239.34.157
988	9.686526	172.20.8.221	142.251.221.106	QUIC	1292	Initial, DCID=e64bd94fc4fbcef0, PKN: 1, CRYPTO, CRYPTO, PING, CRYPTO, PING, PING, CRYPTO, PING
989	9.686656	172.20.8.221	142.251.221.106	QUIC	1292	Initial, DCID=e64bd94fc4fbcef0, PKN: 2, PING, CRYPTO, PING, CRYPTO, PING, PING
992	9.688723	172.20.8.221	142.251.221.106	QUIC	124	0-RTT, DCID=e64bd94fc4fbcef0
1017	9.698721	172.20.8.221	104.18.11.207	QUIC	1292	Initial, DCID=2c6b748139c083b9, PKN: 1, PADDING, CRYPTO, PING, PING, PADDING, CRYPTO, PING, PADDING, CRYPTO, CRYPTO, PADDING, CRYPTO, PADDING, PING, P…
1018	9.698813	172.20.8.221	104.18.11.207	QUIC	1292	Initial, DCID=2c6b748139c083b9, PKN: 2, CRYPTO, CRYPTO, CRYPTO, PING, CRYPTO, PADDING, PING, CRYPTO, PADDING, PING, PADDING, CRYPTO, PADDING, CRYPTO, …
1028	9.704325	142.251.221.106	172.20.8.221	QUIC	86	Initial, SCID=e64bd94fc4fbcef0, PKN: 1, ACK
1029	9.704325	142.251.221.106	172.20.8.221	QUIC	1296	Initial, SCID=e64bd94fc4fbcef0, PKN: 2, ACK, PADDING
1030	9.705311	172.20.8.221	104.17.24.14	QUIC	1292	Initial, DCID=880db394512661cc, PKN: 1, PADDING, CRYPTO, CRYPTO, CRYPTO, CRYPTO, PADDING, CRYPTO, PING, CRYPTO, PING, PADDING, CRYPTO
1031	9.705429	172.20.8.221	104.17.24.14	QUIC	1292	Initial, DCID=880db394512661cc, PKN: 2, PADDING, PING, PING, PADDING, PING, PADDING, CRYPTO, PADDING, PING, PADDING, PING, PADDING, PING, PING, CRYPTO…
1046	9.722728	142.251.221.106	172.20.8.221	QUIC	1296	Initial, SCID=e64bd94fc4fbcef0, PKN: 3, CRYPTO, PADDING
1047	9.723089	142.251.221.106	172.20.8.221	QUIC	347	Protected Payload (KP0)
1048	9.723089	142.251.221.106	172.20.8.221	QUIC	995	Protected Payload (KP0)
1049	9.723089	142.251.221.106	172.20.8.221	QUIC	74	Protected Payload (KP0)
1050	9.723707	172.20.8.221	142.251.221.106	QUIC	120	Handshake, DCID=e64bd94fc4fbcef0
1051	9.723968	172.20.8.221	142.251.221.106	QUIC	73	Protected Payload (KP0), DCID=e64bd94fc4fbcef0

```
Internet Protocol Version 4, Src: 8.8.8.8, Dst: 172.20.8.221
User Datagram Protocol, Src Port: 53, Dst Port: 50239
    Source Port: 53
    Destination Port: 50239
    Length: 62
    Checksum: 0xa975 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 3]
    [Stream Packet Number: 2]
    [Timestamps]
    UDP payload (54 bytes)
Domain Name System (response)
```

4. ICMP Packets (Ping)

icmp

No.	Time	Source	Destination	Protocc	Lengt	Info
15	2.937459	172.20.8.221	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=8/2048, ttl=128 (reply in 16)
16	2.958069	8.8.8.8	172.20.8.221	ICMP	74	Echo (ping) reply id=0x0001, seq=8/2048, ttl=118 (request in 15)
24	3.961301	172.20.8.221	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=9/2304, ttl=128 (reply in 25)
25	3.981282	8.8.8.8	172.20.8.221	ICMP	74	Echo (ping) reply id=0x0001, seq=9/2304, ttl=118 (request in 24)
31	4.986585	172.20.8.221	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=10/2560, ttl=128 (reply in 32)
32	5.004777	8.8.8.8	172.20.8.221	ICMP	74	Echo (ping) reply id=0x0001, seq=10/2560, ttl=118 (request in 31)
34	6.008778	172.20.8.221	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=11/2816, ttl=128 (reply in 35)
35	6.029327	8.8.8.8	172.20.8.221	ICMP	74	Echo (ping) reply id=0x0001, seq=11/2816, ttl=118 (request in 34)

Request Message

```
Internet Control Message Protocol
    Type: 0 (Echo (ping) reply)
    Code: 0
    Checksum: 0x5553 [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence Number (BE): 8 (0x0008)
    Sequence Number (LE): 2048 (0x0800)
    [Request frame: 15]
    [Response time: 20.610 ms]
    Data (32 bytes)
```

Reply Message

```
Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x4d52 [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence Number (BE): 9 (0x0009)
    Sequence Number (LE): 2304 (0x0900)
    [Response frame: 25]
  Data (32 bytes)
```

5. ARP Frames

arp

No.	Time	Source	Destination	Protocc	Lengt	Info
17…	10.849242	Intel_2f:54:e5	Broadcast	ARP	60	ARP Announcement for 172.20.6.244
17…	12.584865	ChongqingFug_93…	Broadcast	ARP	60	ARP Announcement for 172.20.3.30
17…	12.789591	Intel_2f:54:e5	Broadcast	ARP	60	ARP Announcement for 172.20.6.244
17…	12.996033	fe:36:3b:7f:ae:…	Broadcast	ARP	60	Gratuitous ARP for 172.18.3.29 (Reply)

Reply Message

```
Address Resolution Protocol (reply/gratuitous ARP)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    [Is gratuitous: True]
    Sender MAC address: fe:36:3b:7f:ae:81 (fe:36:3b:7f:ae:81)
    Sender IP address: 172.18.3.29
    Target MAC address: Broadcast (ff:ff:ff:ff:ff:ff)
    Target IP address: 172.18.3.29
```

Request Message

```
▸ Ethernet II, Src: Intel_2f:54:e5 (44:a3:bb:2f:54:e5), Dst: Broadcast
▾ Address Resolution Protocol (ARP Announcement)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    [Is gratuitous: True]
    [Is announcement: True]
    Sender MAC address: Intel_2f:54:e5 (44:a3:bb:2f:54:e5)
    Sender IP address: 172.20.6.244
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 172.20.6.244

Address Resolution Protocol (arp), 28 bytes
```


Submitted By :

Daniel George V M

24UBC125

II BCA A