Prepare R&D Document on working of all the layers in OSI Model

The Open Systems Interconnection (OSI) model describes seven layers that computer systems use to communicate over a network. The OSI model is divided into seven distinct layers, each with specific responsibilities, ranging from physical hardware connections to high-level application interactions.

Each layer of the OSI model interacts with the layer directly above and below it, encapsulating and transmitting data in a structured manner. This approach helps network professionals troubleshoot issues, as problems can be isolated to a specific layer. The OSI model serves as a universal language for networking, providing a common ground for different systems to communicate effectively.

The OSI model was the first standard model for network communications, adopted by all major computer and telecommunication companies in the early 1980s. It was introduced in 1983 by representatives of the major computer and telecom companies, and was adopted by ISO as an international standard in 1984.

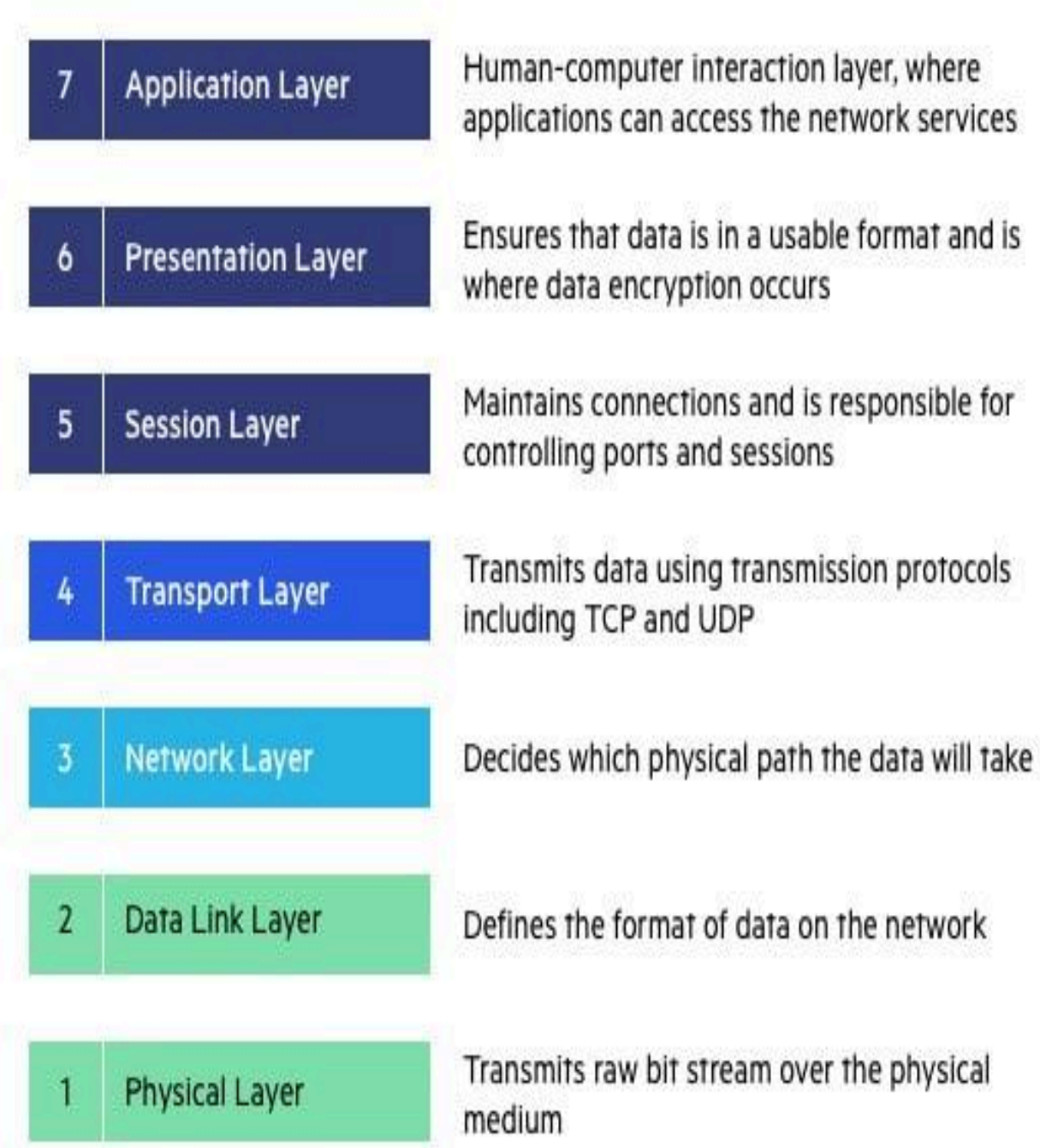
The modern Internet is not based on OSI, but on the simpler TCP/IP model. However, the OSI 7-layer model is still widely used, as it helps visualize and communicate how networks operate.

Why Is the OSI Model Important?

The OSI model provides several advantages for organizations managing networks and communications:

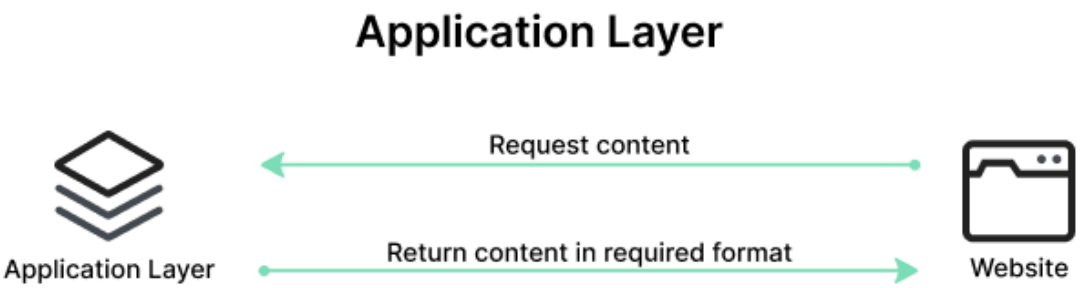
- Shared understanding of complex systems: OSI offers a universal language for networking, enabling different network devices and software to communicate. By dividing communication into seven distinct layers, it allows network professionals to isolate and troubleshoot problems effectively.
- Faster research and development: Developers can focus on improving specific layers without affecting others, leading to more rapid innovations. This modular approach enables specialization and enables different teams to work on various aspects of network communication simultaneously.
- Flexible standardization: The model's layered approach allows for the integration of new technologies at any layer without disrupting the overall network structure. This ensures compatibility across different devices and protocols, ensuring long-term viability and scalability of network infrastructure.

OSI Model : The OSI 7 Layers



We'll describe OSI layers "top down" from the application layer that directly serves the end user, down to the physical layer.

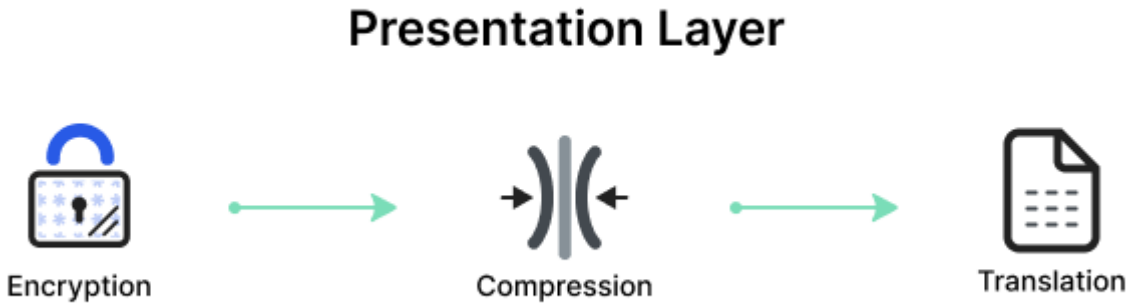
7. Application Layer



The Application Layer serves as the interface between the end-user applications and the underlying network services. This layer provides protocols and services that are directly utilized by end-user applications to communicate across the network. Key functionalities of the Application Layer include resource sharing, remote file access, and network management.

Examples of protocols operating at the Application Layer include Hypertext Transfer Protocol (HTTP) for web browsing, File Transfer Protocol (FTP) for file transfers, Simple Mail Transfer Protocol (SMTP) for email services, and Domain Name System (DNS) for resolving domain names to IP addresses. These protocols ensure that user applications can effectively communicate with each other and with servers over a network.

6. Presentation Layer



The Presentation Layer, also known as the syntax layer, is responsible for translating data between the application layer and the network format. It ensures that data sent from the application layer of one system is readable by the application layer of another system. This layer handles data formatting, encryption, and compression, facilitating interoperability between different systems.

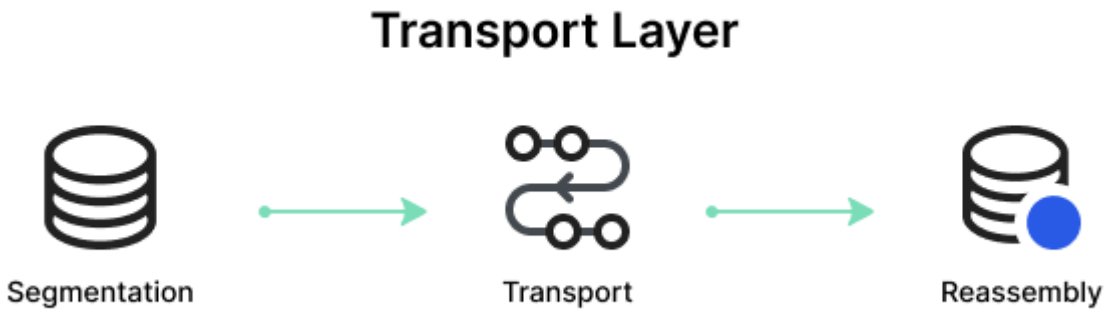
One of the key roles of the Presentation Layer is data translation and code conversion. It transforms data into a format that the application layer can understand. For example, it may convert data from ASCII to EBCDIC. It also includes encryption protocols to ensure data security during transmission and compression protocols to reduce the amount of data for efficient transmission.

5. Session Layer

The Session Layer manages and controls the connections between computers. It establishes, maintains, and terminates connections, ensuring that data exchanges occur efficiently and in an organized manner. The layer is responsible for session checkpointing and recovery, which allows sessions to resume after interruptions.

Protocols operating at the Session Layer include Remote Procedure Call (RPC), which enables a program to execute a procedure on a remote host as if it were local, and the session establishment phase in protocols like NetBIOS and SQL. These services enable reliable communication, especially in complex network environments.

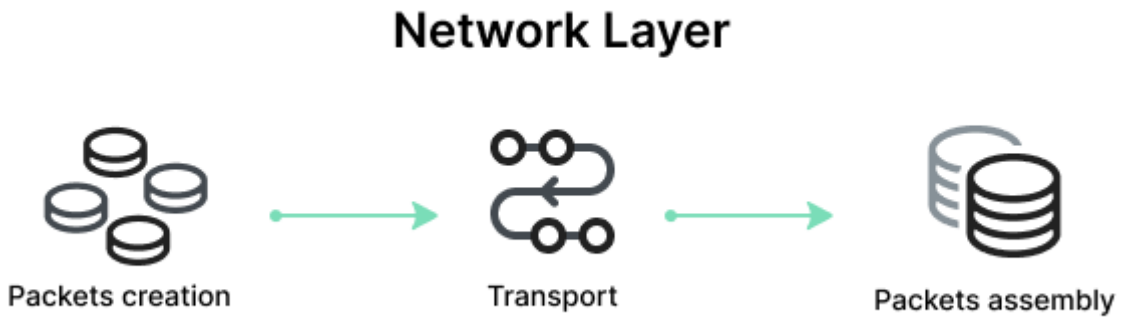
4. Transport Layer



The Transport Layer provides end-to-end communication services for applications. It ensures complete data transfer, error recovery, and flow control between hosts. This layer segments and reassembles data for efficient transmission and provides reliability with error detection and correction mechanisms.

Protocols at this layer include Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). TCP is connection-oriented and ensures reliable data transfer with error checking and flow control, making it suitable for applications like web browsing and email. UDP is connectionless, offering faster, though less reliable, transmission, suitable for applications like video streaming and online gaming.

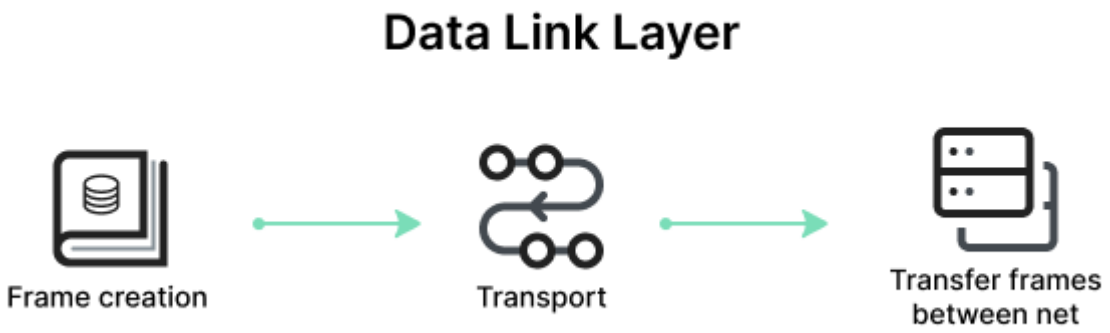
3. Network Layer



The Network Layer is responsible for data routing, forwarding, and addressing. It determines the best physical path for data to reach its destination based on network conditions, the priority of service, and other factors. This layer manages logical addressing through IP addresses and handles packet forwarding.

Key protocols at this layer include the Internet Protocol (IP), which is important for routing and addressing, Internet Control Message Protocol (ICMP) for diagnostic and error-reporting purposes, and routing protocols like Routing Information Protocol (RIP) that manage the routing of data across networks.

2. Data Link Layer



The Data Link Layer is responsible for node-to-node data transfer and error detection and correction. It ensures that data is transmitted to the correct device on a local network segment. This layer manages MAC (Media Access Control) addresses and is divided into two sublayers: Logical Link Control (LLC) and Media Access Control (MAC).

Protocols and technologies at this layer include Ethernet, which defines the rules for data transmission over local area networks (LANs), and Point-to-Point Protocol (PPP) for direct connections between two network nodes. It also includes mechanisms for detecting and possibly correcting errors that may occur in the Physical Layer.

1. Physical Layer

The Physical Layer is responsible for the physical connection between devices. It defines the hardware elements involved in the network, including cables, switches, and other physical components. This layer also specifies the electrical, optical, and radio characteristics of the network.

Functions of the Physical Layer include the modulation, bit synchronization, and transmission of raw binary data over the physical medium. Technologies such as Fiber Optics and Wi-Fi operate at this layer, ensuring that the data physically moves from one device to another in the network.

How Does Communication Happen in the OSI Model? A Practical Example

Let's consider how OSI layers play a role in an everyday activity like sending an email to a person overseas:

When a user in New York sends an email to a colleague in London, the process starts at the Application Layer (Layer 7). The user's email client, such as Outlook, uses SMTP (Simple Mail Transfer Protocol) to handle the email message.
The email is then passed to the Presentation Layer (Layer 6), where it is formatted and encrypted to ensure proper transmission.
Next, the email moves to the Session Layer (Layer 5), where a session is established between the sender's email server in New York and the receiver's email server in London. This layer manages the session, keeping the connection open long enough to send the email.
The email data then reaches the Transport Layer (Layer 4), where it is divided into smaller packets. TCP ensures these packets are sent reliably and in the correct order.
At the Network Layer (Layer 3), each packet is assigned source and destination IP addresses, allowing it to be routed through multiple networks, including routers and switches, to reach the recipient in London.
The Data Link Layer (Layer 2) then uses MAC addresses to handle the packets' journey across local networks and correcting any errors that occur.
Finally, the Physical Layer (Layer 1) converts the data into electrical signals, which are transmitted over fiber-optic cables under the Atlantic Ocean.
Upon reaching the recipient's server in London, the process is reversed:

The Physical Layer converts the signals back into data packets, which are reassembled at the Data Link Layer.
The Network Layer ensures the packets have arrived correctly, and the Transport Layer reorders them if necessary.
The Session Layer maintains the session until the email is fully received.
The Presentation Layer decrypts and formats the email, and the Application Layer delivers the email to the client, where it appears in their inbox.

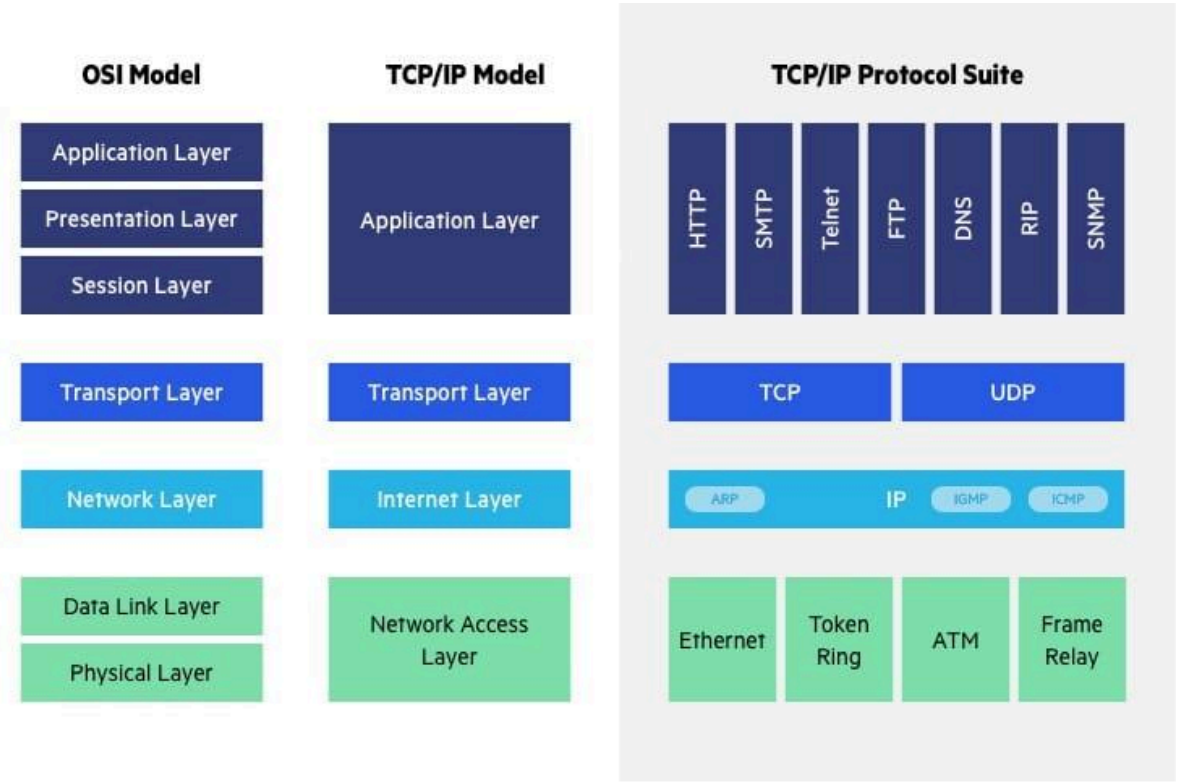
Advantages of OSI Model

The OSI model helps users and operators of computer networks:

- Determine the required hardware and software to build their network.
- Understand and communicate the process followed by components communicating across a network.
- Perform troubleshooting, by identifying which network layer is causing an issue and focusing efforts on that layer.
- The OSI model helps network device manufacturers and networking software vendors.
- Create devices and software that can communicate with products from any other vendor, allowing open interoperability
- Define which parts of the network their products should work with.

- Communicate to users at which network layers their product operates – for example, only at the application layer, or across the stack.

OSI vs. TCP/IP Model



The Transfer Control Protocol/Internet Protocol (TCP/IP) is older than the OSI model and was created by the US Department of Defense (DoD). A key difference between the models is that TCP/IP is simpler, collapsing several OSI layers into one:

OSI layers 5, 6, 7 are combined into one Application Layer in TCP/IP
OSI layers 1, 2 are combined into one Network Access Layer in TCP/IP – however TCP/IP does not take responsibility for sequencing and acknowledgement functions, leaving these to the underlying transport layer.
Other important differences:

TCP/IP is a functional model designed to solve specific communication problems, and which is based on specific, standard protocols. OSI is a generic, protocol-independent model intended to describe all forms of network communication.
In TCP/IP, most applications use all the layers, while in OSI simple applications do not use all seven layers. Only layers 1, 2 and 3 are mandatory to enable any data communication.

Prepare R&D Document on working & functionality of TCP/IP Model

The TCP/IP Model (Transmission Control Protocol/Internet Protocol) is the foundational suite of communication protocols used to interconnect network devices on the internet. It standardizes how data should be packaged, addressed, transmitted, routed, and received.

Unlike the OSI Model (7 layers), the TCP/IP Model is composed of 4 layers, each with specific functionalities that ensure reliable data transmission between devices over a network.

Layers of TCP/IP Model

The TCP/IP Model has the following 4 layers, from top to bottom:

1. Application Layer
2. Transport Layer
3. Internet Layer
4. Network Access Layer (or Link Layer)

1. Application Layer

Function:

- Provides user services and network-based applications like HTTP, FTP, SMTP, DNS, etc.
- Interfaces directly with user-facing applications and processes.

Responsibilities:

- Data formatting
- Encryption/decryption
- Communication protocols like:

  HTTP (for web)

  SMTP (email)

FTP (file transfer)

DNS (domain resolution)

Example: When a user types a URL in the browser, the HTTP protocol in this layer initiates a request to retrieve the web page.

2. Transport Layer

Function:

- Ensures reliable or unreliable data transmission between host devices.

Main Protocols:

- TCP (Transmission Control Protocol) – Reliable, connection-oriented.
- UDP (User Datagram Protocol) – Unreliable, connectionless.

Responsibilities:

- Segmentation and Reassembly of data
- Flow Control
- Error Checking
- Acknowledgments and Retransmissions (TCP only)

TCP Features:

- Three-way handshake (SYN, SYN-ACK, ACK)
- Ensures ordered and reliable delivery

UDP Use Cases:

- Streaming, VoIP, gaming – where speed > reliability

3. Internet Layer

Function:

- Handles logical addressing and routing of data packets across networks.

Main Protocols:

- IP (Internet Protocol) – IPv4 and IPv6
- ICMP (Internet Control Message Protocol) – For error messages (e.g., ping)
- ARP (Address Resolution Protocol) – Maps IP addresses to MAC addresses

Responsibilities:

- Packet creation, addressing, and forwarding
- Routing decisions based on destination IP

Working:

Each packet gets a source and destination IP. Routers use this info to forward the packet to the correct destination.

4. Network Access Layer (Link Layer)

Function:

- Responsible for the physical transmission of data over network hardware.

Sub-layers:

- Data Link Layer (e.g., Ethernet, PPP)
- Physical Layer (e.g., cables, Wi-Fi signals)

Responsibilities:

- MAC addressing
- Framing
- Error detection (via CRC)
- Physical media signaling (electrical/optical signals)

Example:When a packet is sent over Ethernet, the Link Layer adds the destination MAC address and prepares it for transmission on the wire.

Working of TCP/IP Model (End-to-End Communication)

Step-by-Step Flow:

1. User Action: User sends an email using Outlook (Application Layer).
2. Application Layer: SMTP prepares email data.
3. Transport Layer: TCP breaks it into segments, adds port numbers.
4. Internet Layer: IP adds source & destination IP addresses.
5. Link Layer: Converts data into frames, adds MAC address.
6. Physical Transmission: Data travels through cables/routers to the destination.
7. At Receiver: Reverse process reconstructs the original message.

Key Features of TCP/IP Model:

- Modular & Scalable
- Supports Internetworking
- Vendor-Neutral Standards
- Widely Adopted (Internet Backbone)

Prepare R&D Document working of TCP & UDP Protocols, working of HTTP, HTTPs & ICMP Protocol

1. Transmission Control Protocol (TCP)

Transmission Control Protocol (TCP) is a transport layer protocol that enables reliable communication between devices on a network. It is a connection-oriented protocol, which means a connection must be established between the sender and receiver before data transmission begins. The process starts with a three-way handshake, where the client sends a SYN (synchronize) message to the server, the server replies with a SYN-ACK (synchronize-acknowledge), and the client responds with an ACK (acknowledge). This establishes a reliable connection.

Once the connection is established, data is transmitted in segments. Each segment contains a sequence number, which helps the receiver reorder any out-of-order packets. The receiver sends back acknowledgements (ACKs) to inform the sender that the data has been received successfully. If an ACK is not received, TCP assumes packet loss and retransmits the data. TCP uses flow control through a sliding window mechanism to avoid overwhelming the receiver and employs congestion control algorithms to prevent network congestion. When the communication is complete, TCP uses a four-step process to terminate the connection gracefully using FIN and ACK messages.

TCP is widely used for applications that require accuracy and reliability, such as web browsing, email, file transfers, and remote administration. Its key features include guaranteed data delivery, error detection and correction, and proper sequencing of data.

2. User Datagram Protocol (UDP)

User Datagram Protocol (UDP) is a transport layer protocol that offers a faster, simpler method of data transmission compared to TCP. It is a connectionless protocol, meaning it does not establish a connection before sending data. This allows UDP to transmit data with minimal delay, but without any guarantee of delivery, ordering, or error correction.

In UDP, data is sent in discrete units called datagrams. Each datagram contains destination information and is transmitted independently of others. Unlike TCP, UDP does not perform handshaking or acknowledgements. It does include a basic checksum for error detection, but there is no mechanism for retransmitting lost or corrupted packets. This makes UDP suitable for applications where speed is critical and occasional data loss is acceptable.

UDP is commonly used in real-time applications such as voice over IP (VoIP), video conferencing, online gaming, and Domain Name System (DNS) queries, where low latency is more important than reliability. Its minimal overhead and fast transmission make it ideal for time-sensitive communications.

3. HyperText Transfer Protocol (HTTP)

HyperText Transfer Protocol (HTTP) is an application layer protocol used for transmitting data over the World Wide Web. It is based on a request-response model, where a client, typically a web browser, sends a request to a web server, and the server responds with the requested resource such as an HTML page, image, or file.

When a client initiates a request, it uses HTTP methods like GET, POST, PUT, or DELETE. The server processes this request and returns an HTTP response that includes a status code (e.g., 200 OK, 404 Not Found), headers (containing metadata), and sometimes a message body (the requested content). HTTP operates over TCP, usually on port 80, and relies on TCP's reliability for data transmission.

HTTP is a stateless protocol, meaning each request is independent and has no knowledge of previous requests. This statelessness simplifies server design but can limit interactivity, which is often mitigated through the use of cookies or sessions. HTTP is the foundation of data communication for the web, enabling users to access websites and interact with web applications.

4. HyperText Transfer Protocol Secure (HTTPS)

HyperText Transfer Protocol Secure (HTTPS) is the secure version of HTTP. It combines HTTP with SSL/TLS protocols to provide encrypted and authenticated communication between a client and server. HTTPS ensures that data transmitted over the internet is protected from eavesdropping, tampering, and man-in-the-middle attacks.

When a user connects to a website using HTTPS, the communication begins with a TLS handshake. During this process, the client and server exchange digital certificates and agree upon encryption keys using public-key cryptography. Once the handshake is complete, all subsequent data is encrypted using symmetric encryption, ensuring confidentiality and integrity.

HTTPS uses port 443 instead of port 80, which is used by HTTP. The use of HTTPS is essential for sensitive transactions such as online banking, e-commerce, login systems, and any website that handles personal data. It builds user trust by displaying a padlock icon in the browser and often requiring certificates issued by trusted Certificate Authorities (CAs). Overall, HTTPS enhances security without compromising the functionality of standard HTTP.

5. Internet Control Message Protocol (ICMP)

Internet Control Message Protocol (ICMP) is a network layer protocol used primarily for diagnostics and error reporting within IP networks. Unlike TCP and UDP, ICMP is not used to exchange data between applications but rather to send control messages related to network operations.

ICMP messages are typically generated by network devices like routers or gateways in response to network errors or issues. For example, if a packet cannot be delivered to its destination, the router may send an ICMP "Destination Unreachable" message back to the sender. Similarly, if a packet's time-to-live (TTL) value reaches zero, an ICMP "Time Exceeded" message is sent. ICMP also supports tools like ping, which uses Echo Request and Echo Reply messages to test connectivity between hosts.

ICMP packets are encapsulated within IP packets and do not guarantee delivery. Despite being lightweight, ICMP plays a crucial role in network maintenance and troubleshooting. It allows network administrators to identify routing problems, unreachable hosts, or latency issues, making it an essential protocol in the TCP/IP suite.

Case Study: Real-World Applications of TCP, UDP, HTTP, HTTPS & ICMP

- In an online banking system, TCP is used to ensure that all data sent between the client and the bank server is delivered accurately and in order. For example, when a customer submits a transaction, the data packets containing the transaction details must reach the bank server without loss or duplication. TCP guarantees this through its reliable delivery mechanism, retransmission strategies, and error-checking features, making it suitable for critical financial applications.
- UDP is commonly used in live sports streaming platforms such as Disney+ Hotstar. In such real-time applications, speed is more important than perfect accuracy. A few lost packets are tolerable and go unnoticed by users. UDP allows fast, lag-free transmission of audio and video content without the overhead of connection setup or retransmission, which makes it ideal for live events and broadcasts.
- HTTP is used by news websites like BBC or Times of India. When a user opens a news article, the browser sends an HTTP GET request to the web server, which responds with the article content, images, and style sheets. Since HTTP is stateless, each request is independent, making it efficient for delivering large volumes of content quickly to many users simultaneously.
- HTTPS is essential in e-commerce websites like Amazon. During the checkout process, sensitive data such as payment information is transmitted between the user's device and the server. HTTPS ensures this communication is encrypted and secure by using TLS, which protects the transaction from being intercepted or modified. The server's digital certificate also assures the client of the website’s authenticity.
- ICMP is useful in diagnosing connectivity issues within a network. For instance, an IT team in a company may use the ping command to check if a remote server is reachable. The ping utility sends ICMP Echo Request messages and receives Echo Replies. If the server is unreachable, ICMP may return an error such as “Destination Unreachable,” helping the team identify and fix routing or connectivity issues promptly.

Conclusion

The protocols TCP, UDP, HTTP, HTTPS, and ICMP serve as critical building blocks for internet communication. TCP provides reliable and ordered data transfer, UDP offers high-speed communication with minimal overhead, HTTP enables standard web access, HTTPS secures web traffic, and ICMP supports network diagnostics. The case studies demonstrate how these protocols operate in real-world scenarios across banking, entertainment, news, e-commerce, and IT infrastructure. A solid understanding of these protocols is essential for network engineers, developers, and IT professionals involved in designing and maintaining modern networked systems.