PCET's

Pimpri Chinchwad University

School of Computer Application

PMC113A Computer Networks

Unit No. 01 Introduction

Introduction:

A computer network is a collection of interconnected devices that communicate and share resources with each other. These devices, called nodes, can include computers, servers, mobile devices, printers, and other hardware components.

1. Purpose of Computer Networks

- Resource Sharing: Allows sharing of hardware (printers, storage) and software resources (applications).
- Communication: Facilitates email, messaging, video conferencing, and file sharing.
- Data Storage and Management: Enables centralized data storage, backup, and security.
- Remote Access: Supports remote working and cloud computing.
- Scalability and Flexibility: Allows easy addition of new devices without significant changes.

2. Types of Computer Networks

1. **Personal Area Network (PAN)**:
   - Small network for personal devices (e.g., Bluetooth, Wi-Fi).
2. **Local Area Network (LAN)**:
   - Covers a small area, such as an office or building.
   - High speed and low cost.
3. **Metropolitan Area Network (MAN)**:
   - Covers a city or large campus.
   - Connects multiple LANs within a region.
4. **Wide Area Network (WAN)**:
   - Spans large geographical areas, such as countries or continents.
   - Example: The Internet.
5. **Wireless Networks (WLAN, WWAN)**:
   - Uses radio waves for connectivity, eliminating the need for physical cables.

3. Components of Computer Networks

- **Nodes**: Devices that communicate, such as computers and routers.
- **Switches and Hubs**: Devices that connect multiple devices within a LAN.

- **Routers**: Connect different networks and direct data between them.
- **Cables and Media**: Physical links like Ethernet cables, fiber optics, or wireless signals.
- **Protocols**: Rules that govern communication, such as TCP/IP, HTTP, and FTP.

4. Network Models

1. **OSI Model (Open Systems Interconnection)** - Divides networking tasks into 7 layers:
   - Physical, Data Link, Network, Transport, Session, Presentation, and Application.
2. **TCP/IP Model** - Simplified 4-layer model:
   - Network Interface, Internet, Transport, and Application layers.

5. Applications of Computer Networks

- **Internet**: Global network enabling communication and resource sharing.
- **Cloud Computing**: Provides services and storage via the Internet.
- **IoT (Internet of Things)**: Connects smart devices for automation.
- **Industry 4.0**: Supports smart factories and edge computing systems.

6. Security in Computer Networks

- **Encryption**: Protects data during transmission.
- **Firewalls**: Prevent unauthorized access.
- **Antivirus Software**: Detects and removes malware.
- **VPN (Virtual Private Network)**: Ensures secure remote access.

Definition of Computer Network

A **computer network** is a system of interconnected devices, such as computers, servers, and peripherals, that communicate and share resources (e.g., data, files, applications, and hardware) using wired or wireless connections. These networks operate based on established protocols to ensure seamless data transmission and communication.

Goals of Computer Networks

1. **Resource Sharing**
   - Enable multiple users to share hardware (e.g., printers) and software resources.
   - Minimize redundancy and reduce costs by centralizing resources.
2. **Communication**
   - Facilitate data exchange through emails, video conferencing, and messaging services.
   - Support collaborative work environments.
3. **Reliability**

- Provide backup and fault tolerance to ensure data availability even during failures.
- Enable redundancy mechanisms for high availability.

4. **Scalability**
   - Allow easy addition of new devices or users without major changes to the network.
   - Support growing demands for bandwidth and performance.
5. **Data Sharing**
   - Enable efficient access and transfer of files and databases.
   - Allow centralized data storage for better management and backup.
6. **Remote Access and Connectivity**
   - Support remote working through VPNs and cloud services.
   - Enable global access to resources via the Internet.
7. **Security**
   - Protect data through encryption, authentication, and firewalls.
   - Ensure confidentiality, integrity, and availability of information.
8. **Cost Efficiency**
   - Reduce costs by sharing resources instead of duplicating them.
   - Optimize hardware usage and reduce maintenance expenses.
9. **Improved Performance**
   - Enable load balancing and distributed computing to enhance processing power.
   - Support fast data transfer and communication.
10. **Automation and Control**
    - Support automation in industries through IoT and smart systems.
    - Facilitate remote monitoring and control of devices.

Computer Networks Design Issues

Designing a computer network involves addressing several critical issues to ensure efficiency, scalability, security, and reliability. The major design issues in computer networks are outlined below:

1. Performance

- **Throughput**: The amount of data transmitted successfully per unit time. The design should optimize bandwidth utilization.
- **Latency**: The delay in data transfer from source to destination should be minimized.
- **Congestion Control**: Prevents network overload by managing data flow effectively.

- **Quality of Service (QoS)**: Ensures prioritization of critical data, such as voice and video traffic, over less sensitive data.

2. Scalability

- Networks must be designed to handle future growth without requiring major structural changes.
- They should support adding new devices and users without performance degradation.

3. Security

- **Data Privacy**: Ensures sensitive information is protected from unauthorized access using encryption.
- **Authentication and Authorization**: Controls access to network resources using identity verification mechanisms.
- **Firewall and Intrusion Detection Systems**: Defend against external threats and attacks.
- **Data Integrity**: Ensures data consistency and prevents unauthorized modifications.

4. Reliability

- Networks must ensure high availability with minimal downtime through redundancy and fault tolerance.
- Backup systems and failover mechanisms should be included to recover from failures.

5. Addressing and Routing

- **Addressing Scheme**: Provides unique identification for devices (IP addresses) to enable communication.
- **Routing Protocols**: Efficiently determine paths for data transfer based on shortest paths, reliability, and congestion.

6. Interoperability

- Networks must support heterogeneous devices and protocols to facilitate communication across different platforms.
- Standards like TCP/IP and OSI models should be implemented for compatibility.

7. Resource Allocation

- Efficiently allocate resources, such as bandwidth and storage, based on demand.
- Load balancing mechanisms distribute network traffic evenly to prevent bottlenecks.

8. Topology Selection

- Choosing the appropriate network topology (e.g., star, ring, mesh) based on cost, performance, and fault tolerance requirements.
- Flexibility for topology changes should be considered.

9. Energy Efficiency

- Modern networks, especially IoT and edge computing, require energy-efficient designs to minimize power consumption.
- Energy-saving protocols should be implemented in hardware and software.

10. Cost Constraints

- The network should meet performance and security requirements while staying within budget.
- Balancing high performance and low costs is critical.

11. Mobility Support

- Networks must support mobile devices and ensure seamless handoff between access points without data loss.
- Mobile IP protocols are used for managing mobility.

12. Quality of Experience (QoE)

- Ensures end-user satisfaction with network performance, especially for multimedia applications like VoIP, streaming, and gaming.

Network Architecture-Broadcast and Point to Point

Network Architecture: Broadcast vs. Point-to-Point

Computer network architectures can be broadly classified into **Broadcast** and **Point-to-Point** types, based on how data is transmitted between devices.

1. Broadcast Network Architecture

Definition:

In a **broadcast network**, a single communication channel is shared by all connected devices (nodes). When a device sends a message, it is broadcasted to all devices in the network.

Key Features:

- **Shared Medium:** All nodes share the same communication channel.
- **Broadcast Transmission:** Data is sent to all devices, but only the intended recipient processes it based on its address.
- **Collision Handling:** Protocols like **CSMA/CD (Carrier Sense Multiple Access with Collision Detection)** and **Token Passing** handle collisions and ensure orderly access.

Examples:

- **Ethernet (LANs):** Broadcast frames within a network segment.
- **Wi-Fi Networks:** Wireless communication inherently uses broadcasting.
- **Satellite Communication:** Signals are broadcast to all receivers in the coverage area.

Advantages:

1. **Simple Implementation:** Easy setup and cost-effective for small networks.
2. **Resource Sharing:** Suitable for applications like file and printer sharing.
3. **Scalability for Small Networks:** Additional devices can be added without major reconfiguration.

Disadvantages:

1. **Collisions:** High traffic leads to collisions, affecting performance.
2. **Security Issues:** Broadcasted data can be intercepted by unintended recipients without encryption.
3. **Inefficiency for Large Networks:** Broadcasting wastes bandwidth when only a few devices need the data.

2. Point-to-Point Network Architecture

Definition:

In a **point-to-point network**, there is a dedicated communication link between two devices. Data is sent directly between the sender and the receiver without involving other devices.

Key Features:

- **Dedicated Links:** Each pair of nodes is connected by a private link.
- **Direct Communication:** No broadcasting; data travels directly to the intended recipient.
- **Switching Mechanisms:** Use **circuit switching**, **packet switching**, or **message switching** for data transfer.

Examples:

- **Telephone Networks:** Traditional and VoIP calls establish direct connections.
- **Leased Lines:** Dedicated wired connections between offices.
- **Switched Ethernet Networks:** Use switches to create virtual point-to-point links.

Advantages:

1. **Reliable Communication:** No data collisions since links are dedicated.
2. **Security:** Data is not visible to other devices, enhancing privacy.
3. **Scalability for Large Networks:** Better suited for large and complex networks.
4. **Efficiency:** Direct communication reduces unnecessary data transfer.

Disadvantages:

1. **Higher Cost:** Requires more cabling and switches for large-scale networks.
2. **Complex Setup:** Managing multiple dedicated links is more complex than a shared medium.

3. **Limited Resource Sharing:** Requires additional configurations for shared resource access.

Comparison Table: Broadcast vs. Point-to-Point

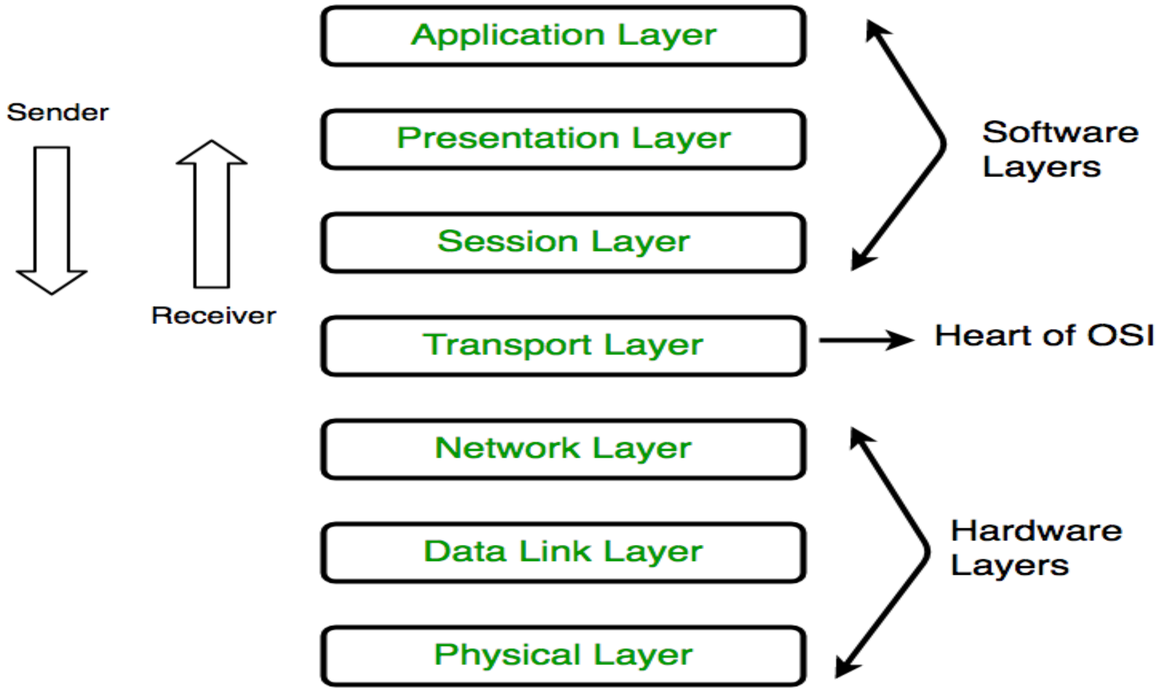
Feature	**Broadcast Network**	**Point-to-Point Network**
Communication Medium	Shared channel for all devices.	Dedicated links between pairs of devices.
Data Transmission	Broadcasted to all devices.	Sent directly to the intended recipient.
Collisions	High possibility, requiring management.	No collisions due to dedicated links.
Scalability	Suitable for small networks.	Scalable for larger networks.
Cost	Low cost and simple setup.	Higher cost due to dedicated connections.
Security	Less secure; data can be intercepted.	Highly secure due to private links.
Examples	Ethernet, Wi-Fi, Satellite communication.	Telephone networks, Leased lines, VPNs.

Conclusion

- **Broadcast networks** are effective for small-scale applications where shared resources are needed, such as LANs or IoT systems.
- **Point-to-point networks** are more suitable for larger and more complex systems requiring scalability, reliability, and security.

OSI Reference Model

OSI stands for Open Systems Interconnection. It has been developed by ISO – ‘International Organization of Standardization ‘, in the year 1974. It is 7-layer architecture with each layer having specific functionality to perform. All these 7 layers work collaboratively to transmit the data from one person to another across the globe.



1. Physical Layer (Layer 1):

The lowest layer of the OSI reference model is the physical layer. It is responsible for the actual physical connection between the devices. The physical layer contains information in the form of bits. It is responsible for the actual physical connection between the devices. When receiving data, this layer will get the signal received and convert it into 0s and 1s and send them to the Data Link layer, which will put the frame back together.

The functions of the physical layer are:

1. Bit synchronization: The physical layer provides the synchronization of the bits by providing a clock. This clock controls both sender and receiver thus providing synchronization at bit level.

2. Bit rate control: The Physical layer also defines the transmission rate i.e. the number of bits sent per second.

3. Physical topologies: Physical layer specifies the way in which the different, devices/nodes are arranged in a network i.e. bus, star or mesh topology.

4. Transmission mode: Physical layer also defines the way in which the data flows between the two connected devices. The various transmission modes possible are: Simplex, half-duplex and full-duplex.

* Hub, Repeater, Modem, Cables are Physical Layer devices.

** Network Layer, Data Link Layer and Physical Layer are also known as Lower Layers or Hardware Layers.

2. Data Link Layer (DLL) (Layer 2):

The data link layer is responsible for the node-to-node delivery of the message. The main function of this layer is to make sure data transfer is error free from one node to another, over the physical layer. When a packet arrives in a network, it is the responsibility of DLL to transmit it to the Host using its MAC address.

Data Link Layer is divided into two sub layers:

1. Logical Link Control (LLC)

2. Media Access Control (MAC)

The packet received from Network layer is further divided into frames depending on the frame size of NIC (Network Interface Card). DLL also encapsulates Sender and Receiver's MAC address in the header.

The Receiver's MAC address is obtained by placing an ARP (Address Resolution Protocol) request onto the wire asking "Who has that IP address?" and the destination host will reply with its MAC address.

The functions of the data Link layer are:

1. Framing: Framing is a function of the data link layer. It provides a way for a sender to transmit a set of bits that are meaningful to the receiver. This can be accomplished by attaching special bit patterns to the beginning and end of the frame.

2. Physical addressing: After creating frames, Data link layer adds physical addresses (MAC address) of sender and/or receiver in the header of each frame.

3. Error control: Data link layer provides the mechanism of error control in which it detects and retransmits damaged or lost frames.

4. Flow Control: The data rate must be constant on both sides else the data may get corrupted thus; flow control coordinates that amount of data that can be sent before receiving acknowledgement.

5. Access control: When a single communication channel is shared by multiple devices, MAC sub-layer of data link layer helps to determine which device has control over the channel at a given time.

*** Packet in Data Link layer is referred as Frame.**

** Data Link layer is handled by the NIC (Network Interface Card) and device drivers of host machines.

*** Switch & Bridge are Data Link Layer devices.

3. Network Layer (Layer 3):

Network layer works for the transmission of data from one host to the other located in different networks. It also takes care of packet routing i.e. selection of the shortest path to transmit the packet, from the number of routes available. The sender & receiver's IP address are placed in the header by network layer.

The functions of the Network layer are:

1. Routing: The network layer protocols determine which route is suitable from source to destination. This function of network layer is known as routing.

2. Logical Addressing: In order to identify each device on internetwork uniquely, network layer defines an addressing scheme. The sender & receiver's IP address are placed in the header by network layer. Such an address distinguishes each device uniquely and universally.

*** Segment in Network layer is referred as Packet.**

Network layer is implemented by networking devices such as routers.

4. Transport Layer (Layer 4):

Transport layer provides services to application layer and takes services from network layer. The data in the transport layer is referred to as Segments. It is responsible for the End-to-End delivery of the complete message. Transport layer also provides the acknowledgment of the successful data transmission and re-transmits the data if an error is found.

• At sender's side:

Transport layer receives the formatted data from the upper layers, performs Segmentation and also implements Flow & Error control to ensure proper data transmission. It also adds Source and Destination port number in its header and forwards the segmented data to the Network Layer.

Note: The sender needs to know the port number associated with the receiver's application. Generally, this destination port number is configured, either by default or manually. For example, when a web application makes a request to a web server, it typically uses port number 80, because this is the default port assigned to web applications. Many applications have default port assigned.

• At receiver's side:

Transport Layer reads the port number from its header and forwards the Data which it has received to the respective application. It also performs sequencing and reassembling of the segmented data.

The functions of the transport layer are:

1. Segmentation and Reassembly: This layer accepts the message from the (session) layer, breaks the message into smaller units. Each of the segment produced has a header associated with it. The transport layer at the destination station reassembles the message.

2. Service Point Addressing: In order to deliver the message to correct process, transport layer header includes a type of address called service point address or port address. Thus, by specifying this address, transport layer makes sure that the message is delivered to the correct process.

The services provided by transport layer:

1. Connection Oriented Service: It is a three-phase process which include

– Connection Establishment

– Data Transfer

– Termination / disconnection

In this type of transmission, the receiving device sends an acknowledgment, back to the source after a packet or group of packets is received. This type of transmission is reliable and secure.

2. Connection less service: It is a one phase process and includes Data Transfer. In this type of transmission, the receiver does not acknowledge receipt of a packet. This approach allows for much faster communication between devices. Connection oriented Service is more reliable than connection less Service.

*** Data in the Transport Layer is called as Segments.**

** Transport layer is operated by the Operating System. It is a part of the OS and communicates with the Application Layer by making system calls. Transport Layer is called as Heart of OSI model.

5. Session Layer (Layer 5):

This layer is responsible for establishment of connection, maintenance of sessions, authentication and also ensures security.

The functions of the session layer are:

1. Session establishment, maintenance, and termination: The layer allows the two processes to establish, use and terminate a connection.

2. Synchronization: This layer allows a process to add checkpoints which are considered as synchronization points into the data. These synchronization point help to identify the error so that the data is re-synchronized properly, and ends of the messages are not cut prematurely and data loss is avoided.

3. Dialog Controller: The session layer allows two systems to start communication with each other in half-duplex or full-duplex.

**All the below 3 layers (including Session Layer) are integrated as a single layer in TCP/IP model as “Application Layer”.

**Implementation of these 3 layers is done by the network application itself. These are also known as Upper Layers or Software Layers.

SCENARIO:

Let’s consider a scenario where a user wants to send a message through some Messenger application running in his browser. The “Messenger” here acts as the application layer which provides the user with an interface to create the data. This message or so-called Data is compressed, encrypted (if any secure data) and converted into bits (0’s and 1’s) so that it can be transmitted.

6. Presentation Layer (Layer 6):

Presentation layer is also called the Translation layer. The data from the application layer is extracted here and manipulated as per the required format to transmit over the network.

The functions of the presentation layer are:

1. Translation: For example, ASCII to EBCDIC.

2. Encryption/ Decryption: Data encryption translates the data into another form or code. The encrypted data is known as the cipher text and the decrypted data is known as plain text. A key value is used for encrypting as well as decrypting data.

3. Compression: Reduces the number of bits that need to be transmitted on the network.

7. Application Layer (Layer 7):

At the very top of the OSI Reference Model stack of layers, we find Application layer which is implemented by the network applications. These applications produce the data, which has to be transferred over the network. This layer also serves as a window for the application services to access the network and for displaying the received information to the user.

Ex: Application – Browsers, Skype Messenger etc.

****Application Layer is also called as Desktop Layer.**

The functions of the Application layer are:

1. Network Virtual Terminal

2. FTAM-File transfer access and management

3. Mail Services

4. Directory Services

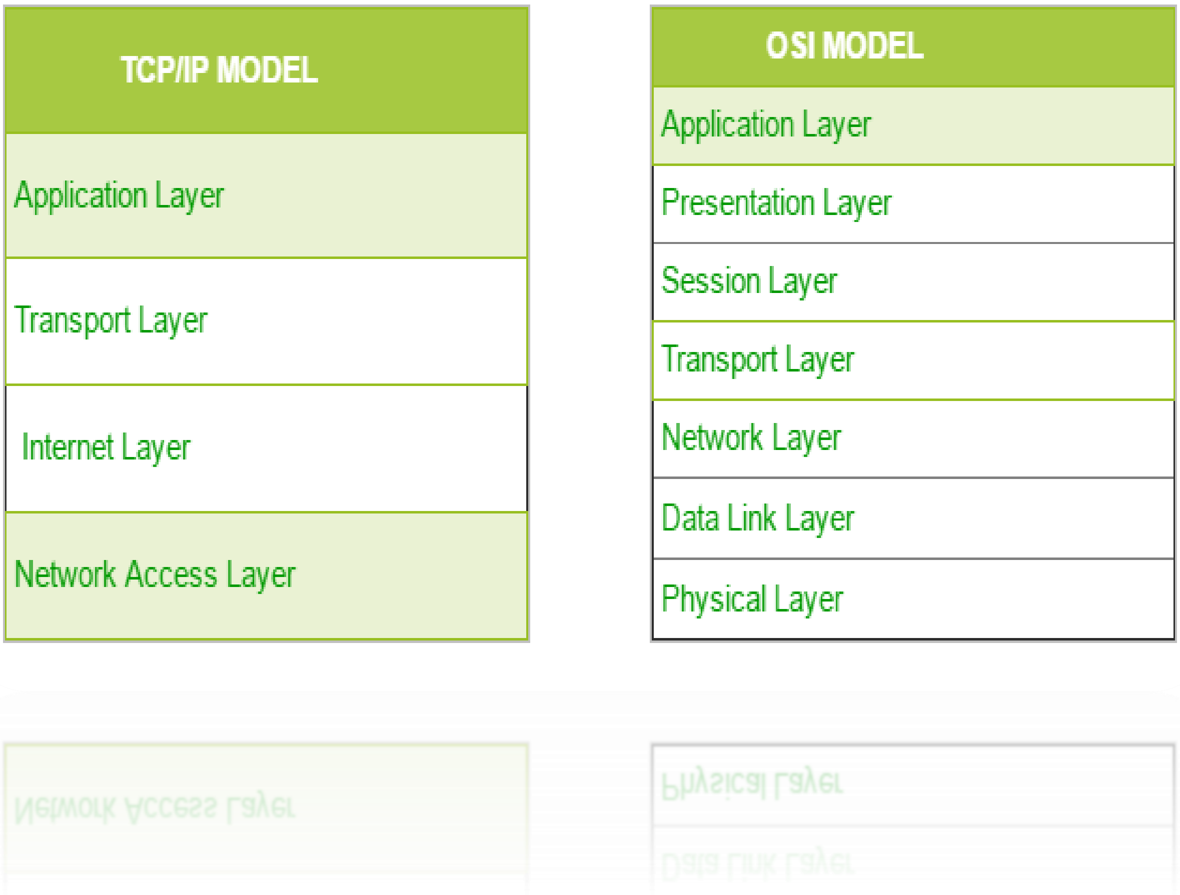
OSI model acts as a reference model and is not implemented in Internet because of its late invention. Current model being used is the TCP/IP model.

TCP/IP Model

The **OSI Model** we just looked at is just a reference/logical model. It was designed to describe the functions of the communication system by dividing the communication procedure into smaller and simpler components. But when we talk about the TCP/IP model, it was designed and developed by Department of Defense (DoD) in 1960s and is based on standard protocols. It stands for Transmission Control Protocol/Internet Protocol. The **TCP/IP model** is a concise version of the OSI model. It contains four layers, unlike seven layers in the OSI model. The layers are:

1. Process/Application Layer
2. Host-to-Host/Transport Layer
3. Internet Layer
4. Network Access/Link Layer (Host to Network)

The diagrammatic comparison of the TCP/IP and OSI model is as follows:



TCP/IP MODEL
Application Layer
Transport Layer
Internet Layer
Network Access Layer

OSI MODEL
Application Layer
Presentation Layer
Session Layer
Transport Layer
Network Layer
Data Link Layer
Physical Layer

Difference between TCP/IP and OSI Model:

TCP/IP	OSI
1) TCP/IP refers to Transmission Control Protocol. Internet protocol	1) OSI refers to Open Systems Interconnection.
2) TCP/IP has 4 layers.	2) OSI has 7 layers.
3) TCP/IP is more reliable	3) OSI is less reliable
4) TCP/IP does not have very strict boundaries.	4) OSI has strict boundaries
5) TCP/IP follows a horizontal approach.	5) OSI follows a vertical approach.
6) TCP/IP uses both session and presentation layer in the application layer itself.	6) OSI uses different session and presentation layers.
7) TCP/IP developed protocols then model.	7) OSI developed model then protocol.

1. Host to Network Layer –

This layer corresponds to the combination of Data Link Layer and Physical Layer of the OSI model. It looks out for hardware addressing and the protocols present in this layer allows for the physical transmission of data.

We just talked about ARP being a protocol of Internet layer, but there is a conflict about declaring it as a protocol of Internet Layer or Network access layer. It is described as residing in layer 3, being encapsulated by layer 2 protocols.

2. Internet Layer –

This layer parallels the functions of OSI's Network layer. It defines the protocols which are responsible for logical transmission of data over the entire network. The main protocols residing at this layer are:

IP – stands for Internet Protocol and it is responsible for delivering packets from the source host to the destination host by looking at the IP addresses in the packet headers.

IP has 2 versions: IPv4 and IPv6. IPv4 is the one that most of the websites are using currently. But IPv6 is growing as the number of IPv4 addresses is limited in number when compared to the number of users.

ICMP – stands for Internet Control Message Protocol. It is encapsulated within IP datagrams and is responsible for providing hosts with information about network problems.

ARP – stands for Address Resolution Protocol. Its job is to find the hardware address of a host from a known IP address. ARP has several types: Reverse ARP, Proxy ARP, Gratuitous ARP and Inverse ARP.

3. Transport Layer –

This layer is analogous to the transport layer of the OSI model. It is responsible for end-to-end communication and error-free delivery of data. It shields the upper-layer applications from the complexities of data. The two main protocols present in this layer are:

Transmission Control Protocol (TCP) – It is known to provide reliable and error-free communication between end systems. It performs sequencing and segmentation of data. It also has acknowledgment feature and controls the flow of the data through flow control mechanism. It is a very effective protocol but has a lot of overhead due to such features. Increased overhead leads to increased cost.

User Datagram Protocol (UDP) – On the other hand does not provide any such features. It is the go-to protocol if your application does not require reliable transport as it is very cost-effective. Unlike TCP, which is connection-oriented protocol, UDP is connectionless.

4. Application Layer –

This layer performs the functions of top three layers of the OSI model: Application, Presentation and Session Layer. It is responsible for node-to-node communication and controls user-interface specifications. Some of the protocols present in this layer are: HTTP, HTTPS, FTP, TFTP, Telnet, SSH, SMTP, SNMP, NTP, DNS, DHCP, NFS, X Window, and LPD. Have a look at Protocols in Application Layer for some information about these protocols. Protocols other than those present in the linked article are:

1) HTTP and HTTPS – HTTP stands for Hypertext transfer protocol. It is used by the World Wide Web to manage communications between web browsers and servers. HTTPS stands for HTTP-Secure. It is a combination of HTTP with SSL (Secure Socket Layer). It is efficient in cases where the browser need to fill out forms, sign in, authenticate and carry out bank transactions.

2) SSH – SSH stands for Secure Shell. It is terminal emulations software similar to Telnet. The reason SSH is more preferred is because of its ability to maintain the encrypted connection. It sets up a secure session over a TCP/IP connection.

3) NTP – NTP stands for Network Time Protocol. It is used to synchronize the clocks on our computer to one standard time source. It is very useful in situations like bank transactions. Assume the following situation without the presence of NTP. Suppose you carry out a transaction, where your computer reads the time at 2:30 PM while the server records it at 2:28 PM. The server can crash very badly if it's out of sync.

Network Classification LAN MAN WAN

Network Classification: LAN, MAN, and WAN

Computer networks are classified based on their **geographical coverage**, **size**, and **purpose** into three main types—**LAN (Local Area Network)**, **MAN (Metropolitan Area Network)**, and **WAN (Wide Area Network)**.

1. Local Area Network (LAN)

Definition:

A **LAN** is a network that connects computers and devices within a limited area, such as an office building, campus, or home.

Key Features:

- **Geographical Range:** Covers a small area (up to a few kilometers).
- **High Speed:** Typically operates at speeds of **100 Mbps to 1 Gbps** or more.
- **Ownership:** Usually owned and managed by a single organization.
- **Topology:** Common topologies include star, bus, and ring.
- **Communication Medium:** Wired (Ethernet) or Wireless (Wi-Fi).

Examples:

- Office networks for file sharing and printing.
- Home Wi-Fi networks.
- Computer labs in educational institutions.

Advantages:

1. **High Speed:** Fast data transfer rates.
2. **Low Cost:** Inexpensive setup and maintenance.
3. **Resource Sharing:** Easy sharing of files, printers, and applications.
4. **Security Control:** Easier to manage security policies.

Disadvantages:

1. **Limited Coverage:** Restricted to a small area.
2. **Scalability Issues:** Expanding beyond its designed range can be complex.

2. Metropolitan Area Network (MAN)

Definition:

A **MAN** is a network that spans a city or large campus, interconnecting multiple **LANs** within that area.

Key Features:

- **Geographical Range:** Covers an area between **5–50 kilometers** (city-level).
- **Moderate Speed:** Speeds range from **10 Mbps to 1 Gbps**.

- **Ownership:** May be owned by a service provider or a consortium of organizations.
- **Communication Medium:** Uses high-speed fiber optics, wireless technologies, or leased lines.

Examples:

- Cable TV networks in a city.
- University campuses connecting different departments.
- City-wide Wi-Fi services.

Advantages:

1. **Wider Coverage:** Supports networks across multiple buildings or city blocks.
2. **High-Speed Connectivity:** Suitable for applications like video conferencing and VoIP.
3. **Cost-Effective:** Provides shared connectivity for organizations within the region.

Disadvantages:

1. **Higher Costs:** More expensive than LANs due to infrastructure requirements.
2. **Complex Management:** Requires skilled personnel to manage interconnections and routing.

3. Wide Area Network (WAN)

Definition:

A **WAN** is a network that spans a large geographical area, such as a country or even globally, and connects multiple **LANs** and **MANs**.

Key Features:

- **Geographical Range:** Covers **100 kilometers to thousands of kilometers**.
- **Lower Speed:** Data transfer speeds are slower compared to LANs and MANs, typically **1 Mbps to 100 Mbps**, though modern WANs can achieve **1 Gbps** or higher.
- **Ownership:** Often leased or managed by third-party service providers.
- **Communication Medium:** Uses satellite links, leased telephone lines, fiber optics, and wireless technologies.

Examples:

- The **Internet** is the largest WAN.
- Corporate networks connecting offices in different cities or countries.
- Global cloud services.

Advantages:

1. **Global Coverage:** Connects devices across vast distances.
2. **Scalability:** Easily scales to include new regions or offices.
3. **Resource Sharing:** Enables communication and data sharing across countries.

Disadvantages:

1. **Higher Costs:** Requires significant investment in hardware, infrastructure, and service providers.
2. **Slower Speeds:** May face latency issues over long distances.
3. **Security Risks:** Greater vulnerability to hacking and data breaches.

Comparison Table: LAN, MAN, and WAN

Feature	**LAN**	**MAN**	**WAN**
Coverage Area	Small (home, office, campus)	Medium (city or large campus)	Large (country or worldwide)
Speed	High (100 Mbps to 1 Gbps+)	Moderate (10 Mbps to 1 Gbps)	Lower (1 Mbps to 1 Gbps)
Ownership	Private ownership.	Shared ownership.	Public or leased infrastructure.
Communication Medium	Ethernet cables, Wi-Fi.	Fiber optics, leased lines, Wi-Fi.	Fiber optics, satellites, microwaves.
Cost	Low setup and maintenance costs.	Moderate costs.	High costs for setup and maintenance.
Examples	Office networks, home Wi-Fi.	University campus networks, Cable TV.	Internet, cloud networks, MPLS links.

Types of services Connection Oriented and Connection Less

Types of Services in Computer Networks: Connection-Oriented vs. Connectionless

Network services are classified into **Connection-Oriented** and **Connectionless** communication models based on how data is transmitted between devices. These services define the way data packets are delivered over a network.

1. Connection-Oriented Service

Definition:

A **connection-oriented service** establishes a **dedicated connection** (virtual circuit) between the sender and receiver before transmitting data. It ensures that data is delivered in the **same order** and without loss.

Key Features:

- **Setup Phase:** Requires a connection to be established before communication starts.
- **Reliable Delivery:** Guarantees data integrity and sequence.

- **Error Checking:** Includes mechanisms for error detection, acknowledgment, and retransmission.
- **Flow Control:** Ensures data is sent at a rate that the receiver can handle.

Phases in Connection-Oriented Communication:

1. **Connection Establishment:** A session is created between sender and receiver.
2. **Data Transfer:** Data is sent in order, ensuring reliability.
3. **Connection Termination:** The connection is closed after data transfer is complete.

Examples:

- **TCP (Transmission Control Protocol):** Used for reliable communication in email, web browsing, and file transfers.
- **Virtual Circuits in ATM Networks (Asynchronous Transfer Mode):** Provide dedicated paths for voice or video calls.

Advantages:

1. **Reliable Communication:** Ensures complete and error-free data delivery.
2. **Order Preservation:** Data packets arrive in the order they were sent.
3. **Congestion Control:** Manages traffic effectively to avoid network overload.

Disadvantages:

1. **Setup Overhead:** Establishing and maintaining a connection adds delays.
2. **Higher Resource Usage:** Requires more bandwidth and memory to manage connections.
3. **Less Flexible:** Not suitable for applications needing quick and dynamic communication.

2. Connectionless Service

Definition:

A **connectionless service** does **not establish a dedicated connection** between the sender and receiver. Each data packet is transmitted independently and may take different routes to reach the destination.

Key Features:

- **No Setup Phase:** Data transmission starts immediately without establishing a connection.
- **Best-Effort Delivery:** Does not guarantee reliability, order, or error checking.
- **Faster Communication:** Suitable for real-time or low-latency applications.
- **Stateless Protocol:** No information about previous communications is stored.

Examples:

- **UDP (User Datagram Protocol):** Used for applications like video streaming, online gaming, and VoIP.
- **IP (Internet Protocol):** Routes packets in networks without connection establishment.

Advantages:

1. **Low Overhead:** Faster and more efficient for short, bursty data transmissions.
2. **Scalable:** Suitable for large networks like the Internet.
3. **Flexible Routing:** Packets can take alternative routes if paths are congested.

Disadvantages:

1. **Unreliable Delivery:** Packets may be lost, duplicated, or arrive out of order.
2. **No Error Recovery:** Errors must be handled at the application level.
3. **No Flow Control:** Cannot adapt to varying receiver speeds, leading to potential data loss.

Comparison Table: Connection-Oriented vs. Connectionless Services

Feature	**Connection-Oriented**	**Connectionless**
Setup Phase	Requires connection establishment.	No setup phase; sends data directly.
Reliability	Reliable, guarantees delivery.	Unreliable, no delivery guarantees.
Order of Data	Data arrives in order.	Data may arrive out of order.
Error Handling	Errors detected and corrected.	Errors not handled by the protocol.
Speed	Slower due to connection setup.	Faster due to no setup delays.
Overhead	High overhead (headers, acknowledgments).	Low overhead (minimal headers).
Flow Control	Provides flow control.	No flow control mechanisms.
Example Protocols	TCP, ATM.	UDP, IP.
Applications	Email, FTP, Web Browsing.	Video streaming, VoIP, Gaming.

Use Cases in Modern Technologies

- **Connection-Oriented Services:** Used where reliability and data integrity are critical, such as **file transfers**, **email**, and **secure payments**.
- **Connectionless Services:** Ideal for applications that prioritize **speed** over reliability, such as **live streaming**, **IoT sensors**, and **real-time analytics** in **Industry 4.0**.

Transmission Media: Twisted Pair, Co-axial Cable And Fiber Optic Cable

Transmission media refers to the physical pathways through which data is transmitted in a network. The choice of transmission media affects the speed, reliability, and cost of the communication system. Common types of transmission media include **Twisted Pair**, **Co-axial Cable**, and **Fiber Optic Cable**. Below is a detailed comparison and explanation of each.

1. Twisted Pair Cable

Definition:

A **twisted pair cable** consists of pairs of insulated copper wires that are twisted together to reduce electromagnetic interference (EMI) from external sources and from other pairs in the same cable.

Types:

- **Unshielded Twisted Pair (UTP):** No additional shielding.
- **Shielded Twisted Pair (STP):** Includes shielding to protect from interference.

Key Features:

- **Conductivity:** Uses copper wires to carry electrical signals.
- **Bandwidth:** Generally supports lower bandwidth (up to 1 Gbps) compared to other media types.
- **Distance Limitations:** Effective up to 100 meters for high-speed applications.
- **Cost:** Inexpensive and widely available.

Examples:

- **Ethernet Networks (LANs):** Commonly used in **Cat5e** and **Cat6** cables.
- **Telephony Networks:** Used in traditional landline phone systems.

Advantages:

1. **Cost-Effective:** Cheapest and easiest to install.
2. **Flexibility:** Easy to route through buildings and networks.
3. **Widely Used:** Used in many types of network setups, including local area networks (LANs).

Disadvantages:

1. **Susceptible to Interference:** Especially in UTP, which can lead to data loss.

2. **Limited Distance and Speed:** Suitable for short to medium distances with moderate data transfer speeds.
3. **Lower Bandwidth:** Limited to moderate data rates, not ideal for high-performance applications.

2. Co-axial Cable

Definition:

A **coaxial cable** consists of a central core wire made of copper, which is surrounded by an insulating layer, a metal shield (often braided or foil), and an outer insulating layer. The shield helps protect the signal from external interference.

Key Features:

- **Conductivity:** Copper core for signal transmission.
- **Bandwidth:** Higher bandwidth than twisted pair cables, supporting data rates from **10 Mbps to 10 Gbps**.
- **Distance Limitations:** Effective for medium-range distances (up to 500 meters).
- **Shielding:** Enhanced shielding protects against electromagnetic interference (EMI) and crosstalk.

Examples:

- **Cable Television (CATV):** Used for broadband internet and TV services.
- **Older Ethernet Networks (10BASE2 and 10BASE5).**

Advantages:

1. **Reduced Interference:** Shielding protects against external noise and crosstalk.
2. **Better Performance Over Distance:** Can cover longer distances than twisted pair cables.
3. **Higher Bandwidth:** Suitable for broadband applications and high-speed data transmission.

Disadvantages:

1. **Costlier than Twisted Pair:** More expensive and harder to install due to its rigidity.
2. **Bulky:** Less flexible compared to twisted pair cables, making installation in tight spaces difficult.
3. **Less Common:** Becoming obsolete for modern high-speed networks like Ethernet.

3. Fiber Optic Cable

Definition:

A **fiber optic cable** transmits data using light signals rather than electrical signals. It consists of glass or plastic fibers that carry light signals, with multiple layers of protective coatings.

Types:

- **Single-Mode Fiber (SMF):** Transmits light directly down the fiber with very little signal loss, ideal for long distances.
- **Multi-Mode Fiber (MMF):** Uses multiple light modes to transmit data; better for short to medium distances.

Key Features:

- **Conductivity:** Uses light to carry signals, which eliminates electrical interference.
- **Bandwidth:** Extremely high bandwidth (from **1 Gbps to 100 Gbps** or more).
- **Distance Limitations:** Can transmit data over **long distances** (up to several kilometers) without signal degradation.
- **Cost:** More expensive compared to twisted pair and coaxial cables, but costs have been decreasing.

Examples:

- **High-Speed Internet Connections:** Used in backbone networks and internet service provider (ISP) connections.
- **Data Centers:** To interconnect servers and storage devices.
- **Long-Distance Communication:** Telephone systems and undersea cables for international communication.

Advantages:

1. **High Bandwidth:** Ideal for large data transfer applications such as cloud computing and video streaming.
2. **Long Range:** Can transmit data over several kilometers without signal loss.
3. **Immunity to Interference:** Fiber optics are immune to electromagnetic interference (EMI).
4. **Future-Proof:** Provides high capacity for future network expansion.

Disadvantages:

1. **Expensive:** High initial cost for installation and materials.
2. **Fragility:** Glass fibers can break easily if not handled properly.
3. **Installation Complexity:** Requires specialized tools and expertise for installation.

Comparison Table: Twisted Pair, Co-axial Cable, and Fiber Optic Cable

Feature	Twisted Pair Cable	Co-axial Cable	Fiber Optic Cable
Material	Copper wires	Copper core, metal shield, insulation	Glass or plastic fibers
Bandwidth	Low to moderate (10 Mbps – 1 Gbps)	Moderate to high (10 Mbps – 10 Gbps)	Very high (1 Gbps – 100 Gbps or more)
Distance Limitation	Short (up to 100 meters)	Medium (up to 500 meters)	Long (several kilometers)
Susceptibility to EMI	High (especially in UTP)	Moderate (shielded against EMI)	None (immune to interference)
Cost	Low	Moderate	High
Installation	Easy and flexible	Rigid and less flexible	Complex and requires expertise
Example Use	LANs, telephone systems	Cable TV, older Ethernet	High-speed internet, data centers, undersea cables

PCET's

Pimpri Chinchwad University

School of Computer Application

PMC113A Computer Networks

Unit No. 02 Wireless Transmission

Introduction

Wireless transmission refers to the process of sending and receiving data, voice, and video signals over the air using electromagnetic waves instead of physical cables. This method is the foundation of wireless communication systems such as Wi-Fi, Bluetooth, cellular networks, and satellite communication. Below are the key aspects of wireless transmission:

1. Principles of Wireless Transmission

- **Medium**: Electromagnetic waves, including radio waves, microwaves, infrared, and visible light.
- **Frequency Range**: Ranges from a few kHz (kilohertz) to several GHz (gigahertz) depending on the technology.
- **Modulation**: Data is encoded onto carrier waves using techniques such as amplitude modulation (AM), frequency modulation (FM), or phase modulation (PM).

2. Components of Wireless Transmission

- **Transmitter**: Converts information into signals and transmits them.
- **Receiver**: Captures the transmitted signals and converts them back into usable data.
- **Antenna**: Sends and receives electromagnetic waves.
- **Medium**: Open space or air through which signals travel.

3. Types of Wireless Transmission

- **Radio Communication**: Used in AM/FM radio, television broadcasting, and mobile networks.
- **Microwave Transmission**: Used in long-distance communication, satellite links, and radar.
- **Infrared Transmission**: Short-range communication like remote controls and optical fiber.
- **Bluetooth**: Short-range wireless technology for connecting devices like headphones and keyboards.
- **Wi-Fi**: Medium-range communication for connecting to the internet and networks.
- **Cellular Networks**: Large-scale networks used in mobile phones (e.g., 4G, 5G).

4. Advantages

- Flexibility: No need for physical cables.
- Mobility: Users can move freely while staying connected.
- Scalability: Easily expand networks.
- Cost-Effective: Reduces infrastructure costs compared to wired systems.

5. Challenges

- **Interference**: Signals can be affected by other electromagnetic sources.
- **Security**: Wireless systems are susceptible to eavesdropping and unauthorized access.
- **Bandwidth Limitations**: Wireless spectrum is finite and shared among multiple users.
- **Signal Degradation**: Environmental factors like buildings, weather, and distance can weaken signals.

6. Applications

- **Telecommunication**: Cellular phones, satellite communication.
- **IoT (Internet of Things)**: Smart devices and edge computing.
- **Military and Défense**: Secure communication and surveillance.
- **Healthcare**: Wireless monitoring devices.
- **Industrial Automation**: Wireless control systems in Industry 4.0

Communication Satellites in Telephone System

Communication satellites play a crucial role in global telephone systems by facilitating long-distance and international voice communication. These satellites act as relay stations in space, receiving signals from ground stations, amplifying them, and transmitting them to the desired location on Earth.

Key Features of Satellite-Based Telephone Systems

1. **Global Coverage**: Satellites provide coverage to areas where terrestrial networks are impractical, such as remote locations and oceans.
2. **High Capacity**: Satellites can handle a large volume of simultaneous calls.
3. **Interconnectivity**: Seamlessly link users across different continents.
4. **Reliability**: Redundancy in satellite networks ensures consistent communication.

How It Works

1. **Uplink**: The ground station (or telephone exchange) sends the voice signal to the satellite.
2. **Satellite Relay**: The satellite processes the signal and amplifies it.
3. **Downlink**: The satellite transmits the signal to another ground station near the recipient.

4. **Distribution**: The ground station routes the signal to the local telephone network, which delivers it to the intended recipient.

Types of Satellite Orbits in Telephone Systems

1. **Geostationary Orbit (GEO)**:
   - Positioned at 35,786 km above the equator.
   - Appears stationary relative to Earth.
   - Ideal for fixed-location telephony but has a slight signal delay (latency ~250 ms round-trip).
2. **Medium Earth Orbit (MEO)**:
   - Located at altitudes of 2,000–20,000 km.
   - Offers lower latency and better signal quality than GEO satellites.
   - Used in some modern satellite telephone systems.
3. **Low Earth Orbit (LEO)**:
   - Positioned at 500–2,000 km altitude.
   - Very low latency and supports mobile satellite telephony.
   - Examples include constellations like Starlink and Iridium.

Advantages of Satellite Telephony

1. **Wide Coverage**: Connects even the most isolated regions.
2. **Scalability**: Accommodates growth in users without significant infrastructure changes.
3. **Disaster Recovery**: Provides communication during natural disasters when terrestrial networks fail.

Challenges

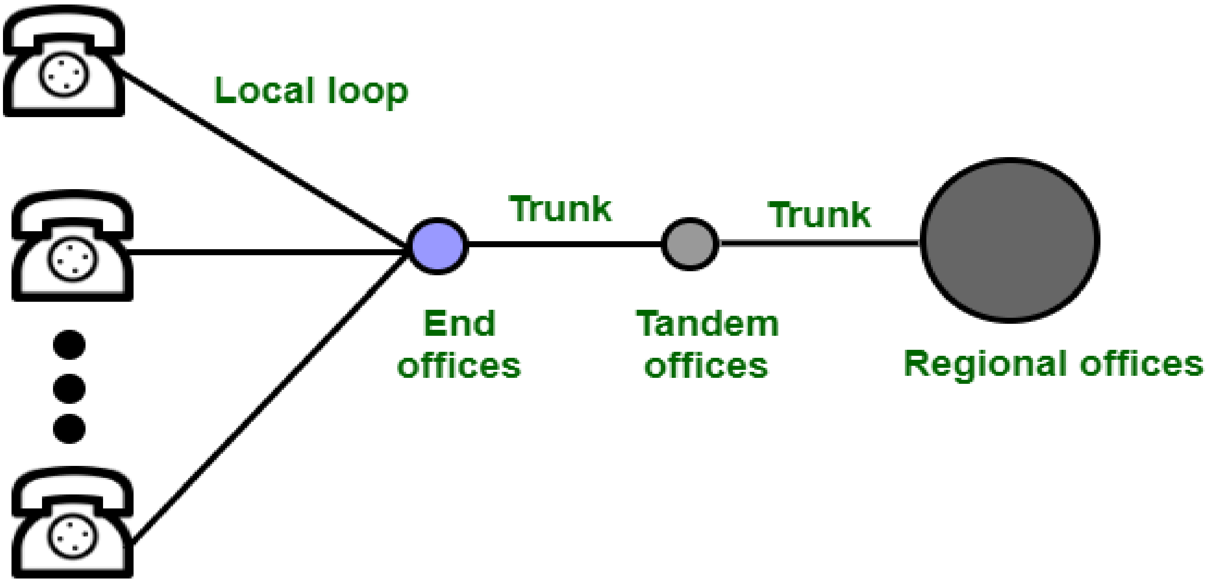
1. **Latency**: GEO satellites introduce noticeable delays in voice communication.
2. **Cost**: High deployment and maintenance costs for satellites and ground stations.
3. **Signal Interference**: Atmospheric conditions or physical obstacles can degrade signal quality.
4. **Power Consumption**: Mobile satellite phones require more power than traditional cell phones.

Applications

1. **Maritime Communication**: Connecting ships and offshore installations.
2. **Emergency Services**: Reliable communication during disasters.
3. **Remote Area Access**: Telephone services in rural or isolated regions.
4. **Military and Défense**: Secure communication for armed forces.

Telephone System

A telephone system is a network that allows people to make calls to other people, send faxes, and access the internet. It's made up of many components, including cell towers, base stations, and switching offices.



How it works

Analog and digital

Telephone systems use both analog and digital technologies. Local loops use analog twisted pair lines, while trunks between switching offices use digital Fiber optics.

Switching

Switches are used in central offices and long-distance centres to connect calls.

Time and timing

Switching, transmission, and billing equipment are synchronized to very close intervals.

Types of telephone systems

- **Plain Old Telephone System (POTS)**: This is the original analog voice communication system.
- **Cellular telephone system**: This is a wireless network that allows people to make calls while driving.
- **Digital telephone system**: This system converts analog signals to digital signals.

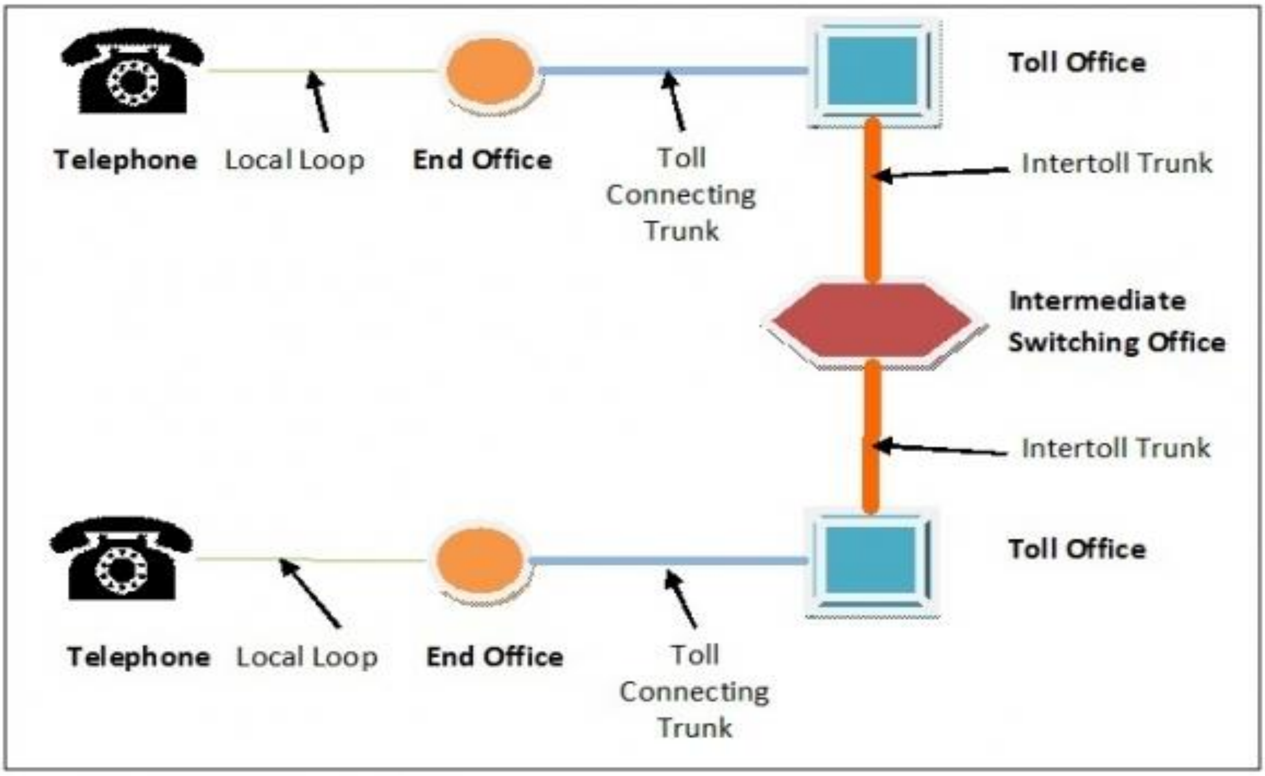
History

- The first crossbar system was demonstrated in 1919 by Televerket, the Swedish government-owned telephone company.

- The first commercially successful crossbar system was installed in Brooklyn, New York in 1938.

Structure of the Telephone System

The model can be diagrammatically represented as follows −



The telephone system model is organized as a highly redundant, multilevel hierarchy. It comprises of the following components –

- Telephone of the subscriber or end user
- End office − Local central office directly connected to end user at a distance of 1 – 10km.
- Local loop − A two-way connection between the telephone and the end office.
- Toll office − switching centres which are called tandem offices when located within the same local area.
- Toll connecting trunk − Lines that connect end offices with toll offices.
- Intermediate switching offices − Interconnected non-hierarchical switching offices for connecting toll offices.
- Inter toll trunk − Very high bandwidth channels that connect either two toll offices via intermediate switching offices.

Communication for different Transmissions

- Both caller and callee are attached to same end office −

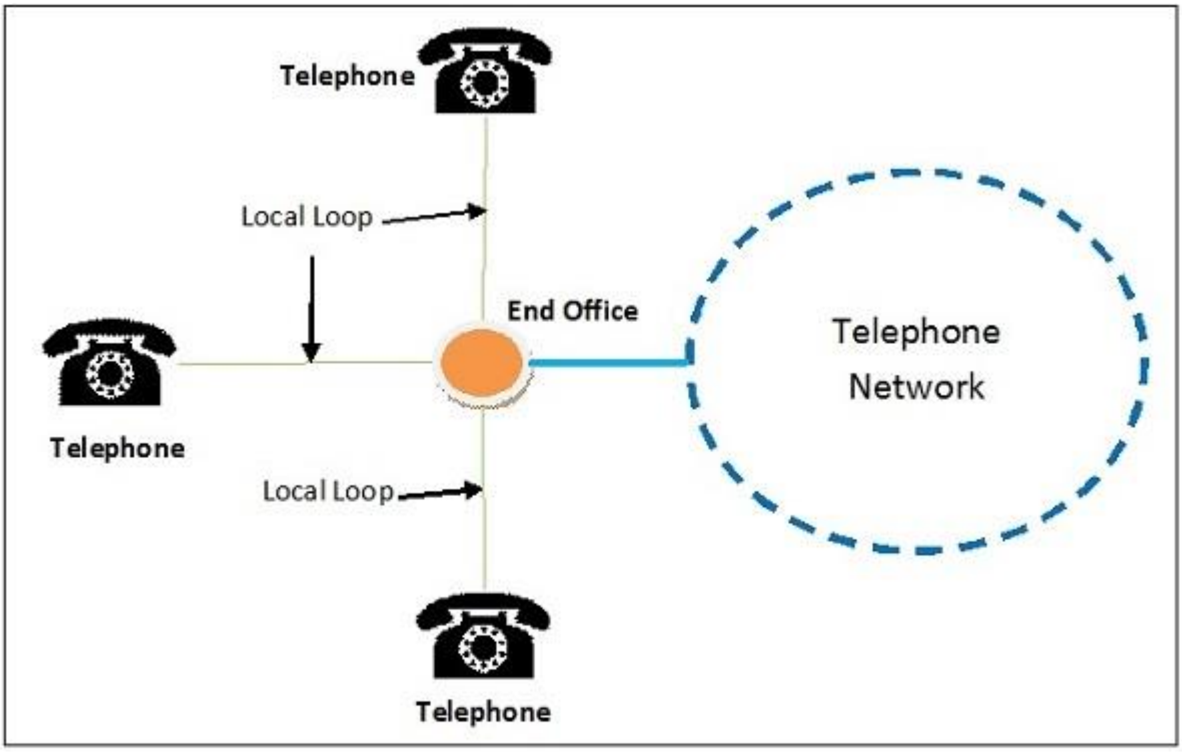
- In this case, a direct electrical connection is set up between the local loops of the subscribers by the switching mechanism of the end office.
- Caller and callee are attached to different end offices −
- In this case, the end office of the caller sets up a connection with one or more connected toll offices, which then performs the switching job. This again has two cases −
  - If the end offices of the caller and the callee have a common toll office, then the toll office establishes a connection within itself.
  - If there are no common toll office between the caller and callee, then a path is established between the different toll offices, through intermediate switching office via intertoll trunks.

Transmission Media Used

- Local loop − analog twisted pair cables.
- Toll connecting trunk − fibre optic links
- Between switching offices − fibre optic cables and microwaves

The Local Loop

In a telephone system, the local loop is a two-wire connection between the subscriber's house and the end office of the telephone company. It is commonly referred to as the "last mile" of the telephone network. The loop may run from 1km to as far as 10 km.



Composition

Traditionally, local loops are composed of twisted pair copper cables. The old local loops have several limitations − narrow bandwidth, high attenuation, distortion of symbols, crosstalk's etc. In recent times, copper wires are being replaced by fiber optic cables for faster and more accurate performance. Installation of fiber cables is popularly known as FttH (Fibre to the Home).

Data Transmission

Telephone networks are widely used for data transmission along with voice, calling for connecting computers as end users, along with telephones. The digital data of the computers need to be converted to analog signals in the telephone lines. So, telephone modems are used. Another method is to use digital loops, popularly called digital subscriber lines.

Trunks and Multiplexing and Switching

Trunks

Definition

Trunks are a large-bandwidth communication channels connecting switching centres, forming the core of the telephone network. A trunk comprises of a cluster of broadcast frequencies that are suitable for long haul communication of many voice and data signals simultaneously.

Features

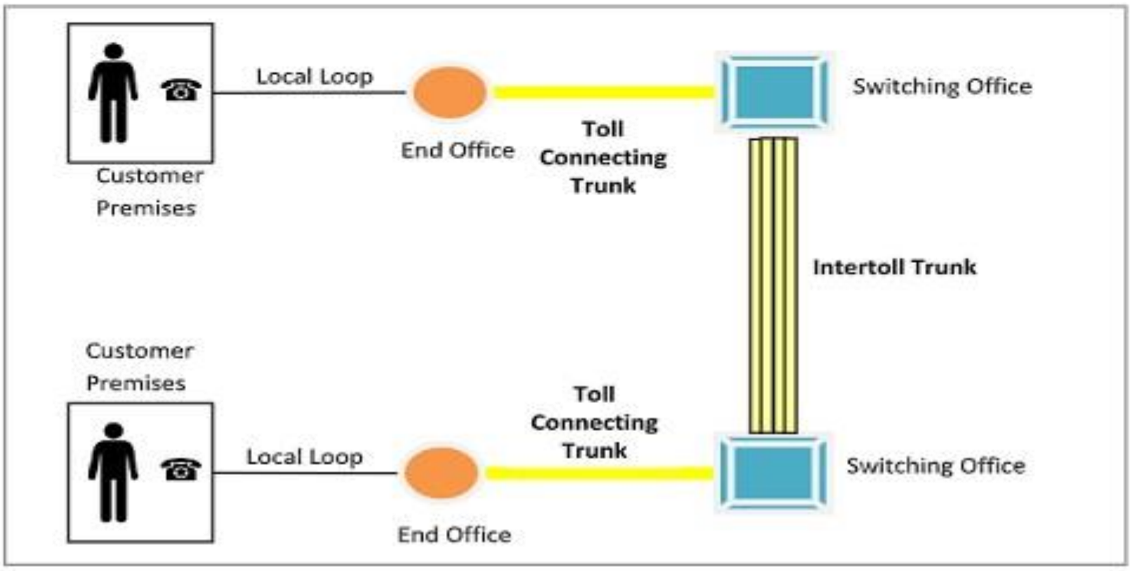
The characteristic features of a trunk are −

- They carry digital information as opposed to analog signals carried by local loops.
- They are designed for both voice and data transmission.
- They can carry millions of calls simultaneously over long distances.
- They have very high bandwidth.
- They generally comprise of a cluster of fiber optic cables bundled together to maximize bandwidth.
- Another method of achieving high bandwidth is to use a single high-capacity link that can carry many signals multiplexed together.

Trunks used in Telephone Networks

Toll connecting Trunk − Lines that connect end offices with toll offices. Fiber optic cables are used here.

Intertoll Trunk − Very high bandwidth channels that connect either two toll offices via intermediate switching offices. Fiber optic cables and microwaves are used here.



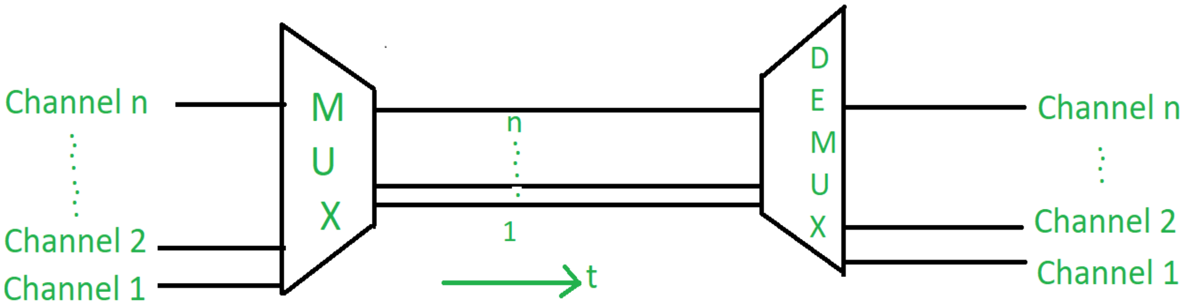
Multiplexing in Trunks

Trunks are required to carry a huge number of calls simultaneously. An option is to provide a large number of low-bandwidth trunks. However, the cost of installation and maintenance of a low-bandwidth trunk is almost same as cost of installing a higher bandwidth counterpart. So, installing a large number of low-bandwidth trunks is not an economic decision.

A feasible solution is to use a high bandwidth channel and use multiplexing for simultaneous transmission of several signals. Multiplexing is a method of combining more than one signal over a shared medium. The commonly used multiplexing techniques in trunks are time division multiplexing (TDM) and frequency division multiplexing (FDM). In TDM, the users are allowed the total available bandwidth on time sharing basis. In FDM, signals of different frequencies are combined for concurrent transmission.

Frequency Division Multiplexing

In this, a number of signals are transmitted at the same time, and each source transfers its signals in the allotted frequency range. There is a suitable frequency gap between the 2 adjacent signals to avoid over-lapping. Since the signals are transmitted in the allotted frequencies so this decreases the probability of collision. The frequency spectrum is divided into several logical channels, in which every user feels that they possess a particular bandwidth. A number of signals are sent simultaneously at the same time allocating separate frequency bands or channels to each signal. It is used in radio and TV transmission. Therefore, to avoid interference between two successive channels Guard bands are used.

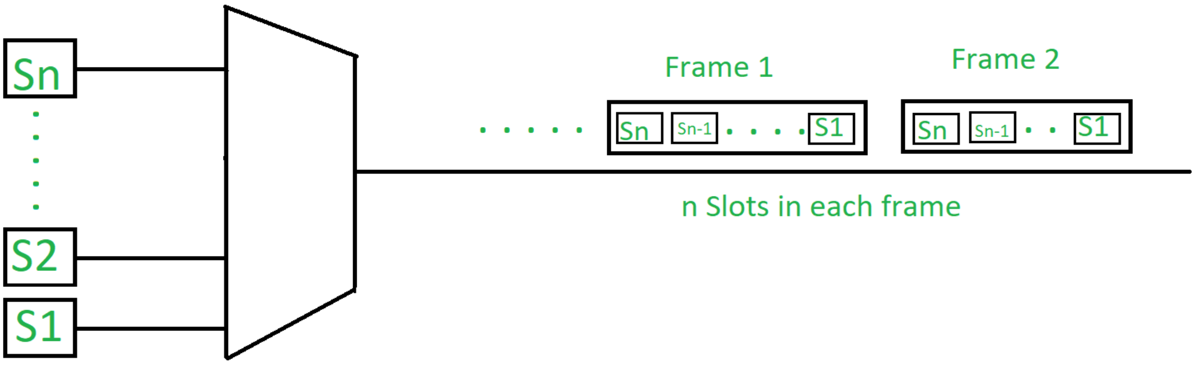


Application of FDM:

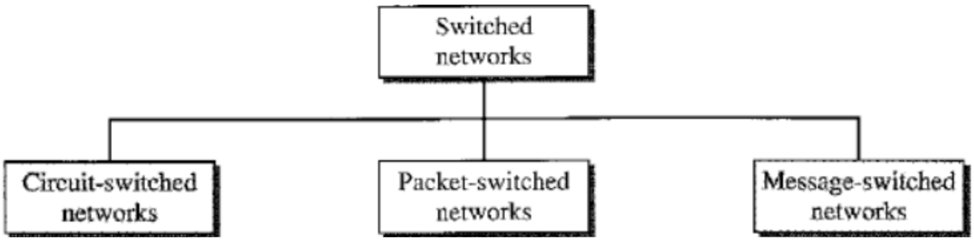
1. In the first generation of mobile phones, FDM was used.
2. The use of FDM in television broadcasting
3. FDM is used to broadcast FM and AM radio frequencies.

Time Division Multiplexing

This happens when the data transmission rate of media is greater than that of the source, and each signal is allotted a definite amount of time. These slots are so small that all transmissions appear to be parallel. In frequency division multiplexing all the signals operate at the same time with different frequencies, but in time-division multiplexing, all the signals operate with the same frequency at different times.



Switching:



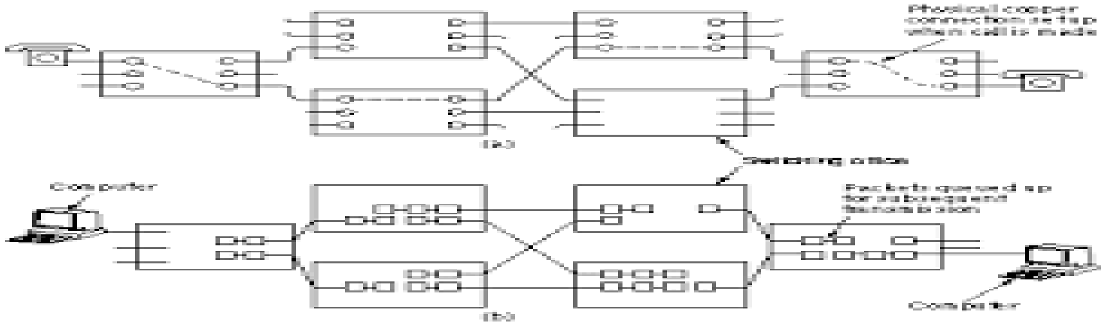
The phone system is divided into two principal parts: outside plant (the local loops and trunks, since they are physically outside the switching offices) and inside plant (the switches), which are inside the switching offices.

Two different switching techniques are used nowadays: circuit switching and packet switching.

Circuit Switching:

When you or your computer places a telephone call, the switching equipment within the telephone system seeks out a physical path all the way from your telephone to the receiver's telephone. This technique is called circuit switching. A switching technology that establishes an electrical connection between stations using a dedicated path.

Each of the six rectangles represents carrier **switching office (end office, toll office etc.).** When a call passes through a switching office, a physical connection is (conceptually) established between the line on which the call came in and one of the output lines, as shown below:


Physical copper connection set up when call is made
(a)
Switching office
Computer
Packets queued up for subsequent transmission
(b)
Computer

Parts of the physical path between the two telephones may, in fact, be microwave or fiber links onto which thousands of calls multiplexed.

An important property of circuit switching is the need to set up an end-to-end path before any data can be sent. The elapsed time between the end of dialling and the start of ringing can be easily be 10 sec, more on long distance or international calls.

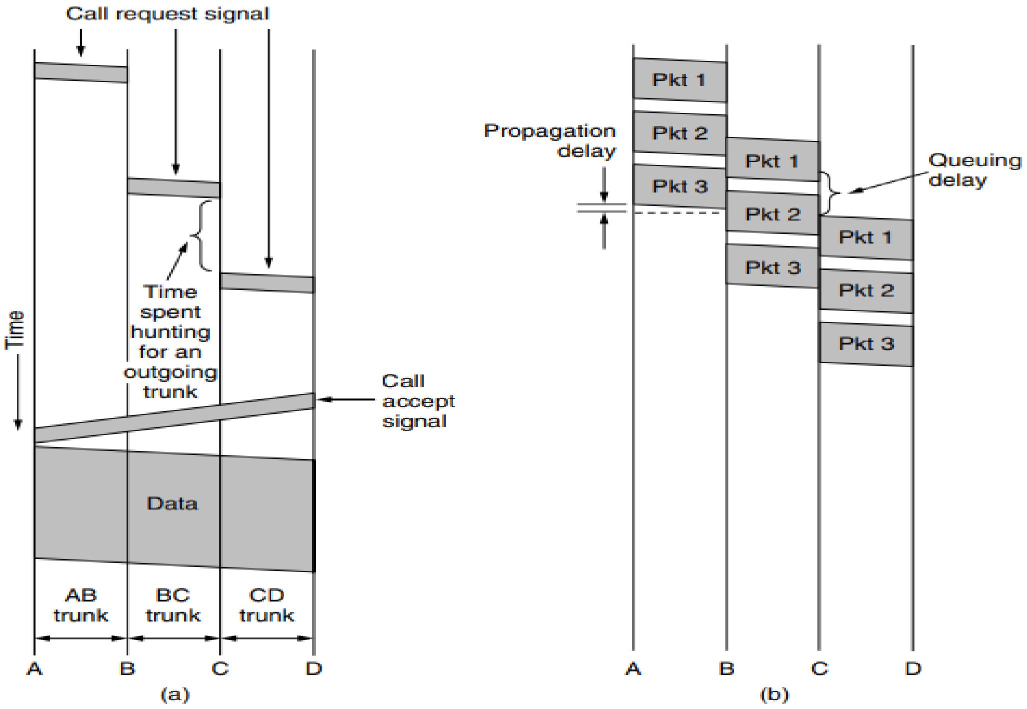


Figure 2-43. Timing of events in (a) circuit switching, (b) packet switching.

Packet Switching:

- Packet switching networks place a tight on upper limit on block size, allowing packets to be buffered in router main memory instead of on disk.
- By making sure that no user can monopolize any transmission line very long, packets switching networks are well suited for handling interactive traffic.
- A further advantage of packet switching over message switching is shown the first packet of a multi-packet message can be forwarded before the second one has fully arrived, reducing delay and improving throughput. For these reasons, computer networks are usually packet switched, occasionally circuit switched, but never message switched.
- Circuit switching requires that a circuit be set up end-to-end before communication begins. Packet switching does not require any advance setup. Result of the connection setup with circuit switching is the reservation of bandwidth all the way from the sender to receiver. All packets follow this path.
- With packet switching there is no path, so different packets can follow different paths depending on network conditions at the time they are sent. Packet switching is more fault tolerant than circuit switching.

- If a circuit has been reserved for a particular user and there is no traffic to send the bandwidth of that circuit is wasted. It cannot be used for other traffic. Packet switching does not waste bandwidth and thus is more efficient from a system-wide perspective.

Data Link Layer: Design Issues

Data-link layer is the second layer after the physical layer. The data link layer is responsible for maintaining the data link between two hosts or nodes.

Before going through the design issues in the data link layer. Some of its sub-layers and their functions are as following below.

The data link layer is divided into two sub-layers:

Logical Link Control Sub-layer (LLC) –

Provides the logic for the data link, thus it controls the synchronization, flow control, and error checking functions of the data link layer. Functions are –

(i) Error Recovery.

(ii) It performs the flow control operations.

(iii) User addressing.

Media Access Control Sub-layer (MAC) –

It is the second sub-layer of data-link layer. It controls the flow and multiplexing for transmission medium. Transmission of data packets is controlled by this layer. This layer is responsible for sending the data over the network interface card.

Functions are –

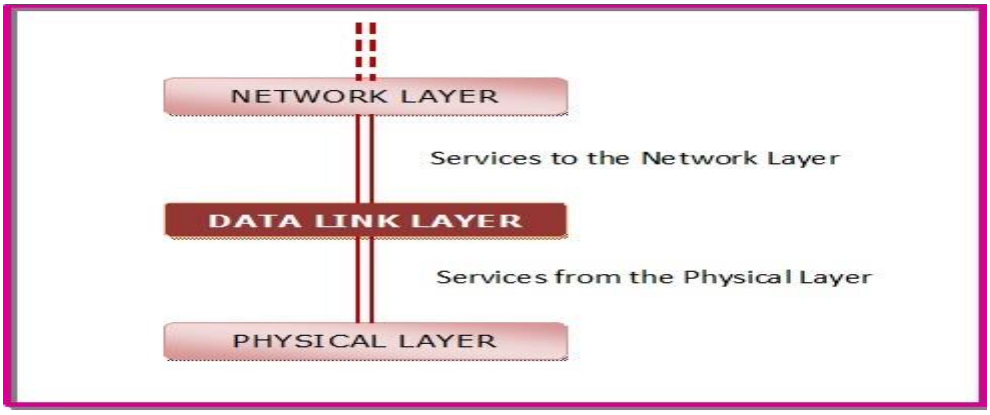
(i) To perform the control of access to media.

(ii) It performs the unique addressing to stations directly connected to LAN.

(iii) Detection of errors.

Design issues with data link layer are:

1. Services provided to the network layer –


NETWORK LAYER
Services to the Network Layer
DATA LINK LAYER
Services from the Physical Layer
PHYSICAL LAYER

The data link layer act as a service interface to the network layer. The principle service is transferring data from network layer on sending machine to the network layer on destination machine. This transfer also takes place via DLL (Data link-layer).

It provides three types of services:

1. Unacknowlwdged and connectionless services.
2. Acknowledged and connectionless services.
3. Acknowledged and connection-oriented services

1 Unacknowlwdged and connectionless services

- Here the sender machine sends the independent frames without any acknowledgement from the sender.
- There is no logical connection established.

2. Acknowledged and connectionless services

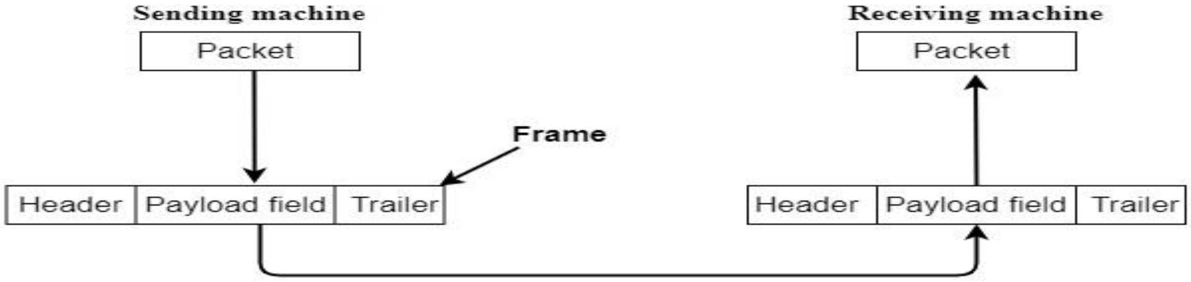
- There is no logical connection between sender and receiver established.
- Each frame is acknowledged by the receiver.
- If the frame didn't reach the receiver in a specific time interval it has to be sent again.
- It is very useful in wireless systems.

3. Acknowledged and connection-oriented services

- A logical connection is established between sender and receiver before data is trimester.
- Each frame is numbered so the receiver can ensure all frames have arrived and exactly once.

2. Frame synchronization –

- The source machine sends data in the form of blocks called frames to the destination machine. The starting and ending of each frame should be identified so that the frame can be recognized by the destination machine.



3. Flow control –

- Flow control is done to prevent the flow of data frame at the receiver end. The source machine must not send data frames at a rate faster than the capacity of destination machine to accept them.

4. Error control –

Error control is done to prevent duplication of frames. The errors introduced during transmission from source to destination machines must be detected and corrected at the destination machine.

- Dealing with transmission errors
- Sending acknowledgement frames in reliable connections
- Retransmitting lost frames
- Identifying duplicate frames and deleting them
- Controlling access to shared channels in case of broadcasting

Error Detection and Correction

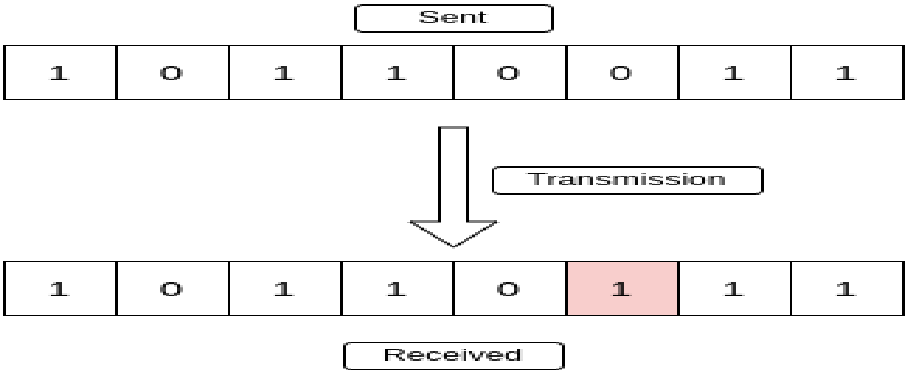
Error is a condition when the receiver's information does not match the sender's. Digital signals suffer from noise during transmission that can introduce errors in the binary bits traveling from sender to receiver. That means a 0 bit may change to 1 or a 1 bit may change to 0.

Data may get scrambled by noise or get corrupted whenever a message is transmitted. To prevent such errors, error-detection codes are added as extra data to digital messages. This helps in detecting any errors that may have occurred during message transmission.

Types of Errors

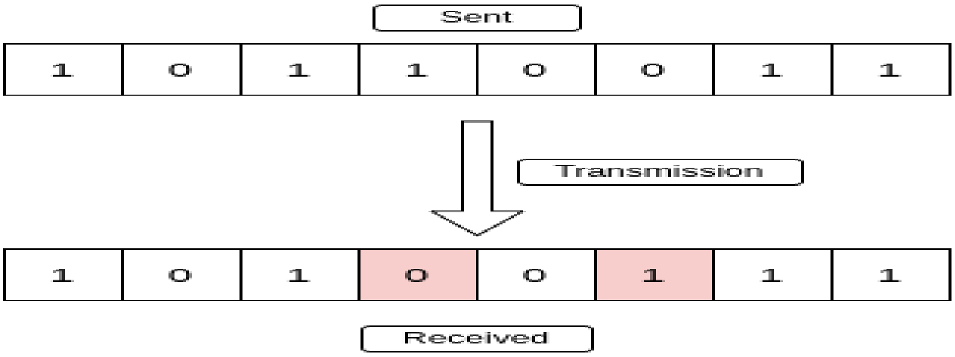
Single-Bit Error

A single-bit error refers to a type of data transmission error that occurs when one bit (i.e., a single binary digit) of a transmitted data unit is altered during transmission, resulting in an incorrect or corrupted data unit.



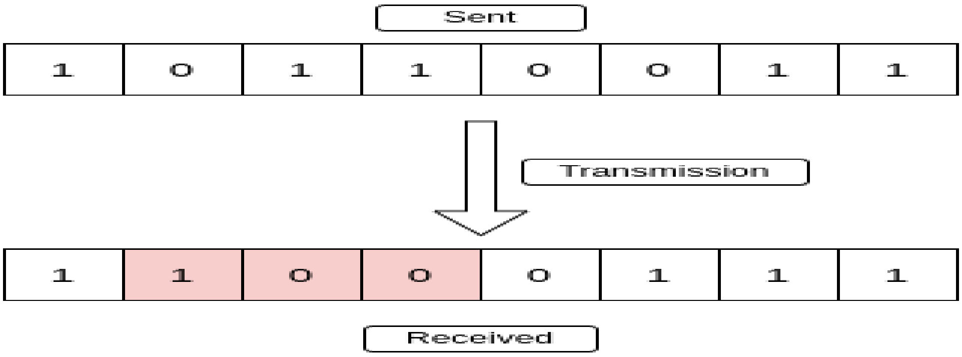
Multiple-Bit Error

A multiple-bit error is an error type that arises when more than one bit in a data transmission is affected. Although multiple-bit errors are relatively rare when compared to single-bit errors, they can still occur, particularly in high-noise or high-interference digital environments.



Burst Error

When several consecutive bits are flipped mistakenly in digital transmission, it creates a burst error. This error causes a sequence of consecutive incorrect values.



Error Detection Methods

To detect errors, a common technique is to introduce redundancy bits that provide additional information. Various techniques for error detection include:

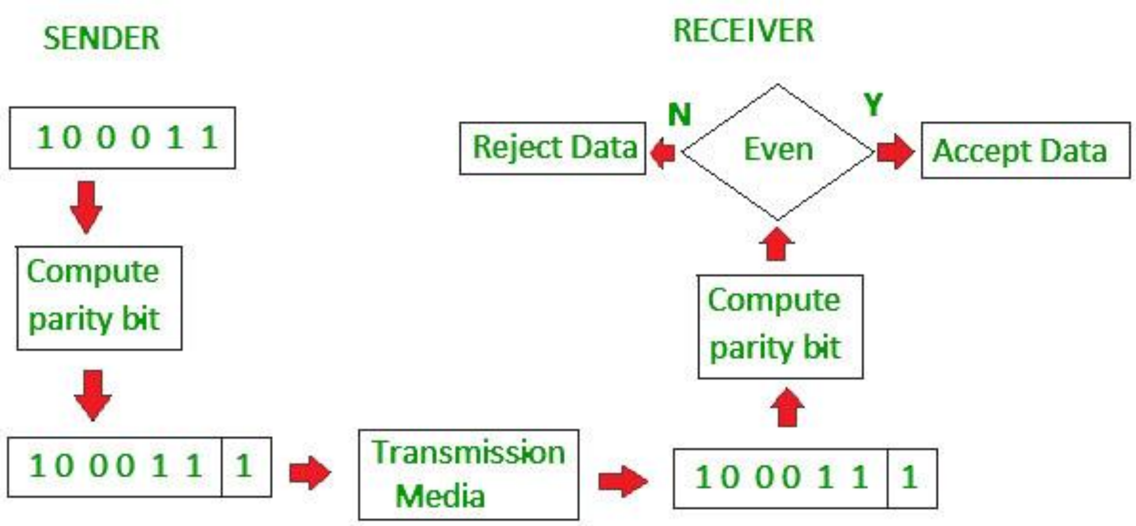
1. Simple Parity Check
2. Two-Dimensional Parity Check
3. Cyclic Redundancy Check (CRC)

Simple Parity Check

Simple-bit parity is a simple error detection method that involves adding an extra bit to a data transmission. It works as:

- 1 is added to the block if it contains an odd number of 1's, and
- 0 is added if it contains an even number of 1's

This scheme makes the total number of 1's even, that is why it is called even parity checking.



Advantages of Simple Parity Check

- Simple parity check can detect all single bit error.
- Simple parity check can detect an odd number of errors.
- **Implementation**: Simple Parity Check is easy to implement in both hardware and software.
- **Minimal Extra Data**: Only one additional bit (the parity bit) is added per data unit (e.g., per byte).
- **Fast Error Detection**: The process of calculating and checking the parity bit is quick, which allows for rapid error detection without significant delay in data processing or communication.
- **Single-Bit Error Detection**: It can effectively detect single-bit errors within a data unit, providing a basic level of error detection for relatively low-error environments.

Disadvantages of Simple Parity Check

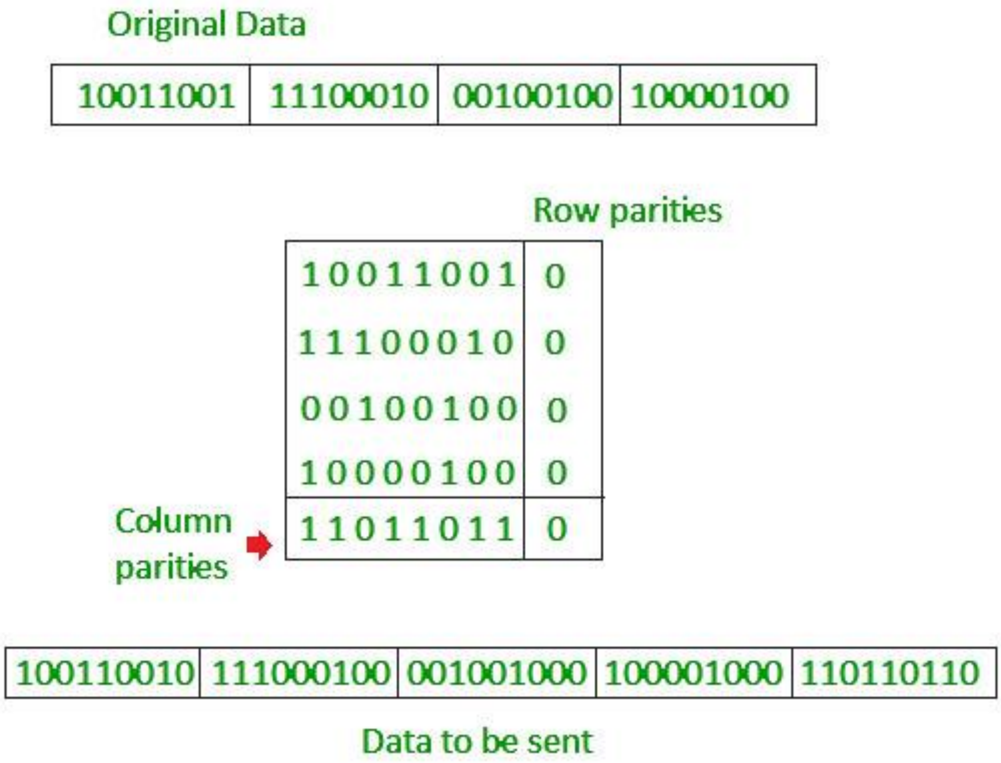
- Single Parity check is not able to detect even no. of bit error.
- **For example,** the Data to be transmitted is **101010**. Codeword transmitted to the receiver is 1010101.

  Let’s assume that during transmission, two of the bits of code word flipped to 111101.

  On receiving the code word, the receiver finds the no. of ones to be even and ence **no error,** *which is a wrong assumption.*

Two-Dimensional Parity Check

Two-dimensional Parity check bits are calculated for each row, which is equivalent to a simple parity check bit. Parity check bits are also calculated for all columns, then both are sent along with the data. At the receiving end, these are compared with the parity bits calculated on the received data.



Original Data

10011001	11100010	00100100	10000100

Row parities

	10011001	0
	11100010	0
	00100100	0
	10000100	0
Column parities	11011011	0

100110010	111000100	001001000	100001000	110110110

Data to be sent

Advantages of Two-Dimensional Parity Check

- Two-Dimensional Parity Check can detect and correct all single bit error.
- Two-Dimensional Parity Check can detect two- or three-bit error that occur anywhere in the matrix.

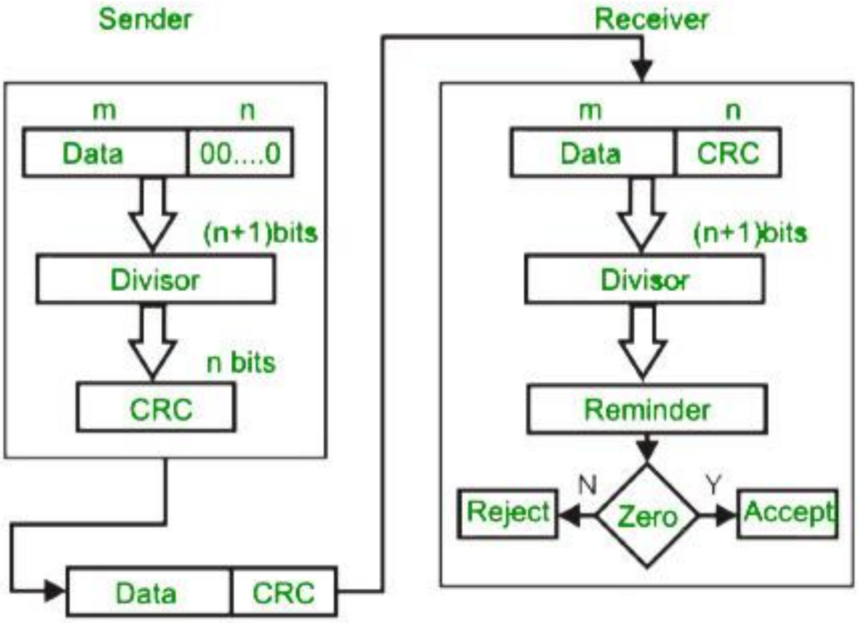
Disadvantages of Two-Dimensional Parity Check

- Two-Dimensional Parity Check cannot correct two- or three-bit error. It can only detect two- or three-bit error.
- If we have a error in the parity bit then this scheme will not work.

Cyclic Redundancy Check (CRC)

- Unlike the checksum scheme, which is based on addition, CRC is based on binary division.
- In CRC, a sequence of redundant bits, called cyclic redundancy check bits, are appended to the end of the data unit so that the resulting data unit becomes exactly divisible by a second, predetermined binary number.
- At the destination, the incoming data unit is divided by the same number. If at this step there is no remainder, the data unit is assumed to be correct and is therefore accepted.

- A remainder indicates that the data unit has been damaged in transit and therefore must be rejected.



CRC Working

We have given dataword of length n and divisor of length k.

Step 1: Append (k-1) zero's to the original message

Step 2: Perform modulo 2 division

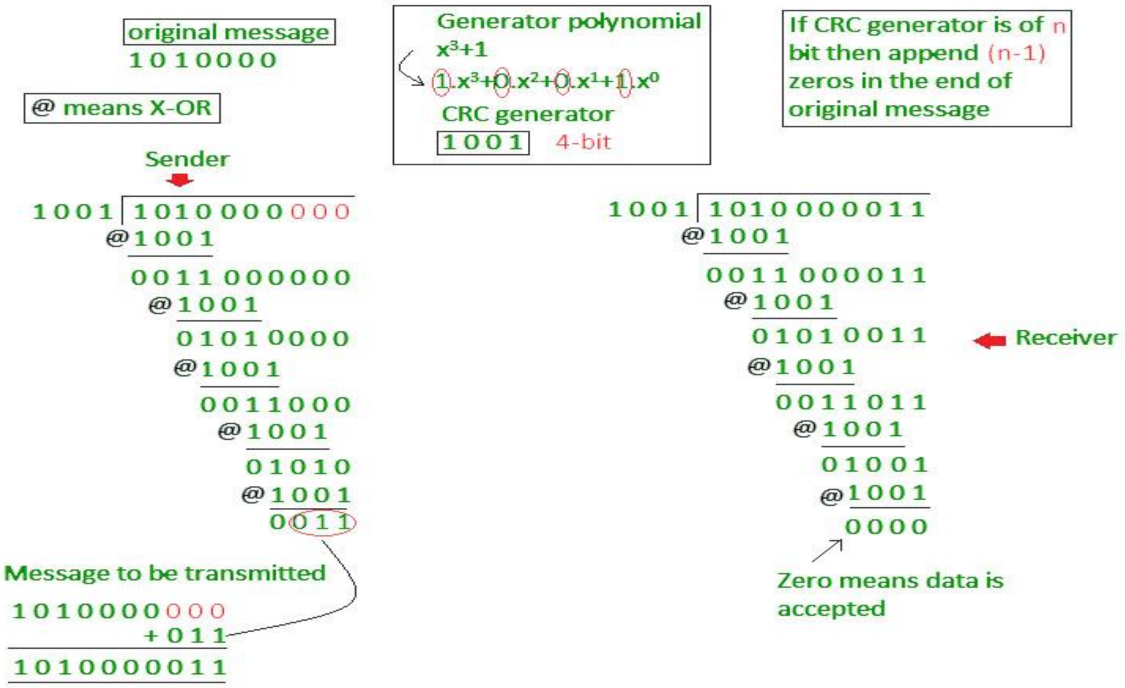
Step 3: Remainder of division = CRC

Step 4: Code word = Data with append k-1 zero's + CRC

Note:

- CRC must be k-1 bits
- Length of Code word = n+k-1 bits

Example: Let's data to be send is 1010000 and divisor in the form of polynomial is x3+1. CRC method discussed below.



Advantages of Error Detection

- **Increased Data Reliability:** Error detection ensures that the data transmitted over the network is reliable, accurate, and free from errors. This ensures that the recipient receives the same data that was transmitted by the sender.
- **Improved Network Performance:** Error detection mechanisms can help to identify and isolate network issues that are causing errors. This can help to improve the overall performance of the network and reduce downtime.
- **Enhanced Data Security:** Error detection can also help to ensure that the data transmitted over the network is secure and has not been tampered with.

Disadvantages of Error Detection

- **Overhead**: Error detection requires additional resources and processing power, which can lead to increased overhead on the network. This can result in slower network performance and increased latency.
- **False Positives:** Error detection mechanisms can sometimes generate false positives, which can result in unnecessary retransmission of data. This can further increase the overhead on the network.
- **Limited Error Correction:** Error detection can only identify errors but cannot correct them. This means that the recipient must rely on the sender to retransmit the data, which can lead to further delays and increased network overhead.

PCET's

Pimpri Chinchwad University

School of Computer Application

PMC113A Computer Networks

Unit No. 03 Elementary Data Link Protocols

Introduction:

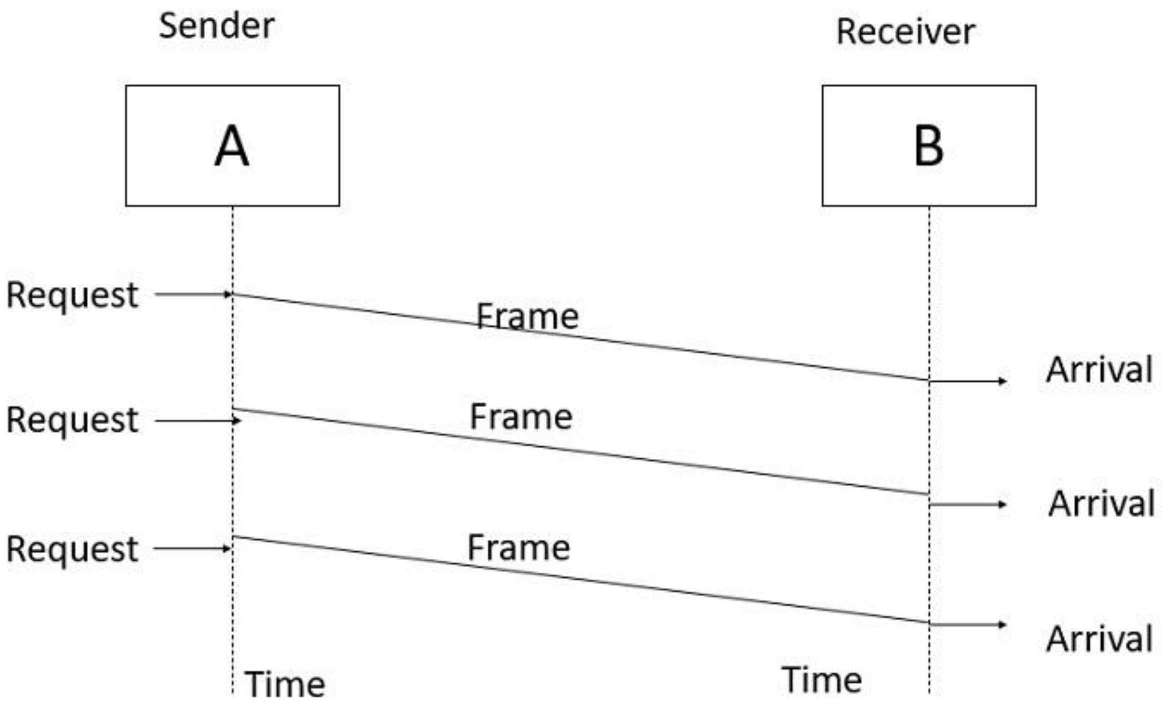
Elementary Data Link Protocols are fundamental methods used in the data link layer of the OSI model to ensure reliable communication between directly connected devices. These protocols manage framing, error detection, and flow control in a simple and efficient manner.

Types of Elementary Data Link Protocols

1. Unrestricted Simplex Protocol

Data transmitting is carried out in one direction only. The transmission (Tx) and receiving (Rx) are always ready and the processing time can be ignored. In this protocol, infinite buffer space is available, and no errors are occurring that is no damage frames and no lost frames.

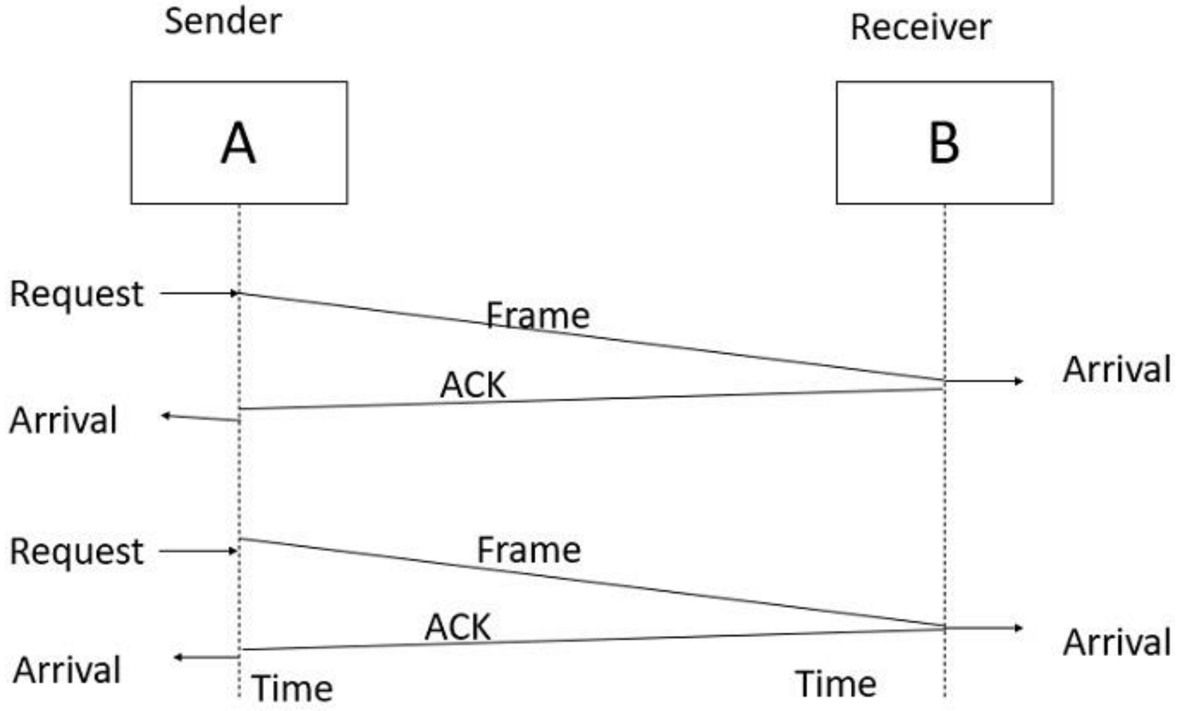
The Unrestricted Simplex Protocol is diagrammatically represented as follows –



2. Stop-and-Wait Protocol

In this protocol we assume that data is transmitted in one direction only. No error occurs; the receiver can only process the received information at finite rate. These assumptions imply that the transmitter cannot send frames at rate faster than the receiver can process them.

The main problem here is how to prevent the sender from flooding the receiver. The general solution for this problem is to have the receiver send some sort of feedback to sender, the process is as follows −



3. Stop-and-Wait ARQ (Automatic Repeat reQuest)

Extends the **Stop-and-Wait Protocol** by handling **error detection** using ACK (Acknowledgment) and NAK (Negative Acknowledgment).

If the sender does not receive an ACK within a timeout period, it **resends** the frame.

Uses mechanisms like **checksums** and **timers**.

4. Sliding Window Protocols

Allow multiple frames to be sent before requiring an acknowledgment.

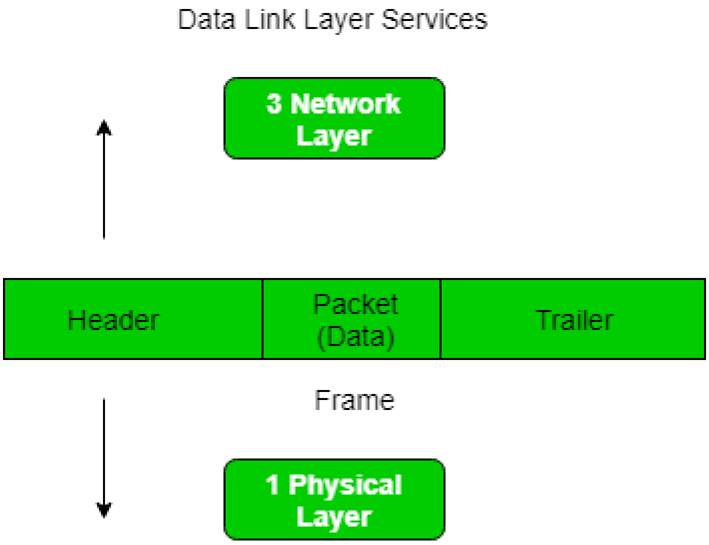
Uses a **window size** to control the number of unacknowledged frames.

Two common versions:

Go-Back-N ARQ → If an error occurs, all subsequent frames are retransmitted.

Selective Repeat ARQ → Only the erroneous frames are retransmitted.

Framing: In an elementary data link protocol, "framing" refers to the process of dividing a stream of data bits into smaller, manageable units called "frames," which include header information to identify the sender and receiver, allowing for reliable transmission and error detection at the data link layer of a network; essentially, it's how data is organized into distinct blocks to be sent across a communication channel, ensuring the receiver can interpret the information correctly.



At the data link layer, it extracts the message from the sender and provides it to the receiver by providing the sender's and receiver's addresses. The advantage of using frames is that data is broken up into recoverable chunks that can easily be checked for corruption.

The process of dividing the data into frames and reassembling it is transparent to the user and is handled by the data link layer.

Framing is an important aspect of data link layer protocol design because it allows the transmission of data to be organized and controlled, ensuring that the data is delivered accurately and efficiently.

Problems in Framing

- **Detecting start of the frame:** When a frame is transmitted, every station must be able to detect it. Station detects frames by looking out for a special sequence of bits that marks the beginning of the frame i.e. SFD (Starting Frame Delimiter).
- **How does the station detect a frame:** Every station listens to link for SFD pattern through a sequential circuit. If SFD is detected, sequential circuit alerts station. Station checks destination address to accept or reject frame.
- **Detecting end of frame:** When to stop reading the frame.
- **Handling errors:** Framing errors may occur due to noise or other transmission errors, which can cause a station to misinterpret the frame. Therefore, error detection and correction mechanisms, such as cyclic redundancy check (CRC), are used to ensure the integrity of the frame.

- **Framing overhead:** Every frame has a header and a trailer that contains control information such as source and destination address, error detection code, and other protocol-related information. This overhead reduces the available bandwidth for data transmission, especially for small-sized frames.
- **Framing incompatibility:** Different networking devices and protocols may use different framing methods, which can lead to framing incompatibility issues. For example, if a device using one framing method sends data to a device using a different framing method, the receiving device may not be able to correctly interpret the frame.
- **Framing synchronization:** Stations must be synchronized with each other to avoid collisions and ensure reliable communication. Synchronization requires that all stations agree on the frame boundaries and timing, which can be challenging in complex networks with many devices and varying traffic loads.
- **Framing efficiency:** Framing should be designed to minimize the amount of data overhead while maximizing the available bandwidth for data transmission. Inefficient framing methods can lead to lower network performance and higher latency.

Types of framing

There are two types of framing:

1. Fixed-size: The frame is of fixed size and there is no need to provide boundaries to the frame, the length of the frame itself acts as a delimiter.

- **Drawback:** It suffers from internal fragmentation if the data size is less than the frame size
- **Solution:** Padding (**Padding** in framing refers to the addition of extra bits or bytes to a frame to meet a required minimum size.)

2. Variable size: In this, there is a need to define the end of the frame as well as the beginning of the next frame to distinguish. This can be done in two ways:

1. **Length field** – We can introduce a length field in the frame to indicate the length of the frame. Used in **Ethernet (802.3)**. The problem with this is that sometimes the length field might get corrupted.
2. **End Delimiter (ED)** – We can introduce an ED (pattern) to indicate the end of the frame. Used in **Token Ring**.

Causes of Error:

Errors in elementary data link protocols within computer networks primarily occur due to transmission impairments like signal attenuation, distortion, and noise during data transmission over the physical medium, leading to bit flips or corrupted data where individual bits within a frame are altered or lost, causing the receiver to interpret the information incorrectly.

Key causes of errors in data link protocols:

- **Signal Attenuation:**

As signals travel through the transmission medium, their strength weakens, making them susceptible to noise interference at the receiver.

- **Signal Distortion:**

The signal shape can be altered during transmission due to different propagation speeds within the medium, causing data corruption.

- **Noise:**

Random electrical interference from external sources like other devices or environmental factors can disrupt the received signal, leading to bit errors.

- **Synchronization Issues:**

Improper timing synchronization between sender and receiver can cause frame delimitation problems, where the receiver may not correctly identify the start and end of a frame.

- **Physical Layer impairments:**

Issues with the physical transmission medium itself, such as cable damage, poor connection quality, or electromagnetic interference can introduce errors.

- **Collisions in shared media:**

In networks using shared media like Ethernet, multiple devices transmitting data simultaneously can cause collisions, resulting in data loss or corruption.

How data link protocols handle errors:

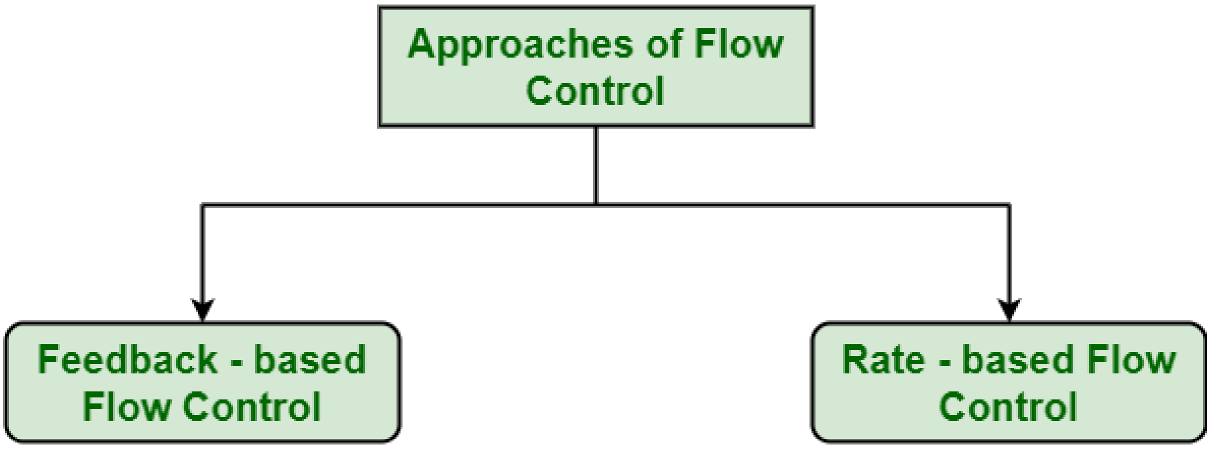
- **Error Detection Techniques:**
    - **Checksums:** A simple method where a checksum value is calculated based on the data and appended to the frame, allowing the receiver to verify data integrity.
    - **Cyclic Redundancy Check (CRC):** A more robust error detection technique using polynomial division to generate a check value that is more reliable in detecting burst errors.
- **Error Correction Techniques:**
    - **Automatic Repeat Request (ARQ):** A common method where the receiver sends a negative acknowledgment (NAK) if an error is detected, prompting the sender to retransmit the corrupted frame.

- **Forward Error Correction (FEC):** Redundant data is added to the transmitted frame, allowing the receiver to reconstruct the original data even if some bits are corrupted.

Flow Control in Elementary Data Link Protocol

Flow control is a technique that generally observes the proper flow of data from sender to receiver. It is very essential because it is possible for sender to transmit data or information at very fast rate and hence receiver can receive this information and process it. This can happen only if receiver has very high load of traffic as compared to sender, or if receiver has power of processing less as compared to sender. Flow control is basically a technique that gives permission to two of stations that are working and processing at different speeds to just communicate with one another. Flow control in Data Link Layer simply restricts and coordinates number of frames or amount of data sender can send just before it waits for an acknowledgement from receiver. Flow control is actually set of procedures that explains sender about how much data or frames it can transfer or transmit before data overwhelms receiver. The receiving device also contains only limited amount of speed and memory to store data. This is why receiving device should be able to tell or inform the sender about stopping the transmission or transferring of data on temporary basis before it reaches limit. It also needs buffer, large block of memory for just storing data or frames until they are processed.

Flow control can also be understood as a speed matching mechanism for two stations.


Approaches of Flow Control
Feedback - based Flow Control
Rate - based Flow Control

Approaches to Flow Control: Flow Control is classified into two categories:

- **Feedback – based Flow Control:** In this control technique, sender simply transmits data or information or frame to receiver, then receiver transmits data back to sender and also allows sender to transmit more amount of data or tell sender about how receiver is processing or doing. This simply means that sender transmits data or frames after it has received acknowledgements from user.

- **Rate – based Flow Control:** In this control technique, usually when sender sends or transfer data at faster speed to receiver and receiver is not being able to receive data at the speed, then mechanism known as built-in mechanism in protocol will just limit or restricts overall rate at which data or information is being transferred or transmitted by sender without any feedback or acknowledgement from receiver.

Techniques of Flow Control in Data Link Layer: There are basically two types of techniques being developed to control the flow of data

1. Stop-and-Wait Flow Control: This method is the easiest and simplest form of flow control. In this method, basically message or data is broken down into various multiple frames, and then receiver indicates its readiness to receive frame of data. When acknowledgement is received, then only sender will send or transfer the next frame. This process is continued until sender transmits EOT (End of Transmission) frame. In this method, only one of frames can be in transmission at a time. It leads to inefficiency i.e. less productivity if propagation delay is very much longer than the transmission delay and ultimately in this method sender sent single frame and receiver take one frame at a time and sent acknowledgement (which is next frame number only) for new frame.

Advantages –

- This method is very easiest and simple and each of the frames is checked and acknowledged well.
- This method is also very accurate.

Disadvantages –

- This method is fairly slow.
- In this, only one packet or frame can be sent at a time.
- It is very inefficient and makes the transmission process very slow.

2. Sliding Window Flow Control: This method is required where reliable in-order delivery of packets or frames is very much needed like in data link layer. It is point to point protocol that assumes that none of the other entity tries to communicate until current data or frame transfer gets completed. In this method, sender transmits or sends various frames or packets before receiving any acknowledgement. In this method, both the sender and receiver agree upon total number of data frames after which acknowledgement is needed to be transmitted. Data Link Layer requires and uses this method that simply allows sender to have more than one unacknowledged packet “in-flight” at a time. This increases and improves network throughput. and Ultimately In this method sender sent multiple frames but receiver take one by one and after completing one frame acknowledge (which is next frame number only) for new frame.

Advantages –

- It performs much better than stop-and-wait flow control.
- This method increases efficiency.
- Multiples frames can be sent one after another.

Disadvantages –

- The main issue is complexity at the sender and receiver due to the transferring of multiple frames.
- The receiver might receive data frames or packets out the sequence.

Comparison of Flow Control Methods

Flow Control Method	**Efficiency**	**Error Handling**	**Best Used in**
Stop-and-Wait	Low	Retransmits single frame	Simple, low-speed networks
Sliding Window (Go-Back-N)	Medium	Retransmits from error onward	Medium-speed networks
Sliding Window (Selective Repeat)	High	Retransmits only corrupted frames	High-speed, reliable networks

Piggybacking: Piggybacking is the technique of delaying outgoing acknowledgment temporarily and attaching it to the next data packet. When a data frame arrives, the receiver waits and does not send the control frame (acknowledgment) back immediately. The receiver waits until its network layer moves to the next data packet. Acknowledgment is associated with this outgoing data frame. Thus, the acknowledgment travels along with the next data frame.

Why Piggybacking?

Efficiency can also be improved by making use of full-duplex transmission. Full Duplex transmission is a transmission that happens with the help of two half-duplex transmissions which helps in communication in both directions. Full Duplex Transmission is better than both simplex and half-duplex transmission modes.

There are two ways through which we can achieve full-duplex transmission:

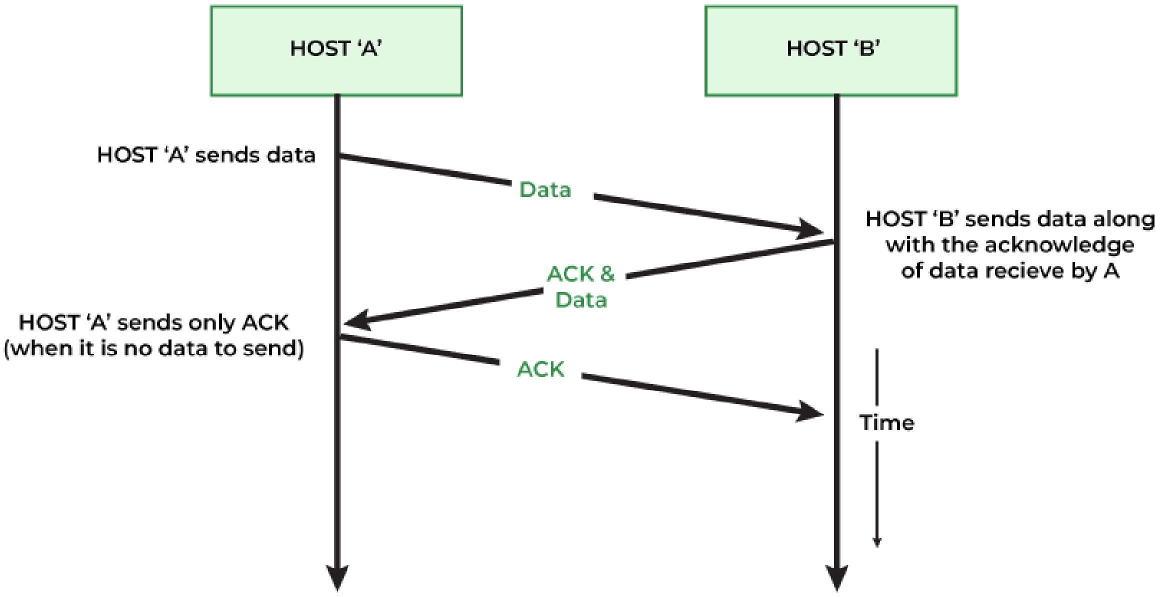
1. **Two Separate Channels:** One way to achieve full-duplex transmission is to have two separate channels with one for forwarding data transmission and the other for reverse data transfer (to accept). But this will almost completely waste the bandwidth of the reverse channel.

2. Piggybacking: A preferable solution would be to use each channel to transmit the frame (front and back) both ways, with both channels having the same capacity. Assume that A and B are users. Then the data frames from A to B are interconnected with the acknowledgment from A to B. and can be identified as a data frame or acknowledgment by checking the sort field in the header of the received frame.

One more improvement can be made. When a data frame arrives, the receiver waits and does not send the control frame (acknowledgment) back immediately. The receiver waits until its network layer moves to the next data packet.

Acknowledgment is associated with this outgoing data frame. Thus, the acknowledgment travels along with the next data frame.

Working of Piggybacking


HOST 'A'
HOST 'B'
HOST 'A' sends data
Data
HOST 'B' sends data along with the acknowledge of data recieve by A
ACK & Data
HOST 'A' sends only ACK (when it is no data to send)
ACK
Time

- If Host A has both acknowledgment and data, which it wants to send, then the data frame will be sent with the ack field which contains the sequence number of the frame.
- If Host A contains only one acknowledgment, then it will wait for some time, then in the case, if it finds any data frame, it piggybacks the acknowledgment, otherwise, it will send the ACK frame.
- If Host A left with only a data frame, then it will add the last acknowledgment to it. Host A can send a data frame with an ack field containing no acknowledgment bit.

Advantages of Piggybacking

1. The major advantage of piggybacking is the better use of available channel bandwidth. This happens because an acknowledgment frame does not need to be sent separately.
2. Usage cost reduction.
3. Improves latency of data transfer.

4. To avoid the delay and rebroadcast of frame transmission, piggybacking uses a very short-duration timer.

Disadvantages of Piggybacking

1. The disadvantage of piggybacking is the additional complexity.
2. If the data link layer waits long before transmitting the acknowledgment (blocks the ACK for some time), the frame will rebroadcast.

Multiple access protocol- ALOHA, CSMA, CSMA/CA and CSMA/CD

Data Link Layer

The data link layer is used in a computer network to transmit the data between two devices or nodes. It divides the layer into parts such as **data link control** and the **multiple access resolution/protocol**. The upper layer has the responsibility to flow control and the error control in the data link layer, and hence it is termed as **logical of data link control**. Whereas the lower sub-layer is used to handle and reduce the collision or multiple access on a channel. Hence it is termed as **media access control** or the multiple access resolutions.

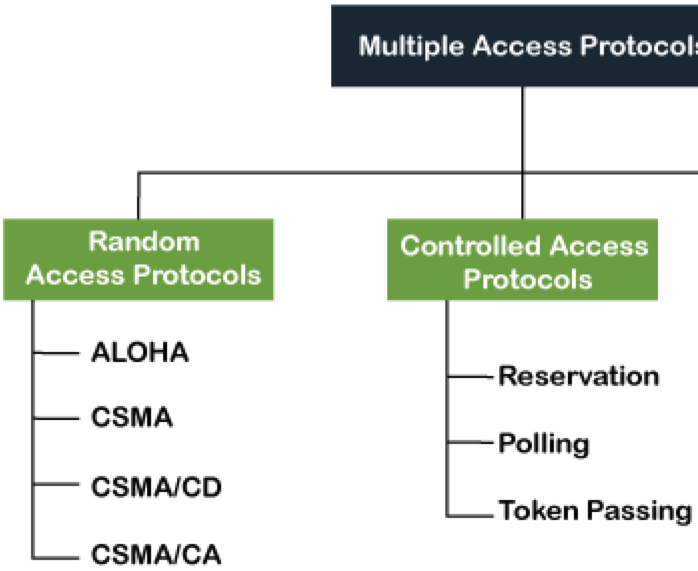
Data Link Control

A data link control is a reliable channel for transmitting data over a dedicated link using various techniques such as framing, error control and flow control of data packets in the computer network.

What is a multiple access protocol?

When a sender and receiver have a dedicated link to transmit data packets, the data link control is enough to handle the channel. Suppose there is no dedicated path to communicate or transfer the data between two devices. In that case, multiple stations access the channel and simultaneously transmits the data over the channel. It may create collision and cross talk. Hence, the multiple access protocol is required to reduce the collision and avoid crosstalk between the channels.

Following are the types of multiple access protocol that is subdivided into the different process as:



A. Random Access Protocol

In this protocol, all the station has the equal priority to send the data over a channel. In random access protocol, one or more stations cannot depend on another station nor any station control another station. Depending on the channel's state (idle or busy), each station transmits the data frame. However, if more than one station sends the data over a channel, there may be a collision or data conflict. Due to the collision, the data frame packets may be lost or changed. And hence, it does not receive by the receiver end.

Following are the different methods of random-access protocols for broadcasting frames on the channel.

- Aloha
- CSMA
- CSMA/CD
- CSMA/CA

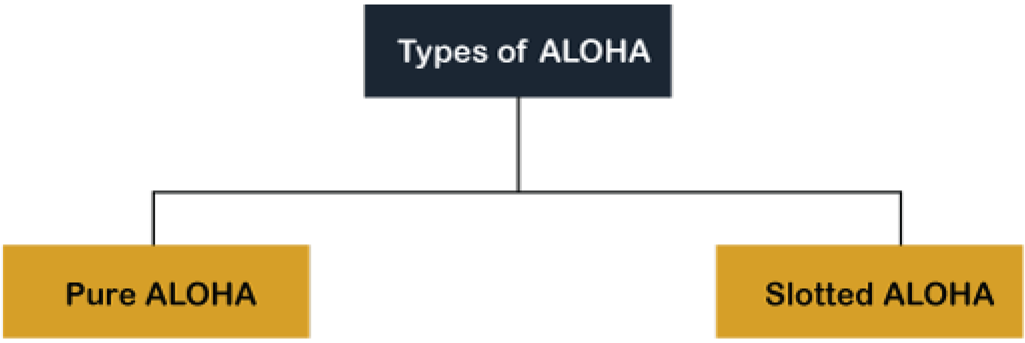
ALOHA Random Access Protocol

It is designed for wireless LAN (Local Area Network) but can also be used in a shared medium to transmit data. Using this method, any station can transmit data across a network simultaneously when a data frameset is available for transmission.

Aloha Rules

1. Any station can transmit data to a channel at any time.
2. It does not require any carrier sensing.
3. Collision and data frames may be lost during the transmission of data through multiple stations.

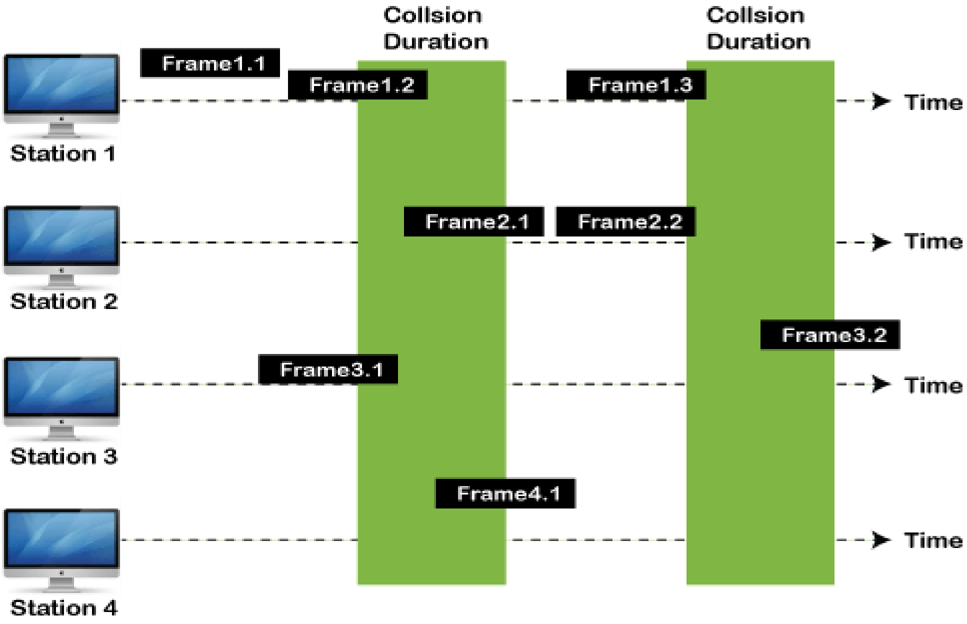
4. Acknowledgment of the frames exists in Aloha. Hence, there is no collision detection.
5. It requires retransmission of data after some random amount of time.



Pure Aloha

Whenever data is available for sending over a channel at stations, we use Pure Aloha. In pure Aloha, when each station transmits data to a channel without checking whether the channel is idle or not, the chances of collision may occur, and the data frame can be lost. When any station transmits the data frame to a channel, the pure Aloha waits for the receiver's acknowledgment. If it does not acknowledge the receiver end within the specified time, the station waits for a random amount of time, called the backoff time (Tb). And the station may assume the frame has been lost or destroyed. Therefore, it retransmits the frame until all the data are successfully transmitted to the receiver.

1. The total vulnerable time of pure Aloha is 2 * Tfr.
2. Maximum throughput occurs when G = 1/ 2 that is 18.4%.
3. Successful transmission of data frame is S = G * e ^ - 2 G.



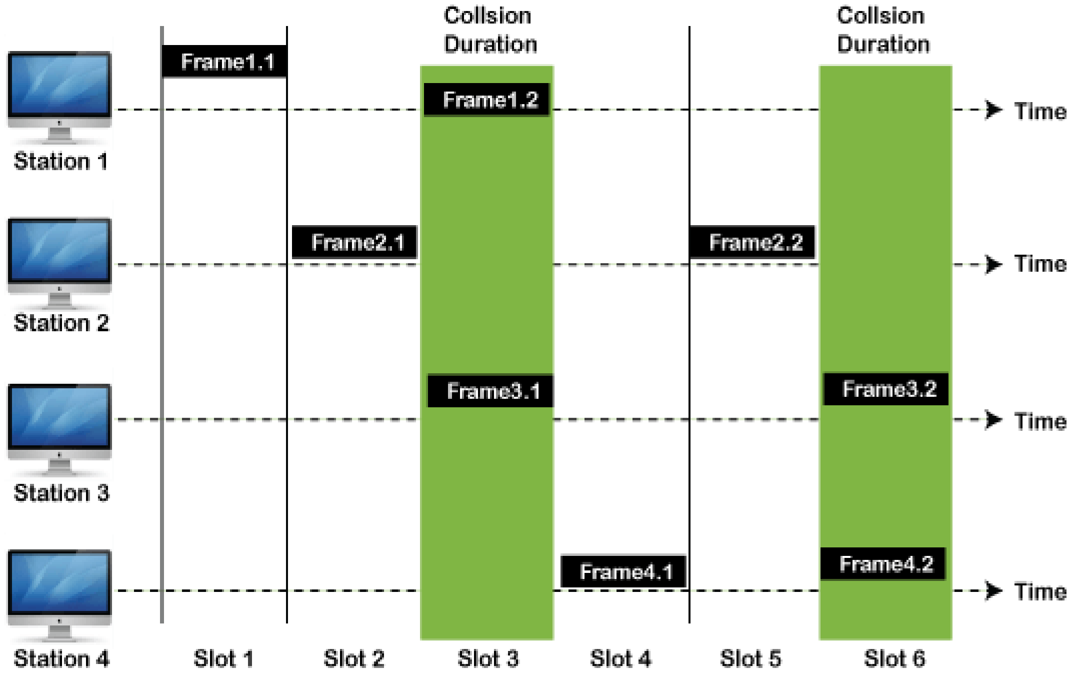
Frames in Pure ALOHA

As we can see in the figure above, there are four stations for accessing a shared channel and transmitting data frames. Some frames collide because most stations send their frames at the same time. Only two frames, frame 1.1 and frame 2.2, are successfully transmitted to the receiver end. At the same time, other frames are lost or destroyed. Whenever two frames fall on a shared channel simultaneously, collisions can occur, and both will suffer damage. If the new frame's first bit enters the channel before finishing the last bit of the second frame. Both frames are completely finished, and both stations must retransmit the data frame.

Slotted Aloha

The slotted Aloha is designed to overcome the pure Aloha's efficiency because pure Aloha has a very high possibility of frame hitting. In slotted Aloha, the shared channel is divided into a fixed time interval called **slots**. So that, if a station wants to send a frame to a shared channel, the frame can only be sent at the beginning of the slot, and only one frame is allowed to be sent to each slot. And if the stations are unable to send data to the beginning of the slot, the station will have to wait until the beginning of the slot for the next time. However, the possibility of a collision remains when trying to send a frame at the beginning of two or more station time slot.

1. Maximum throughput occurs in the slotted Aloha when G = 1 that is 37%.
2. The probability of successfully transmitting the data frame in the slotted Aloha is S = G * e ^ - 2 G.
3. The total vulnerable time required in slotted Aloha is Tfr.



Frames in Slotted ALOHA

CSMA (Carrier Sense Multiple Access)

It is a **carrier sense multiple access** based on media access protocol to sense the traffic on a channel (idle or busy) before transmitting the data. It means that if the channel is idle, the station can send data to the channel. Otherwise, it must wait until the channel becomes idle. Hence, it reduces the chances of a collision on a transmission medium.

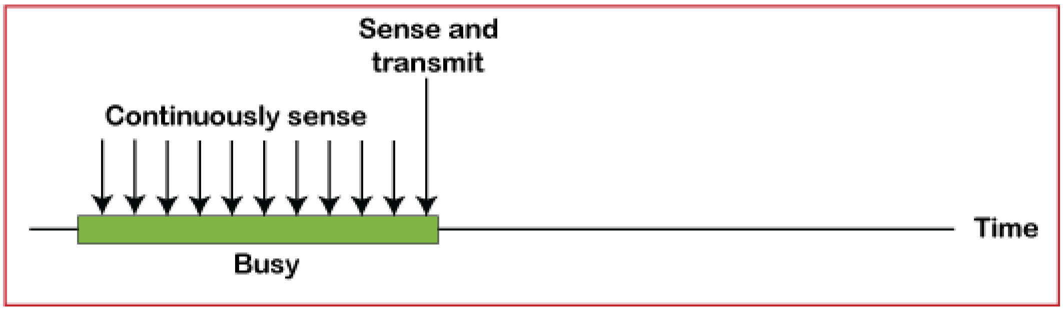
CSMA Access Modes

1-Persistent: In the 1-Persistent mode of CSMA that defines each node, first sense the shared channel and if the channel is idle, it immediately sends the data. Else it must wait and keep track of the status of the channel to be idle and broadcast the frame unconditionally as soon as the channel is idle.

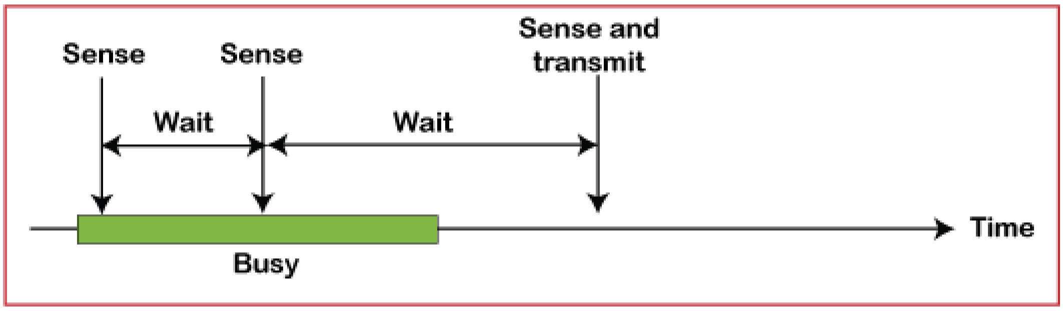
Non-Persistent: It is the access mode of CSMA that defines before transmitting the data, each node must sense the channel, and if the channel is inactive, it immediately sends the data. Otherwise, the station must wait for a random time (not continuously), and when the channel is found to be idle, it transmits the frames.

P-Persistent: It is the combination of 1-Persistent and Non-persistent modes. The P-Persistent mode defines that each node senses the channel, and if the channel is inactive, it sends a frame with a **P** probability. If the data is not transmitted, it waits for a (**q = 1-p probability**) random time and resumes the frame with the next time slot.

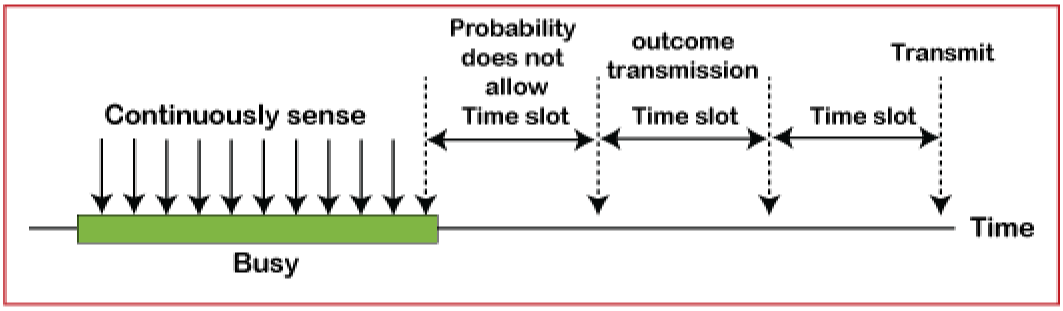
O- Persistent: It is an O-persistent method that defines the superiority of the station before the transmission of the frame on the shared channel. If it is found that the channel is inactive, each station waits for its turn to retransmit the data.



a. 1-persistent



b. Nonpersistent



c. p-persistent

CSMA/ CD

It is a **carrier sense multiple access/ collision detection** network protocol to transmit data frames. The CSMA/CD protocol works with a medium access control layer. Therefore, it first senses the shared channel before broadcasting the frames, and if the channel is idle, it transmits a frame to check whether the transmission was successful. If the frame is successfully received, the station sends another frame. If any collision is detected in the CSMA/CD, the station sends a jam/ stop signal to the shared channel to terminate data transmission. After that, it waits for a random time before sending a frame to a channel.

CSMA/ CA

It is a **carrier sense multiple access/collision avoidance** network protocol for carrier transmission of data frames. It is a protocol that works with a medium access control layer. When a data frame is sent to a channel, it receives an acknowledgment to check whether the channel is clear. If the station receives only a single (own) acknowledgment, that means the data frame has been successfully transmitted to the receiver. But if it gets two signals (its own and one more in which the collision of frames), a collision of the frame occurs in the shared channel. Detects the collision of the frame when a sender receives an acknowledgment signal. Following are the methods used in the CSMA/ CA to avoid the collision:

Interframe space: In this method, the station waits for the channel to become idle, and if it gets the channel is idle, it does not immediately send the data. Instead of this, it waits for some time, and this time period is called the **Interframe** space or IFS. However, the IFS time is often used to define the priority of the station.

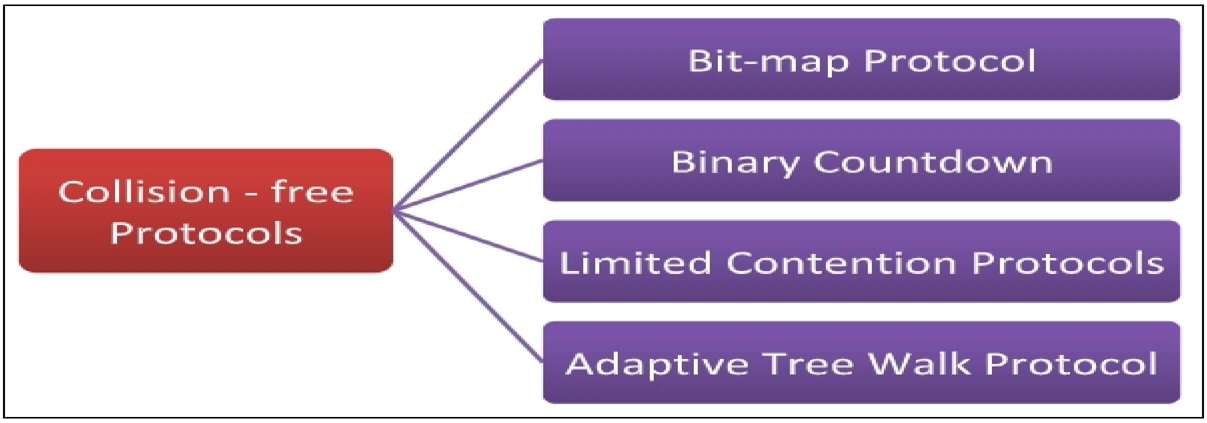
Contention window: In the Contention window, the total time is divided into different slots. When the station/ sender is ready to transmit the data frame, it chooses a random slot number of slots as **wait time**. If the channel is still busy, it does not restart the entire process, except that it restarts the timer only to send data packets when the channel is inactive.

Acknowledgment: In the acknowledgment method, the sender station sends the data frame to the shared channel if the acknowledgment is not received ahead of time.

Collision Free Protocols

In computer networks, when more than one station tries to transmit simultaneously via a shared channel, the transmitted data is garbled. This event is called collision. The Medium Access Control (MAC) layer of the OSI model is responsible for handling collision of frames. Collision – free protocols are devised so that collisions do not occur. Protocols like CSMA/CD and CSMA/CA nullifies the possibility of collisions once the transmission channel is acquired by any station. However, collision can still occur during the contention period if more than one stations starts to transmit at the same time. Collision – free protocols resolves collision in the contention period and so the possibilities of collisions are eliminated.

Types of Collision – free Protocols


Collision - free Protocols
Bit-map Protocol
Binary Countdown
Limited Contention Protocols
Adaptive Tree Walk Protocol

Bit – map Protocol

In bit map protocol, the contention period is divided into N slots, where N is the total number of stations sharing the channel. If a station has a frame to send, it sets the corresponding bit in the slot. So, before transmission, each station knows whether the other stations want to transmit. Collisions are avoided by mutual agreement among the contending stations on who gets the channel.

Binary Countdown

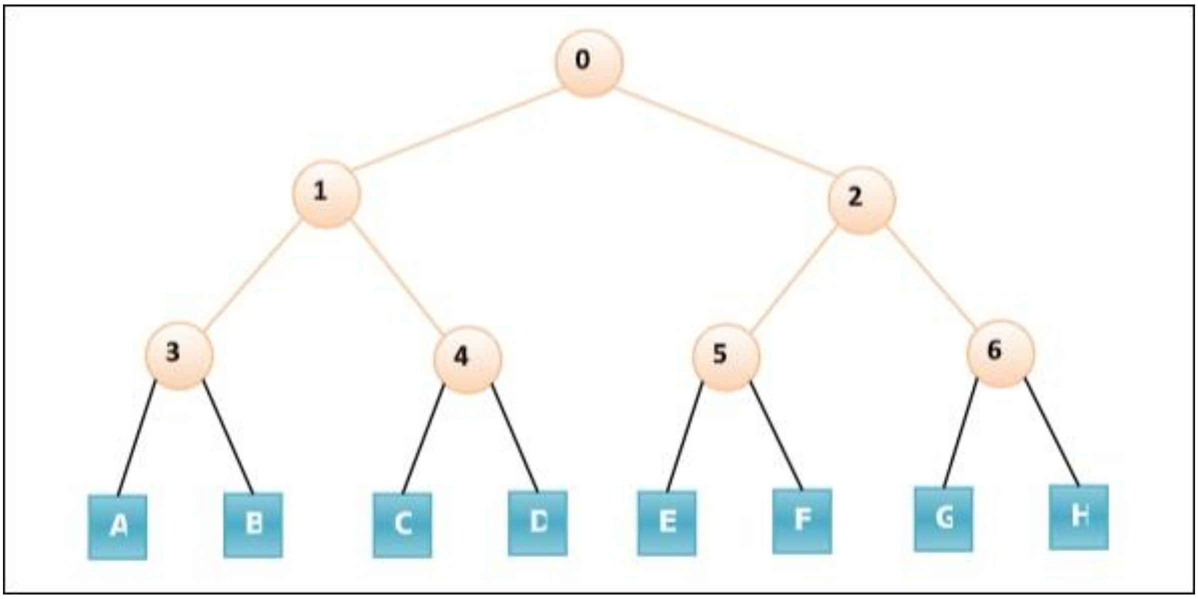
This protocol overcomes the overhead of 1 bit per station of the bit – map protocol. Here, binary addresses of equal lengths are assigned to each station. For example, if there are 6 stations, they may be assigned the binary addresses 001, 010, 011, 100, 101 and 110. All stations wanting to communicate broadcast their addresses. The station with higher address gets the higher priority for transmitting.

Limited Contention Protocols

These protocols combine the advantages of collision-based protocols and collision free protocols. Under light load, they behave like ALOHA scheme. Under heavy load, they behave like bitmap protocols.

Adaptive Tree Walk Protocol

In adaptive tree walk protocol, the stations or nodes are arranged in the form of a binary tree as follows -



Initially all nodes (A, B ……. G, H) are permitted to compete for the channel. If a node is successful in acquiring the channel, it transmits its frame. In case of collision, the nodes are divided into two groups (A, B, C, D in one group and E, F, G, H in another group). Nodes belonging to only one of them is permitted for competing. This process continues until successful transmission occurs.

IEEE 802 Protocols

The IEEE 802 family of standards deals with local area networks (LAN) and metropolitan area networks (MAN). IEEE 802 protocols ensure that networking components from different manufacturers work seamlessly together. Here's a brief overview of some key IEEE 802 protocols:

1. **IEEE 802.3 (Ethernet)**: This protocol defines the physical and data link layers for wired Ethernet networks. It's the most widely used LAN technology, supporting data transfer rates ranging from 10 Mbps to 100 Gbps.
2. **IEEE 802.11 (Wi-Fi)**: This set of standards defines the protocols for implementing wireless local area network (WLAN) communication. The various amendments (e.g., 802.11a, 802.11b, 802.11g, 802.11n, 802.11ac, 802.11ax) introduce different enhancements in terms of speed, frequency bands, and features.
3. **IEEE 802.1Q (VLANs)**: This standard defines how virtual LANs (VLANs) are implemented in Ethernet networks. VLANs allow network administrators to segment a physical LAN into multiple logical networks, improving security and traffic management.
4. **IEEE 802.15 (Wireless Personal Area Networks):** This set of standards defines protocols for wireless personal area networks (WPANs), which enable communication between devices over short distances. Notable protocols include Bluetooth (IEEE 802.15.1) and Zigbee (IEEE 802.15.4).
5. **IEEE 802.16 (WiMAX):** This standard defines protocols for broadband wireless access networks, providing high-speed internet access over long distances. WiMAX is designed to serve as an alternative to traditional wired broadband technologies.
6. **IEEE 802.1X (Port-Based Network Access Control):** This standard provides a framework for secure network access control, ensuring that only authorized devices and users can access the network.

Standard Topologies

Standard topologies refer to the various ways in which a network is designed and organized. Here are some common network topologies:

1. **Bus Topology**: In this topology, all devices are connected to a single central cable, known as the bus. Data travels along the bus, and each device can send and receive data. It's simple and cost-effective but can be problematic if the central cable fails.
2. **Ring Topology**: Devices are connected in a circular fashion, forming a ring. Data travels in one direction (or both directions in a dual ring topology) until it reaches its

destination. Each device has exactly two neighbours. Ring topologies are reliable but can be slow if the network grows large.

3. **Star Topology**: All devices are connected to a central hub or switch. Data is sent to the hub, which then forwards it to the intended device. This topology is easy to set up and manage, and if one connection fails, it doesn't affect the others. However, if the central hub fails, the entire network goes down.
4. **Mesh Topology**: Each device is connected to every other device in the network. This provides high redundancy and reliability, as data has multiple paths to reach its destination. However, it can be expensive and complex to implement, especially in large networks.
5. **Tree Topology**: This topology combines elements of star and bus topologies. Devices are connected in a hierarchical manner, with a central root node and branches connecting multiple star-configured networks. It's scalable and easy to manage but relies on the stability of the root node.
6. **Hybrid Topology**: A combination of two or more different topologies to leverage the advantages of each. For example, a star-bus hybrid topology might use a star layout within each segment and connect these segments using a bus structure.

Network Management

Network management is the procedure of administering, managing and working a data network using a network management system. Current network management systems use software and hardware to constantly collect and analyse data and push out configuration changes for increasing performance, reliability, and security.

It involves configuring monitoring and possibly reconfiguring components in a network with the goal of providing optimal performance, minimum downtime, proper security, accountability, and flexibility.

Features

There are various features of network management which are as follows −

1. **Network automation**

One defining feature of a modern network management system is network automation. This is the procedure of automating the configuring, handling, testing, deploying, and operating of physical and virtual devices inside a network. Network service availability increases when everyday network tasks and functions are automated and repetitive processes are controlled and managed automatically.

2. **Network administration**

Network administration encompasses tracking network resources, including switches, routers, and servers. It also includes performance monitoring and software updates.

3. **Network Operation**

This contains smooth network functioning as created and intended, including close monitoring of activities to quickly and effectively address and fix problems as they occur and preferably even before users are aware of the problem.

4. **Network assurance**

Network assurance features are often included in modern network management systems. These features help improve network performance, customer experience, and security. Assurance systems help network analytics, application analytics, and policy analytics, as well as AI and ML, to achieve full network assurance.

5. **Network provisioning**

Network provisioning involves network resource configuration for the purposes of supporting any given service, like voice functions or accommodating additional users.

6. **Network maintenance**

Network maintenance covers upgrades and fixes to network resources. It also consists of proactive and remediation activities executed by working with network administrators, such as replacing network gear like routers and switches.

7. **Network analytics**

Network analytics is a software tool that compares incoming information against preprogrammed operational models and makes functional decisions for improving network performance.

MAC Addressing Frame Format

What is MAC Address?

To communicate or transfer data from one computer to another, we need an address. A MAC address, which stands for Media Access Control Address, is a physical address that works at the Data Link Layer.

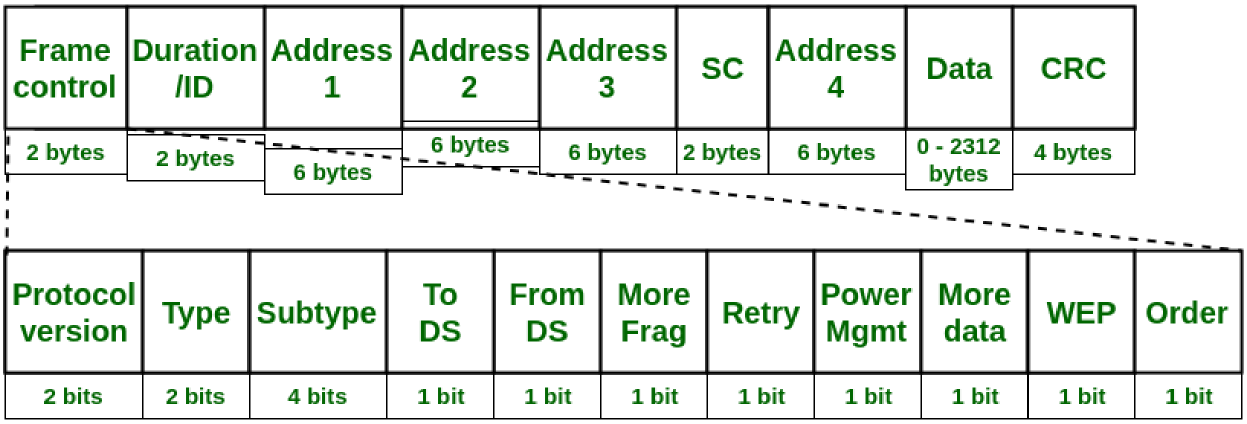
MAC Addresses are unique 48-bit hardware numbers of a computer that are embedded into a network interface card during manufacturing.

In the IEEE 802 standard, the data link layer is divided into two sublayers:

- Logical Link Control (LLC) Sublayer
- Media Access Control (MAC) Sublayer

Format of MAC Address

MAC Frame: The MAC layer frame consists of 9 fields. The following figure shows the basic structure of an IEEE 802.11 MAC data frame along with the content of the frame control field.



IEEE 802.11 MAC Frame Structure

Frame Control (FC) – It is 2 bytes long field which defines type of frame and some control information. Various fields present in FC are:

1. **Version:** It is a 2 bit long field which indicates the current protocol version which is fixed to be 0 for now.
2. **Type:** It is a 2 bit long field which determines the function of frame i.e management (00), control (01) or data (10). The value 11 is reserved.
3. **Subtype:** It is a 4 bit long field which indicates sub-type of the frame like 0000 for association request, 1000 for beacon.
4. **To DS:** It is a 1 bit long field which when set indicates that destination frame is for DS (distribution system).
5. **From DS:** It is a 1 bit long field which when set indicates frame coming from DS.
6. **More frag (More fragments):** It is 1 bit long field which when set to 1 means frame is followed by other fragments.
7. **Retry:** It is 1-bit long field, if the current frame is a retransmission of an earlier frame, this bit is set to 1.
8. **Power Mgmt (Power management):** It is 1-bit long field that indicates the mode of a station after successful transmission of a frame. Set to 1 the field indicates that the station goes into power-save mode. If the field is set to 0, the station stays active.
9. **More data:** It is 1-bit long field that is used to indicate receiver that a sender has more data to send than the current frame. This can be used by an access point to indicate to a

station in power-save mode that more packets are buffered or it can be used by a station to indicate to an access point after being polled that more polling is necessary as the station has more data ready to transmit.

10. **WEP:** It is 1 bit long field which indicates that the standard security mechanism of 802.11 is applied.
11. **Order:** It is 1 bit long field, if this bit is set to 1 the received frames must be processed in strict order.

Duration/ID – It is 4 bytes long field which contains the value indicating the period of time in which the medium is occupied(in µs).

Address 1 to 4 – These are 6 bytes long fields which contain standard IEEE 802 MAC addresses (48 bit each). The meaning of each address depends on the DS bits in the frame control field.

SC (Sequence control) – It is 16 bits long field which consists of 2 sub-fields, i.e., Sequence number (12 bits) and Fragment number (4 bits). Since acknowledgement mechanism frames may be duplicated hence, a sequence number is used to filter duplicate frames.

Data – It is a variable length field which contain information specific to individual frames which is transferred transparently from a sender to the receiver(s).

CRC (Cyclic redundancy check) – It is 4 bytes long field which contains a 32 bit CRC error detection sequence to ensure error free frame.

features of the IEEE 802.11 MAC frame:

Frame Control Field: The frame control field contains information about the type of frame, the data rate, and the power management status.

Duration Field: The duration field specifies the length of time that the channel will be occupied by the transmission.

Address Fields: The address fields specify the source and destination MAC addresses of the Wi-Fi devices involved in the communication.

Sequence Control Field: The sequence control field is used to identify and manage the transmission sequence of the frames.

Frame Body: The frame body contains the actual data being transmitted between Wi-Fi devices, such as IP packets, TCP segments, or UDP datagrams.

Frame Check Sequence: The frame check sequence (FCS) is used to check the integrity of the data transmitted in the frame and to detect any transmission errors.

Management, Control, and Data Frames: The IEEE 802.11 MAC frame defines three types of frames: management frames, control frames, and data frames. Management frames are used for network management, control frames are used for coordination between Wi-Fi devices, and data frames are used for the transmission of actual data.

Fragmentation: The IEEE 802.11 MAC frame supports fragmentation, which allows large data packets to be divided into smaller fragments for transmission.

Acknowledgments: The IEEE 802.11 MAC frame uses acknowledgments to confirm the successful transmission of frames and to request the retransmission of any frames that were not successfully received.

PCET's

Pimpri Chinchwad University

School of Computer Application

PMC113A Computer Networks

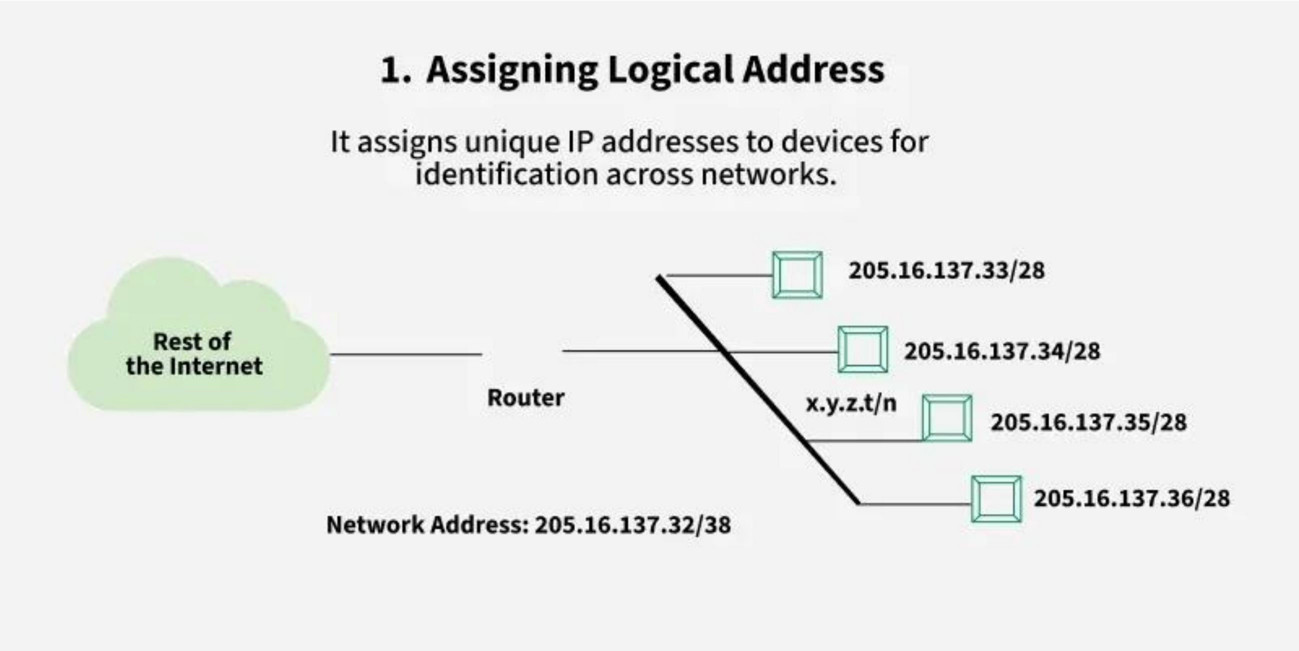
Unit No. 05 Transport Layer

Introduction:

The network layer is a part of the communication process in computer networks. Its main job is to move data packets between different networks. It helps route these packets from the sender to the receiver across multiple paths and networks. Network-to-network connections enable the Internet to function. These connections happen at the network layer which sends data packets between different networks. In the 7-layer OSI model, the network layer is layer 3. The Internet Protocol (IP) is a key protocol used at this layer, along with other protocols for routing, testing, and encryption.

Services Offered by Network Layer

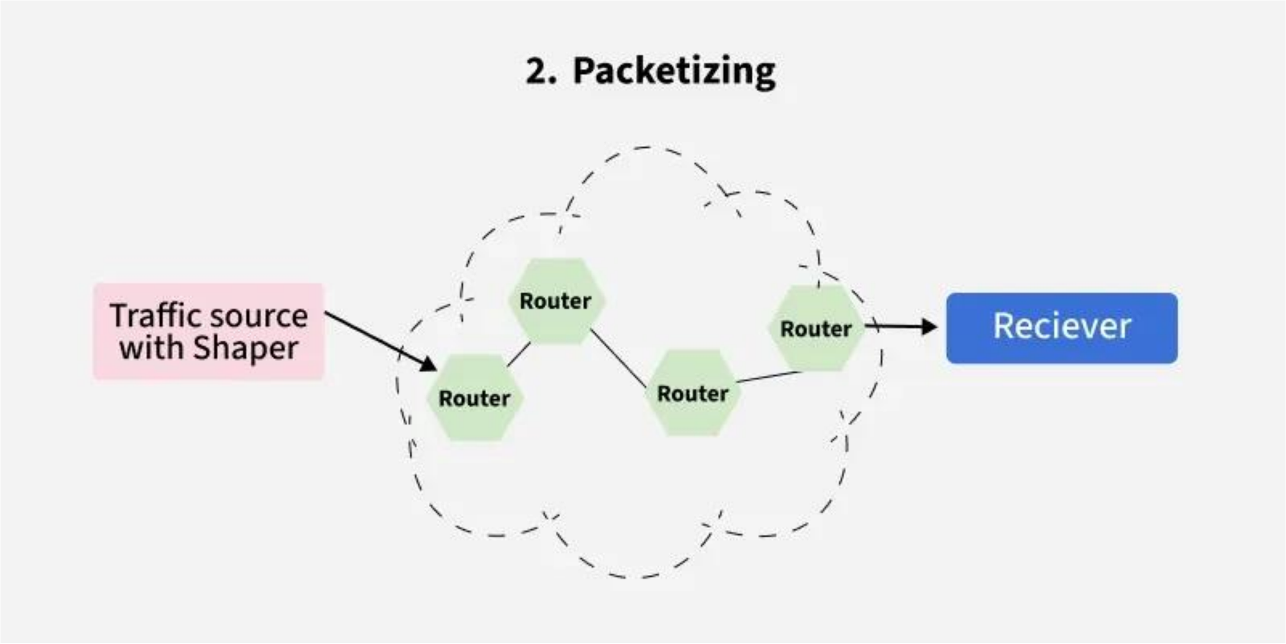
The **services** which are offered by the network layer are as follows:


1. Assigning Logical Address
It assigns unique IP addresses to devices for identification across networks.
Rest of the Internet
Router
x.y.z.t/n
205.16.137.33/28
205.16.137.34/28
205.16.137.35/28
205.16.137.36/28
Network Address: 205.16.137.32/38

1. Assigning Logical Address

Logical addressing is the process of assigning unique IP addresses (IPv4 or IPv6) to devices within a network. Unlike physical addresses (MAC addresses), logical addresses can change based on network configurations. These addresses are hierarchical and help identify both the network and the device within that network. Logical addressing is important for:

- Enabling communication between devices on different networks.
- Facilitating routing by providing location-based information.



2. Packetizing

The process of encapsulating the data received from the upper layers of the network (also called payload) in a network layer packet at the source and decapsulating the payload from the network layer packet at the destination is known as packetizing.

The source host adds a header that contains the source and destination address and some other relevant information required by the network layer protocol to the payload received from the upper layer protocol and delivers the packet to the data link layer.

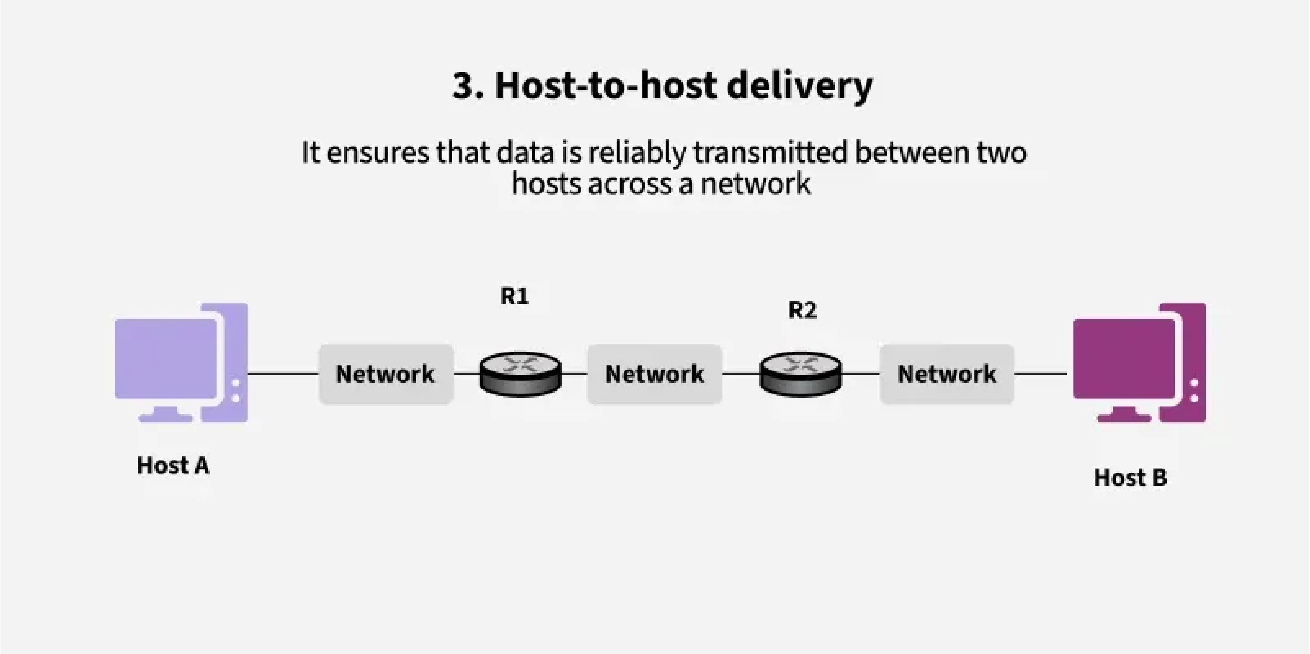
The destination host receives the network layer packet from its data link layer, decapsulates the packet, and delivers the payload to the corresponding upper layer protocol. The routers in the path are not allowed to change either the source or the destination address. The routers in the path are not allowed to decapsulate the packets they receive unless they need to be fragmented.

3. Host-to-Host Delivery

The network layer ensures data is transferred from the source device (host) to the destination device (host) across one or multiple networks. This involves:

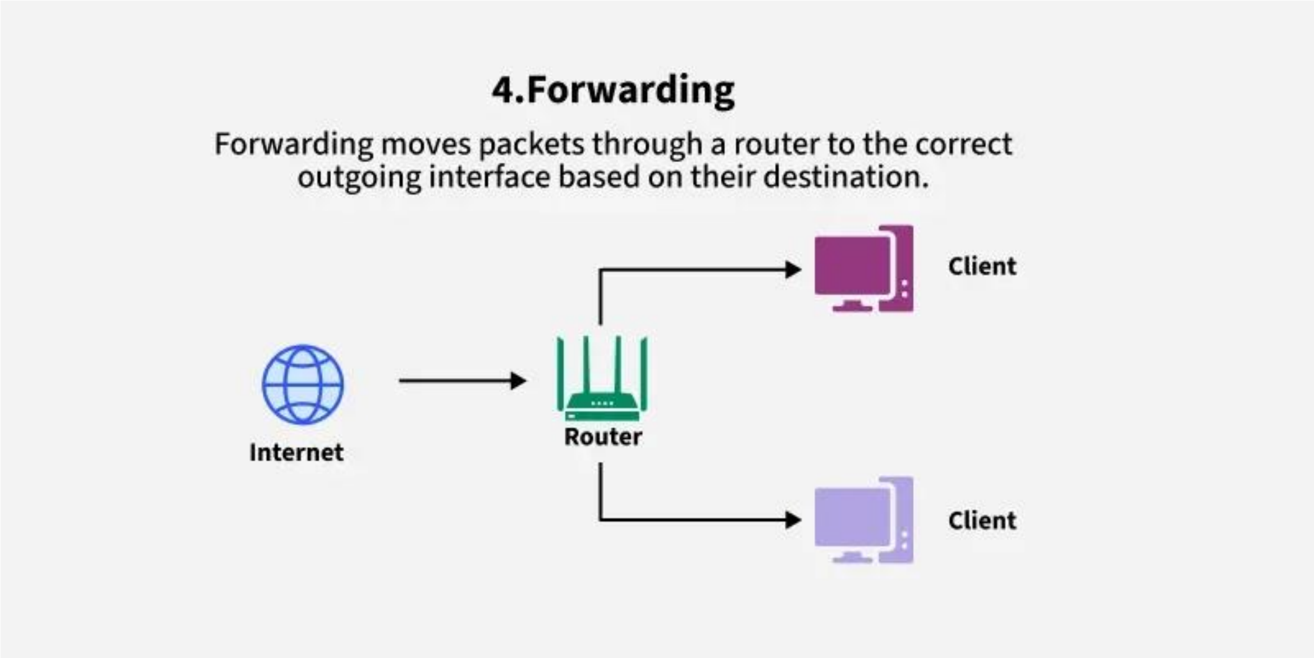
- Determining the destination address.
- Ensuring that data is transmitted without duplication or corruption.

Host-to-host delivery is a foundational aspect of communication in large-scale, interconnected systems like the internet.



3. Host-to-host delivery

It ensures that data is reliably transmitted between two hosts across a network



4.Forwarding

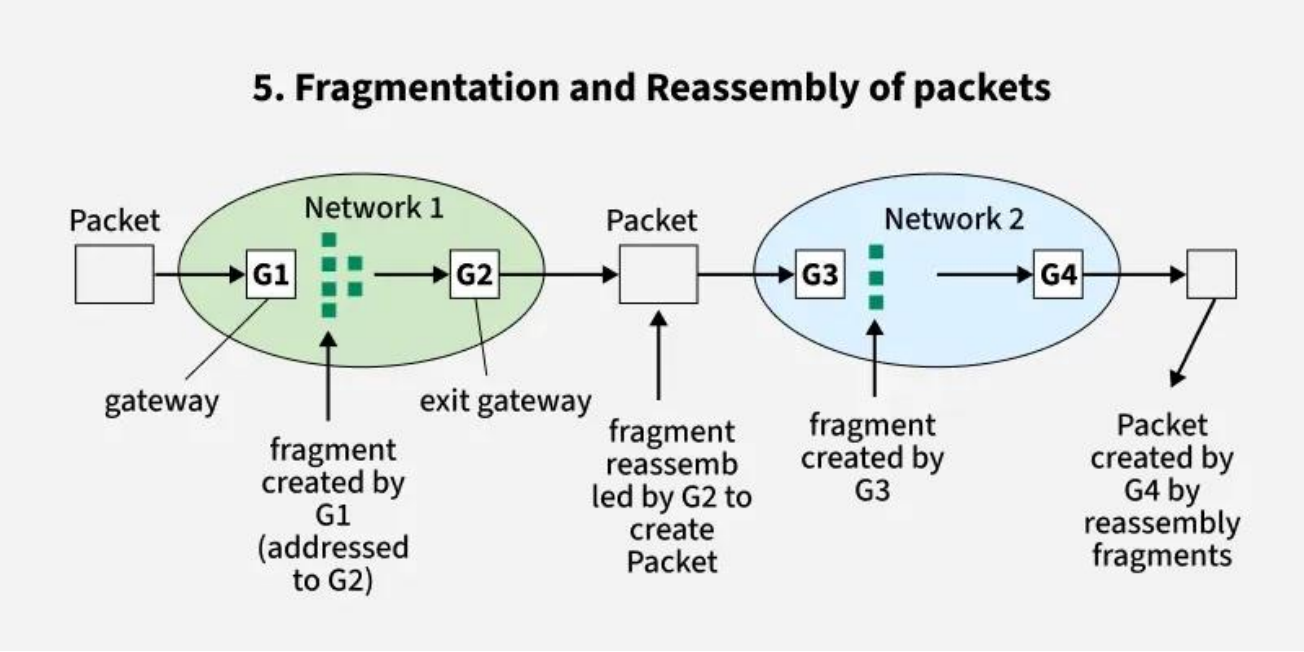
Forwarding moves packets through a router to the correct outgoing interface based on their destination.

4. Forwarding

Forwarding is the process of transferring packets between network devices such as routers, which are responsible for directing the packets toward their destination. When a router receives a packet from one of its attached networks, it needs to forward the packet to another attached network (unicast routing) or to some attached networks.

The router uses:

- Routing tables: These tables store information about possible paths to different networks.
- Forwarding decisions: Based on the destination IP address in the packet header. Forwarding ensures that packets move closer to their destination efficiently.


5. Fragmentation and Reassembly of packets
Packet
Network 1
G1
G2
gateway
exit gateway
fragment created by G1 (addressed to G2)
Packet
fragment reassemb led by G2 to create Packet
Network 2
G3
G4
fragment created by G3
Packet created by G4 by reassembly fragments

5. Fragmentation and Reassembly of Packets

Some networks have a maximum transmission unit (MTU) that defines the largest packet size they can handle. If a packet exceeds the MTU, the network layer:

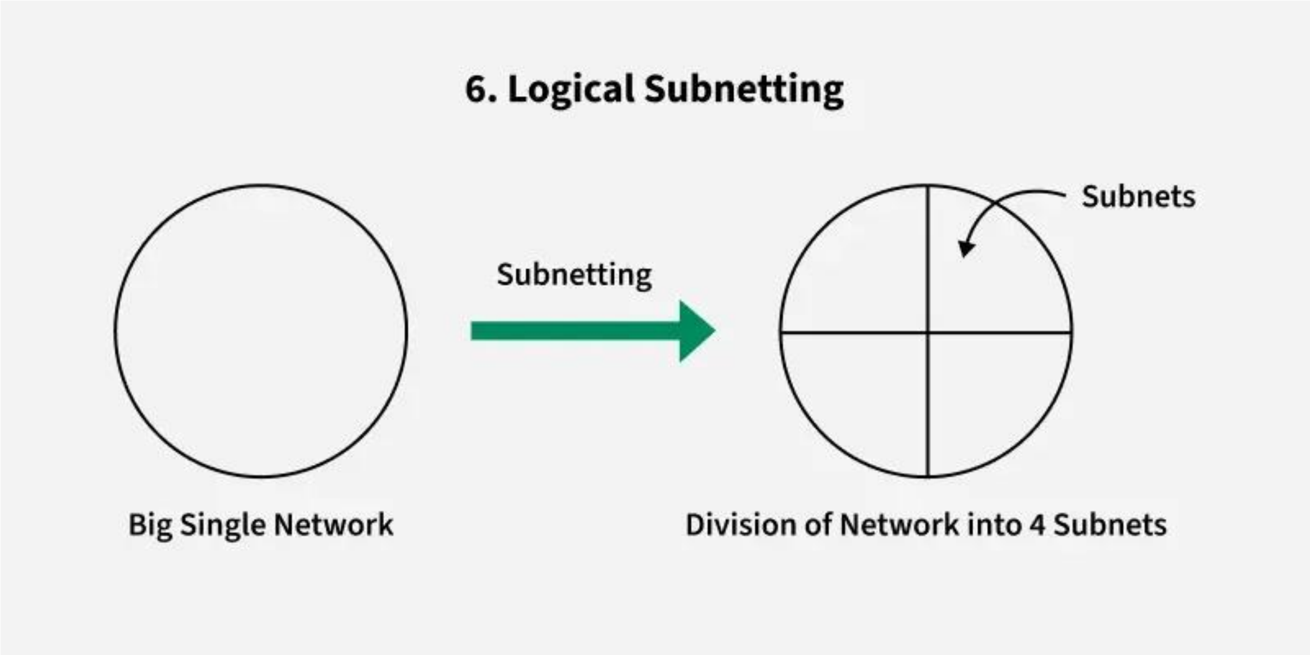
- Fragments the packet into smaller pieces.
- Adds headers to each fragment for identification and sequencing. At the destination, the fragments are reassembled into the original packet. This ensures compatibility with networks of varying capabilities without data loss.

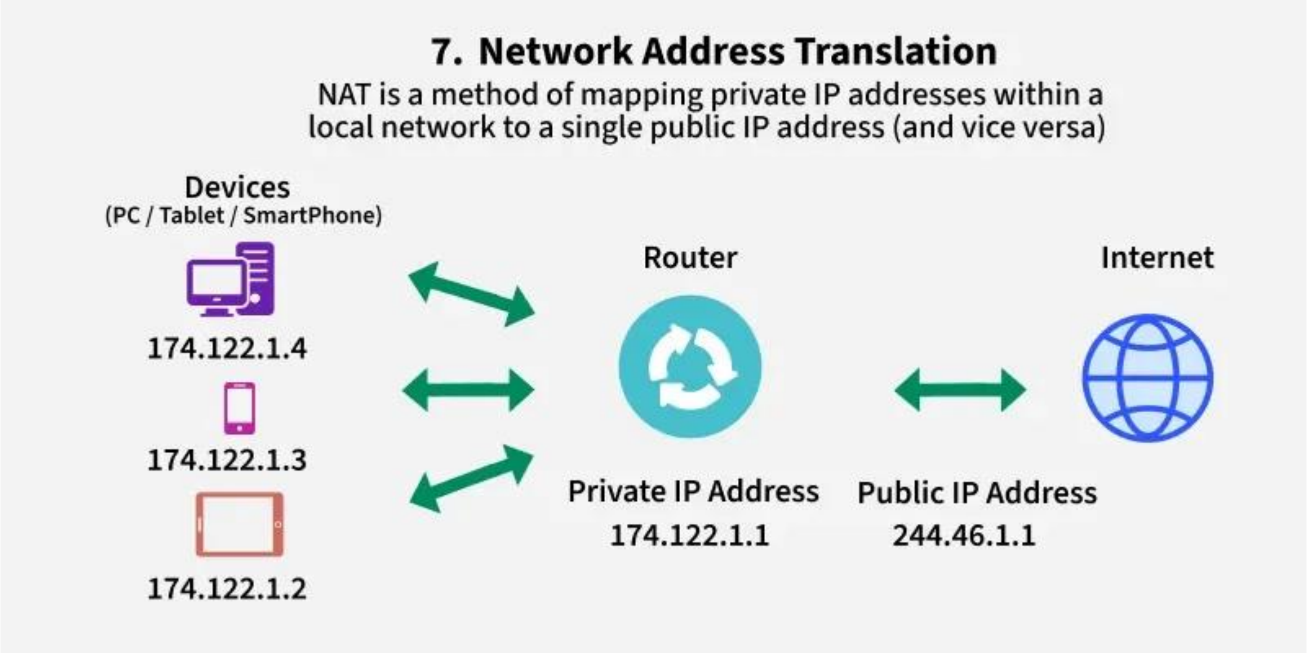
6. Logical Subnetting

Logical subnetting involves dividing a large IP network into smaller, more manageable sub-networks (subnets). Subnetting helps:

- Improve network performance by reducing congestion.
- Enhance security by isolating parts of a network.

- Simplify network management and troubleshooting. Subnetting uses subnet masks to define the range of IP addresses within each subnet, enabling efficient address allocation and routing.



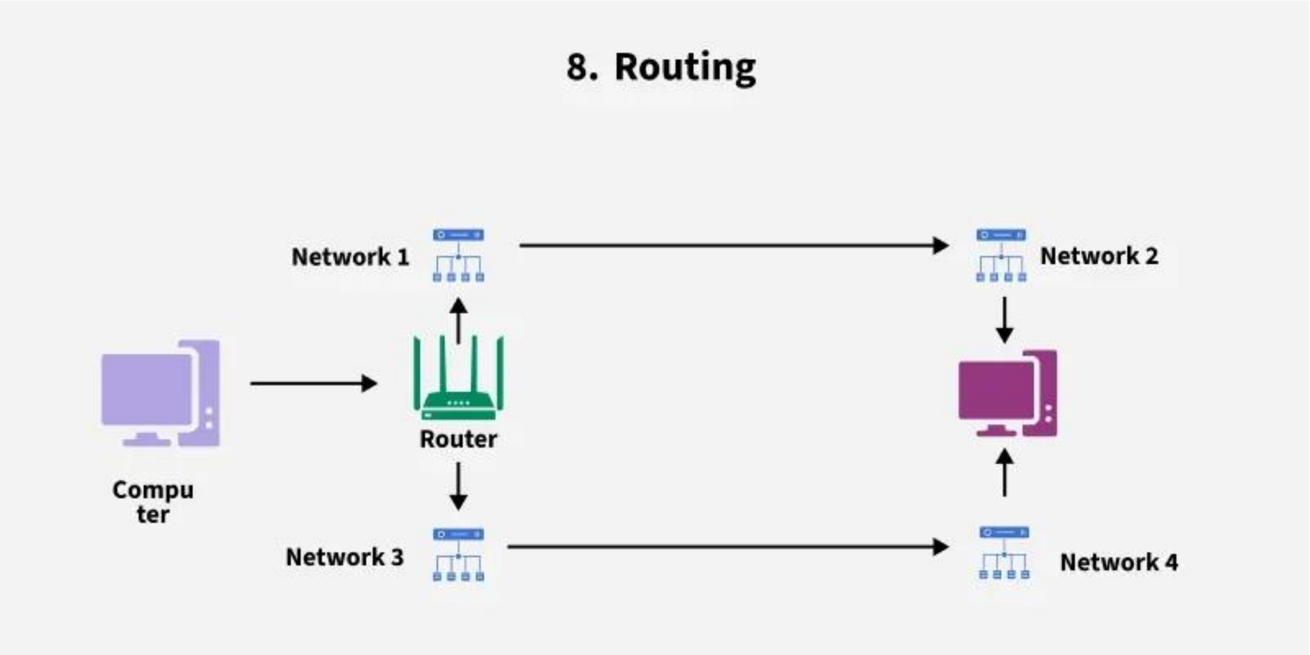

7. Network Address Translation
NAT is a method of mapping private IP addresses within a local network to a single public IP address (and vice versa)
Devices
(PC / Tablet / SmartPhone)
174.122.1.4
174.122.1.3
174.122.1.2
Router
Private IP Address
174.122.1.1
Public IP Address
244.46.1.1
Internet

7. Network Address Translation (NAT)

NAT allows multiple devices in a private network to share a single public IP address for internet access. This is achieved by:

- Translating private IP addresses to a public IP address for outbound traffic.
- Reversing the process for inbound traffic. Benefits of NAT include:

- Conserving IPv4 addresses by reducing the need for unique public IPs for each device.
- Enhancing security by masking internal IP addresses from external networks.



8. Routing

Routing is the process of moving data from one device to another device. These are two other services offered by the network layer. In a network, there are several routes available from the source to the destination. The network layer specifies some strategies which find out the best possible route. This process is referred to as routing. There are several routing protocols that are used in this process and they should be run to help the routers coordinate with each other and help in establishing communication throughout the network.

Advantages of Network Layer Services

- Packetization service in the network layer provides ease of transportation of the data packets.
- Packetization also eliminates single points of failure in data communication systems.
- Routers present in the network layer reduce network traffic by creating collision and broadcast domains.
- With the help of Forwarding, data packets are transferred from one place to another in the network.

Disadvantages of Network Layer Services

- There is a lack of flow control in the design of the network layer.

- Congestion occurs sometimes due to the presence of too many datagrams in a network that is beyond the capacity of the network or the routers. Due to this, some routers may drop some of the datagrams, and some important pieces of information may be lost.
- Although indirect error control is present in the network layer, there is a lack of proper error control mechanisms as due to the presence of fragmented data packets, error control becomes difficult to implement.

Features and Design issues of Network Layer

The network layer in computer networks, responsible for logical addressing and routing packets, faces design issues like addressing, routing protocols, and fragmentation/reassembly, which are crucial for efficient and reliable data transmission across networks.

Key Features:

Logical Addressing:

The network layer assigns unique addresses (like IP addresses) to devices on a network, enabling communication between different networks.

Routing:

It determines the best path for packets to travel from the source to the destination, using routing protocols and tables.

Fragmentation and Reassembly:

The network layer handles situations where packets exceed the maximum transmission unit (MTU) of a network, breaking them into smaller fragments for transmission and reassembling them at the destination.

Inter-networking:

The network layer enables communication between different networks, allowing devices on one network to communicate with devices on another.

Connection-Oriented vs. Connectionless:

The network layer can operate in connection-oriented (like X.25) or connectionless (like IP) modes, influencing how packets are handled and routed.

Quality of Service (QoS):

The network layer can implement mechanisms to prioritize certain types of traffic, ensuring that critical applications receive better performance.

Design Issues:

Addressing:

- **Address Allocation:** How IP addresses are assigned to devices (e.g., static, dynamic).

- **Address Resolution:** How a device can determine the physical address (MAC address) of a destination device based on its IP address.

Routing:

- **Routing Protocol Design:** Choosing the right routing protocol (e.g., RIP, OSPF, BGP) for a specific network topology and requirements.
- **Routing Table Management:** How routing tables are maintained and updated efficiently.
- **Congestion Control:** How to handle network congestion and prevent packet loss.

Fragmentation and Reassembly:

- **Fragment Size:** Determining the optimal fragment size for efficient transmission.
- **Fragmentation Overhead:** Minimizing the overhead introduced by fragmentation.
- **Reassembly Challenges:** Ensuring reliable and efficient reassembly of fragments at the destination.

Security:

- **Network Layer Security:** Implementing security mechanisms at the network layer to protect against attacks and unauthorized access.
- **Firewalling:** Using firewalls to control network traffic and protect against malicious activity.

Scalability:

- **Network Layer Scalability:** Designing the network layer to handle large networks and high traffic volumes.
- **Routing Protocol Scalability:** Choosing routing protocols that can scale to large networks.

Interoperability:

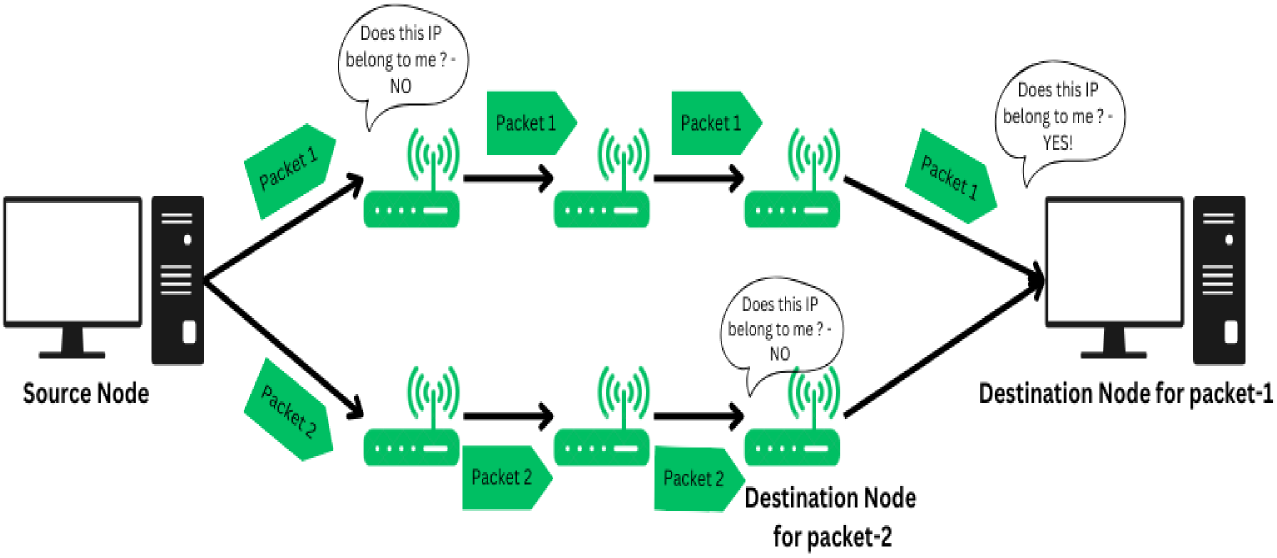
- **Network Layer Interoperability:** Ensuring that different network devices and protocols can interoperate effectively.
- **Protocol Standardization:** Adhering to network layer protocols and standards to ensure interoperability.

What is Routing?

Routing refers to the process of directing a data packet from one node to another. It is an autonomous process handled by the network devices to direct a data packet to its intended destination. Note that, the node here refers to a network device called - 'Router'.

Routing is a crucial mechanism that transmits data from one location to another across a network. The process of routing involves making various routing decisions to ensure reliable & efficient delivery of the data packet by finding the shortest path using various routing metrics which we will be discussing in this article.

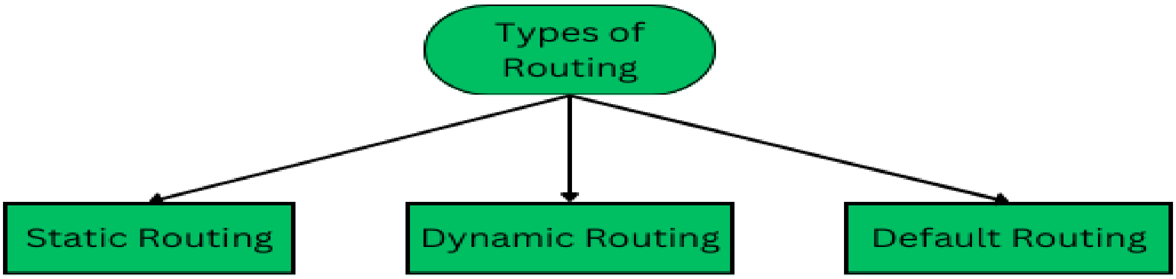
Routing of a data packet is done by analysing the destination IP Address of the packet. Look at the below image:


Does this IP belong to me ? - NO
Packet 1
Packet 1
Packet 1
Does this IP belong to me ? - YES!
Packet 1
Source Node
Does this IP belong to me ? - NO
Destination Node for packet-1
Packet 2
Packet 2
Packet 2
Destination Node for packet-2

- The Source Node (Sender) sends the data packet on the network, embedding the IP in the header of the data packet.
- The nearest router receives the data packet, and based on some metrics, further routes the data packet to other routers.
- Step 2 occurs recursively till the data packet reaches its intended destination.

What are Different Types of Routing?

Routing is typically of 3 types, each serving its purpose and offering different functionalities.



1. Static Routing

Static routing is also called as "non-adaptive routing". In this, routing configuration is done manually by the network administrator. Let's say for example, we have 5 different routes to transmit data from one node to another, so the network administrator will have to manually enter the routing information by assessing all the routes.

- A network administrator has full control over the network, routing the data packets to their concerned destinations
- Routers will route packets to the destination configured manually by the network administrator.
- Although this type of routing gives fine-grained control over the routes, it may not be suitable for large-scale enterprise networks.

2. Dynamic Routing

Dynamic Routing is another type of routing in which routing is an autonomous procedure without any human intervention. Packets are transmitted over a network using various shortest-path algorithms and pre-determined metrics. This type of routing is majorly preferred in modern networks as it offers more flexibility and versatile functionality.

- It is also known as adaptive routing.
- In this, the router adds new routes to the routing table based on any changes made in the topology of the network.
- The autonomous procedure of routing helps in automating every routing operation from adding to removing a route upon updates or any changes made to the network.

3. Default Routing

Default Routing is a routing technique in which a router is configured to transmit packets to a default route that is, a gateway or next-hop device if no specific path is defined or found. It is commonly used when the network has a single exit point. The IP Router has the following address as the default route: 0.0.0.0/0.

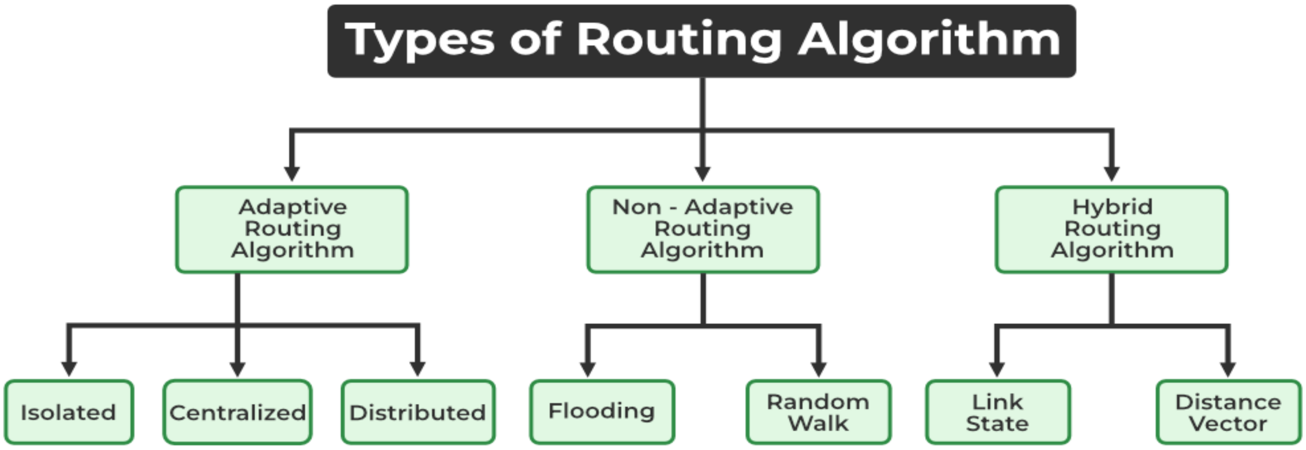
Routing Algorithms

Routing is the process of establishing the routes that data packets must follow to reach the destination. In this process, a routing table is created which contains information regarding routes that data packets follow. Various routing algorithms are used for the purpose of deciding which route an incoming data packet needs to be transmitted on to reach the destination efficiently.

Classification of Routing Algorithms

The routing algorithms can be classified as follows:

- Adaptive Algorithms
- Non-Adaptive Algorithms
- Hybrid Algorithms



Routing algorithms can be classified into various types such as distance vector, link state, and hybrid routing algorithms. Each has its own strengths and weaknesses depending on the network structure.

1. Adaptive Algorithms

These are the algorithms that change their routing decisions whenever network topology or traffic load changes. The changes in routing decisions are reflected in the topology as well as the traffic of the network. Also known as dynamic routing, these make use of dynamic information such as current topology, load, delay, etc. to select routes. Optimization parameters are distance, number of hops, and estimated transit time.

Further, these are classified as follows:

- **Isolated:** In this method each, node makes its routing decisions using the information it has without seeking information from other nodes. The sending nodes don’t have information about the status of a particular link. The disadvantage is that packets may be sent through a congested network which may result in delay. Examples: Hot potato routing, and backward learning.
- **Centralized:** In this method, a centralized node has entire information about the network and makes all the routing decisions. The advantage of this is only one node is required to keep the information of the entire network and the disadvantage is that if the central node goes down the entire network is done. The link state algorithm is

referred to as a centralized algorithm since it is aware of the cost of each link in the network.

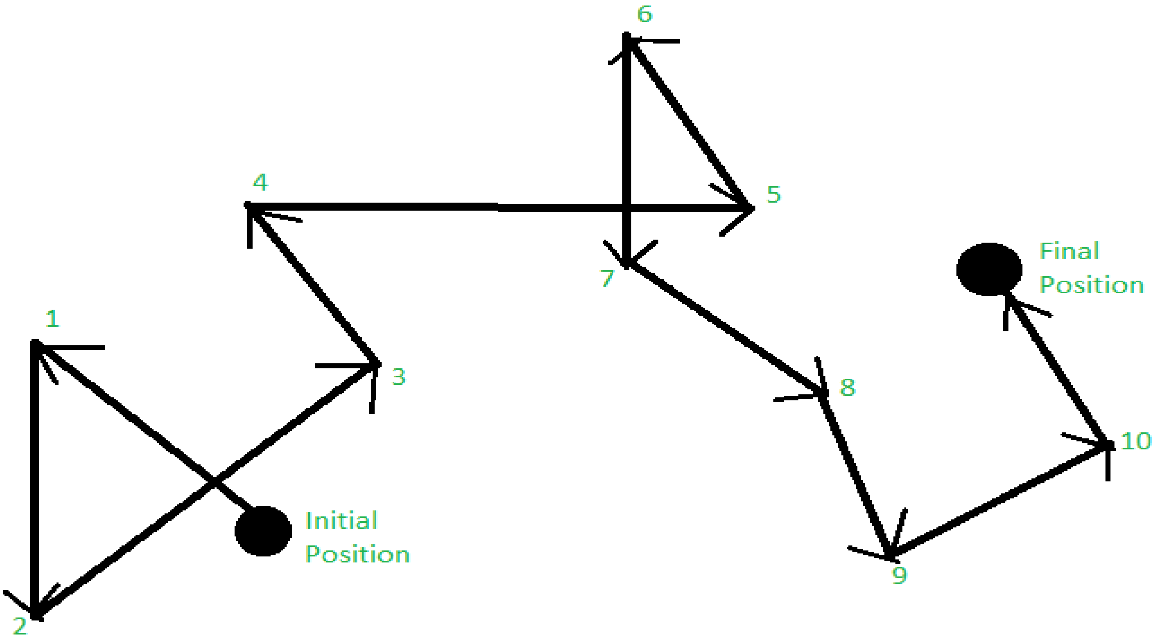
- **Distributed:** In this method, the node receives information from its neighbors and then takes the decision about routing the packets. A disadvantage is that the packet may be delayed if there is a change in between intervals in which it receives information and sends packets. It is also known as a decentralized algorithm as it computes the least-cost path between source and destination.

2. Non-Adaptive Algorithms

These are the algorithms that do not change their routing decisions once they have been selected. This is also known as static routing as a route to be taken is computed in advance and downloaded to routers when a router is booted.

Further, these are classified as follows:

- **Flooding:** This adapts the technique in which every incoming packet is sent on every outgoing line except from which it arrived. One problem with this is that packets may go in a loop and as a result of which a node may receive duplicate packets. These problems can be overcome with the help of sequence numbers, hop count, and spanning trees.
- **Random walk:** In this method, packets are sent host by host or node by node to one of its neighbors randomly. This is a highly robust method that is usually implemented by sending packets onto the link which is least queued.



3. Hybrid Algorithms

As the name suggests, these algorithms are a combination of both adaptive and non-adaptive algorithms. In this approach, the network is divided into several regions, and each region uses a different algorithm.

Further, these are classified as follows:

- Link-state: In this method, each router creates a detailed and complete map of the network which is then shared with all other routers. This allows for more accurate and efficient routing decisions to be made.
- Distance vector: In this method, each router maintains a table that contains information about the distance and direction to every other node in the network. This table is then shared with other routers in the network. The disadvantage of this method is that it may lead to routing loops.

Internetworking-Concept and Architecture

What is Internetworking in a Computer Network?

The word “internetworking,” which combines the words “inter” and “networking,” denotes a connection between completely distinct nodes/segments. This connection is made possible by intermediary hardware like routers or gateways. Catenet was the initial title for associate degree internetwork. Private, public, commercial, industrial, and governmental networks frequently connect to one another. Therefore, a degree of internetwork could be a collection of several networks that operate as a single large network and are connected by intermediate networking devices. The trade, goods, and methods used to address the difficulty of creating and managing internet works are referred to as internetworking.

How Does Internetworking Work?

Every network node or phase is built using a similar protocol or a communication logic, such as TCP (Transfer Control Protocol) or IP (Internet Protocol), to enable communication. It is referred to as “internetworking” when a network interacts with another network using ongoing communication protocols. A packet of information must be delivered across numerous links, which is a problem that internetworking was created to address.

The distinction between expanding the network and internetworking is quite slight. A simple extension of a LAN is the use of a switch or hub to join two local area networks, but connecting them via a router is an example of internetworking. The OSI-ISO model’s Layer 3 (Network Layer) enforces internetworking. The internet is the most prominent famous example of internetworking.

Types of Internetworking

Internetworking primarily consists of three units: Extranet, Internet, and Intranet. Internet connections may or may not be present on intranets and extranets. The computer network or the extranet area unit is typically protected from being accessed from the internet if it is not approved and if there is a link to the internet. Although it should serve as a portal for access to portions of the associate degree extranet, the internet is not considered to be a part of the computer network or extranet.

Extranet

It's a network of the internetwork with a confined scope to one organisation or institution but with limited links to one or more other networks on occasion; however, this is not always the case. It is the lowest degree of internet usage and is typically prohibited in extremely private areas. An extranet may also be referred to as a MAN, WAN, or another type of network, but it cannot include a single local area network; rather, it must make at least one mention of an external network.

Internet

Internet is a specific internetworking that connects governmental, academic, public, and private networks on a global scale. It is based on the ARPANET, which was created by the ARPA (Advanced Research Projects Agency) of the U.S. Defense Department. It is also the location of the World Wide Web (WWW) and is referred to as the "Internet" to distinguish it from other generic internetworking. Internet users and their service providers utilise IP addresses obtained from address registries that control assignments.

Intranet

This computer network can be a collection of interconnected networks that employ the Internet Protocol and IP-based software like web browsers as well as FTP tools, all of which are controlled by a single body entity. This body entity blocks access to the computer network for the rest of the world and only allows a select few users. This network most frequently refers to the internal network of a business or other enterprise. To provide users with browseable data, a large computer network can typically have its own internet server.

Why Internetworking?

A few major issues, including duplicated resources, separated LANs, and a lack of network administration, have led to the evolution of internetworking. Transmission issues across completely distinct offices or departments were caused by isolated LANs. Duplication of resources required distinct support staff and continual hardware and code to be delivered to each business or department. Due to the absence of network administration, there was no centralised system for managing or troubleshooting networks.

At the link layer of a networking model, which is the hardware-centric layer present below the number of TCP/IP logical interfaces, another type of network interconnection typically occurs between businesses. Network switches and bridges are used to connect the various networks. However, the resulting system is essentially a larger, single subnetwork, no internetworking protocol, such as web protocol, is required to traverse these devices. It is often wrongly referred to as “internetworking.”

However, by segmenting the network into phases and then logically splitting the segment traffic using routers, one electronic network can also be reborn into an associated internetwork. The Internet Protocol is designed to provide a network with an unstable packet service. The approach avoids using intermediate network components to keep the network in any state. Instead, the endpoints of each communication session are given this task. Applications should use some suitable Transport Layer protocol, like TCP, which offers a dependable stream, in order to send information correctly. User Datagram Protocol (UDP), a less complex, connection-free transport protocol, is used by some apps for tasks like voice chat and video streaming that don’t require timely or accurate information transmission.

Internetwork Addressing

The internetwork addresses set up devices singly or collectively. Depending on the protocol family and because of the OSI layer, addressing strategies vary. DLL, MAC addresses, and network-layer addresses are the three types of internetwork address area units that are typically employed.

DLL Addresses

All the physical network associations of network devices are clearly identified by a data-link layer address. Area units are frequently used as physical addresses or hardware addresses in data-link addresses. Data-link addresses can occasionally be found within a flat address space and are pre-configured with a fixed relationship to a particular device. End systems typically only have one data-link address since they only have one physical network association. As a result of having many physical network connections, routers and other internetworking equipment frequently have various data-link addresses.

MAC Addresses

Data-link layer addresses are included in MAC addresses. MAC addresses create network entities in LANs that use the data-link layer’s IEEE MAC addresses. For each local area network interface, a unique MAC address designates a particular area unit. MAC addresses are expressed as twelve hexadecimal numbers and are forty-eight bits long. The Organisational Unique Identifier (OUI) is made up of the first 12 hexadecimal digits, which are typically managed by the IEEE and identify the maker or seller.

The interface serial variety or the other price set by a specific merchant would be represented by the final half a dozen positional notation digits. When an interface card initialises, MAC addresses are routinely traced into RAM from ROM, where they are known as burned-in addresses (abbreviated as BIAs).

Network Layer Addresses

The network addresses can occasionally be seen in both gradable address areas and the more common virtual or logical address area units. The relationship between the network address and the tool is logical and flexible; it typically depends either on the properties of the physical network or on groupings without any physical foundation. For each network-layer protocol that a finished system supports, a network-layer address is required. For each supported network-layer protocol, routers and other internetworking devices require a single network-layer address for every physical network association.

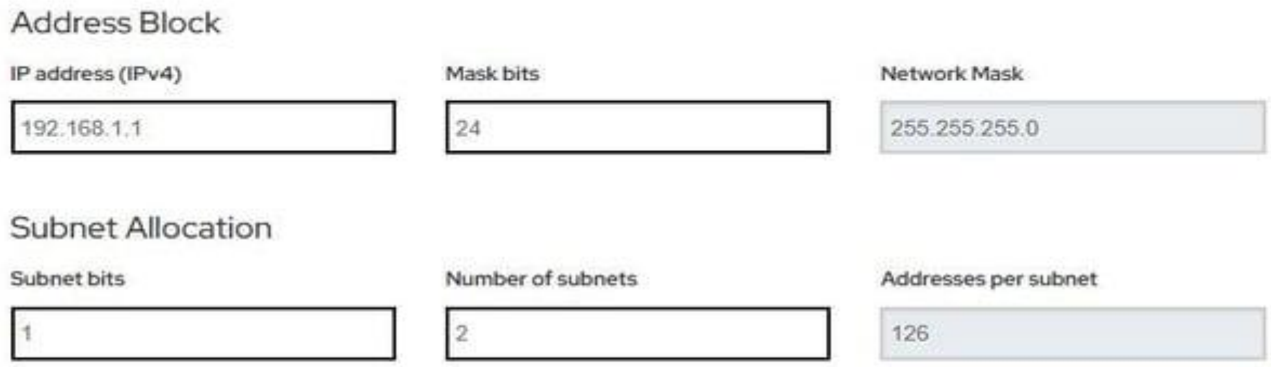
Challenges to Internetworking

There is no guarantee that useful internetwork will be implemented. There are many difficult fields, especially in the ones of dependability, connection, adaptability, and network management. However, each and every one of these fields is crucial to the creation of an efficient and cost-effective internetwork. The challenges to internetworking include:

- The first difficulty arises when we attempt to link several systems in order to allow communication among various technologies. For instance, completely distinct websites may employ various media or function at various speeds.
- Reliable service that must be maintained in the internetwork is another crucial consideration. Organisations as a whole and individual users alike rely on regular, dependable access to network resources.
- Centralised assistance and internet network troubleshooting should be provided via network management. For the network to operate smoothly, configuration, security, performance, and other issues need to be addressed properly.
- The most significant factor, flexibility, is crucial for network expansion as well as new applications and services.

Addressing-IP Addressing and Subnet Masking

IP addressing and subnetting are central to this communication, and they play a critical role in ensuring efficient network management, performance optimization, and security. IP addressing provides a unique identifier for devices on a network, allowing them to communicate effectively, while subnetting divides larger networks into smaller, manageable segments.



Address Block

IP address (IPv4)	Mask bits	Network Mask
192.168.1.1	24	255.255.255.0

Subnet Allocation

Subnet bits	Number of subnets	Addresses per subnet
1	2	126

This segmentation not only enhances performance by reducing congestion but also improves security by limiting access to sensitive data. As networks grow and become increasingly complex, mastering IP addressing and subnetting becomes crucial for optimizing resources, maintaining seamless connectivity, and safeguarding against potential threats.

Difference Between IPv4 and IPv6

The two versions of IP addresses most commonly used today are IPv4 and IPv6.

- **IPv4 (Internet Protocol version 4)**: This is the most widely used format and consists of four sets of numbers separated by periods (e.g., 192.168.1.1). IPv4 provides approximately 4.3 billion unique addresses, which is no longer sufficient given the exponential growth of internet-connected devices.
- **IPv6 (Internet Protocol version 6)**: To address the limitations of IPv4, IPv6 was developed. It uses a much larger address space with hexadecimal numbers separated by colons (e.g., 2001:0db8:85a3:0000:0000:8a2e:0370:7334). IPv6 can support an almost infinite number of devices, making it the future of IP addressing.

What is IP Addressing?

IP addressing refers to the system used to assign unique identifiers to devices on a network. Every device that connects to the internet or a local network requires an IP (Internet Protocol) address to communicate.

IP addresses are crucial because they help route data from one device to another, ensuring that packets of information reach the correct destination.

In simple terms, an IP address is like a home address for devices in a network, allowing them to send and receive data accurately.

Example of IP Addressing:

Every house on your street has a street number: 123 FIrst street

This street number is used by the US postal service, delivery drivers, and police and fire to make sure that everyone knows your unique “address”. Nobody has the same address on your street.

Ex. 192.168.10.107

What is Subnetting?

Subnetting is the process of dividing a large network into smaller, more manageable sub-networks, or "subnets." By breaking a network into subnets, organizations can efficiently allocate IP addresses, optimize network performance, and enhance security.

Each subnet operates as its own smaller network, reducing congestion and improving overall efficiency.

Example of Subnetting

If you needed to add 10,000 more houses to our city, you wouldn’t want to put them on the same street – you’d have a VERY long street.

The solution to this is to create more streets, and cut them up into manageable chunks that are logical.

Maybe you have 300 houses on First Street, and 400 houses on Second street, etc.

With this type of design, every house is still completely unique, you just need a street name and the street number.

Why is Subnetting Important in Networking?

Subnetting is essential for several reasons:

- **Efficient IP Management**: Subnetting helps prevent the exhaustion of IP addresses by dividing large networks into smaller segments, allowing more precise allocation of IPs.
- **Performance Optimization**: By creating smaller, segmented networks, data traffic is reduced, leading to improved performance and minimized network congestion.
- **Enhanced Security**: Subnetting allows organizations to isolate critical systems and data into secure subnets, reducing the risk of unauthorized access.

Understanding IP Address Classes and Ranges

IP addresses are divided into five classes, each designed for different network sizes and use cases.

- **Class A**: Supports large networks (1.0.0.0 to 126.255.255.255).
- **Class B**: For medium-sized networks (128.0.0.0 to 191.255.255.255).
- **Class C**: For smaller networks (192.0.0.0 to 223.255.255.255).
- **Class D**: Reserved for multicasting (224.0.0.0 to 239.255.255.255).
- **Class E**: Reserved for experimental purposes (240.0.0.0 to 255.255.255.255).

To determine the network and host portions of an IP address, subnet masks are used. The network portion identifies the specific subnet, while the host portion identifies individual devices within that subnet.

How to Subnet a Network

Subnetting a network involves a step-by-step process:

1. **Determine the number of required subnets**: Identify how many subnets are needed based on organizational requirements.
2. **Calculate the subnet mask**: Use CIDR (Classless Inter-Domain Routing) notation to calculate the correct subnet mask (e.g., /24).
3. **Divide the network**: Based on the subnet mask, break the larger network into smaller subnets.
4. **Assign IP addresses**: Allocate the appropriate IP ranges to each subnet.

- Understanding CIDR notation is essential for this process. For example, a /24 network allows for 256 IP addresses, while a /28 network allows for only 16 IP addresses.
- For this sample subnet you will have 256 IP addresses, but there are addresses that cannot be used. For example, you cannot use the .0 address (ex: 192.168.1.0) as an address, as it is the network "name".
- You also cannot use the broadcast address. This is the highest address in the range (ex: 192.168.1.255) as this address is used to address all the computers on that subnet.
- As a result of the above two, there are now only 254 usable IP addresses in this subnet.
- You will also need to reserve one IP address as the router to get to other subnets. Typically, most people reserve the first address for the router: 192.168.1.1.
- Thus, for client computers, there are only 253 IP addresses that can be allocated.

Calculating Subnet Masks and IP Ranges

To calculate subnet masks, follow these steps

Step 1: Identify the required number of subnets and hosts.

Step 2: Use the formula 2^n (where n is the number of bits borrowed) to calculate the number of subnets.

Step 3: Calculate the valid IP range for each subnet, ensuring the network and broadcast addresses are excluded.

For example, if you're subnetting a /24 network into smaller subnets, a /28 mask would allow for 16 IP addresses, with 14 usable for hosts.

Best Practices for Subnetting in Large Networks for Enterprise

Subnetting in large networks can be challenging. Here are some best practices to consider:

- **Efficient IP management**: Use hierarchical subnetting to allocate IPs based on organizational structure (e.g., departments or locations).
- **Security-focused subnetting**: Implement subnetting as part of your security strategy to isolate sensitive data or critical systems.
- **Regular audits**: Continuously monitor and audit IP address allocations to ensure efficient usage.

Common Subnetting Mistakes and How to Avoid Them

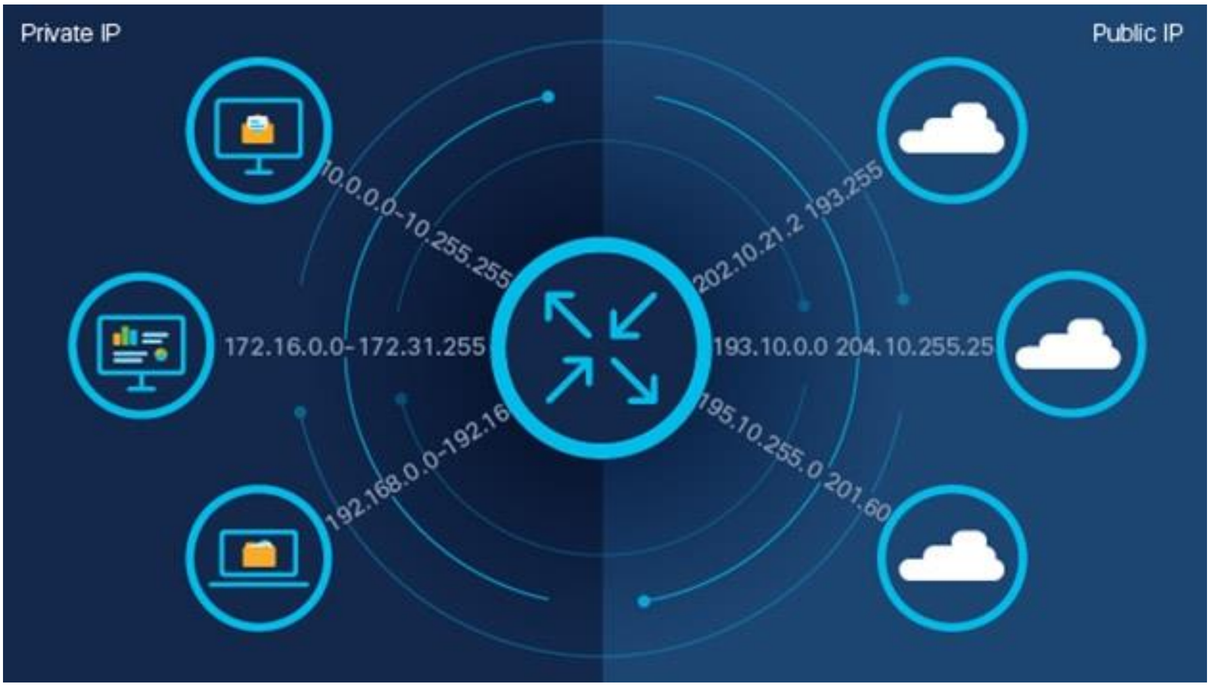
Some common mistakes in subnetting include:

- **Overlapping subnets**: Ensure that subnets are properly segmented to avoid conflicts.
- **Inaccurate subnet mask calculations**: Double-check calculations to prevent misallocation of IP addresses.
- **Wasting IP addresses**: Optimize IP usage by selecting subnet sizes that closely match your needs.

Avoid these mistakes by carefully planning and reviewing your subnetting strategy.

Network Address Translation (NAT)

Network Address Translation (NAT) is a service that enables private IP networks to use the internet and cloud. NAT translates private IP addresses in an internal network to a public IP address before packets are sent to an external network.



How does NAT work?

Network Address Translation (NAT) is a service that operates on a router or edge platform to connect private networks to public networks like the internet. NAT is often implemented at the WAN edge router to enable internet access in core, campus, branch, and colocation sites.

With NAT, an organization needs one IP address or one limited public IP address to represent an entire group of devices as they connect outside their network. Port Address Translation (PAT) enables one single IP to be shared by multiple hosts using IP and port address translation.

What are Private IP Addresses?

As the internet became more popular years ago, the organization that manages IP addresses, known as the Internet Assigned Numbers Authority (IANA) realized that they needed to do something. So, they created a network address translation scheme. This scheme is described in a document called Request for Comments (RFC) 1918. This is just one document of thousands that define how the internet works. If you want to learn about NAT, this is the document that all router manufactures must implement. No matter what type of NAT you use, you will be using RFC 1918 addresses.

If you were to try to send an RFC 1918 private IP address onto the internet, it would be much like sending a physical piece of mail with the return address of "anonymous," yet requesting return service notification. If you were to try doing that with a snail mail service, you would never get that return service notification, because the service wouldn't be able to tell where "anonymous" even is.

NAT Types

There are three different types of NATs. People and organizations use them for different reasons, but they all still work as a NAT.

Static NAT

When the local address is converted to a public one, this NAT chooses the same one. This means there will be a consistent public IP address associated with that router or NAT device.

Dynamic NAT

Instead of choosing the same IP address every time, this NAT goes through a pool of public IP addresses. This results in the router or NAT device getting a different address each time the router translates the local address to a public address.

PAT

PAT stands for port address translation. It's a type of dynamic NAT, but it bands several local IP addresses to a singular public one. Organizations that want all their employees' activity to use a singular IP address use a PAT, often under the supervision of a network administrator.

Why Use NAT?

NAT is a straightforward process. Most routing equipment you purchase at a store will implement it automatically, or with a simple click of a mouse. Let's get a bit deeper into NAT's role in IP conservation and explain its limited role in providing security services.

Advantages of Network Address Translation

Advantages of NAT

Address conservation. NAT conserves IP addresses that are legally registered and prevents their depletion.

Network address translation security. NAT offers the ability to access the internet with more security and privacy by hiding the device IP address from the public network, even when sending and receiving traffic. NAT rate-limiting allows users to limit the maximum number of concurrent NAT operations on a router and rate limit the number of NAT translations. This provides more control over the use of NAT addresses, but can also be used to limit the effects of worms, viruses, and denial-of-service (DoS) attacks. Dynamic NAT implementation creates a firewall between the internal network and the internet automatically. Some NAT routers offer traffic logging and filtering.

Flexibility. NAT provides flexibility; for example, it can be deployed in a public wireless LAN environment. Inbound mapping or static NAT allows external devices to initiate connections to computers on the stub domain in some cases.

Simplicity. Eliminates the need to renumber addresses when a network changes or merges. Network address translation allows you to create an inside network virtual host to coordinate TCP load-balancing for internal network servers.

Speed. Compared to proxy servers, NAT is transparent to both destination and source computers, allowing for quicker direct dealing. In addition, proxy servers typically work at the transport layer or layer 4 of the OSI Reference Model or higher, making them slower than network address translation, which is a network layer or layer 3 protocol.

Scalability. NAT and dynamic host configuration protocol (DHCP) work well together, with the DHCP server doling out unregistered IP addresses for the stub domain from the list as necessary. Scaling up is easier, since you can increase the available range of IP addresses the DHCP configures to make room for additional network computers immediately instead of requesting more IP addresses from IANA as needs increase.

Multi-homing. Multiple connections to the internet, called multi-homing, helps maintain a reliable connection and reduces the chance of a shutdown in case of a failed connection. This also enables load-balancing via reducing the number of computers using any single connection. Multi-homed networks often connect to multiple ISPs, each assigning a range of IP addresses or a single IP address to the organization. Routers use network address translation to route between networks using different network address translation protocols. In a multi-homed network, the router uses part of the TCP/IP protocol suite, the border gateway protocol (BGP), to communicate; the stub domain side uses internal BGP or IBGP, and routers communicate with each other using external BGP or EBGP. Multi-homing reroutes all data through another router should one of the connections to an ISP fail.

Disadvantages of NAT

Resource consumption. Network address translation is a technology that consumes memory resources and processor space, because it must translate IPv4 addresses for all outgoing and incoming IPv4 datagrams and retain the details from translation in memory.

Delays. Path delays are caused by translation results in switching path delays.

Functionality. Some applications and technologies will not function as expected with NAT enabled.

Traceability. Network address translation complicates protocols for tunneling. IPsec is the secure protocol recommended for network address translation.

Layer issue. A router is a device for the network layer, yet as a NAT device it is required to tamper with the transport layer in the form of port numbers.

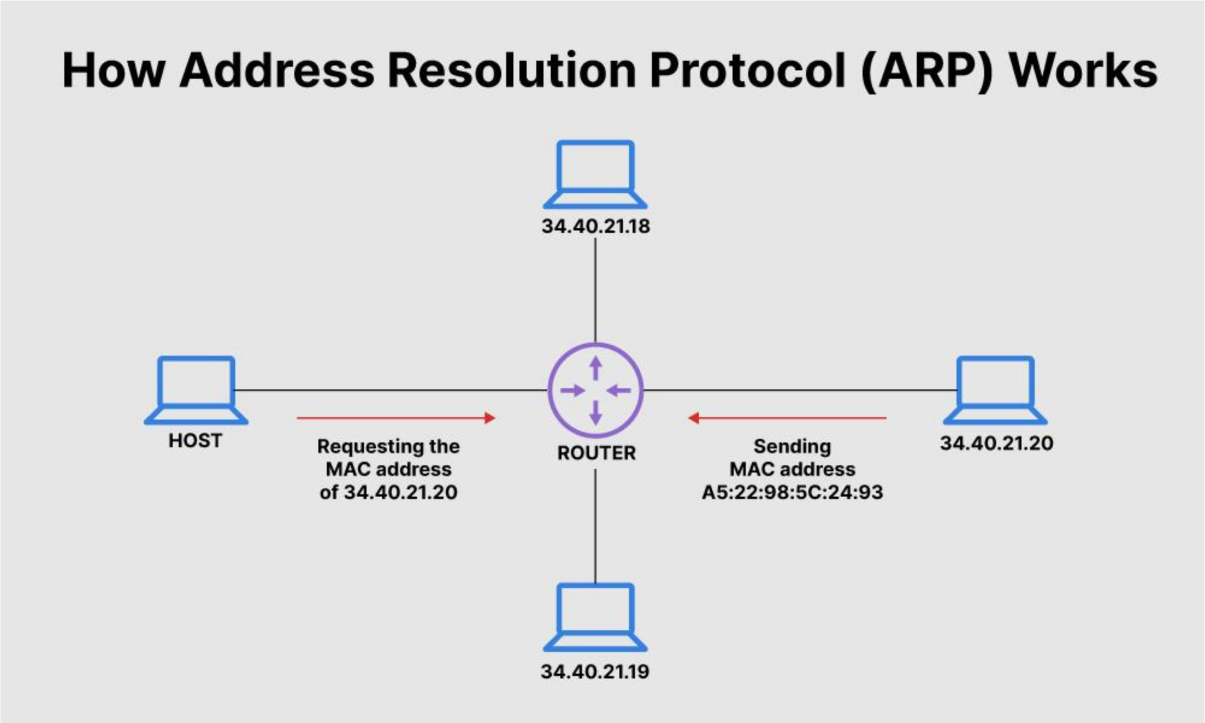
Address Resolution Protocol (ARP)

Address Resolution Protocol (ARP) is a protocol or procedure that connects an ever-changing Internet Protocol (IP) address to a fixed physical machine address, also known as a media access control (MAC) address, in a local-area network (LAN).

This mapping procedure is important because the lengths of the IP and MAC addresses differ, and a translation is needed so that the systems can recognize one another. The most used IP today is IP version 4 (IPv4). An IP address is 32 bits long. However, MAC addresses are 48 bits long. ARP translates the 32-bit address to 48 and vice versa.

There is a networking model known as the Open Systems Interconnection (OSI) model. First developed in the late 1970s, the **OSI model** uses layers to give IT teams a visualization of what is going on with a particular networking system. This can be helpful in determining which layer affects which application, device, or software installed on the network, and further, which IT or engineering professional is responsible for managing that layer.

The MAC address is also known as the data link layer, which establishes and terminates a connection between two physically connected devices so that data transfer can take place. The IP address is also referred to as the network layer or the layer responsible for forwarding packets of data through different routers. ARP works between these layers.


How Address Resolution Protocol (ARP) Works
34.40.21.18
HOST
Requesting the MAC address of 34.40.21.20
ROUTER
Sending MAC address A5:22:98:5C:24:93
34.40.21.20
34.40.21.19

What Does ARP Do and How Does It Work?

When a new computer joins a local area network (LAN), it will receive a unique IP address to use for identification and communication.

Packets of data arrive at a gateway, destined for a particular host machine. The gateway, or the piece of hardware on a network that allows data to flow from one network to another, asks the ARP program to find a MAC address that matches the IP address. The ARP cache keeps a list of each IP address and its matching MAC address. The ARP cache is dynamic, but users on a network can also configure a static ARP table containing IP addresses and MAC addresses.

ARP caches are kept on all operating systems in an IPv4 Ethernet network. Every time a device requests a MAC address to send data to another device connected to the LAN, the device verifies its ARP cache to see if the IP-to-MAC-address connection has already been completed. If it exists, then a new request is unnecessary. However, if the translation has not yet been carried out, then the request for network addresses is sent, and ARP is performed.

An ARP cache size is limited by design, and addresses tend to stay in the cache for only a few minutes. It is purged regularly to free up space. This design is also intended for privacy and security to prevent IP addresses from being stolen or spoofed by cyberattackers. While MAC addresses are fixed, IP addresses are constantly updated.

In the purging process, unutilized addresses are deleted; so is any data related to unsuccessful attempts to communicate with computers not connected to the network or that are not even powered on.

What Is Address Resolution Protocol's Relationship with DHCP And DNS? How Do They Differ?

ARP is the process of connecting a dynamic IP address to a physical machine's MAC address. As such, it is important to have a look at a few technologies related to IP.

As mentioned previously, IP addresses, by design, are changed constantly for the simple reason that doing so gives users security and privacy. However changes on IP addresses should not be completely random. There should be rules that allocate an IP address from a defined range of numbers available in a specific network. This helps prevent issues, such as two computers receiving the same IP address. The rules are known as DHCP or Dynamic Host Configuration Protocol.

IP addresses as identities for computers are important because they are needed to perform an internet search. When users search for a domain name or Uniform Resource Locator (URL), they use an alphabetical name. Computers, on the other hand, use the numerical IP address to associate the domain name with a server. To connect the two, a Domain Name System (DNS) server is used to translate an IP address from a confusing string of numbers into a more readable, easily understandable domain name, and vice versa.

What Are the Types of ARP?

There are different versions and use cases of ARP. Let us take a look at a few.

Proxy ARP

Proxy ARP is a technique by which a proxy device on a given network answers the ARP request for an IP address that is not on that network. The proxy is aware of the location of the traffic's destination and offers its own MAC address as the destination.

Gratuitous ARP

Gratuitous ARP is almost like an administrative procedure, carried out as a way for a host on a network to simply announce or update its IP-to-MAC address. Gratuitous ARP is not prompted by an ARP request to translate an IP address to a MAC address.

Reverse ARP (RARP)

Host machines that do not know their own IP address can use the Reverse Address Resolution Protocol (RARP) for discovery.

Inverse ARP (IARP)

Whereas ARP uses an IP address to find a MAC address, IARP uses a MAC address to find an IP address.

PCET's

Pimpri Chinchwad University

School of Computer Application

PMC113A Computer Networks

Unit No. 05 Transport Layer

Introduction:

The transport layer is the fourth layer of the OSI model and is the core of the Internet model. It responds to service requests from the session layer and issues service requests to the network Layer. The transport layer provides transparent transfer of data between hosts. It provides end-to-end control and information transfer with the quality of service needed by the application program. It is the first true end-to-end layer, implemented in all End Systems (ES).

Transport Layer Functions / Services

The transport layer is located between the network layer and the application layer. The transport layer is responsible for providing services to the application layer; it receives services from the network layer. The services that can be provided by the transport layer are

1. Process-to-Process Communication
2. Addressing: Port Numbers
3. Encapsulation and Decapsulation
4. Multiplexing and Demultiplexing
5. Flow Control
6. Error Control
7. Congestion Control

Transport Layer: Design issues

The Transport Layer in networking plays a critical role in ensuring reliable end-to-end communication between applications running on different hosts across a network. Its design involves addressing several key issues to achieve efficient and effective data transmission.

Here are the design issues in detail:

1. Reliability:

- **Objective:** Ensure that data delivered from the sender to the receiver is accurate and in the correct order, without errors or duplication.
- **Techniques:**

    - **Acknowledgment and Retransmission:** Use of acknowledgments (ACKs) from the receiver to confirm receipt of data segments. If ACK is not received within a timeout period, the sender retransmits the segment.
    - **Sequence Numbers:** Assign sequence numbers to data segments to ensure they are received in the correct order.
    - **Checksums:** Include error-checking mechanisms (like CRC or checksums) to detect errors and corrupted data segments.

2. Flow Control:

- **Objective:** Regulate the amount of data sent by the sender to match the receiver's processing capability, preventing overflow and ensuring smooth transmission.
- **Techniques:**
    - **Sliding Window Protocol:** Allows the sender to transmit multiple segments before receiving acknowledgments, optimizing throughput while adhering to the receiver's buffer capacity.
    - **Buffer Management:** Manage buffer sizes at both sender and receiver ends to handle varying network conditions and traffic loads effectively.

3. Multiplexing and Demultiplexing:

- **Objective:** Enable multiple applications or services on the same host to communicate simultaneously over the network.
- **Techniques:**
    - **Port Numbers:** Use port numbers to distinguish different communication streams (e.g., TCP/UDP ports like 80 for HTTP, 443 for HTTPS).
    - **Socket Pair:** Combination of IP address and port number uniquely identifies a communication endpoint (socket) in a network.

4. Connection Management:

- **Objective:** Establish, maintain, and terminate logical connections between applications running on different hosts.
- **Techniques:**
    - **Connection-Oriented Protocols:** Like TCP establish a reliable connection through a three-way handshake (SYN, SYN-ACK, ACK) before data exchange and tear down the connection gracefully.
    - **Connectionless Protocols:** Like UDP do not establish a dedicated connection but simply send data packets independently.

5. Error Handling and Retransmission:

- **Objective:** Detect and recover from errors that occur during data transmission.
- **Techniques:**
    - **Selective Repeat and Go-Back-N:** Two types of ARQ (Automatic Repeat reQuest) protocols used in sliding window mechanisms to handle lost or corrupted data segments.
    - **Timeouts:** Set timers to wait for acknowledgments (ACKs). If an ACK is not received within the timeout period, the sender assumes the segment is lost and retransmits it.

6. Quality of Service (QoS):

- **Objective:** Prioritize traffic and allocate network resources based on application requirements (e.g., latency-sensitive applications like VoIP or video streaming).
- **Techniques:**
    - **Traffic Prioritization:** Assign different priorities to data packets based on QoS parameters (e.g., delay, throughput, jitter).
    - **Congestion Control:** Adjust transmission rates dynamically to avoid network congestion and ensure fair resource allocation among competing flows.

7. Segmentation and Reassembly:

- **Objective:** Divide large data units received from the upper layers into smaller segments for efficient transmission over the network.
- **Techniques:**
    - **Maximum Segment Size (MSS):** Largest amount of data that TCP can send in one segment.
    - **Fragmentation:** Splitting of data packets into smaller units to fit within the maximum transmission unit (MTU) size of the underlying network.

8. Security:

- **Objective:** Ensure data confidentiality, integrity, and availability during transmission.
- **Techniques:**
    - **Encryption:** Use of cryptographic techniques (e.g., SSL/TLS for TCP) to encrypt data to protect against eavesdropping and unauthorized access.
    - **Authentication:** Verification of identities using digital certificates or passwords to prevent unauthorized access or data tampering.

Transport Layer Addressing

Transport Layer addressing is a concept used in the Transport Layer (Layer 4) of the OSI (Open Systems Interconnection) model. This layer is responsible for providing communication services between devices over a network and ensures the reliable transfer of data. The primary components used in Transport Layer addressing are **ports** and **sockets**.

Key Concepts in Transport Layer Addressing:

1. **Port Numbers**:
   - Each device on a network is identified by an **IP address** at the Network Layer (Layer 3). However, when multiple applications or services are running on the same device, the Transport Layer needs a way to distinguish between them. This is where **port numbers** come into play.
   - Port numbers are used to identify specific services or applications on a device, and each port number corresponds to a particular service or process.
   - **Well-known Ports**: Ports 0–1023 are reserved for well-known services (e.g., HTTP on port 80, HTTPS on port 443, FTP on port 21).
   - **Registered Ports**: Ports 1024–49151 are used by applications that are not as widely recognized as the well-known ports.
   - **Dynamic or Private Ports**: Ports 49152–65535 are used for ephemeral connections, often chosen dynamically by the operating system for temporary use.
2. **Socket**:
   - A **socket** is the combination of an **IP address** and a **port number**. It uniquely identifies a communication endpoint in the Transport Layer.
   - A socket enables a process on one machine to communicate with a process on another machine over a network.
   - The format for a socket is typically written as: IP_address: port_number
   - For example, the socket 192.168.1.10:8080 refers to the machine with IP address 192.168.1.10 and the application running on port 8080.
3. **Transport Protocols**:
   - Two main transport protocols are used for communication at the Transport Layer:

        - **TCP (Transmission Control Protocol)**: Provides reliable, connection-oriented communication. It ensures data is delivered in order and retransmits lost data.
        - **UDP (User Datagram Protocol)**: Provides connectionless communication. It does not guarantee reliability or order of data delivery, making it faster but less reliable than TCP.

4. **Communication Process**:
    - **Connection-oriented (TCP)**: A connection is established between two devices using a three-way handshake, where port numbers and IP addresses are exchanged to create a dedicated communication channel.
    - **Connectionless (UDP)**: Data is sent as packets without a prior connection setup, and port numbers still help identify the destination application.

Example of Transport Layer Addressing:

When a user accesses a website, the browser on the user's computer typically uses the following:

- The **IP address** of the web server (say, 192.168.1.100).
- The **port number** for HTTP (port 80).
- This information forms a socket, such as 192.168.1.100:80, which is used to establish communication.

Buffering

Buffering refers to the process of temporarily storing data in memory (a buffer) before it is sent or received over a network. The buffer helps to smooth out any variations in the rate at which data can be sent or received, and ensures that the sender and receiver can work at their own pace without interruption.

For example, when streaming a video, the video player may buffer some of the video before it starts playing. This is done so that if the network connection is slow, the video can continue playing without interruption. Similarly, when a computer is sending data over a network, it may buffer the data before sending it to prevent the network from becoming overloaded.

There are two main types of buffering: input buffering and output buffering.

1. Input buffering occurs when data is received faster than it can be processed and it is temporarily stored in a buffer until it can be used.

2. Output buffering occurs when data is generated faster than it can be sent or written to storage, so it is temporarily stored in a buffer until it can be sent or written.

Need of Buffering

Buffering is used to smooth out variations in the rate at which data is received or generated, and to ensure that data is processed efficiently and without interruption. There are several reasons why buffering is necessary −

- **Speed mismatches** − In many computer systems and applications, data is received or generated at different rates. For example, a network connection may be faster than the rate at which data can be processed by the computer, or data may be generated faster than it can be written to storage. Buffering helps to mitigate these speed mismatches by temporarily storing data in memory.
- **Network congestion** − When sending data over a network, it is important to avoid overwhelming the network with too much data at once. Buffering helps to prevent network congestion by temporarily storing data in memory before it is sent.
- **Error handling** − Buffering can also be used to help handle errors that may occur when transmitting or processing data. For example, if an error occurs while sending data over a network, the data can be retransmitted from the buffer rather than being lost.
- **Synchronization** − Buffering also helps to synchronize different processes that may be running simultaneously. For example, when video is being streamed, the video player may buffer some of the video before it starts playing. This ensures that the video can continue playing without interruption even if the network connection is slow.
- **User Experience** − Buffering allows for smooth experience for the user. For example, in video streaming, if buffering is not implemented, video can freeze or get buffered frequently which can be annoying for the user.
- **Power optimization** − buffering allows the system to work at its own pace and avoid power consumption while fetching or sending large chunks of data.

How Buffering is important

Buffering is an important technique for ensuring the efficient and smooth operation of a wide range of computer systems and applications. Some of the ways in which buffering is important include −

- **Maintaining the performance and throughput of computer systems** − Buffering helps to ensure that data can be processed smoothly, even if the rate at which it is received or generated varies. This can improve the overall performance and throughput of a computer system.
- **Improving the quality of user experience** − Buffering helps to ensure that data can be processed and delivered to the user without interruption. This can improve the quality

of the user experience, especially in applications such as video streaming and online gaming.

- **Ensuring data integrity and reliability** − Buffering can also be used to help ensure the integrity and reliability of data transmission and storage. For example, if an error occurs while sending data over a network, the data can be retransmitted from the buffer rather than being lost.
- **Optimizing power consumption** − Buffering can be used to limit the amount of data that needs to be transferred at any given time, thus reducing power consumption for devices.
- **Facilitating effective communication** − Buffering allows data to be temporarily stored before it is sent or received, which can help to prevent network congestion and ensure that data can be transmitted or received efficiently.
- **Supporting multiple processes** − Buffering can also be used to synchronize different processes that may be running simultaneously, which can improve the overall performance and functionality of a computer system.

Multiplexing

Multiplexing in the Transport Layer refers to the technique of combining multiple communication streams (data from different applications) into a single shared communication channel, while still maintaining the isolation and integrity of each stream. This process is crucial for managing multiple connections and ensuring that data from different applications can be transmitted simultaneously over the same network.

In the context of the **Transport Layer** (Layer 4 in the OSI model), multiplexing allows different processes or applications to communicate over the network simultaneously, even if they share the same underlying physical resources. It is achieved by using unique identifiers, like **port numbers**, to distinguish between multiple communication streams.

Key Aspects of Multiplexing in the Transport Layer:

1. **Port Numbers**:
    - Transport layer protocols (e.g., TCP, UDP) use port numbers to identify different services or applications on a host.
    - Each process (application) running on a system is assigned a unique port number. When a message is sent, it contains the destination port number, allowing the receiving system to know which process should handle the data.
2. **Connection-Oriented vs. Connectionless**:

    - In **TCP (Transmission Control Protocol)**, multiplexing ensures that each stream of data is directed to the correct application using port numbers while establishing a connection and maintaining reliable delivery.
    - In **UDP (User Datagram Protocol)**, multiplexing allows multiple applications to communicate without establishing a connection, using port numbers to route data.

3. **Segmenting Data**:
    - In TCP, the data from an application is segmented into packets. The Transport Layer assigns each segment with a header that contains source and destination port numbers, ensuring that data from multiple applications can be transmitted simultaneously without confusion.
4. **Demultiplexing**:
    - On the receiving side, demultiplexing is the reverse process. The transport layer reads the destination port number in the packet and passes the data to the correct application process based on that port number.

Example:

If a user is running a web server (HTTP) and a file transfer protocol (FTP) server on the same machine:

- The HTTP server might listen on port 80.
- The FTP server might listen on port 21. Even though both servers use the same underlying network connection, multiplexing ensures that the data for each service is directed to the correct server process.

Benefits of Multiplexing:

- **Efficient Use of Resources**: Multiple applications can share the same network infrastructure, avoiding the need for separate physical connections.
- **Scalability**: It allows numerous applications to run on the same machine and communicate over the same network connection.
- **Isolation**: Even if multiple connections are active, each application’s data remains isolated, avoiding cross-application interference.

Real-World Example:

- When you browse a website, your browser is making multiple connections to different servers (e.g., downloading images, scripts, etc.), all using HTTP (usually over port 80). Each connection is multiplexed over the same transport layer connection but identified by the corresponding port number.

Recovery

Recovery in the Transport Layer refers to mechanisms used to ensure reliable data transfer, detect and correct errors, and recover from transmission failures. These mechanisms are essential for maintaining data integrity and performance, especially in the presence of network problems such as packet loss, duplication, or corruption.

In the context of **Transport Layer** protocols, **TCP (Transmission Control Protocol)** is typically the focus when discussing recovery mechanisms, as it provides reliable communication. UDP (User Datagram Protocol), on the other hand, is connectionless and does not provide built-in recovery mechanisms, leaving error handling to the application layer.

Here are the main **recovery mechanisms** used in the **Transport Layer**:

1. Error Detection and Correction:

- **Checksums**: Both TCP and UDP use checksums to detect errors in transmitted data. The checksum is a value computed from the data and is sent along with it. The receiver also computes the checksum and compares it to the received checksum. If they don't match, the data is considered corrupted, and the receiver requests a retransmission.
  - In TCP, the checksum is part of the header and is applied to both the data and header.
  - In UDP, the checksum is optional in IPv4 but mandatory in IPv6.

2. Retransmission of Lost Data (TCP):

- **Timeout and Retransmission**: If a segment is lost or if the sender does not receive an acknowledgment (ACK) for a specific segment within a given time window, the sender will **retransmit** the segment. This is done to recover from packet loss.
- **Selective Acknowledgments (SACK)**: TCP can use selective acknowledgments to allow the receiver to inform the sender about exactly which segments were received successfully. This allows the sender to retransmit only the lost segments, rather than all unacknowledged segments.

3. Acknowledgments (TCP):

- **Positive Acknowledgment with Retransmission (PAR)**: In TCP, each segment sent is acknowledged by the receiver. The sender waits for an acknowledgment (ACK) from the receiver, which confirms that the segment has been successfully received.
- **Cumulative Acknowledgment**: TCP uses cumulative acknowledgment, where the receiver acknowledges the highest sequence number of all segments received in order. If any segment is lost or not received, the receiver will not send an acknowledgment for it, prompting the sender to retransmit the missing segment.

4. Flow Control (TCP):

- **Sliding Window**: TCP uses a sliding window mechanism for flow control. It helps manage the amount of data sent before receiving an acknowledgment. If the receiver is overwhelmed, the window size can be adjusted, and the sender must wait for more space before sending more data.
- **Window Size Adjustments**: This mechanism ensures that the sender does not overwhelm the receiver, improving recovery and preventing network congestion.

5. Congestion Control (TCP):

- **Slow Start**: TCP begins transmission with a small amount of data and gradually increases the data rate. If packet loss is detected (often indicating congestion), the sender reduces the transmission rate.
- **Congestion Avoidance**: Once TCP detects congestion (using mechanisms like packet loss or delay), it adjusts its transmission rate by decreasing the sending window size.
- **Fast Retransmit and Fast Recovery**: When TCP detects three duplicate ACKs (which indicates that a packet is lost but other packets have been received), it will retransmit the lost packet immediately and enter the fast recovery phase. This helps to recover from losses more quickly.

6. Sequence Numbers:

- **Sequencing**: TCP uses sequence numbers to label each byte of data in a connection. This helps the receiver properly reorder the data if packets arrive out of order and ensures that no data is missed or duplicated.
- **Duplicate Packet Detection**: If a packet with the same sequence number arrives, it is identified as a duplicate, and the receiver can ignore it, preventing errors in data recovery.

7. Connection Establishment and Termination (TCP):

- **Three-Way Handshake**: The reliable establishment of a TCP connection ensures that both sides are ready to transmit and that data can be recovered if necessary.
  - SYN: The sender sends a synchronization request.
  - SYN-ACK: The receiver acknowledges and agrees to the connection.
  - ACK: The sender acknowledges the connection is established.
- **Graceful Connection Termination**: The termination process (four-way handshake) ensures that data is not lost during the connection shutdown phase.

8. Duplicate Data Handling:

- TCP ensures that duplicate data, which can occur due to retransmissions, is handled by discarding redundant data or reordering out-of-sequence packets. Sequence numbers help in identifying and removing duplicates.

Example of Recovery Process in TCP:

1. A sender sends a series of packets to the receiver.
2. The receiver acknowledges the first three packets but not the fourth, which gets lost in the network.
3. The sender waits for an acknowledgment, but after a timeout, it retransmits the lost fourth packet.
4. The receiver sends a cumulative acknowledgment for all packets up to the fourth one, and the communication continues.

Why Recovery in the Transport Layer is Important:

- **Reliability**: The Transport Layer ensures that data is reliably delivered even in the presence of network errors. Without these mechanisms, data could be lost, delayed, or corrupted.
- **Error Handling**: Transmission errors (e.g., packet loss, corruption) are common in networks, and the Transport Layer provides the tools to recover from these errors efficiently.
- **Efficient Use of Resources**: Through mechanisms like flow control and congestion control, the Transport Layer can recover from congestion or network bottlenecks, ensuring that resources are used efficiently without overwhelming the network.

TCP and IP Suit of Protocols

List of Protocols associated with TCP and IP

1. HTTP (HyperText Transfer Protocol) – Port 80
2. HTTPS (HyperText Transfer Protocol Secure) – Port 443
3. FTP (File Transfer Protocol) – Ports 20, 21
4. SMTP (Simple Mail Transfer Protocol) – Port 25
5. IMAP (Internet Message Access Protocol) – Port 143
6. POP3 (Post Office Protocol v3) – Port 110
7. SSH (Secure Shell) – Port 22
8. Telnet – Port 23
9. DNS (Domain Name System) – Port 53 (for zone transfers)
10. SNMP (Simple Network Management Protocol) – Port 161 (rarely over TCP)
11. TCP (Transmission Control Protocol) – Protocol Number 6
12. UDP (User Datagram Protocol) – Protocol Number 17
13. SCTP (Stream Control Transmission Protocol) – Protocol Number 132

14. IPv4 (Internet Protocol v4) – Protocol Number 4
15. IPv6 (Internet Protocol v6) – Protocol Number 41
16. ICMP (Internet Control Message Protocol) – Protocol Number 1
17. ICMPv6 (Internet Control Message Protocol for IPv6) – Protocol Number 58
18. IGMP (Internet Group Management Protocol) – Protocol Number 2
19. RIP (Routing Information Protocol) – Uses UDP (Port 520)
20. OSPF (Open Shortest Path First) – Protocol Number 89
21. BGP (Border Gateway Protocol) – Uses TCP (Port 179)
22. EIGRP (Enhanced Interior Gateway Routing Protocol) – Protocol Number 88
23. IPSec (Internet Protocol Security) – Protocol Numbers 50 (ESP) & 51 (AH)

HTTP (Hypertext Transfer Protocol)

HTTP (Hypertext Transfer Protocol) is a stateless application-layer protocol used for communication between web browsers and web servers. It follows a client-server model where a client (browser) sends requests to a server, and the server responds with the requested data.

Key Features of HTTP:

1. **Stateless:** Each request is independent, meaning the server does not retain session information between requests.
2. **Request-Response Model:** Clients (e.g., browsers) send requests, and servers send responses with requested resources like HTML, images, and videos.
3. **Human-Readable:** Messages use text-based commands, making debugging easy.
4. **Supports Caching:** Can cache responses to reduce network load and improve performance.
5. **Supports Authentication:** Can use headers for authentication via mechanisms like Basic Auth, Bearer tokens, and OAuth.

HTTP Request Methods (Verbs)

HTTP defines multiple methods to perform different actions:

- GET → Retrieves data from the server (e.g., loading a webpage).
- POST → Sends data to the server (e.g., submitting a form).
- PUT → Updates existing resources or creates them if they don’t exist.
- DELETE → Removes resources from the server.
- PATCH → Partially updates a resource.
- HEAD → Retrieves only headers without the response body.
- OPTIONS → Retrieves supported HTTP methods for a resource.

HTTP Status Codes

Responses from the server include status codes to indicate success or errors:

- 1xx (Informational) → Processing (e.g., 100 Continue).
- 2xx (Success) → Request was successful (e.g., 200 OK, 201 Created).
- 3xx (Redirection) → Further action needed (e.g., 301 Moved Permanently, 302 Found).
- 4xx (Client Errors) → Issue with the request (e.g., 400 Bad Request, 404 Not Found).
- 5xx (Server Errors) → Server-side issues (e.g., 500 Internal Server Error, 503 Service Unavailable).

HTTPS (Hypertext Transfer Protocol Secure)

HTTPS is the secure version of HTTP, where all data transmitted between the client (browser) and server is encrypted using SSL/TLS (Secure Sockets Layer / Transport Layer Security). This ensures that the communication remains private, secure, and tamper-proof.

Key Features of HTTPS

1. **Encryption:** Uses **SSL/TLS** to encrypt data, preventing interception (man-in-the-middle attacks).
2. **Authentication:** Uses **SSL certificates** to verify the server's identity, ensuring users are connecting to the legitimate website.
3. **Data Integrity:** Ensures that data is not altered during transmission.
4. **SEO Benefits:** Search engines like **Google prioritize HTTPS websites** in search rankings.
5. **User Trust:** Websites with HTTPS display a **padlock** icon in the browser, increasing user confidence.

How HTTPS Works

1. **Client (Browser) requests a secure connection** to the server.
2. **Server responds with its SSL certificate** issued by a trusted **Certificate Authority (CA)**.
3. **Client verifies the certificate** (validity, issuer, expiration date, etc.).
4. **TLS Handshake occurs:** Client and server establish a secure encryption key.
5. **Encrypted Communication Begins** – All transmitted data is securely encrypted.

HTTPS vs. HTTP

Feature	HTTP (Port 80)	HTTPS (Port 443)
Security	No encryption	Uses SSL/TLS encryption
Data Protection	Vulnerable to attacks	Protects against eavesdropping and tampering

Feature	HTTP (Port 80)	HTTPS (Port 443)
Authentication	No identity verification	Uses SSL certificates to verify website authenticity
SEO Impact	Lower ranking	Google gives preference to HTTPS
User Trust	No padlock, less secure	Padlock icon, increased trust

FTP (File Transfer Protocol)

FTP (File Transfer Protocol) is a **network protocol** used to transfer files between a **client** and a **server** over a TCP-based network, such as the internet. It operates on **port 21** by default and can be used to upload, download, and manage files on a remote server.

Key Features of FTP

1. **Supports File Upload & Download** – Transfer files between client and server.
2. **User Authentication** – Can require a username and password for access.
3. **Supports Both Active & Passive Modes** – Determines how the connection is established.
4. **Multiple File Operations** – Create, delete, rename, and list directories.
5. **No Encryption (Plain FTP)** – Data is sent in plain text unless secured via FTPS or SFTP.

How FTP Works

FTP follows a **client-server model**, where:

1. The **client** initiates a connection to the **FTP server**.
2. The server **authenticates** the client (if required).
3. The client can **upload/download files** or perform other file operations.
4. Once finished, the client **disconnects** from the server.

SMTP (Simple Mail Transfer Protocol)

SMTP (Simple Mail Transfer Protocol) is a **protocol used to send emails** over the Internet. It works at the **application layer** and ensures that emails are properly delivered from the sender’s mail server to the recipient’s mail server.

SMTP is used for sending emails, not receiving them. For receiving emails, protocols like IMAP (Internet Message Access Protocol) and POP3 (Post Office Protocol v3) are used.

Key Features of SMTP

1. **Email Sending Protocol** – Transfers emails from the sender’s client to the recipient’s server.
2. **Uses TCP for Reliable Communication** – Typically works on **port 25, 465, or 587**.

3. **Works in a Store-and-Forward Model** – Emails are sent to an SMTP server, which then forwards them to the destination.
4. **Supports Authentication (SMTP AUTH)** – Requires a username and password for security.
5. **Supports Secure Communication via SSL/TLS (SMTPS)** – Encrypts email transmission.

How SMTP Works

SMTP follows a **client-server model** and works in three main steps:

1. **Mail Submission:** The email client (Outlook, Gmail, etc.) sends an email to the sender's SMTP server.
2. **Mail Relay & Forwarding:** The SMTP server processes the email and **relays** it to the recipient's email server using **DNS (MX Records)**.
3. **Mail Delivery:** The recipient's mail server **stores the email** until it is retrieved via **IMAP or POP3**.

IMAP (Internet Message Access Protocol)

IMAP (Internet Message Access Protocol) is a protocol used to retrieve and manage emails from a mail server. Unlike POP3, IMAP allows users to access and manage emails directly on the server without downloading them. This is useful for accessing email from multiple devices.

POP3 (Post Office Protocol v3)

POP3 (Post Office Protocol v3) is an email retrieval protocol that allows users to download emails from a mail server to a local device and then delete them from the server. Unlike IMAP, POP3 does not support multi-device email synchronization, making it ideal for users who access email from only one device.

POP3 is used for receiving emails, while SMTP is used for sending emails.

Key Features of PoP3

- Emails are downloaded and stored locally – No need for a constant internet connection.
- Faster access – Since emails are saved locally, retrieval is quick.
- Reduces server storage usage – Emails are removed from the server after downloading.
- Simple and lightweight – Requires minimal server resources.

DNS (Domain Name System)

DNS (Domain Name System) is the phonebook of the internet that translates human-readable domain names (e.g., www.google.com) into IP addresses (e.g., 142.250.183.78) so computers can communicate with each other.

Without DNS, users would need to remember numeric IP addresses instead of domain names.

Key Features of DNS

- Resolves domain names to IP addresses
- Reduces the need to remember IP addresses
- Improves website performance with caching
- Supports load balancing and redundancy
- Enables email delivery with MX records

How DNS Works (DNS Resolution Process)

- User enters a domain name (e.g., www.example.com) in a web browser.
- The request is sent to a DNS resolver (usually provided by the ISP).
- The resolver checks its cache for a stored IP address.
- If not found, it queries the root DNS server.
- The root server directs the resolver to the TLD server (.com, .org, etc.).
- The TLD server provides the IP of the authoritative DNS server for example.com.
- The authoritative DNS server responds with the IP address of www.example.com.
- The resolver caches the response and returns the IP to the browser.
- The browser connects to the web server using the IP address.

SNMP (Simple Network Management Protocol)

SNMP (Simple Network Management Protocol) is a network management protocol used to monitor and manage network devices such as routers, switches, servers, printers, and IoT devices. It operates over UDP and enables administrators to collect performance data, configure devices, and detect network issues.

SNMP is widely used in network monitoring systems like Nagios, Zabbix, and PRTG.

Key Features of SNMP

- Monitors network devices in real-time
- Collects performance metrics like CPU usage, bandwidth, and uptime
- Supports remote device configuration
- Sends alerts when issues are detected (traps and notifications)
- Uses a hierarchical structure to manage network elements

TCP and UDP Network Applications

TCP (Transmission Control Protocol) is used for reliable, ordered, and error-checked data transmission, while UDP (User Datagram Protocol) prioritizes speed and is used for applications where some data loss is acceptable, such as streaming and online gaming.

TCP Applications:

- **Web Browsing (HTTP/HTTPS):** TCP ensures that web pages are delivered in the correct order and without errors.
- **Email (SMTP, POP3, IMAP):** TCP guarantees reliable email delivery and retrieval.
- **File Transfer (FTP):** TCP ensures complete and error-free file transfers.
- **Secure Shell (SSH):** TCP provides a secure connection for remote access.
- **Streaming Media (e.g., Netflix, YouTube):** While streaming often uses UDP for real-time delivery, TCP is used for establishing the connection and ensuring the integrity of the stream.
- **Peer-to-Peer File Sharing:** TCP is used to establish connections and transfer files reliably.

UDP Applications:

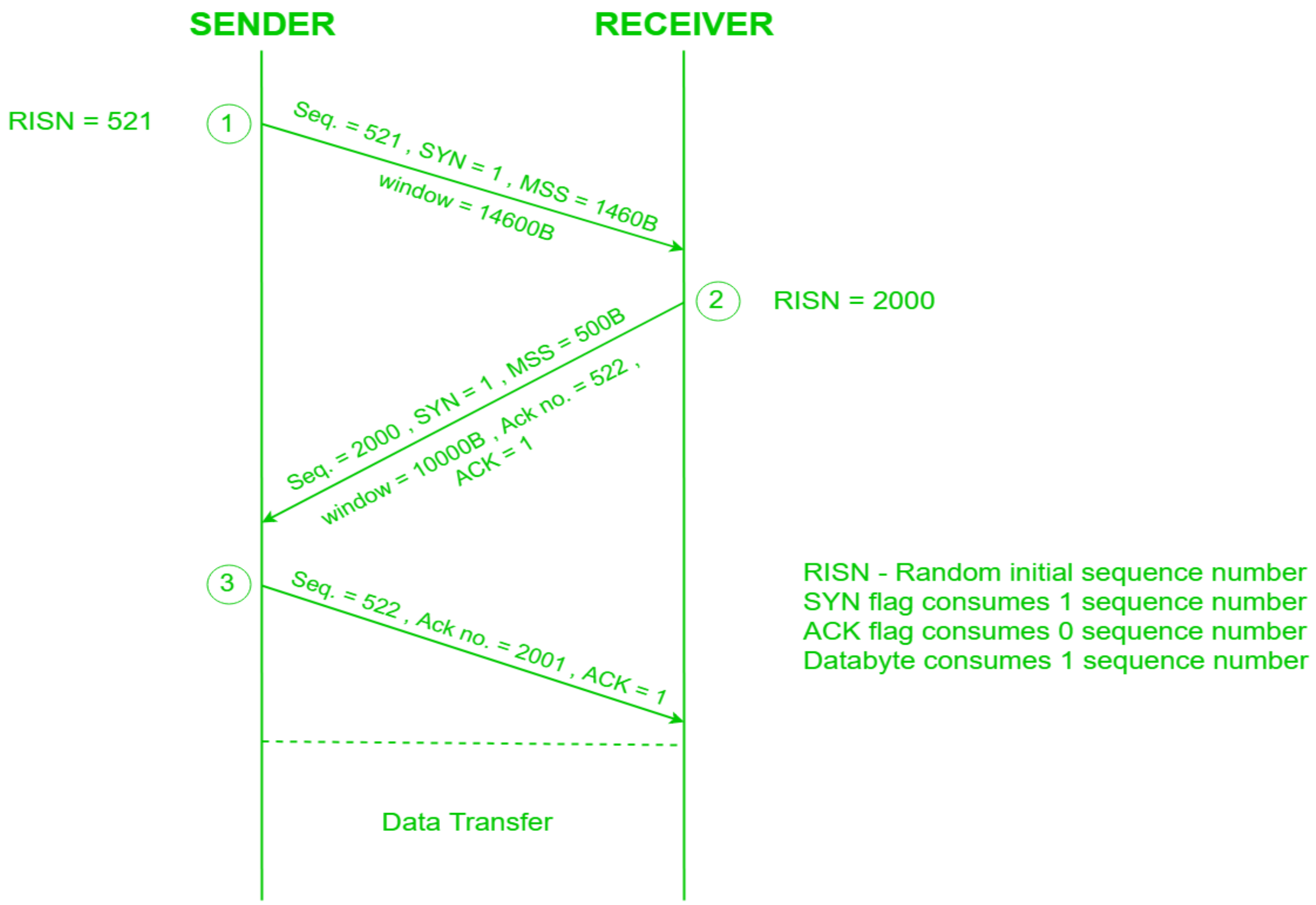
- **Online Gaming:** UDP allows for low-latency, real-time gameplay, even if some packets are lost.
- **Live Streaming:** UDP is used to transmit video and audio streams with minimal delay, even if some data loss occurs.
- **Domain Name System (DNS):** UDP is used for DNS lookups, where speed is more important than absolute reliability.
- **VoIP (Voice over IP):** UDP is used for real-time voice communication, where some packet loss is acceptable.
- **Network Time Protocol (NTP):** UDP is used for synchronizing network clocks, where speed is important.

Transport Layer Connection Establishment

In the transport layer, connection establishment involves a three-way handshake (SYN, SYN-ACK, ACK) to ensure a reliable connection, while connection release uses a four-way handshake (FIN, FIN-ACK, ACK) to gracefully terminate the connection.

Connection Establishment (Three-Way Handshake):

- **SYN (Client to Server):** The client initiates the connection by sending a SYN (synchronize) packet to the server, including a random initial sequence number (A).
- **SYN-ACK (Server to Client):** The server responds with a SYN-ACK packet, acknowledging the client's SYN and including its own random initial sequence number (B) and an acknowledgment number (A+1).
- **ACK (Client to Server):** The client sends an ACK packet to the server, acknowledging the server's SYN-ACK and including an acknowledgment number (B+1).
- Once this three-way handshake is complete, a TCP connection is established.

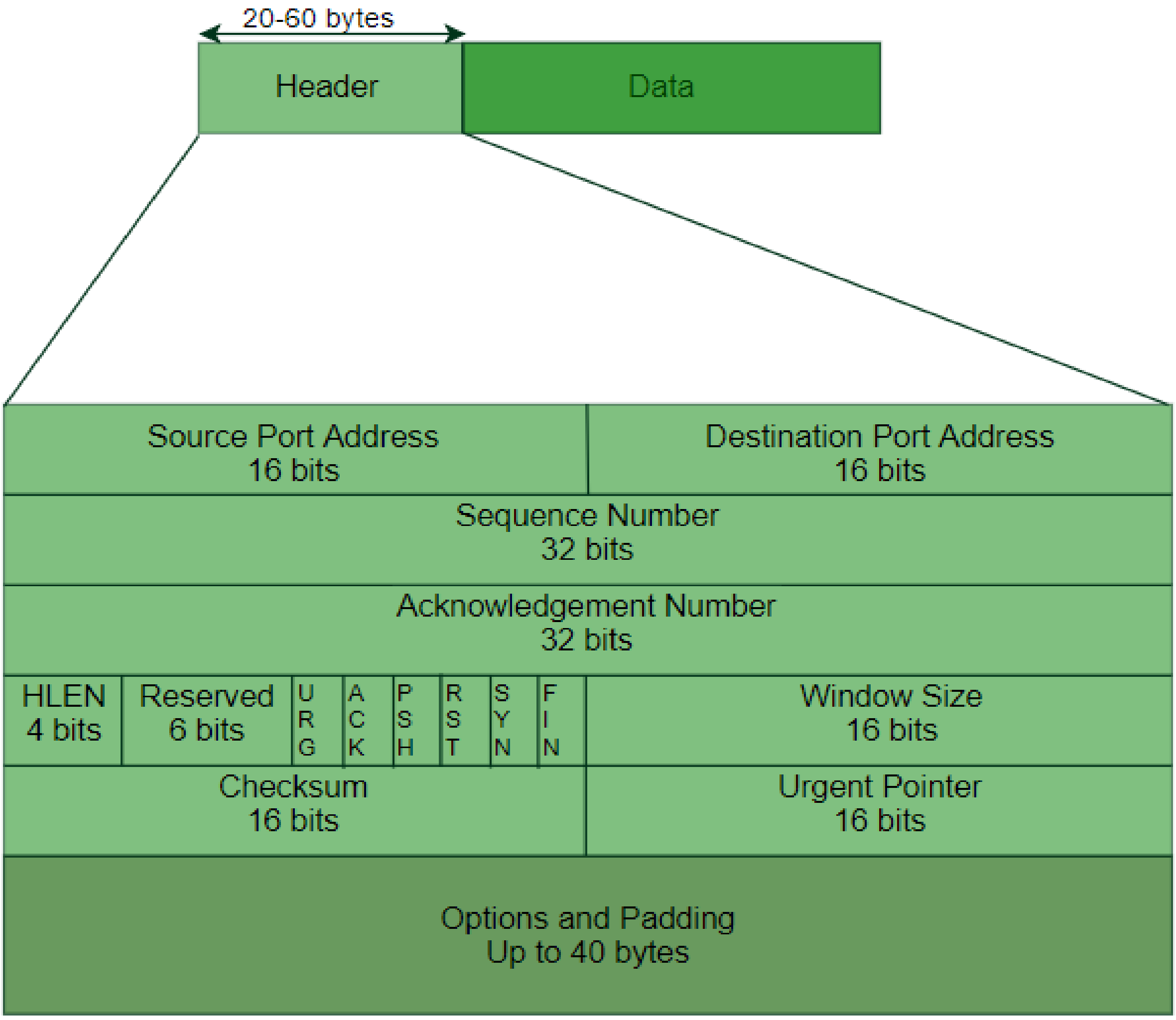

SENDER
RECEIVER
RISN = 521
1
Seq. = 521 , SYN = 1 , MSS = 1460B
window = 14600B
2
RISN = 2000
Seq. = 2000 , SYN = 1 , MSS = 500B
window = 10000B , Ack no. = 522 ,
ACK = 1
3
Seq. = 522 , Ack no. = 2001 , ACK = 1
RISN - Random initial sequence number
SYN flag consumes 1 sequence number
ACK flag consumes 0 sequence number
Databyte consumes 1 sequence number
Data Transfer

Transport Layer Connection Release

Connection Release (Four-Way Handshake):

- **FIN (Initiator to Receiver):** The initiator (client or server) sends a FIN (finish) packet to the receiver, indicating that it has no more data to send.
- **FIN-ACK (Receiver to Initiator):** The receiver acknowledges the FIN by sending a FIN-ACK packet, indicating that it has received the FIN.
- **ACK (Initiator to Receiver):** The initiator acknowledges the receiver's FIN-ACK with an ACK packet.
- **FIN (Receiver to Initiator):** The receiver then sends a FIN packet to the initiator, indicating that it has no more data to send.
- **ACK (Initiator to Receiver):** The initiator acknowledges the receiver's FIN with an ACK packet.
- Once these four packets are exchanged, the connection is fully released.

TCP Header



A TCP (Transmission Control Protocol) header is a structured block of information included in each TCP segment that provides essential details for data transmission.

Purpose:

Reliable Data Delivery: TCP headers ensure the reliable and ordered delivery of data packets by providing mechanisms for error detection, retransmission, and flow control.

Connection Management: They facilitate the establishment, maintenance, and termination of TCP connections.

Data Identification: They help identify the source and destination of the data, enabling proper routing and delivery.

TCP Segment structure:

A TCP segment consists of data bytes to be sent and a header that is added to the data by TCP as shown above:

Source Port Address – A 16-bit field that holds the port address of the application that is sending the data segment.

Destination Port Address – A 16-bit field that holds the port address of the application in the host that is receiving the data segment.

Sequence Number – A 32-bit field that holds the sequence number, i.e, the byte number of the first byte that is sent in that particular segment. It is used to reassemble the message at the receiving end of the segments that are received out of order.

Acknowledgement Number – A 32-bit field that holds the acknowledgement number, i.e, the byte number that the receiver expects to receive next. It is an acknowledgement for the previous bytes being received successfully.

Header Length (HLEN) – This is a 4-bit field that indicates the length of the TCP header by a number of 4-byte words in the header, i.e if the header is 20 bytes (min length of TCP header), then this field will hold 5 (because 5 x 4 = 20) and the maximum length: 60 bytes, then it'll hold the value 15(because 15 x 4 = 60). Hence, the value of this field is always between 5 and 15.

Control flags – These are 6 1-bit control bits that control connection establishment, connection termination, connection abortion, flow control, mode of transfer etc. Their function is:

URG: Urgent pointer is valid

ACK: Acknowledgement number is valid (used in case of cumulative acknowledgement)

PSH: Request for push

RST: Reset the connection

SYN: Synchronize sequence numbers

FIN: Terminate the connection

Window size – This field tells the window size of the sending TCP in bytes.

Checksum – This field holds the checksum for error control. It is mandatory in TCP as opposed to UDP.

Urgent pointer – This field (valid only if the URG control flag is set) is used to point to data that is urgently required that needs to reach the receiving process at the earliest. The value of this field is added to the sequence number to get the byte number of the last urgent byte.