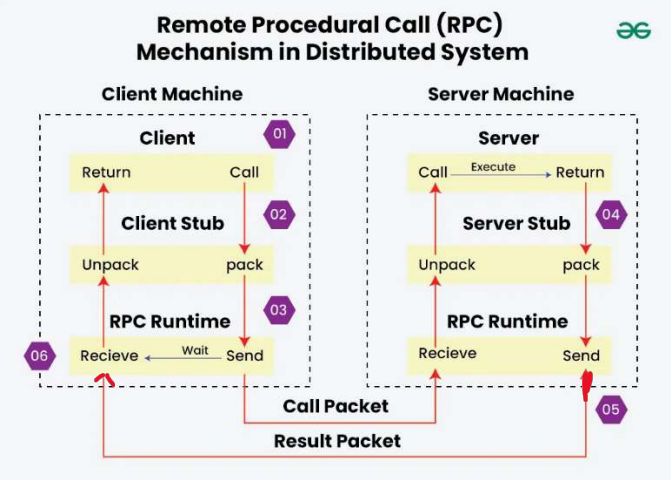
Remote Procedure Call (RPC)

- Remote Procedure Call (RPC) is a service commonly associated with the Session Layer, as it helps manage remote function execution within a communication session.
- RPC allows a program on one computer to execute a procedure (function) on another computer as if it is local.
- It hides the details of network communication.



- **Client Calls**

The client program calls a function.

The call goes to the **Client Stub** instead of running locally.

- **Client Stub Packs**

The stub **packs** (marshals) the function parameters into a network-friendly format.

- **RPC Runtime Sends**

The **RPC Runtime** on the client sends the call packet to the server.

The client waits for the result.

- **Server Stub Unpacks & Executes**

The **Server Stub** receives the packet and **unpacks** (unmarshals) the parameters.

The server then **executes** the function.

- **Server Packs Result**

The server stub **packs** the result and sends it back to the client via the server's RPC runtime.

- **Client Receives Result**

The client RPC runtime receives the result packet.

The client stub **unpacks** it and returns the value to the client program.

Advantages of RPC

- **Makes distributed computing simple.**
- **Programmers do not need to write low-level networking code.**
- **Ensures transparency — remote calls feel like local calls.**

Issues/Challenges in RPC

- **Network failures**
- **Delay**
- **Security during call transmission**

Data Compression Techniques

Data compression reduces file size by encoding data more efficiently. Two major categories:

---

1. Lossless Compression

No information is lost.
Original data can be perfectly reconstructed.

Techniques:

a) Run Length Encoding (RLE)

- **Compresses repeated characters.**

- **Example:**
  AAAAABBB → 5A3B

b) Huffman Coding

- **Uses variable-length codes based on character frequency.**
- **Frequently used characters get shorter codes.**

c) Lempel-Ziv-Welch (LZW)

- **Dictionary-based compression.**
- **Used in ZIP files, GIF images.**

d) Arithmetic Coding

- **Represents entire message as a single number between 0 and 1.**

Uses: text, programs, database files.

---

2. Lossy Compression

Some data is lost but not noticeable to human perception.
Used for multimedia.

Techniques:

a) JPEG Compression (Images)

- **Removes details not detectable by human eyes.**
- **Uses DCT (Discrete Cosine Transform).**

b) MPEG / MP4 (Video)

- **Removes redundant video frames.**
- **Uses motion prediction + DCT.**

c) MP3 / AAC (Audio)

- **Removes inaudible sound frequencies.**
- **Very high compression ratio.**

Difference Between Lossless & Lossy

Lossless	**Lossy**
No data lost	**Some data lost**
Reversible	**Irreversible**
Used for text/data	**Used for images, audio, video**
Example: ZIP, PNG	**Example: JPEG, MP3**

Local Access Network Design

An **access network** is a type of network which connects an end system to an intermediate router on a path from the end system to any other end system.

Types of Access Links

1. Residential Access
2. Company Access
3. Wireless Access

1. Residential Access

- Connects home computers to the network using **ordinary telephone lines** or other media.
- **Technologies:**
    1. **Digital Subscriber Line (DSL):**
        - Upstream: 2–5 Mbps
        - Downstream: 8–10 Mbps
    2. **Hybrid Fiber-Coaxial (HFC):**
        - Upstream: 10 Mbps
        - Downstream: 100 Mbps
- **Purpose:** Provide Internet and network services to homes.

2. Company Access Network

- Used in universities, offices, and companies to connect systems to a router.
- **Technology:** Ethernet
- **Speed:** Operates from 10 Mbps up to 1 Gbps (the note says 16 Mbps, probably older reference).

- **Purpose:** Provide reliable network access to employees or users.

3. Wireless Access

- Provides **wireless communication** within a limited range (few meters).
- Uses **high-frequency radio waves** for data transfer.
- **Mobile Wireless Access:** 3G, LTE, and modern Wi-Fi.
- **Purpose:** Allow flexible and mobile connectivity without cabling.

Backbone Network (BBN) Design

A **Backbone Network** is the **high-speed central network that interconnects different parts of a network**, such as multiple LANs or segments within a building/campus.

- Carries **large amounts of data** between network segments.
- Provides a **path for information** to travel from one subnet or LAN to another.
- Can be **wired or wireless**.

Backbone Network Functions

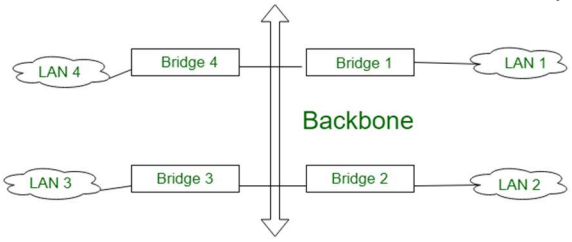
1. **High-Speed Data Transmission** – Fast movement of large volumes of data between network segments.
2. **Interconnection of LANs/Subnets** – Linking multiple LANs to form a bigger network.
3. **Traffic Aggregation** – Collecting data from access networks and forwarding it efficiently. ✅ (This matches your "aggregation")
4. **Reliability and Redundancy** – Maintaining network uptime with multiple paths.
5. **Scalability** – Supporting growth in devices and networks.

Types of Backbone Network Topologies

1) Bus Backbone

- **Structure:** Single high-speed backbone cable connects all segments.
- **Characteristics:**
  - Simple design, easy to install.
  - All devices share the same backbone.
  - Terminators required at both ends.
- **Advantages:**
  - Cost-effective for small networks.
  - Easy to expand.

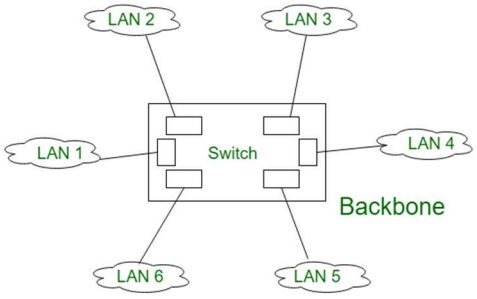
- **Disadvantages:**
    - Performance decreases as more devices are added.
    - Backbone failure can collapse the entire network.



Structure of a Bus backbone

2) Star Backbone

- **Structure:** Each LAN or network segment connects to a **central backbone hub/switch/router**.
- **Characteristics:**
    - Centralized control.
    - Fault in one segment does not affect others.
- **Advantages:**
    - High reliability.
    - Easy to manage and troubleshoot.
- **Disadvantages:**
    - Central device failure affects entire network.
    - More cabling required.



Star Backbone

3) Connecting Remote LANs

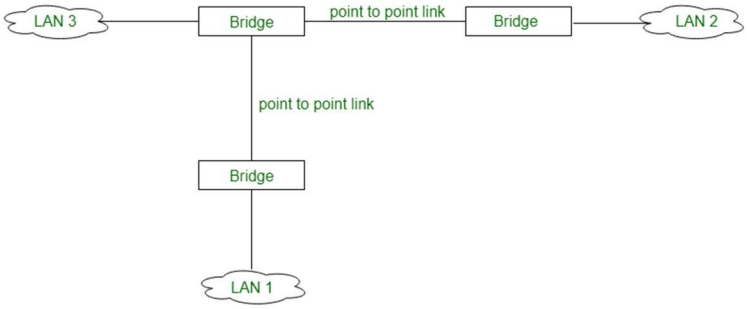
When LANs are geographically separated (different buildings, campuses, or cities), the backbone network extends via **WAN technologies**.

Methods:

1. **Leased Lines:** Dedicated physical connection between LANs.
2. **Frame Relay / MPLS:** Packet-switched connections over WAN.
3. **VPN over Internet:** Secure connections using public internet.
4. **Microwave / Satellite Links:** For very distant locations.

- **Router:** Connects each LAN to the wide-area network.

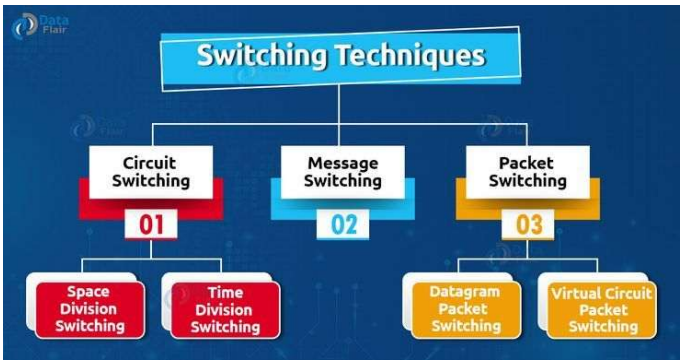
- **WAN/Internet:** Acts as a backbone for remote communication.



Connecting remote LANs to each other

⭐ Switching Techniques

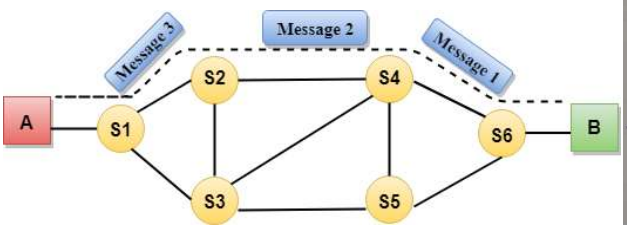
- Switching techniques are methods used in computer networks to transfer data from the sender to the receiver through intermediate network devices.
- They decide how the communication path is created and how information travels across the network.
- There are **three major switching techniques**: Circuit Switching, Message Switching, and Packet Switching.

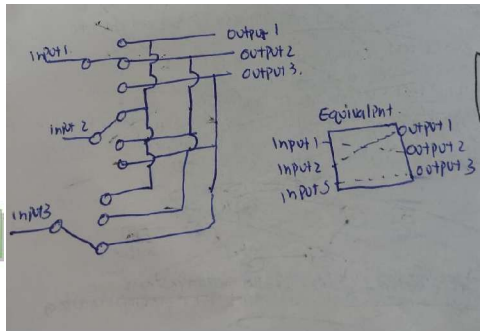


✅ 1. Circuit Switching

- Circuit Switching establishes a **dedicated physical path** between the sender and receiver before communication begins.
- Once this path is established, it is **exclusively reserved** for that communication session.

- No other user can use the same path until the session ends.
- The path is released **only after the connection is terminated**.





Characteristics

1. Connection-oriented technique.
2. Uses a three-phase process:
    - **Circuit establishment**
    - **Data transfer**
    - **Circuit release**
3. Provides guaranteed bandwidth and low delay once the circuit is set.

Types of Circuit Switching

1. **Space-Division Switching**
    - Uses separate **physical paths** (like switches) for each connection.
    - Each circuit has a **dedicated path** through the switch network.
    - Example: Traditional telephone exchanges.

Space Division Switches can be categorized in two ways:

Crossbar Switch

- Uses a **grid of horizontal and vertical bars**.
- A **crosspoint connects** an input line to an output line.
- Simple, direct, and **fast**, but **expensive for large networks**.

Multistage Switch

- Uses **multiple smaller switching stages** to connect inputs to outputs.
- Reduces the number of crosspoints compared to a single large crossbar.
- **More scalable and cost-effective** for large networks.

2. **Time-Division Switching**
    - Shares the **same physical path** by dividing it into **time slots**.

- Each connection gets a **specific time slot** in a repeating cycle.
- Efficient when many users share the same line.

Advantages

- No congestion after the path is created.
- Suitable for real-time and continuous data like voice calls.

Disadvantages

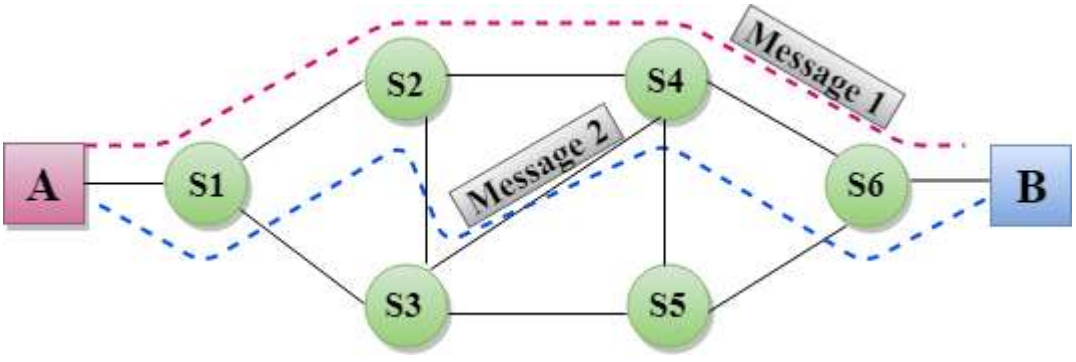
- Wastes bandwidth because the channel remains reserved even when no data is sent.
- Not suitable for bursty data such as internet traffic.

Example

- Traditional telephone networks.

✅ 2. Message Switching

- In message switching, the entire message is treated as a **single unit (block)**.
- Each intermediate node stores the complete message, checks it, and forwards it to the next node.
- The **destination address is appended to the message**, so each intermediate node knows where to send it next.
- This technique is called **store-and-forward switching**.



Characteristics

1. No dedicated path is required.
2. Each node stores the full message before forwarding.
3. High delay because of storage and processing at every node.
4. Large memory is required at intermediate nodes.

Advantages

- Efficient use of bandwidth.
- Useful for non-real-time communication like email, because messages may experience delay at intermediate nodes.

Disadvantages

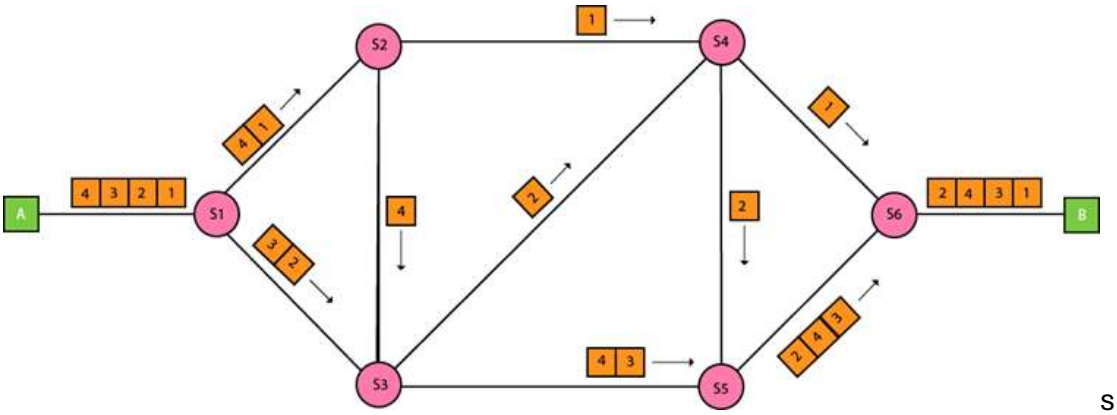
- High end-to-end delay.
- Not suitable for real-time applications like voice or video.

Example

- Old telegraph systems.

✅ 3. Packet Switching

- Long messages are **broken into small packets**.
- Each packet has **data + destination address + sequence number**.
- Packets are **sent independently** and **reassembled** at the receiver using sequence numbers.
- **Lost or corrupted packets** are **retransmitted** individually, not the whole message.



s

Packet switching is of two types:

3.1 Datagram Packet Switching (Connectionless Switching)

- No prior path setup.
- Each packet travels independently and may take different routes.
- Packets may arrive **out of order**.
- Used in **IP networks (Internet)**.

3.2 Virtual Circuit Packet Switching (Connection-Oriented Switching)

- A temporary logical path (virtual circuit) is created before sending packets.
- All packets follow the **same route** and arrive in order.
- More reliable than datagram switching.

Advantages of Packet Switching

- Efficient use of bandwidth.
- Suitable for data networks like the internet.
- Even if one route fails, packets can take another path.

Disadvantages

- Requires complex protocols.
- Packets can be delayed and may need reassembly.

TELNET

- TELNET stands for **Telecommunication Network**.
- It is one of the oldest **remote login protocols** used in computer networks.
- It works using simple **text commands**.
- Telnet is an **Application Layer** protocol.
- It uses **TCP port 23** for communication so it is reliable
- Works on **client–server model**.
- Modern networks have replaced Telnet with **SSH (use port 22) which is secure because it is encrypted**
- Ex- telnet <IP> <port> (telnet google.com 80)

Purpose / Why Telnet is used

- To access a remote computer.
- To configure network devices (routers, switches).
- To test open ports and network connectivity.

- To run commands on a remote system.

Limitations

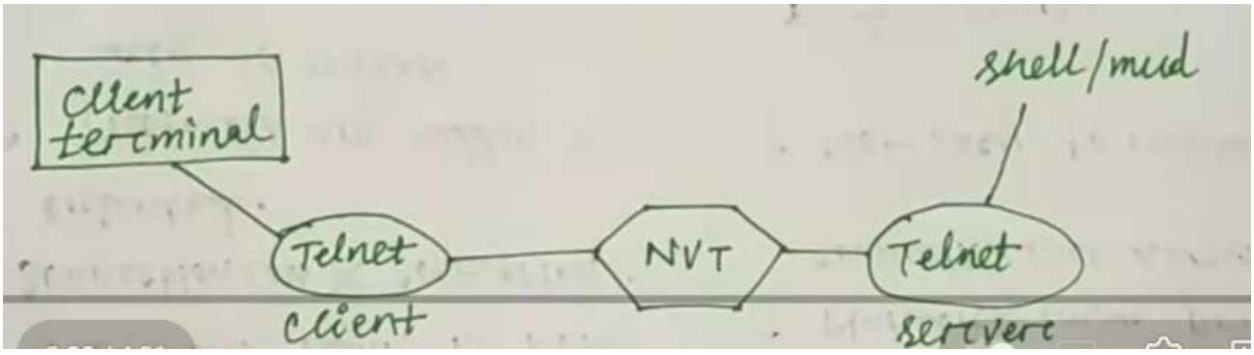
- Data (including password) is sent in **plain text**
- **No security hence** Can be easily hacked therefore Cannot be used safely on the internet

How Telnet Works (Step-by-Step)

1. User opens a terminal and runs the Telnet command.
2. Telnet client creates a TCP connection with the Telnet server on **port 23**.
3. Client sends login details (username & password).
4. Server provides a **remote shell/command prompt**.
5. User types commands → they are executed on the remote machine.
6. Server sends output back to client.
7. Connection is closed when done.

Telnet Architecture

Client Terminal → Telnet Client → NVT → Telnet Server → Shell/Host

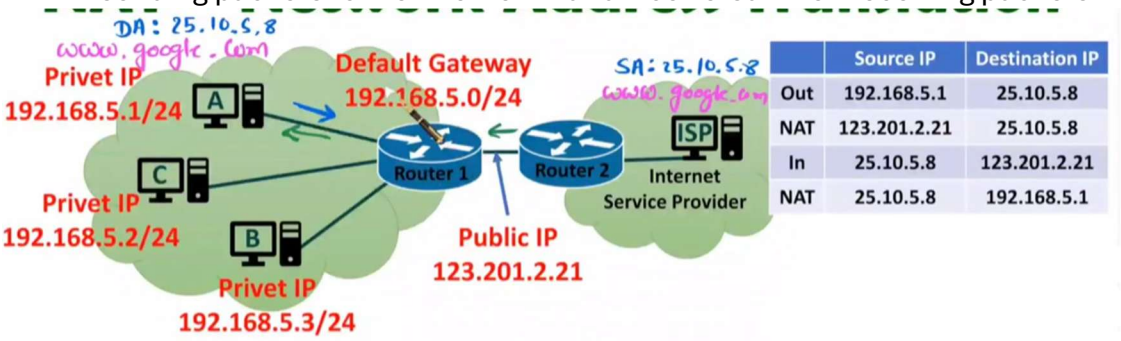


- **Client Terminal:** User's computer.
- **Telnet Client:** Converts user commands into a standard format.
- **NVT (Network Virtual Terminal):**
  A common, universal communication format.
  Makes different systems communicate in a uniform way **so they do not need to know each other's terminal characteristics like character set, keyboard type, or display format.**
- **Telnet Server:** Receives NVT-formatted commands and converts them back to server format.

- **Shell/Host:** The command interpreter on the remote machine that executes user commands.

Network address translation

- NAT operates in the router or gateway that connects a Local Area Network (LAN) to the Internet (WAN)
- **Network Address Translation (NAT)** is a technique used in routers to **translate private IP addresses** used inside a local network into a **public IP address** before sending packets to the internet — and vice versa when receiving packets.



Type	Description	Example
Static NAT	One private IP is mapped to **one public IP** permanently	Used for servers needing constant address
Dynamic NAT	Private IP is mapped to **any available public IP** from a pool	Mapping changes dynamically
PAT (Port Address Translation) / NAPT (network address port translation)Overload	Many private IPs share **one public IP** using **different port numbers**	Most common (used in home routers)

- 🔹 **Why NAT is Needed**
  - IPv4 addresses are limited (only ~4.3 billion). But IPv6 does not need NAT because it has an extremely large address space —
    👉 about 3.4×10^{38} unique IP addresses (that's 340 undecillion!).
  - Every device needs a unique public IP to connect to the Internet.
  - NAT helps conserve IP addresses by allowing many private devices to use one public IP.
  - 🌐 **Private IP Address Ranges**

Class	Range	Default Mask
A	10.0.0.0 – 10.255.255.255	255.0.0.0
B	172.16.0.0 – 172.31.255.255	255.255.0.0
C	192.168.0.0 – 192.168.255.255	255.255.255.0

Internetworking

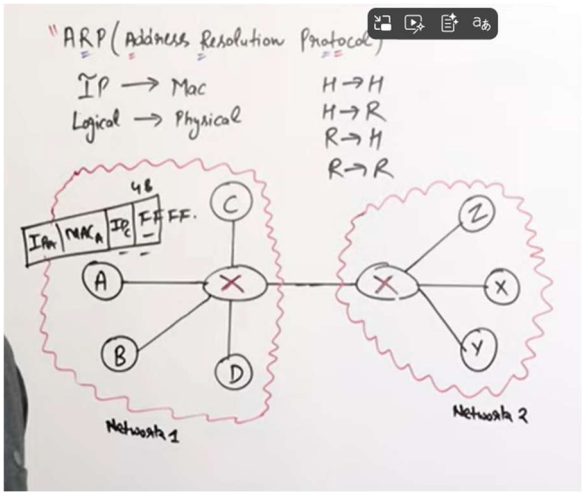
• Connecting different types of networks (LAN, WAN, Wi-Fi, etc.) so they act as one Network.
• Achieved using TCP/IP protocol

INTERNET CONTROL PROTOCOLS

These protocols assist the network in address resolution (ARP/RARP), error handling, and control messaging. (ICMP)

Address resolution protocol

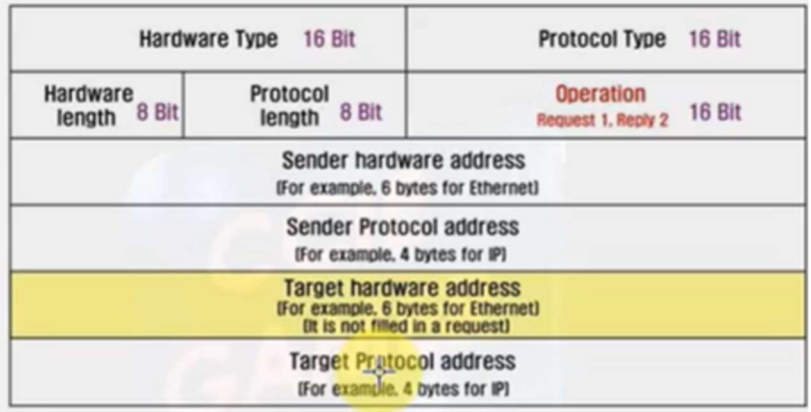
ARP (Address Resolution Protocol) is used to map an IP address to its corresponding MAC (physical) address in a local network.



⭐ **Types of ARP**

- **HH (Host to Host)** A computer (host) asks **another host** for its MAC address.
- **HR (Host to Router)** A host asks a **router** for a MAC address.
- **RH (Router to Host)** A router replies to a **host** with its MAC address.
- **RR (Router to Router)** A router asks **another router** for a MAC address.

Arp header



target Hardware Address Destination MAC (empty in request)

🔹 **How ARP Works (Step-by-Step)**

Example:

Host A (192.168.1.2) wants to send data to Host B (192.168.1.5)

Step 1 — Check ARP Cache

Host A checks its **ARP table** to see if it already knows the MAC address of 192.168.1.5.

If found → uses it directly.

Step 2 — Send ARP Request (Broadcast)

If not found → Host A sends an **ARP Request**:

Who has IP 192.168.1.5? Tell 192.168.1.2

- This message is **broadcast** to all devices on the LAN.

Step 3 — Receive ARP Reply (Unicast)

Host B (with IP 192.168.1.5) replies with:

192.168.1.5 is at MAC address 00:14:22:16:23:A8

- This reply is **unicast** back to Host A.

Step 4 — Update ARP Table

Host A stores the pair in its **ARP table**:

IP Address	MAC Address
192.168.1.5	00:14:22:16:23:A8

Then Host A can send data directly using the MAC address.

ARP Table (Cache)

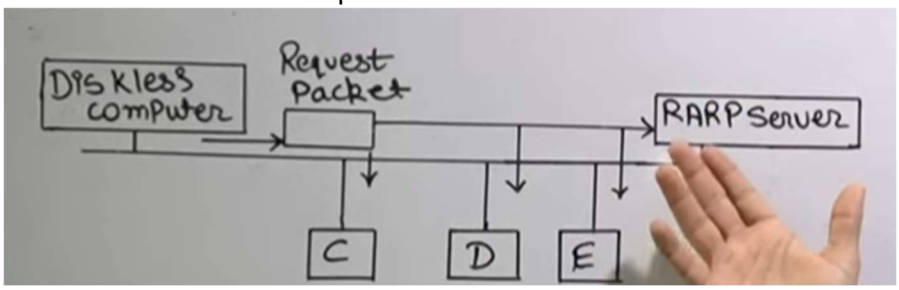
IP Address	MAC Address	TTL
192.168.1.5	00:14:22:16:23:A8	5 min

👉 Stored for a short time to avoid repeated requests.

RARP (Reverse ARP)

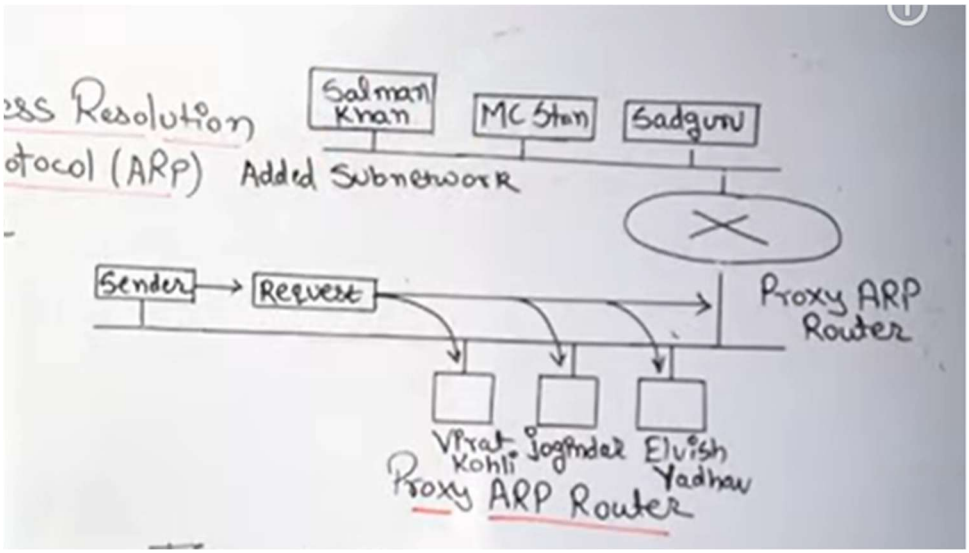
Used by a device to find its **IP address** when it knows only its **MAC address Mainly used by diskless systems that have a NIC(network interface card) (with MAC address) but no storage to keep IP address.**

- The device **broadcasts a RARP request** on the network asking,
  👉 *"This is my MAC address — what is my IP?"*
- A **RARP server** replies with the correct IP address.



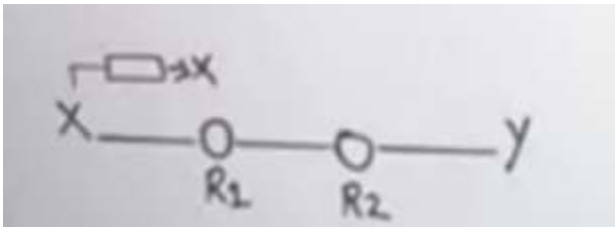
Proxy ARP

Router pretends to be the target (answers ARP with its MAC) and then forwards (x will send its mac address and sender send data to it and then x will send to correct device)



ICMP – Internet Control Message Protocol

ICMP is a **network-layer protocol** used for **error reporting** and **network diagnostics** in IP networks. It helps devices communicate problems or check connectivity.



---

1 ICMP Message Types

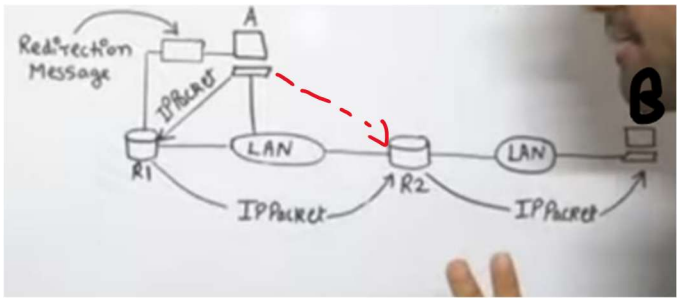
ICMP messages are broadly classified into:

Category	Purpose
Error Reporting	Informs the sender about problems in packet delivery
Query / Diagnostic	Requests or responds to information

---

2 Error Reporting Messages

- **Destination Unreachable:** Triggered by network failures, wrong IP, or unreachable host.
- **Time Exceeded:** Packet TTL expired; used in traceroute.
- **Parameter Problem:** Invalid or missing fields in the IP header.
- **Source Quench (deprecated):** Previously used to slow down packet sending due to congestion
- **Redirection:** Router advises sender to use a better route.



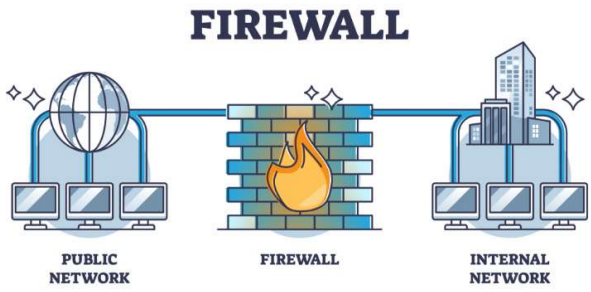
3️⃣ Query Messages – Key Points

- **Echo Request / Reply:** Test reachability (ping).
- **Timestamp Request / Reply:** Measures time taken for packets to travel between hosts.

ICMP HEADER

Type(8 bit)	Code(8 bit)	Check Sum(16 bit)
Extended Header(32 bit)		
Data/Payload(Variable Length)		

FIREWALLS



A firewall is a **security device/software** that monitors and filters incoming/outgoing network traffic based on security rules.

Actions a firewall can take:

- **Accept:** Allow traffic
- **Reject:** Block with an error message
- **Drop:** Block silently (no reply)

Purpose:
Creates a barrier between a trusted internal network and an untrusted external network (Internet).

Generations of Firewalls

1) First Generation – Packet Filtering

- Filters packets based on IP, port, protocol
- Treats each packet individually
- No connection awareness

2) Second Generation – Stateful Inspection

- Remembers connection states
- Filters packets based on context/session

3) Third Generation – Application Layer Firewall

- Filters traffic at any OSI layer
- Can run proxy servers

4) Next-Generation Firewalls

- Detects advanced malware & application layer attacks
- Combines firewall + intrusion detection + deep packet inspection

---

Types of Firewalls

- **Host-based Firewall** – Runs on a single device
- **Network-based Firewall** – Protects an entire network

Cryptography Concepts

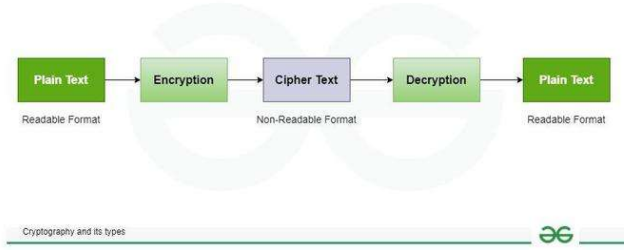
Cryptography is the technique of securing information and communications using coded messages, ensuring:

- **Confidentiality** → Only authorized users can read the data
- **Integrity** → Data is not altered during transmission
- **Authenticity** → Confirms the identity of sender/receiver

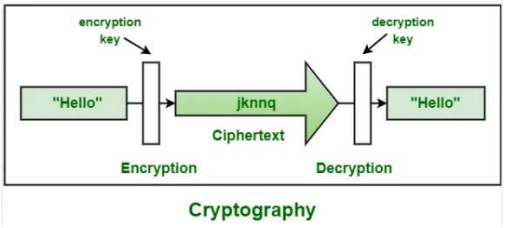
The word comes from Greek:

- **kryptos** = hidden
- **-graphy** = writing

Cryptography uses **mathematical algorithms** and **rule-based calculations** to protect sensitive data.



Basic Operations



1. Encryption

Converts **plaintext → ciphertext** (unreadable form).

2. Decryption

Converts **ciphertext → plaintext** (readable form).

For both encryption and decryption, **the sender and receiver agree on an algorithm and key**.

Types of cryptography

Cryptography can be broadly classified into **three main types**, based on how keys are used and the purpose:

1. Symmetric Key Cryptography /Secret Key

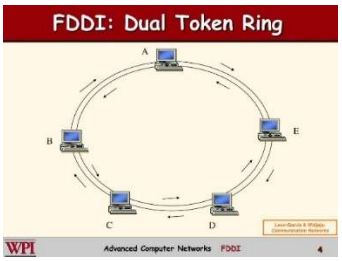
2. Asymmetric Key Cryptography /Public-Key

3. Hash Functions /One-Way Cryptography ex- SHA-256 ((Secure Hash Algorithm 256-bit)))

Feature	Symmetric (Secret Key)	Asymmetric (Public Key)	Hash (One-Way)
Key Used	Same key for encryption & decryption	Two keys: public & private	No key
Purpose	Confidentiality	Confidentiality + Authentication	Data integrity
Speed	Fast	Slower	Very fast
Key Sharing	Must be shared securely	Public key can be shared openly; private key kept secret	Not applicable
Example Algorithms	DES: Data Encryption Standard, AES: Advanced Encryption Standard	RSA: Rivest–Shamir–Adleman, ECC: Elliptic Curve cryptography	SHA-256: Secure Hash Algorithm 256 bit MD5: Message Digest 5
Reversibility	Reversible	Reversible (with correct key)	Not reversible
Use Case	File encryption, VPNs	Digital signatures, secure communication, key exchange	Integrity check, blockchain, password verification

FDDI (Fiber Distributed Data Interface)

- FDDI is a **high-speed LAN technology** that uses **fiber optic cables that** works mainly as a **backbone network** connecting multiple LANs.
- **FDDI** was originally an **ANSI (American National Standards Institute) X3T9.5** standard, later referenced under **IEEE (Institute of Electrical and Electronics Engineers) 802.8**



- **Primary Ring:** Main data transmission

- **Secondary Ring:** Backup (used when fault occurs and can also be used for **large data transmission** in dual ring mode

---

IEEE Standards

The **IEEE (Institute of Electrical and Electronics Engineers)** is an international organization that develops standards for **computer networks, electronics, and communication systems**.

IEEE Standard	Use / Description
IEEE 488	GPIB (general purpose interface bus) – connects electronic instruments.
IEEE 754	Defines floating-point arithmetic in computers.
IEEE 802	define standards for **Local Area Networks (LANs)** and **Metropolitan Area Networks (MANs)**.
IEEE 1394	FireWire – high-speed data transfer interface.
IEEE 1588	Precision Time Protocol for time synchronization.

OSI Layer Relation

IEEE 802 standards mainly define protocols for the **first two layers** of the OSI model:

- **Layer 1 – Physical Layer**
- **Layer 2 – Data Link Layer**, which is further divided into:
  - **LLC (802.2)**
  - **MAC (802.3, 802.5, 802.11, etc.)**

LAN Protocols

LAN protocols are **rules** that control **how computers in a Local Area Network** communicate, share data, and avoid errors or collisions.

⚙️ **Main Functions**

- Framing (dividing data into frames)
- Addressing (using MAC address ensures data reaches the correct device in the network)
- Error detection & correction (Detects transmission errors using **CRC, parity, checksum** and requests retransmission if errors are found.)
- Flow control (Prevents a fast sender from overwhelming a slow receiver ,uses methods like **Stop-and-Wait**)
- Access control (who can send data)

IEEE Lan protocol

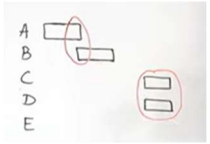
Protocol	IEEE Std	Access Method	Topology	Speed	Medium	Advantage	Use Case
Ethernet	**802.3**	CSMA/CD	Star or Bus	10 Mbps – 100 Gbps	Twisted pair / Fiber	Low cost, simple, high speed	Offices, homes, campus networks
Token Bus	**802.4**	Token Passing	Bus	1 – 10 Mbps	Coaxial	Reliable, predictable access	Industrial and control networks
Token Ring	**802.5**	Token Passing	Ring	4 – 16 Mbps	Twisted pair	No collisions, fair access	Business LANs, factories
FDDI	**ANSI X3T9.5 / IEEE 802.8**	Token Passing	Dual Ring	100 Mbps	Fiber	High speed, fault tolerant	Backbone and large LANs
Wi-Fi	**802.11**	CSMA/CA	Star	Up to several Gbps	Wireless	No cabling, supports mobility	Homes, offices, public hotspots

Attached Resource Computer Network LAN protocol

ARCnet **(Attached Resource Computer Network)**	*(Not IEEE standard, proprietary)*	Token Passing	Star / Bus	2.5 Mbps	Coaxial	Simple, low cost, reliable	Small offices, automation

Collison

- A **collision** occurs when multiple devices send data **simultaneously** on the same network
- Data collision=data loss or data corrupt



- **Type of Collison**
  - Partial collision (A, B)
  - Total collision (C, D)

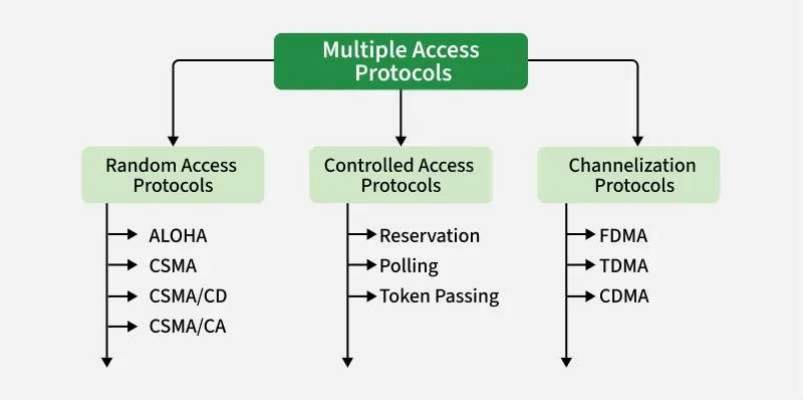
Data Link Protocol

- A **Data Link Protocol** is a **set of rules** (procedures and conventions) used at the **Data Link Layer** of the **OSI model**
- It ensures **reliable data transfer** between two directly connected devices over a physical link.
- It handles **framing, flow control, error control, and access control**.

Sublayers of Data Link Layer

Sublayer	Function	Examples
1. Logical Link Control (LLC)	Provides **flow control** and **error control** between nodes.	Stop & Wait, Sliding Window, ARQ protocols
2. Media Access Control (MAC)	Controls **how devices access the shared medium**.	CSMA/CD, CSMA/CA, Token Passing, TDMA

Multiple access protocols



Throughput (low → high): Throughput means the rate of successful data transmission over a network

👉 **Pure ALOHA (18%) < Slotted ALOHA (37%) < 1-Persistent CSMA (45–50%) < p-Persistent CSMA (50–60%) < Non-Persistent CSMA (60–70%) < O-Persistent CSMA (≈70–75%) < CSMA/CA (70–80%) < CSMA/CD (75–90%)**

Random Access Protocols/contention-based protocols

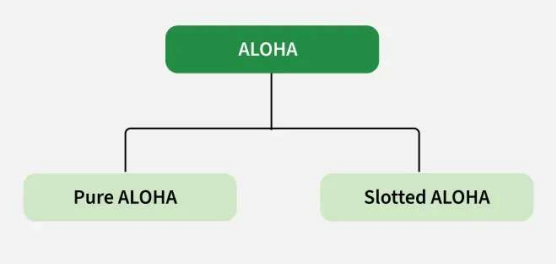
- Random Access Protocols are **contention-based protocols** used in shared communication channels
- **any device can transmit data at any time**
- **collisions may occur** (because **no prior coordination and if two or more device send data simultaneously**)
- **Decentralized**: No master or coordinator.
- **Simple and scala**ble, especially in low-traffic networks.

- Devices **must detect or avoid collisions and retransmit later**.

🔧 Types of Random Access Protocols:

1. ALOHA (ack is used)

- It was designed for wireless LAN but is also applicable for shared medium.
- In this**, multiple stations can transmit data at the same time** and can hence lead to collision and data being garbled.



. Pure ALOHA

- Devices transmit **whenever they have data**.
- If a **collision occurs** (**no acknowledgment received** from the receiver), the sender waits for a **random backoff time** and **retransmits** the frame.
- Very simple but high collision rate.

Efficiency:

- Max throughput ≈ 18% of the channel.

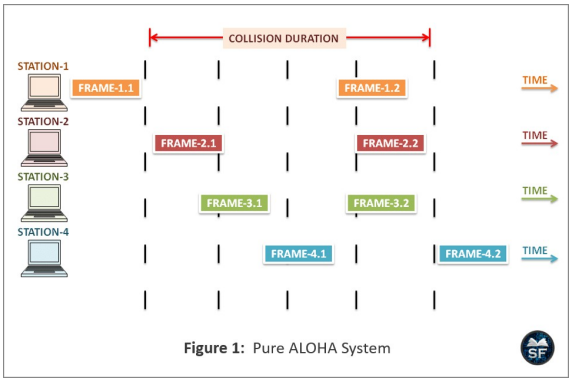


Figure 1: Pure ALOHA System

. Slotted ALOHA

- Time is divided into **equal-sized slots**.
- A device can transmit **only at the start** of a time slot.
- Reduces the chance of collision compared to Pure ALOHA.
- If **two or more stations** start transmitting **in the same slot**, their packets **collide**, and **both are destroyed**.
- Each station waits for a **random number of future slots** before retransmitting.
- Because transmissions start only at slot boundaries, chances of collision are **reduced**.

Efficiency:

- Max throughput ≈ 37% of the channel.

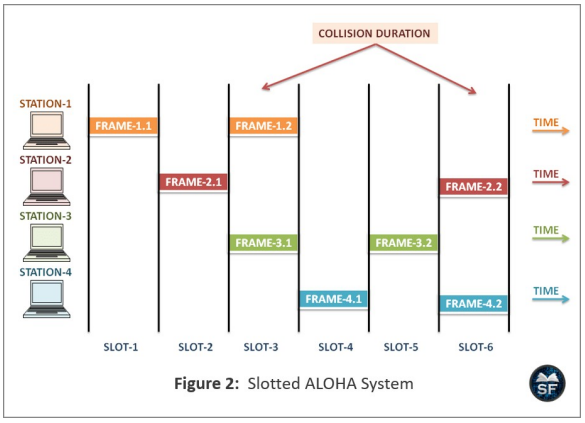


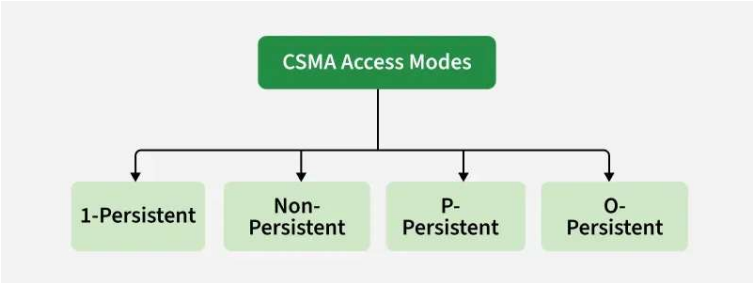
Figure 2: Slotted ALOHA System

---

2. CSMA (Carrier Sense Multiple Access)

- Devices **listen to the channel** before transmitting:
  - If the channel is **idle**, it sends the data.
  - If the channel is **busy**, it waits.

Variants:



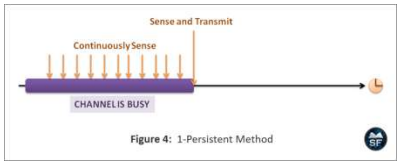


Figure 4: 1-Persistent Method

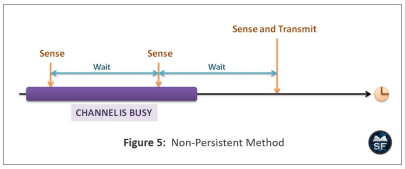


Figure 5: Non-Persistent Method

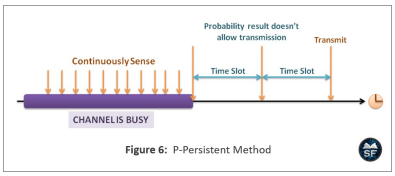


Figure 6: P-Persistent Method

1-Persistent CSMA

- **Behavior**: The station continuously listens to the channel and transmits **immediately** once it's idle.
- **Pros**: Fast response.
- **Cons**: High chance of **collisions** if multiple stations detect the idle channel simultaneously.

Non-Persistent CSMA

- **Behavior**: The station **waits a random time** after detecting a busy channel before sensing again.
- **Pros**: Lower collision probability.
- **Cons**: Higher **delay** compared to 1-persistent.

P-Persistent CSMA (used in slotted channels)

- When the channel is idle,
  the station generates a random number between 0 and 1.
- If number < p, → send the frame.
- If number ≥ p, → wait for the next time slot and try again.
- If a collision occurs, the station waits for a random number of slots (backoff) before retransmitting. like A wait for 2 slot and B wait for 4 slot

- **Pros**: Balances delay and collision probability.
- **Used in**: Slotted networks like some wireless systems.

O-Persistent CSMA (O -ORDERED)

- It is a variant of CSMA where each station is given a priority order/transmission order by supervisory node/controller.

When the channel becomes idle, the highest-priority (first) station gets the chance to transmit first.

- there is no chance of Collison

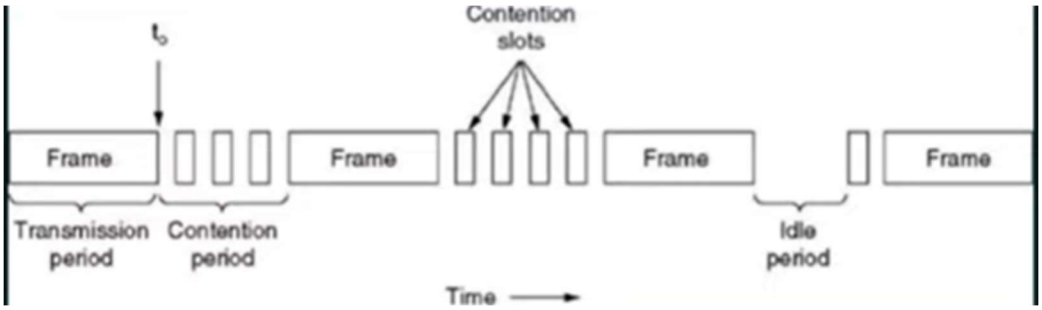
- Working Steps:

  - All stations sense the channel.
  - When the channel is idle, stations follow a predefined order (like Station 1, then 2, then 3...).
  - The first station in order transmits immediately.
  - If it has no data, the next station in the order gets the chance.
  - This continues in sequence — maintaining order and avoiding collision.

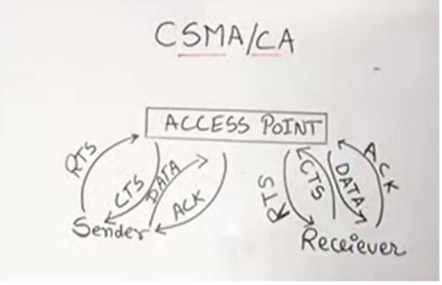
---

2. CSMA with Collision Detection (CSMA/CD)(no ack is used)

- **Used in**: Traditional Ethernet.
- **Behavior**: The station monitors the channel during transmission. If a **collision** is detected, it **stops transmitting**, waits a random time (backoff), then tries again. And The waiting range **doubles (exponentially)** after each collision.
- **Pros**: Efficient in wired media.
- **Note**: Mostly outdated due to full-duplex Ethernet and switches.
- **Contention period**: minimum wait period where devices will check Collison can happen or not



4: CSMA with Collision Avoidance (CSMA/CA) (ack is used)



- **Used in**: **Wi-Fi (IEEE 802.11)**.
- **Behavior**: Instead of detecting collisions, it **avoids** them by using techniques like **RTS** (**Request to Send) /CTS (Clear to Send)** and random backoff before transmitting.
- If the channel is busy or ACK not received, station waits for a **random backoff time**.
- The range again **doubles exponentially** after each failed attempt.
- **Pros**: Better for wireless media where collision detection is hard.

Binary Exponential Backoff is used in both CSMA/CD and CSMA/CA

- Backoff time=Random$(0,2^k-1)$ ×slot time
- **k** = number of retransmission attempts (increases after each failure)
- Slot Time=2×Propagation Delay

◇ *CONTROLLED ACCESS PROTOCOL*

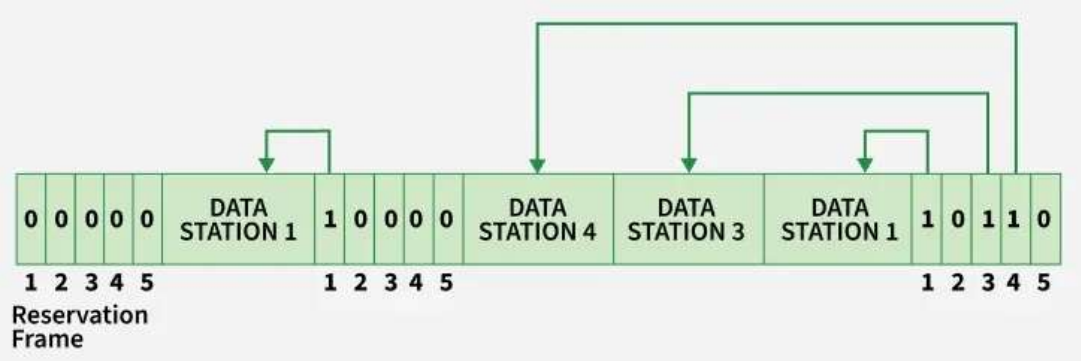
- **Devices take turns to use the channel**.
- A device can **transmit data only when it is authorized**, avoiding collisions.

⚙️ **How Controlled Access Works**

1. **Coordination:** All devices agree to follow a common control rule.
2. **Permission System:** One device gets the right to transmit — others must wait.
3. **Transmission:** The selected device sends its data without interference.
4. **Release:** Control is passed to another device according to the method used.
5. **Repeat:** The process continues in a cyclic or managed way.

🔹 **TYPES OF CONTROLLED ACCESS PROTOCOLS**

🟩 **1. Reservation Protocol**



Concept:
Before sending data, each station **reserves a time slot** in advance.
This ensures only one station transmits in that slot.

Detailed Points:

- A station must **reserve a slot** before sending data.
- Each **time interval** starts with a **reservation frame**, followed by **data frames**.
- If there are **N stations**, there are **N reservation minislots**—each assigned to one station.
- A station that wants to send data marks its **own minislot (sets it to 1)**.
- After reservation, **only the stations that reserved** transmit their data frames in the same order.

Example:
Used in **TDMA (Time Division Multiple Access)** systems (like cellular networks).

Advantages:

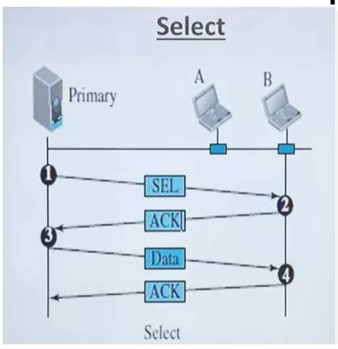
- No collision since each station has a reserved slot.
- Predictable and fair channel access.

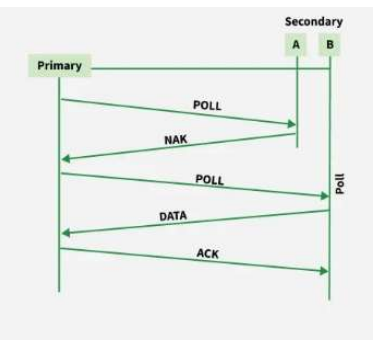
Disadvantages:

- Wastes bandwidth if some reserved stations have no data.
- More control overhead due to reservation signaling.

🟧 2. Polling Protocol

Select function **poll function**





If primary want to send if secondary want to send it wait until not asked

In select function (Primary → Select(B) → B → ACK → Primary → Data → B → ACK)

In poll function (Primary → Poll(A) → A → NAK → Primary → Poll(B) → B → Data → Primary → ACK)

Concept:
A **primary device (master)** controls access and **polls (asks)** each **secondary device (slave)** one by one if it wants to send data.
Detailed Points:

- A **primary station** manages the communication.
- It sends a **poll message** to a secondary station: "Do you have data to send?"
- If the secondary station replies "Yes," it is allowed to transmit.
- If "No," the primary moves to the next device in the list.
- After all devices are polled, the process repeats.

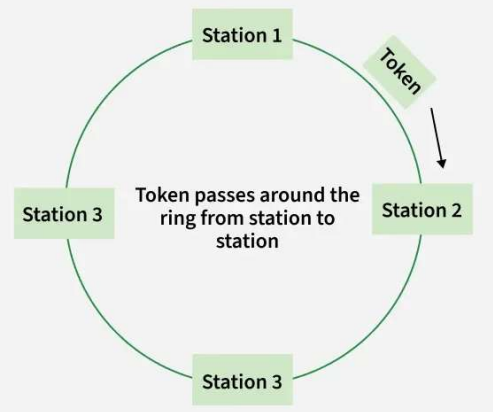
Example:
Used in **Bluetooth networks** (master-slave communication) and some **Wi-Fi infrastructure** modes.
Advantages:

- Completely avoids collisions.
- Good for centralized control systems.

Disadvantages:

- The primary station can become a **bottleneck or single point of failure**.
- **Delay increases** if there are many stations to poll.

🟦 3. Token Passing Protocol



Concept:
A **small control frame** called a **token** circulates among the stations.
Only the device **holding the token** is allowed to transmit data.
Detailed Points:

- The token travels in a **logical ring** from one station to another.
- A station can transmit **only when it receives the token**.
- After sending data, the station **passes the token** to the next device.
- If a token is **lost or damaged**, a new one is generated by a control mechanism.

Example:
Used in **Token Ring LANs**, **FDDI (Fiber Distributed Data Interface)**, and some **industrial control networks**.
Advantages:

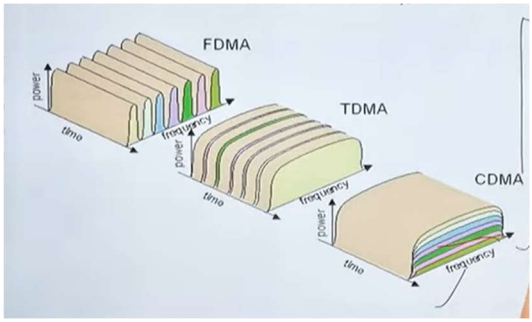
- No collisions at all.
- Each station gets a fair opportunity to transmit.

Disadvantages:

- If a token is lost, the network must pause to regenerate it.
- If one device fails, it can disturb the whole ring.

◇ *CHANNELIZATION PROTOCOLS (Channel Partitioning Protocols)*

Total available communication bandwidth (channel) is **divided among multiple users** based on **frequency, time, or code**.so that each user can transmit **simultaneously** without interference.



🟩 1. FDMA (Frequency Division Multiple Access)

🔸 Concept

- The total bandwidth is **divided into smaller frequency bands**.
- Each user is assigned a **unique frequency channel**.
- All users can transmit **at the same time**, but on **different frequencies**.
- Small guard bands are used between frequencies to avoid overlap.

🔸 Characteristics

- Transmission is **continuous** for each user.
- **No synchronization** between users needed.
- Bandwidth per user is **fixed**.

🔸 Advantages

✅ Simple and easy to implement
✅ No collision between users

🔸 Disadvantages

❌ Wastes bandwidth (due to guard bands)
❌ Not efficient if some users are idle

🔸 **Examples**

- **Radio broadcasting** (each station has its own frequency)
- **Satellite communication**

🟧 **2. TDMA (Time Division Multiple Access)**

🔸 **Concept**

- The whole channel bandwidth is **shared in time**.
- Each user is assigned a **specific time slot** in a repeating frame.
- Users take turns to transmit data — one at a time.
- Time is divided into **frames**, and each frame into **time slots**.
- Each user sends data during its assigned slot.
- Frames repeat continuously so users can send data periodically.

🔸 **Characteristics**

- All users share **the same frequency**, but at **different times**.
- Requires **precise synchronization** between users.
- Data is sent in **bursts**, not continuously.

🔸 **Advantages**

✅ Efficient bandwidth usage

✅ Easier to manage and organize

🔸 **Disadvantages**

❌ Needs accurate time synchronization

❌ Not ideal for continuous or real-time signals

🔸 **Examples**

- **GSM (2G mobile)**
- **Satellite communication**

🟦 **3. CDMA (Code Division Multiple Access)**

🔸 **Concept**

- All users share the **same frequency and time**, but each user's signal is encoded with a **unique code**.
- The receiver uses the same code to **decode only the intended signal**.
- Each bit of data is multiplied by a **unique spreading code** (sequence).
- All coded signals are transmitted together.
- The receiver uses the same code to **extract** the desired signal.
- All users **transmit simultaneously**.

🔸 **Advantages**

✅ High capacity — many users can share the same channel

✅ Efficient bandwidth use

🔸 **Disadvantages**

❌ Complex encoding and decoding process

❌ Requires strict power control and synchronization

🔸 **Examples**

- **3G mobile systems (WCDMA, CDMA2000)**
- **GPS (Global Positioning System)**

Protocol / Technique	Use Case / Example
ALOHA	Satellite communication
Slotted ALOHA	Satellite networks
CSMA/CD	Ethernet (Wired LAN)
CSMA/CA	Wi-Fi (IEEE 802.11)
Reservation	Real-time multimedia networks
Polling	Mainframe–terminal systems
Token Passing	Token Ring / FDDI
FDMA	Analog cellular systems
TDMA	GSM (2G mobile)
CDMA	3G mobile communication

◇ SLIDING WINDOW PROTOCOL

🧩 **Definition:**
The **Sliding Window Protocol** is a **flow control method** used in data communication that allows **multiple frames** to be sent **at once**, instead of one at a time — improving **efficiency**.

⚙️ **Main Purpose**

- To **control the flow** of data between sender and receiver.
- To ensure **no data loss or overflow** at the receiver side.
- To **increase efficiency** of transmission.

📘 **Characteristics**

- **Full-duplex communication** (both sides can send data).
- Uses **sequence numbers** for frame tracking.
- **Window size** controls flow rate.
- Ensures **reliable delivery** using ACKs and retransmission.
- Reduces **waiting time** — increases channel utilization.

✅ **Advantages**

- Efficient and fast transmission
- Prevents sender from overloading receiver

❌ **Disadvantages**

- More **complex** than stop-and-wait protocol
- Requires **memory and buffer** at sender and receiver
- Handling errors and ACKs adds overhead

How It Works (Step by Step)

1. The **sender** can send several frames **before waiting for ACK** — limited by a "window size."
2. Each frame has a **sequence number**.
3. The **receiver** acknowledges** frames as it receives them.
4. As ACKs arrive, the sender's "window" **slides forward**, allowing new frames to be sent.

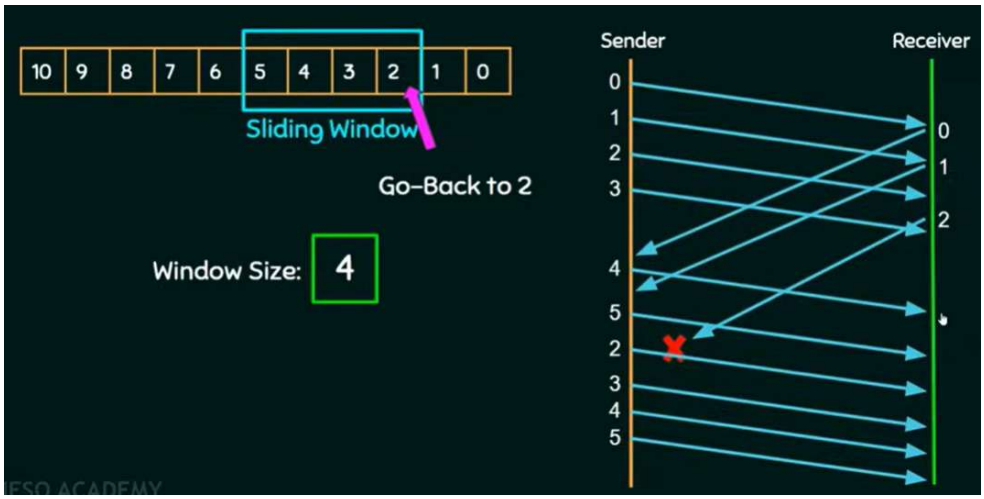
🔸 **Terminology**

- **Window Size:** Number of frames that can be sent before waiting for acknowledgment.
- **ACK (Acknowledgment):** Message from receiver confirming successful reception.
- **Sequence Number:** Unique number given to each frame to keep track.
- **ARQ (Automatic Repeat reQuest)** :A **reliable transmission technique** used for **error control** in data link layer — receiver requests retransmission when an error is detected.

🟩 **Types of Sliding Window Protocols**

Type	Description	Example Use
1. Go-Back-N ARQ	Sender can send multiple frames before ACK. If one frame is lost, all following frames are resent.	High-speed, reliable links
2. Selective Repeat ARQ	Only the **error frame** is resent, not the others.	More efficient, used in TCP

🟧 **1. Go-Back-N ARQ**



Sender Side

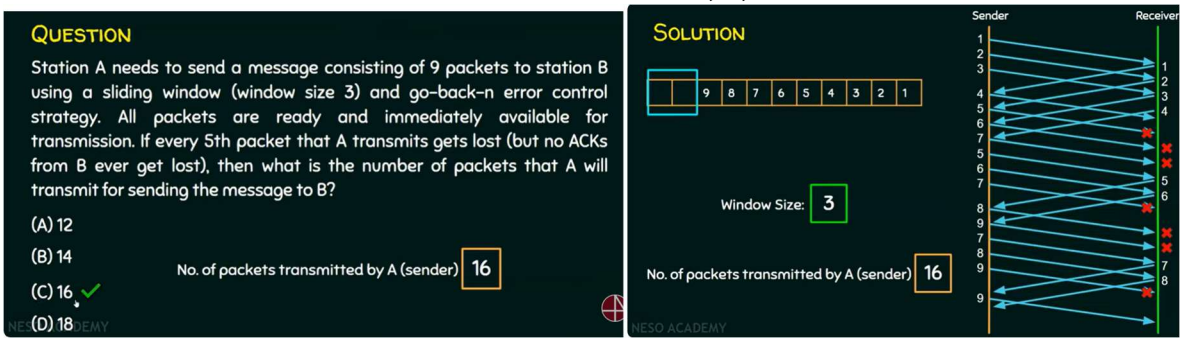
- Can send N frames continuously without waiting for an ACK.
- Maintains a window of size N (called sending window).
- Waits for ACKs from the receiver.
- If ACK is not received within a timeout period, it retransmits all frames from the last unacknowledged frame onward.

Receiver Side

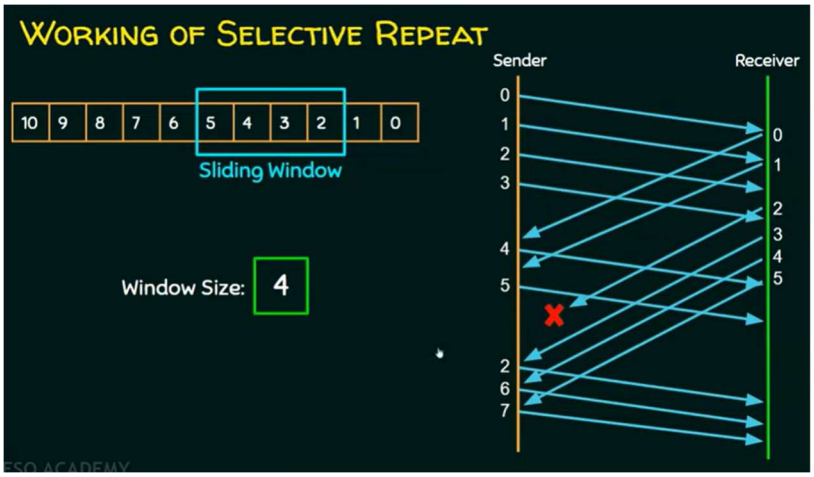
- Has a window size of 1 (in basic form).
- Accepts frames in order only.
- If it gets an out-of-order frame, it discards it and sends ACK for the last correct frame.

Example:

If frames 1–5 are sent and frame 3 is lost → resend 3, 4, and 5.



🟦 2. Selective Repeat ARQ



Sender Side

- Maintains a window of size N (called the sending window).
- Can send N frames before needing an ACK.
- If an ACK is not received for a specific frame within a timeout, only that frame is retransmitted.

Receiver Side

- Has a window of size N (unlike Go-Back-N, which has 1).
- Can accept frames out of order.
- Stores out-of-order frames in a buffer until the missing frame(s) arrive.
- Sends individual ACKs for each correctly received frame.

Example:
If frames 1–5 are sent and frame 3 is lost → resend **only frame 3**.

Flow Control Protocols

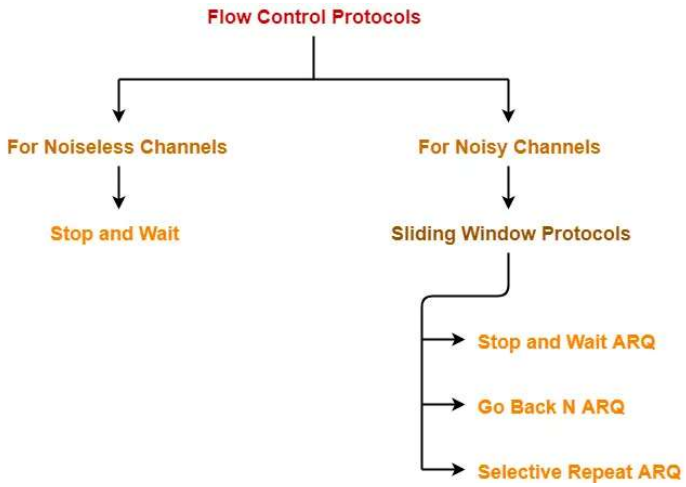
- **Definition**

Flow control protocols are **Data Link Layer** mechanisms that ensure the **sender does not overwhelm the receiver** with too much data at once.
They keep the data transfer **smooth, reliable, and efficient**.

- **Purpose**

  - To **match the data rate** of sender and receiver.

- To **avoid buffer overflow** at the receiver.
- To **maintain reliable communication** between two directly connected nodes.



---

1 For Noiseless Channels

(No errors occur during transmission)

🔹 a) Simplex Protocol (Elementary Protocol)

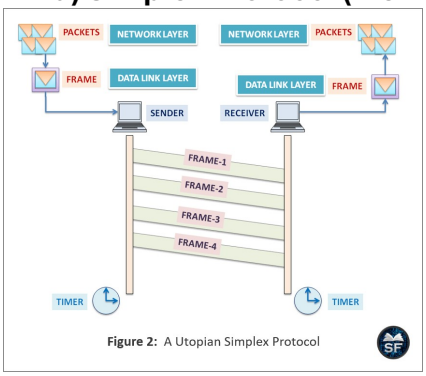
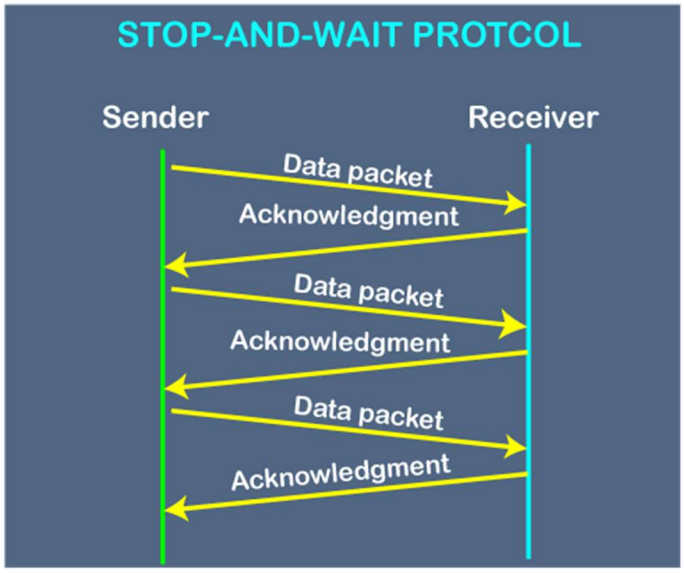


Figure 2: A Utopian Simplex Protocol

- Also called **Unacknowledged Simplex Protocol**.
- Sender keeps sending frames without waiting for acknowledgment.
- **No error control, no flow control.**
- Used only in **perfect channels** (error-free).

  🧠 *Example:* Internal communication between devices inside a computer.

🔹 b) Stop-and-Wait Protocol



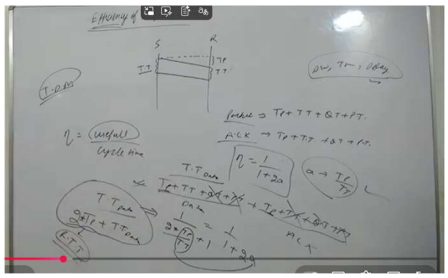
Sender side

- Rule 1 : Send one data packet at a time.
- Rule 2 : Send the next packet only after receiving ACK for the previous.

Receiver side

- Rule 1 : Receive and consume data packet.
- Rule 2 : After consuming packet, ACK need to be sent (Flow Control).

working

- Sender sends **one frame**, then **waits for acknowledgment (ACK)**.
- If ACK is received → send next frame.
- Ensures **no frame is lost or duplicated**.
- **Simple**, but **slow**.
- 

1 Efficiency (η)

Meaning

Efficiency (η) = Fraction of total time used for actual data transmission.

It shows how well the link is utilized.

$$\eta = \frac{\text{Useful Transmission Time}}{\text{Total Cycle Time}}$$

For Stop-and-Wait Protocol

Only one frame is sent per cycle, so:

$$\eta = \frac{T_t}{T_t + 2T_p}$$

Let

$$a = \frac{T_p}{T_t}$$

Then simplified:

$$\eta = \frac{1}{1+2a}$$

2 Throughput (S)

Meaning

Throughput = **Actual data rate** successfully transmitted per second.

It depends on **Efficiency** × **Link Bandwidth**.

$$S = \eta \times R$$

Where:

- S = Throughput (bits per second)
- η = Efficiency (unitless)
- R = Data rate (bits per second)

⚠️ Disadvantages / Problems in Stop-and-Wait Protocol

1️⃣ Lost Data Frame

- The data frame may get lost during transmission.
- Receiver never gets it → no ACK sent.
- Sender keeps waiting forever (if no timeout).

➡️ Both sender and receiver remain idle → transmission stops.

2️⃣ Lost Acknowledgment (ACK)

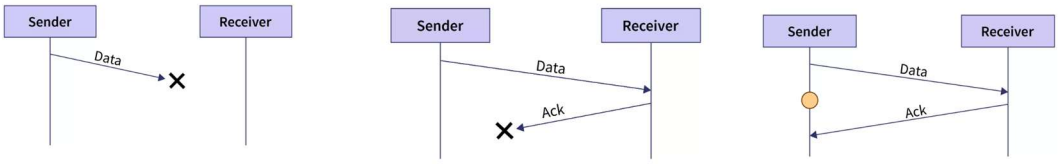
- Receiver gets data and sends ACK, but ACK is lost.
- Sender never receives it → keeps waiting → retransmits same frame.

➡️ Receiver receives duplicate frame → wastes bandwidth.

3️⃣ Delayed Data or ACK

- ACK or data arrives **after timeout**.
- Sender assumes it's lost → retransmits the frame.

➡️ Receiver gets duplicate frame and discards it → causes delay and inefficiency.



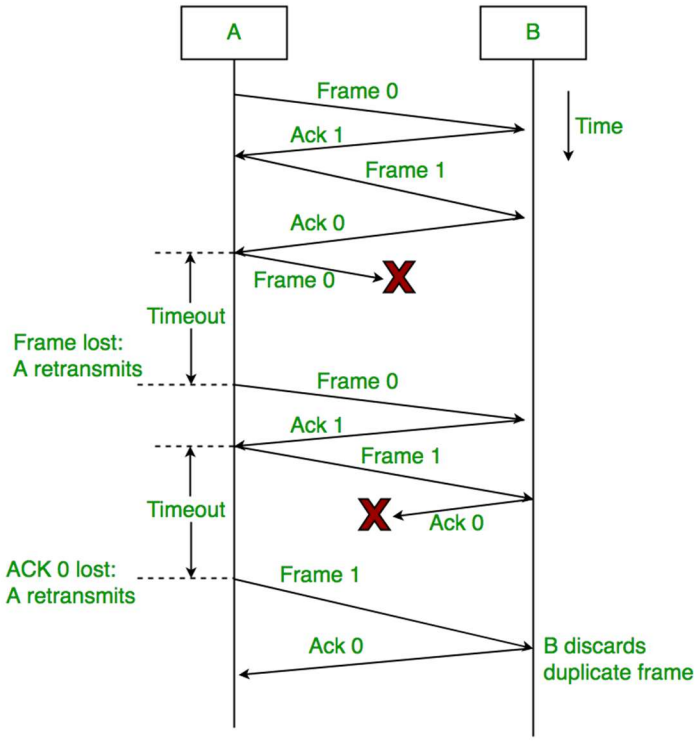
2️⃣ For Noisy Channels

(Errors or losses can occur)

Here, we use **ARQ (Automatic Repeat reQuest)** methods to retransmit lost/damaged frames.

🔹 a) Stop-and-Wait ARQ

- Sender sends one frame → waits for ACK before sending next.
- Ensures error control and flow control.
- It's a Sliding Window Protocol with window size = 1.
- Requires only two sequence numbers (0 and 1).



- Sender sends **one frame**, then **waits for acknowledgment (ACK)**.
- If ACK is received → send next frame.
    - When **Frame 0** arrives → Receiver sends **ACK 1** ✅
    - When **Frame 1** arrives → Receiver sends **ACK 0** ✅

🔹 **Stop-and-Wait ARQ Solves Them Using**

Mechanism	Function
Timeout	Retransmit if no ACK received in time.
Sequence Numbers	Identify duplicates and track next frame.
CRC (Error detection)	Detect corrupted data or ACKs.

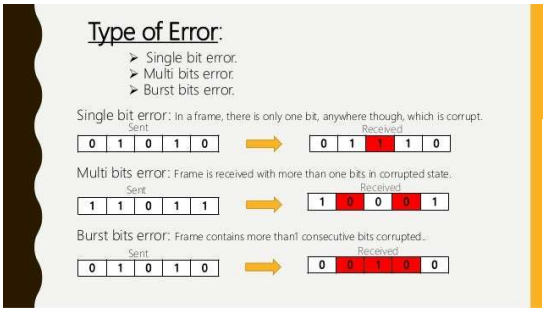
🔹 **How it works**

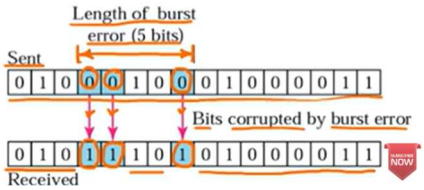
Step	Sender	Receiver	ACK Sent
1	Sends Frame 0	Receives it	ACK 1
2	Sends Frame 1	Receives it	ACK 0
🔁	If timeout	Resend same frame	Receiver discards duplicate, resends ACK

- **Limitations**
  - Inefficient for long-delay or high-bandwidth links (low utilization).
  - Improved by **Go-Back-N** or **Selective Repeat ARQ**.

- **b) Go-Back-N ARQ**
  - Sender can send **multiple frames (N frames)** before waiting for ACK.
  - If an error occurs in one frame, all frames after it are **resent**.
  - Faster than Stop-and-Wait.
- **c) Selective Repeat ARQ**
  - Most **efficient** and **complex**.
  - Sender retransmits **only the frames that had errors**.
  - Uses **sliding window** for both sending and receiving sides.

⚙️ Error Detection and Correction





🧠 *Burst errors are very common in real-world communication channels.*
The **length of the error (burst length)** is the **distance between the first and last corrupted bit**, **including both** — even if bits in between are correct.

- **Error Detection Methods**

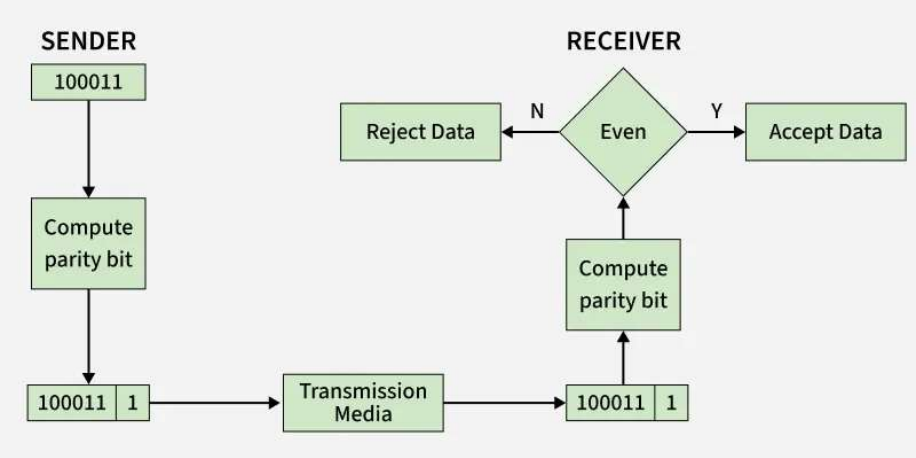
These methods **only find** if an error happened — **not correct it**.

Method	Description	Detects	Used In
1. Single Parity Bit	Adds 1 extra bit to make number of 1s even or odd.	Single-bit errors	Simple serial comm.
2. Two-Dimensional (2D) Parity	Adds parity **for rows and columns** → stronger detection.	Multiple-bit errors	Memory systems
3. Checksum	Adds all data segments and sends the sum → receiver rechecks.	Burst errors	TCP/UDP

Method	Description	Detects	Used In
4. CRC (Cyclic Redundancy Check)	Divides data by a generator polynomial and sends remainder (CRC bits).	Burst errors	Ethernet, USB

🧩 **Detection → Parity, 2D Parity, Checksum, CRC**

(a) Parity Check



- Add an extra bit called a **parity bit** to make the total number of 1's either even or odd.
- "Parity check (even or odd) detects all odd-number-of-bit errors but fails to detect even-number-of-bit errors."
- **total length (codeword) = m + 1 bits**
- **Detection capability:** Can **detect all single-bit errors**.

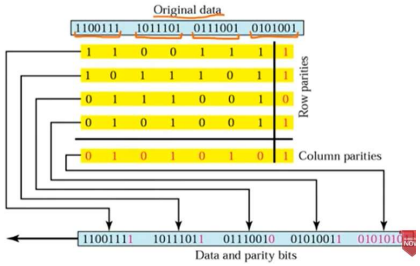
Type	Description
Even Parity	The number of 1s (including parity bit) should be even.
Odd Parity	The number of 1s (including parity bit) should be odd.

Two-Dimensional Parity Check (2D Parity)

🔹 Definition

Two-Dimensional Parity Check is an **error detection technique** that improves on simple (1D) parity.
It adds **both row and column parity bits** to a block of data bits.

Total bits = (m + 1)(n + 1).

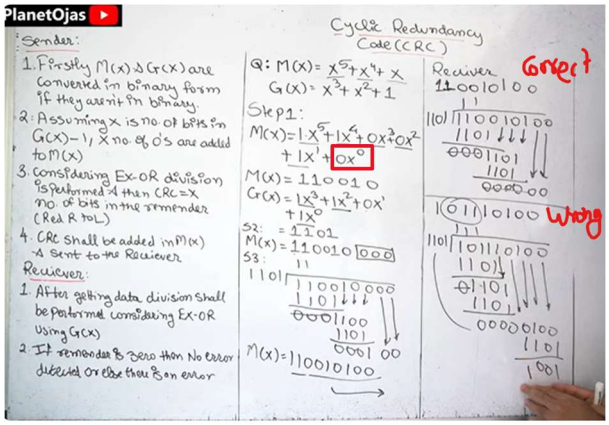


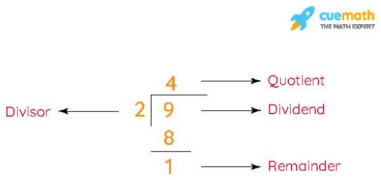
Cyclic redundancy

the total number of bits transmitted = **m + r**.

CRC can detect:

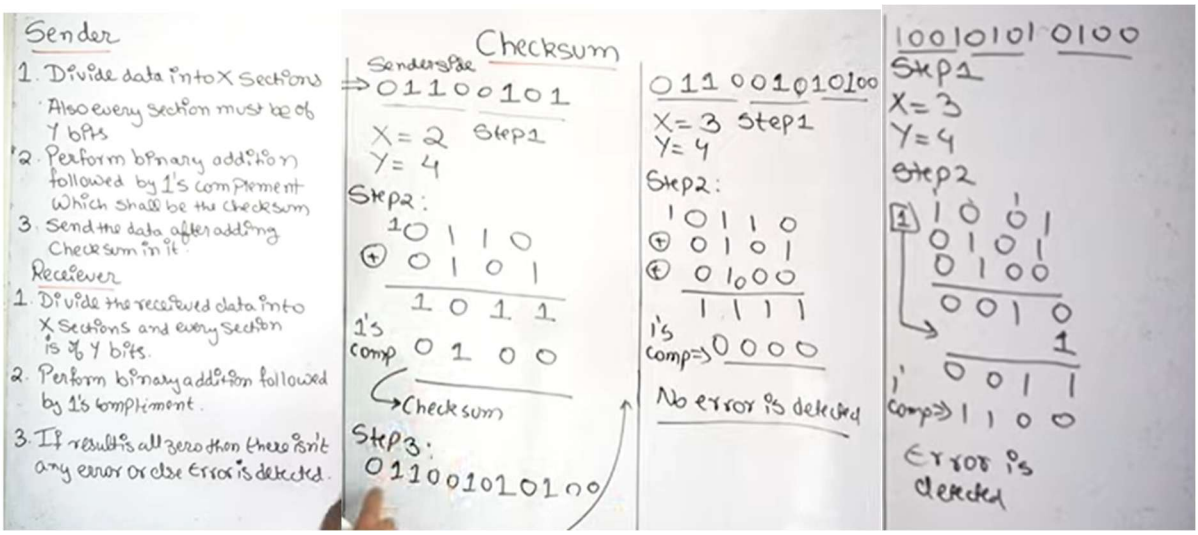
- Single-bit errors
- Double-bit errors
- Odd number of bit errors
- Burst errors (up to a certain length)

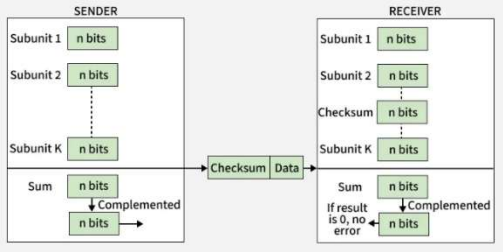




Checksum

if receiver get correct if receiver get wrong





total transmitted bits = **m + r (Number of bits in each segment)**.

🔹 **Error Correction Methods**

These methods **detect and fix** errors.

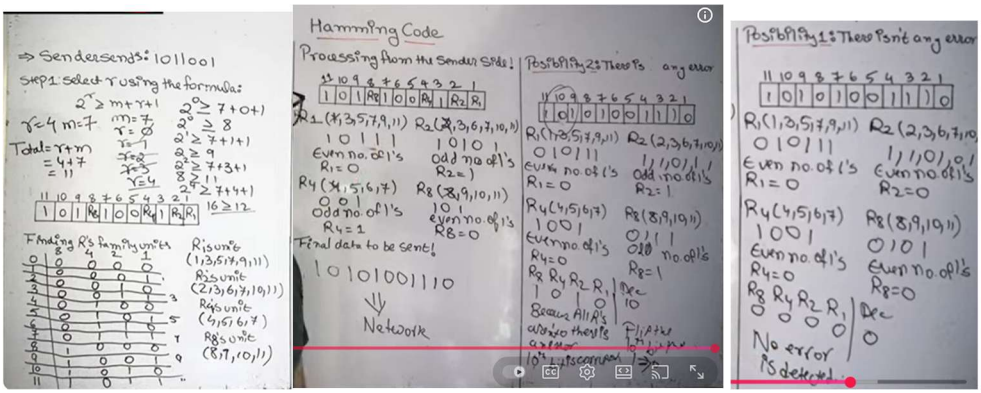
Two main types 👇

Type	Meaning	Example
1. Forward Error Correction (FEC)	Receiver can correct errors using extra redundant bits.	**Hamming Code**, Reed–Solomon
2. Backward Error Correction (ARQ – Automatic Repeat Request)	Receiver detects error and asks **sender to resend** the frame.	**Stop-and-Wait ARQ**, **Go-Back-N**, **Selective Repeat**

🧩 **Correction → FEC (Hamming Code)** and **Backward correction (ARQ)**

Hamming code

correct wrong



Place r at 20 21 22 23 form
Find missing r1 r2 r3 r4
Even no of 1 =0, Odd no of 1=1
Now check if total sum is 0
At last check whether in dec form all Redundant bit is 0 else that dec form bit is error

Formula=2^r>=m+r+1

🧠 Hamming Distance

The **Hamming Distance** between **two binary strings** of equal length is the **number of bit positions** where the two bits are **different**.

🔹 Minimum Hamming Distance (d_min)

When we design **error-detecting or correcting codes**, we use the **minimum Hamming distance** between **any two valid codewords**.

It determines how many errors can be **detected** or **corrected**.

🔹 Error Detection and Correction Relation

Capability	Formula	Meaning
To detect up to 'd' errors	$d_{min} = d + 1$	Can detect up to d bit errors
To correct up to 't' errors	$d_{min} = 2t + 1$	Can correct up to t bit errors

---

🧠 Example:

If $d_{min} = 3$:

- It can **detect** up to **2-bit errors** (since 3 = d + 1 → d = 2)
- It can **correct** up to **1-bit error** (since 3 = 2t + 1 → t = 1)

✅ So, **Hamming codes (d_min = 3)** can **detect 2-bit errors** and **correct 1-bit error**.

Autonomous System (AS)

An **Autonomous System (AS)** is a **group of connected IP networks and routers** that are **controlled by a single organization or authority** and that follow a **common routing policy**.

The Internet is too large for one routing system.So, it's divided into smaller, manageable parts → **Autonomous Systems (ASes)**.

🔹 **Example**

- **AS1:** Airtel Network
- **AS2:** Jio Network
- **AS3:** BSNL Network
  Each is a separate **autonomous system**.
  They connect to each other using **BGP (Border Gateway Protocol)**.

Network Layer Design Issues

Routing – selects the best path.
Forwarding – sends packets to the correct output link.
Congestion Control – prevents overload in the network.
Connection Setup – creates and closes virtual circuits.
Packet Format – defines packet header and fields.
Addressing – assigns IP addresses to devices.
Internetworking – connects different networks together.
Fragmentation & Reassembly – splits large packets and rebuilds them.
Quality of Service (QoS) – ensures priority and low-delay delivery.

Routing

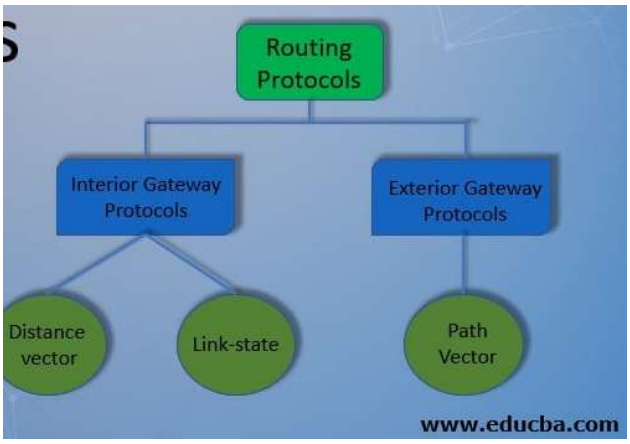
Routing is the process of **selecting a path for data packets to travel** from the **source device to the destination device** across a network.

Type of routing

In **static routing**, routes are entered manually by the network administrator and do not change automatically — this is usually used in small networks.

In **dynamic routing**, routers use protocols such as **RIP**, **OSPF**, or **BGP** to update routes automatically, which is suitable for large and complex networks.

Routing Protocol



- A **routing protocol** is used by routers to **discover, maintain, and update routes** to different networks automatically.
- In simple terms, routing protocols help routers decide **where to send packets**.

✨ 1️⃣ Interior Gateway Protocols (IGP) / Intra-Domain Routing Protocols

Used **inside one Autonomous System (AS)** — like within a single organization.

(a) Distance Vector Routing Protocols

- Uses **distance (hop count)** and **direction (vector)** to find destination.
- Based on **Bellman–Ford Algorithm**.
- Routers send **periodic updates** to neighbors.
- ✅ **Advantages:** Simple and easy to set up.
- ❌ **Disadvantages:** Slow convergence, routing loops may occur.
- **Examples: RIP** → *Routing Information Protocol*

(b) Link State Routing Protocols

- Each router knows the **complete network topology**.
- Based on **Dijkstra's Shortest Path Algorithm**.

- Sends updates **only when a change occurs**.
- ✅ **Advantages:** Fast convergence, more accurate routing.
- ❌ **Disadvantages:** Needs more **memory and CPU**.

Examples: OSPF (Open Shortest Path First)

🌐 2️⃣ Exterior Gateway Protocols (EGP) / Inter-Domain Routing Protocols

Used **between different Autonomous Systems (AS)** — e.g., between ISPs.

- Purpose: Exchange routing info between **independent networks**.
- Type: **Path Vector Protocol**.
- **Example:** BGP (Border Gateway Protocol).

⚙️ 3️⃣ Hybrid Routing Protocols

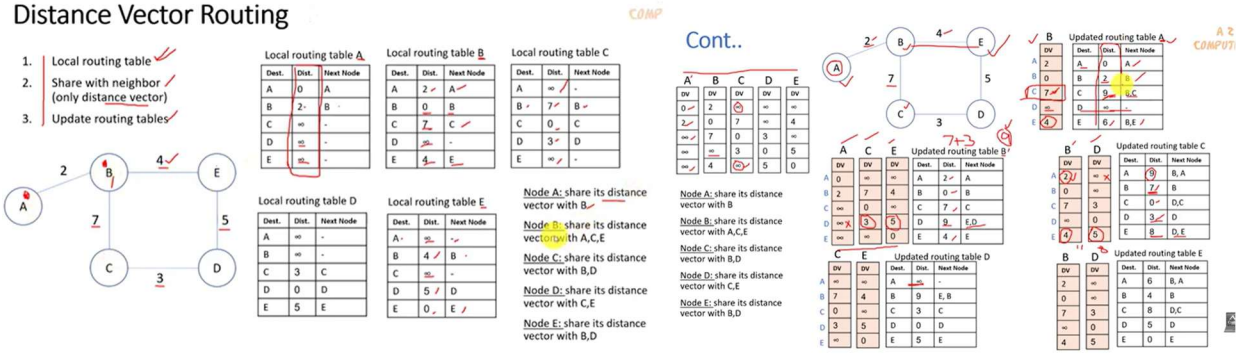
- Mix of **Distance Vector + Link State** features.
- Uses **both hop count and topology** info.
- Converges **faster** than pure Distance Vector.
- **Example:** EIGRP (Enhanced Interior Gateway Routing Protocol – Cisco proprietary).

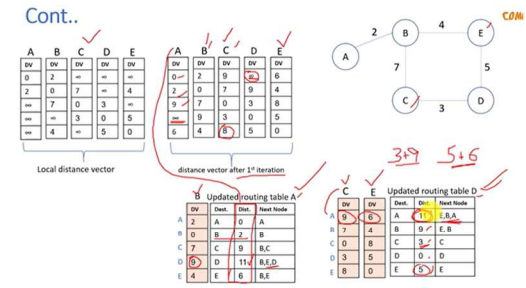
Routing Algorithms / Routing Protocols

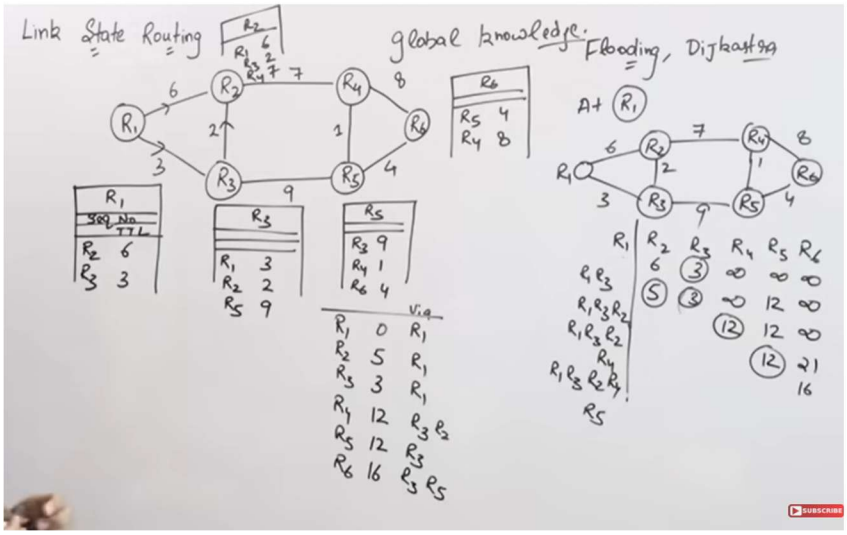
This is where **DVR** and **Link State** come in.

🔹 Two Main Types of Routing Algorithms

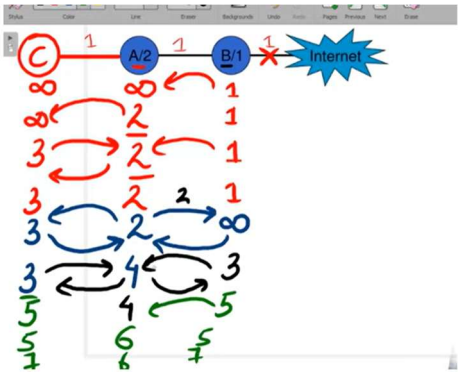
Type	Description	Examples
1. Distance Vector Routing (DVR)	Each router shares its routing table (distance to other networks) with **neighboring routers**. Routing decisions based on distance (hop count).	RIP (Routing Information Protocol)
2. Link State Routing (LSR)	Each router discovers its **neighbor links and costs**, builds a full network map, and calculates shortest path using **Dijkstra's Algorithm**.	OSPF (Open Shortest Path First)







Count to infinity problem in distance vector



⭐ Flooding Algorithm

- Flooding is a **routing technique** where every incoming packet is **sent out on all outgoing lines** except the one on which it arrived.
- Used when no routing table exists or in emergency situations.
- Ensures the packet reaches **all possible paths** to the destination.
- Guarantees delivery if **a path exists**, even if routers have no knowledge of the network.
- Causes **duplicate packets**, **congestion**, and **wastage of bandwidth**.

⭐ Problems with Flooding

- Creates **duplicate packets**.
- Requires **large bandwidth**.
- Causes **network congestion**.
- Increases processing load on routers.

⭐ Solutions to Control Flooding

1. **Hop Count / TTL (Time To Live)**
    - Each packet has a limit of hops after which it is discarded.
2. **Sequence Numbering**
    - Source attaches a unique number; routers discard duplicates.

Congestion Control

• Too much packet in network → congestion → delay / packet loss.
• Control methods:

- **Leaky Bucket** – Sends data **slowly at a steady rate**, extra data waits in a queue.
- **Token Bucket** – Sends data **when tokens are available**, allows bursts of data.

☑ Network Layer Congestion Control (Routers/IP)

- Done **inside routers**.

- Manages congestion in the network path.
- Uses techniques like:

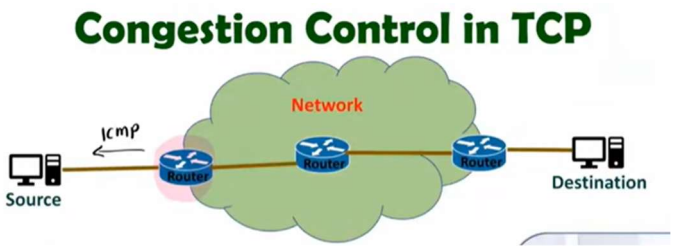
  ✓ Queue management: How routers **manage packets waiting in the queue**.
  - **RED** → drops packets early to avoid congestion
  - **Drop Tail** → drops packets only when queue is full

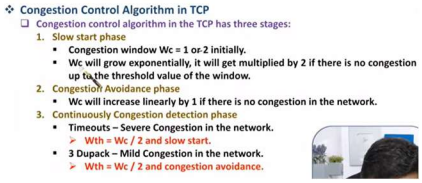
  ✓ Packet dropping: hen there is no space or heavy congestion.

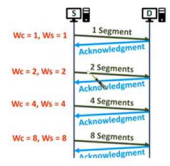
  ✓ ECN (Explicit Congestion Notification): Instead of dropping packets, the router **marks** packets to warn the sender about congestion so it can slow down.

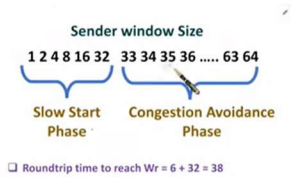
☑ Transport Layer Congestion Control (TCP)

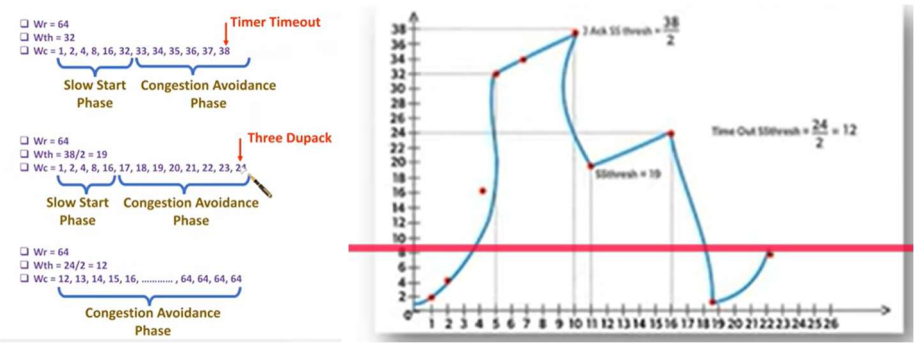
- Done **end-to-end** by sender & receiver.
- Adjusts **sending rate** based on congestion signals.
- Uses algorithms like:
  - Slow Start
  - Congestion Avoidance
  - **Fast Retransmit** → Retransmits lost packets immediately after receiving **3 duplicate ACKs** (without waiting for timeout).
  - **Fast Recovery** → After fast retransmit, **reduces congestion window slightly** instead of starting from scratch, allowing faster recovery.
- Goal: **Control how fast data is sent to avoid overload.**











Transport Layer – Connection Management

Connection management involves establishing, maintaining, and terminating connections between processes.

---

1. Connection Establishment (3-Way Handshake)

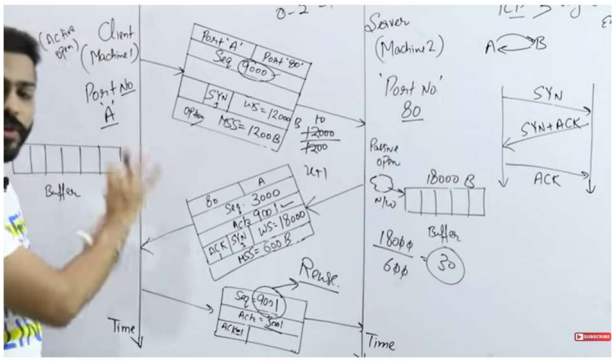
Used in TCP:

1. **SYN – Client sends connection request.**
2. **SYN + ACK – Server acknowledges and sends its own request.**
3. **ACK – Client acknowledges.**

Connection is now established.

Purpose:

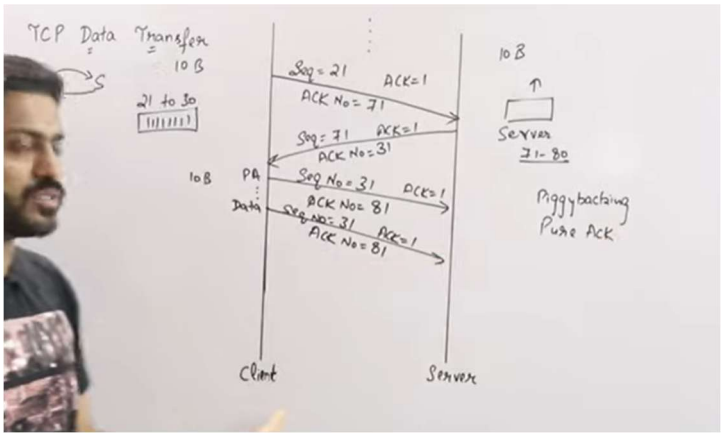
- **Agree on sequence numbers**
- **Ensure both sides are ready**
- **Prevent old duplicate packets from forming connections**



---

2. Data Transfer Phase

- **Reliable transmission using:**
  - **ACKs**
  - **Retransmissions**
  - **Sliding window**
  - **Flow and congestion control**



1️⃣ **Pure Acknowledgment (ACK): The receiver sends a separate acknowledgment for received data.**

2️⃣ **Piggybacking: The receiver sends ACK along with its own data in the same segment and Saves bandwidth**

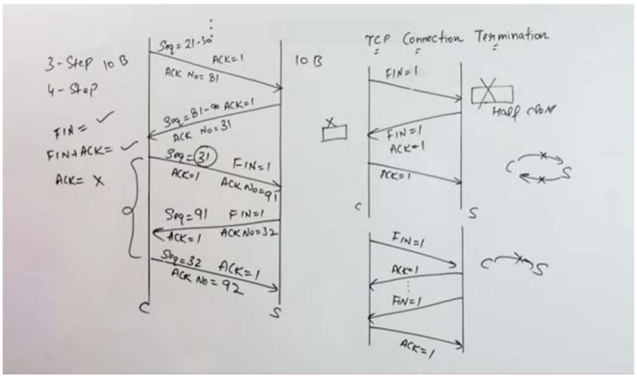
---

3. Connection Termination (4-Way Handshake)

Steps:

1. **FIN – Client sends connection termination request.**
2. **ACK – Server acknowledges.**
3. **FIN – Server sends its own termination request.**
4. **ACK – Client acknowledges termination.**

Connection is closed gracefully.



IPv4 VS IPv6

Feature	IPv4	IPv6
Address Size	32-bit address	128-bit address
Address Space	**Total addresses:** 2^{32}= About **4.3 billion** addresses	**Total addresses: 2^{128} = Almost unlimited** (around 340 undecillion addresses)
Address Format	• Written in **decimal** (e.g.192.168.1.10 → 11000000.10101000.00000001.00001010) • Divided into **4 parts (octets)**. • Each part is **8 bits**, written in decimal numbers separated by dots (.).	• Written in **hexadecimal** (e.g., 2001:0db8:85a3:0000:0000:8a2e:0370:7334) • Divided into **8 parts (groups or blocks)**. • • Each part is **16 bits**, written in hexadecimal (0–9, a–f) and separated by colons (:).

Feature	IPv4	IPv6
Address Range	**0.0.0.0** to **255.255.255.255**	**0000:0000:0000:0000:0000:0000:0000:0000** to **FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF**
Header Complexity	**Complex header** with more fields	**Simpler header**, faster processing
Security	**Security optional** (IPSec can be added manually)	**IPSec (internet Protocol Security like CIA triad) built-in**, provides better data protection

- Total IP addresses

$$2^{\text{host bits}}$$

- Usable (Reusable) IPs

$$2^{\text{host bits}} - 2$$

both IPv4 and IPv6 are connectionless and datagram-based.

✅ **Both IPv4 and IPv6 are connectionless**

- They do **not** set up a connection before sending data.
- They just send packets directly.

✅ **Both IPv4 and IPv6 are datagram-based**

- Data is sent in **independent packets** (datagrams).
- Each packet can take different paths and arrive at different times.

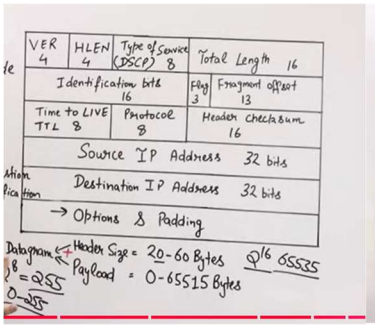
IPv4 Address Types

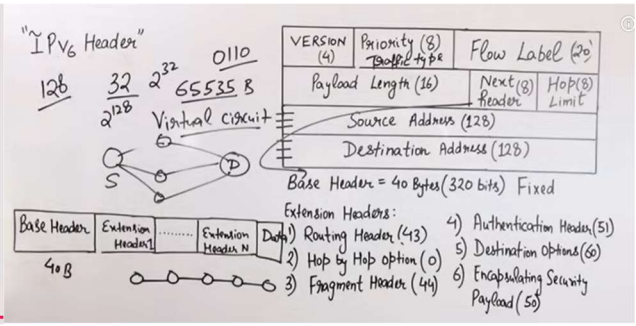
1. **Unicast** – Goes to **one specific device**.
2. **Broadcast** – Goes to **all devices** on a network.
3. **Multicast** – Goes to a **group of devices**.
4. **Anycast** – Goes to the **nearest device** in a group (rare in IPv4).

IPv6 Address Types

1. **Unicast** – Goes to **one interface**.
2. **Multicast** – Goes to a **group of interfaces**.
3. **Anycast** – Goes to the **nearest interface** in a group.
4. **No broadcast** in IPv6 – replaced by multicast.

Ip4 header ipv6





1. Version (4 bits)

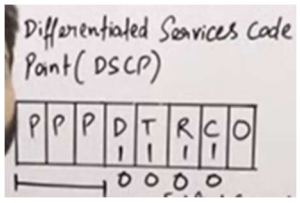
- Tells which IP version is used (IPv4 → value = 4).

2. IHL – Internet Header Length (4 bits) means 15 value means 15*4 =60 byte max

- Size of the header in 32-bit words (minimum 20 bytes).

3. Type of Service / DSCP (8 bits)

- Priority of the packet (QoS).



4. Total Length (16 bits)

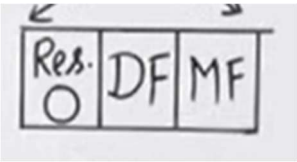
- Total size of packet = header + data.

5. Identification (16 bits)

- Used for uniquely identifying fragments of a packet.

6. Flags (3 bits)

- Controls fragmentation.
    - DF – Don't Fragment
    - MF – More Fragments



7. Fragment Offset (13 bits)

- Tells the position of a fragment in the original data.

8. TTL – Time To Live (8 bits)

- Limits packet lifetime; prevents infinite loops.
- Decreases by 1 at every hop.

9. Protocol (8 bits)

- Which transport protocol?
    - 6 → TCP
    - 17 → UDP
    - 1 → ICMP

10. Header Checksum (16 bits)

- Error checking for the header only.

11. Source IP Address (32 bits)

- Sender's IPv4 address.

12. Destination IP Address (32 bits)

- Receiver's IPv4 address.

13. Options (Variable, optional)

- Rarely used. For things like security

14. Padding

- Extra zeros added to make header length a multiple of 32 bits.

✅ **IPv6 Header – Important Points**

The IPv6 header is fixed at 40 bytes (unlike IPv4 which is variable).

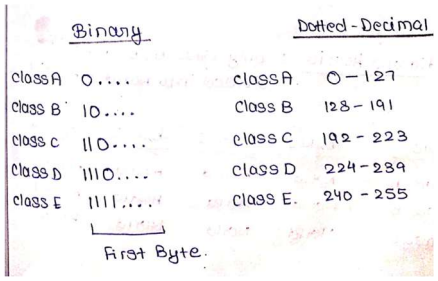
🔹 **IPv6 Header Fields (with one-line meaning)**

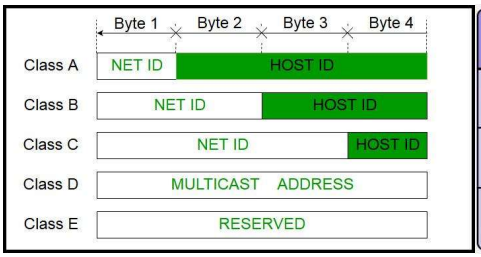
✅ **Version (4 bits): IP version = 6.**
✅ **Traffic Class (8 bits): Priority/QoS of the packet.**
✅ **Flow Label (20 bits): Identifies packets of the same flow for fast routing.(same path is followed)**
✅ **Payload Length (16 bits): Size of data after the header.**
✅ **Next Header (8 bits): Identifies the next protocol (TCP/UDP/extension header).**
✅ **Hop Limit (8 bits): Number of hops allowed (same as TTL in IPv4).**
✅ **Source Address (128 bits): Sender's IPv6 address.**
✅ **Destination Address (128 bits): Receiver's IPv6 address.**

IPv6 Extension Headers

- **Fragment Header (44) → Used when a packet is broken into pieces by the sender (routers never fragment).**
- **Authentication Header (AH)(51) → Adds integrity + authentication (no encryption).**

IPv4 Address Classes





Class	Default Subnet Mask
A	255.0.0.0
B	255.255.0.0
C	255.255.255.255

- **Class A:** 0–127 → Big networks
- **Class B:** 128–191 → Medium networks

- **Class C:** 192–223 → Small networks
- **Class D:** 224–239 → Multicast
- **Class E:** 240–255 → Research/Experimental

👉 **Private IP ranges exist only in Class A, B, and C.**
👉 **Class D and E do NOT have private IP ranges.**

Netid & Hostid

- IPv4 = **Network part + Host part**
- Found using **Subnet Mask**

Examples:

- Class A mask → **255.0.0.0**
- Class B mask → **255.255.0.0**
- Class C mask → **255.255.255.0**

✅ **How to Find NetID and HostID** (IP address: **192.168.1.25)**

🔥 **EASY METHOD 1 (No binary):**

Just look at the subnet mask:

Example: 255.255.255.0

- First 3 octets = 255 → **network**
- Last octet = 0 → **host**

So:

NetID = copy the network part → 192.168.1.0
HostID = copy only the host part → 0.0.0.25

🔥 **METHOD 2 (binary):**

Term	**Formula**
Net ID	**IP AND Subnet Mask**
Host ID	**IP – Net ID**

To find them, you only need:

1. **IP address**
2. **Subnet mask**

The subnet mask tells you **how many bits belong to the network** and how many bits belong to the host.

⭐ STEP 1: Write the IP and the Subnet Mask

Example:
IP address: **192.168.1.25**
Subnet mask: **255.255.255.0**

⭐ STEP 2: Convert Both to Binary

(You don't always need to convert—but this explains the logic.)

IP:

192 168 1 25

11000000.10101000.00000001.00011001

Subnet mask:

255 255 255 0

11111111.11111111.11111111.00000000

⭐ STEP 3: AND the IP with the Subnet Mask

(1 AND 1 = 1, everything else = 0)

IP: 11000000.10101000.00000001.00011001

Mask: 11111111.11111111.11111111.00000000

NetID: 11000000.10101000.00000001.00000000

Convert NetID back to decimal:

192.168.1.0

⭐ **STEP 4: Find the HostID**

HostID = IP − NetID (in each octet)

IP: 192.168.1.25

NetID: 192.168.1.0

HostID: 0.0.0.25

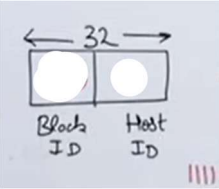
⭐ **Final Answer**

Part	Meaning	Result
NetID	Identifies the network	**192.168.1.0**
HostID	Identifies the device in that network	**0.0.0.25**

Subnet Mask (Address Masking)

- Used to find which part is **network** and which is **host**
- Example:
    - IP: 192.168.10.5
    - Mask: 255.255.255.0
      → Network = 192.168.10.0s
      → Host = 5

Classless Addressing (CIDR)/ CIDR (Classless Inter-Domain Routing)



- No fixed classes (A/B/C).
- Format: **IP/prefix** (example → 192.168.1.0/24) 24 is mask mean 11111111.1111111.111111.00000000 total 24 one in octate form
- **Classless addressing (CIDR)** = allows **supernetting** (and subnetting) using prefixes like /20, /22, etc.
- Benefits:
    - Saves IP addresses
    - Flexible network sizes
    - Efficient allocation

Rules:

- Block size must be **power of 2**
- Addresses must be **contiguous**

Example:

192.168.1.0/24 (NetworkAddress / PrefixLength)

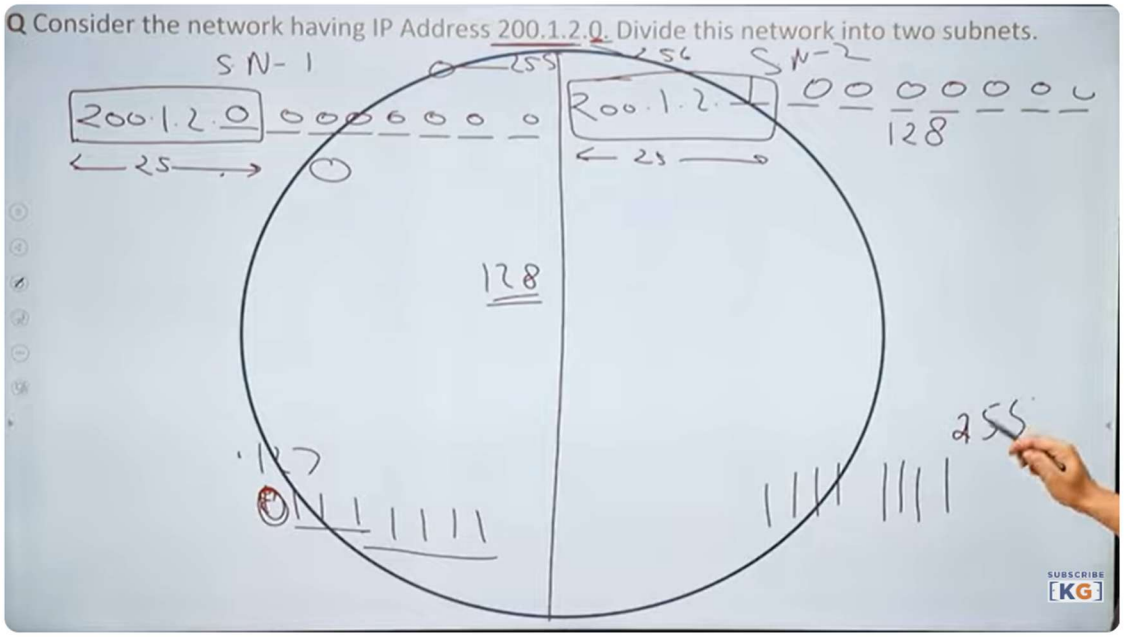
- /24 means first 24 bits are network
- the remaining 8 bits are host

Subnetting (Dividing a Network)

- Break one network into **multiple smaller networks**
- Helps in:
    - Security
    - Traffic control
    - Better organization

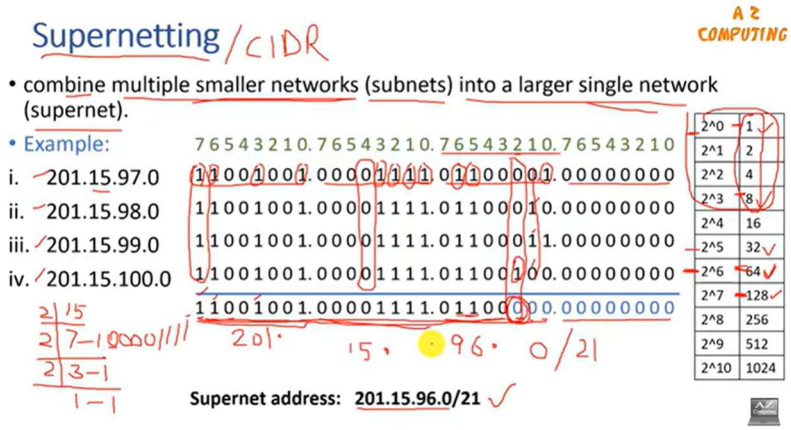
Example:

- 192.168.1.0/24 → subnet to /26 (26-24=2 => 2^2=4)
  → Creates 4 subnets (0, 64, 128, 192)



Supernetting

- Combine many **small networks** into **one big network**
- Used to reduce routing table size



Example:

192.168.1.0/24 + 192.168.2.0/24
→ Supernet = **192.168.0.0/22**

⭐ OSI Layers – Protocols

1️⃣ Physical Layer

- **Ethernet (Physical)** – Defines electrical and signaling standards for network cables.
- **DSL** – Uses telephone lines for high-speed internet.
- **RS-232** – Standard for serial communication.(recommended standard)
- **Bluetooth (Physical)** – Short-range wireless communication standard.

2️⃣ Data Link Layer

- **Ethernet (MAC)** – Handles frame formation and MAC addressing.
- **Wi-Fi MAC (802.11)** – Defines wireless LAN frame handling.
- **PPP** – Provides data link connection between two nodes.
- **ARP** – Maps IP addresses to MAC addresses.
- **Frame Relay** – High-speed WAN protocol for frames.

3️⃣ Network Layer

- **IP (IPv4/IPv6)** – Provides logical addressing and routing of packets.
- **ICMP** – Reports errors and diagnostics in the network.
- **OSPF** – Link-state routing protocol for shortest path.
- **RIP** – Distance-vector routing protocol using hop count.
- **BGP** – Routing protocol used between ISPs.
- **IPsec** – Provides security for IP communication.

4️⃣ Transport Layer

- **TCP** – Reliable, connection-oriented data transport.
- **UDP** – Fast, connectionless transport without reliability.
- **DCCP (Datagram Congestion Control Protocol): Supports congestion control for real-time applications.**

5 Session Layer

- **RPC (Remote Procedure Call):** Allows running a function on another computer.
- **NetBIOS (Network Basic Input/Output System):** Helps computers create and manage sessions in a LAN.
- **PPTP (Point-to-Point Tunneling Protocol):** Creates VPN sessions.
- **SIP (Session Initiation Protocol):** Sets up and manages voice/video call sessions.

---

6 Presentation Layer

- **SSL (Secure Sockets Layer):** Provides encryption and secure communication.
- **TLS (Transport Layer Security):** Newer, stronger version of SSL for secure communication.
- **JPEG (Joint Photographic Experts Group):** Compression standard for images.
- **ASCII (American Standard Code for Information Interchange):** Text character encoding standard.
- **MPEG (Moving Picture Experts Group):** Compression standard for audio and video.

---

7 Application Layer

- **HTTP (tcp-80)/HTTPS (tcp-443)** – Protocol for web browsing.
- **FTP (tcp-20/21)** – Transfers files between client and server.
    - **Port 21 → Commands** (tells the server what you want to do, like login or list files)
    - **Port 20 → Data** (actually sends or receives the files)
- **SFTP(tcp-22) (SSH File Transfer Protocol)-** Transfer files securely over a network.
- **SMTP (tcp-25)** – Sends emails.
- **POP3 (tcp-11))** (Post Office Protocol Version 3): Downloads email to the device and deletes it from the server.
- **IMAP (tcp-143/993)** (Internet Message Access Protocol): Reads and manages email directly from the server using the internet.
- **DNS (udp-53)** – Converts domain names to IP addresses.
- **DHCP (udp-67/68)** – Assigns IP address automatically. (dynamic host configuration protocol)
- **SSH (tcp-22)** – Secure remote login.
- **Telnet (tcp-23)** – Remote login without encryption.

⭐ HUB

What it is:

A hub is a very simple device that connects many computers in one place.

Passive Hub: Just forwards signals.

Active Hub: Forwards + amplifies signals.

Function:

- It sends every incoming signal to **all** connected devices.
- It does not check or filter data.
- It cannot choose the best path, so network becomes slow and wasteful.

OSI Layer:

Physical Layer (Layer 1)

⭐ SWITCH

What it is:

A switch is a smarter device that connects many computers. It is like an upgraded hub.

Function:

- Sends data **only to the correct device** using MAC address.
- Stores data temporarily using buffer memory.
- Checks errors before forwarding.
- Gives better speed and efficiency.

OSI Layer:

Data Link Layer (Layer 2)
(Some advanced switches also work at Layer 3, but basics are Layer 2)

⭐ BRIDGE

What it is:

A bridge connects two LAN segments. It is basically a repeater that can **filter** data.

Function:

- Reads MAC addresses.
- Decides whether to forward or block the data.
- Connects two LANs that use the same protocol.

OSI Layer:

Data Link Layer (Layer 2)

Types:

Transparent Bridge: Forwards data automatically without devices knowing.

Source Routing Bridge: Follows the path set by the sender.

⭐ ROUTER

What it is:

A router is a device that connects different networks.

Function:

- Forwards data using **IP address**.
- Chooses the best path using routing table.
- Connects LANs and WANs.
- Splits broadcast domains to reduce unnecessary traffic.

OSI Layer:

Network Layer (Layer 3)

⭐ GATEWAY

What it is:

A gateway is a device or node that connects **completely different types of networks**.

Function:

- Converts data format/protocol so two different networks can communicate.
- Serves as main entry and exit point of a network.
- Used when networks speak different "languages".

OSI Layer:

Multiple Layers (often **Layer 7 – Application Layer** depending on type)

TCP/IP Reference Model?

- It is a practical model used on the internet.
- name of this model is based on 2 standard protocols used i.e. TCP (Transmission Control Protocol) and IP (Internet Protocol).
- Developed by the **Department of Defence (DoD) in 1960's**.
- Has **4 layers**

4 Layers (most common):

1. **(Netowrk Access)Link** : Decides which links such as serial lines or classic Ethernet must be used

to meet the needs of the connectionless internet layer. Ex - Sonet, Ethernet

2. **Internet** : The internet layer is the most important layer which holds the whole architecture together. It delivers the IP packets where they are supposed to be delivered. Ex - IP, ICMP.

3. **Transport** : Its functionality is almost the same as the OSI transport layer. It enables peer entities on the network to carry on a conversation. Ex - TCP, UDP (User Datagram Protocol)

4. **Application** : It contains all the higher-level protocols. Ex - HTTP, SMTP, RTP,

DNS.

Purpose:
Helps computers communicate over networks using standard protocols.

Important Protocols:

- **TCP**, **UDP**, **IP**, **ICMP**, **DHCP**

Key Characteristics

- Hierarchical model with interactive modules.
- Follows **client-server architecture.**
- Used for **end-to-end data transmission over the internet.**

TCP and IP Working Together

- **TCP**: Breaks data into packets, ensures reliable delivery, reassembles data.
- **IP**: Routes packets to the correct destination.

---

2. TCP (Transmission Control Protocol)

- **Connection-oriented** → connection is made before sending data.
- **Reliable** → ensures data reaches correctly.
- **Sequences** data and **retransmits** if something is lost.
- **Slower**, but accurate.
- **Header size:** 20–60 bytes.
- Used by: **HTTP, FTP, SMTP**

---

3. UDP (User Datagram Protocol)

- **Connectionless** → no connection needed.
- **Unreliable** → no delivery guarantee.
- **No error checking** → faster.
- **Header size:** 8 bytes.
- Used for: **Online games, videos, audio calls, DNS**

---

4. Other Transport Layer Protocols

- **AEP:** Tests whether a network device is reachable.
- **AH:** Provides authentication (IP security).
- **DCCP:** Supports congestion control (used in streaming).
- **FCP:** Used in Fibre Channel networks for high-speed storage.
- **NBP:** Naming protocol used in AppleTalk networks.

AEP – Address Exchange Protocol

AH – Authentication Header

DCCP – Datagram Congestion Control Protocol

FCP – Fibre Channel Protocol

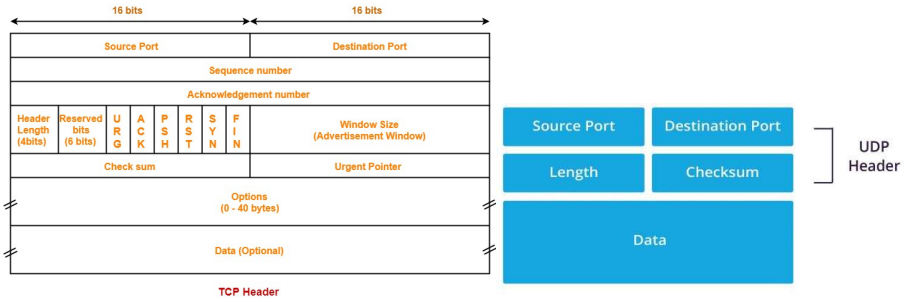
NBP – Name Binding Protocol

---

5. TCP vs UDP (Very Easy Table)

TCP	**UDP**
Connection-oriented	Connectionless
Reliable	Not reliable
Slow	Fast
Retransmits lost data	No retransmission
Heavy protocol	Light protocol
Uses sequencing	No sequencing
Big header (20–60 bytes)	Small header (8 bytes)

TCP	UDP
Used in web, email, file transfer	Used in streaming, gaming, DNS



Key Fields in TCP Header

- Source Port (16 bits): Identifies the sending port.
- Destination Port (16 bits): Identifies the receiving port.
- Sequence Number: Used for data ordering and reliability.
- Acknowledgement Number: Confirms receipt of data.
- Header Length (4 bits): Specifies the size of the header.
- Reserved Bits (6 bits): For future use, set to zero.
- Control Flags (URG, ACK, PSH, RST, SYN, FIN): Control various functions such as connection setup, termination, and data push.
- Window Size: Indicates the size of the sender's receive window (flow control).
- Checksum: Used for error-checking the header and data.
- Urgent Pointer: Points to urgent data, if the URG flag is set.
- Options (0–40 bytes): Can include features like window scaling or timestamps.
- Data (Optional): Actual payload sent over the connection.

Key Fields in UDP Header

- Source Port: Identifies the sending port for the datagram.
- Destination Port: Identifies the receiving port.
- Length: Specifies the total length of the UDP header and data.
- Checksum: Used for error-checking of the header and data.
- Data: Contains the actual payload being sent.

Public network – open to anyone freely like wifi in cafes, library, its scope is local.

Computer Network (group of interconnected computers that share resources, data and services)

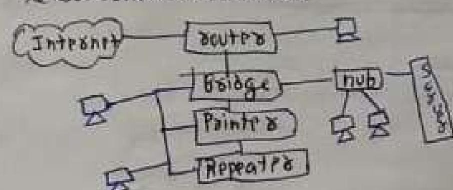
Internet – It is a massive global network connecting many smaller network

First Computer Network
ARPANET – Advanced Research Projects Agency Network in 1969

Main goal of Computer Network – Resource Sharing (Allow multiple computers in a network to use the same hardware, software, data or services)
- Spooling: stores print/data jobs on disk to be processed later in a queue (FIFO)
  Ex – used when multiple print jobs are sent at once
- Buffering: temporarily stores data in memory to match speed between CPU and I/O
  Ex – improve efficiency when devices operate at different speeds.

Characteristics of Computer Network | Goals of CN
1) Resource Sharing – Share devices like printers using spooling and Buffering
2) Communication speed – fast data transmission among connected systems Ex – whatsapp message
3) Backup – Data can be backed up and restored easily over the network Ex – Add new computer to wifi
4) Scalability – Network performance remains stable even if no. of users increases
5) Reliability – Services continue even if one network path crashes, ensures high availability Ex – If one internet cable breaks, data is routed through another path
6) Security – Protect data from unauthorized access using firewalls, encryption etc
7) Software and Hardware Sharing – Applications, storage and peripheral devices can be accessed remotely.

Components of Computer Network
1) Hardware – devices used to connect to network and transfer data between devices
   Ex – server, client, router (directs data), switch & hub (connect devices), bridge (connect two Network), Repeater (Boosts signals), Gateway (connects different network type)
2) Software – controls and manages network operations
   Ex – Network Operating System (Ex – Window server manage network resources & communication), Protocols (TCP/IP, OSI) (set of rules for communication)



Application of Computer Network
- Business Application
  - Resource sharing (file & Printer)
  - Remote work
  - cloud computing
  - Email & messaging
  - video conferencing
- Home Application
  - Internet Browsing
  - entertainment
  - online shopping & Banking
  - Smart home control like Alexa, Smart bulb
- Mobile user Application
  - wifi & hotspots
  - GPS & maps – (GPS: global positioning system)
  - mobile
  - wearables (smartwatch)
- Social & community Application
  - Social Media – Insta
  - Knowledge sharing – Quora

Transmission Technologies (define how data is transmitted b/w devices in a network)

1 Point-to-Point Link Network :- Direct connection b/w two devices only
- Data is sent directly from one sender to one receiver
- used for private, secure and fast data transfer
Ex – USB cable b/w PC and printer

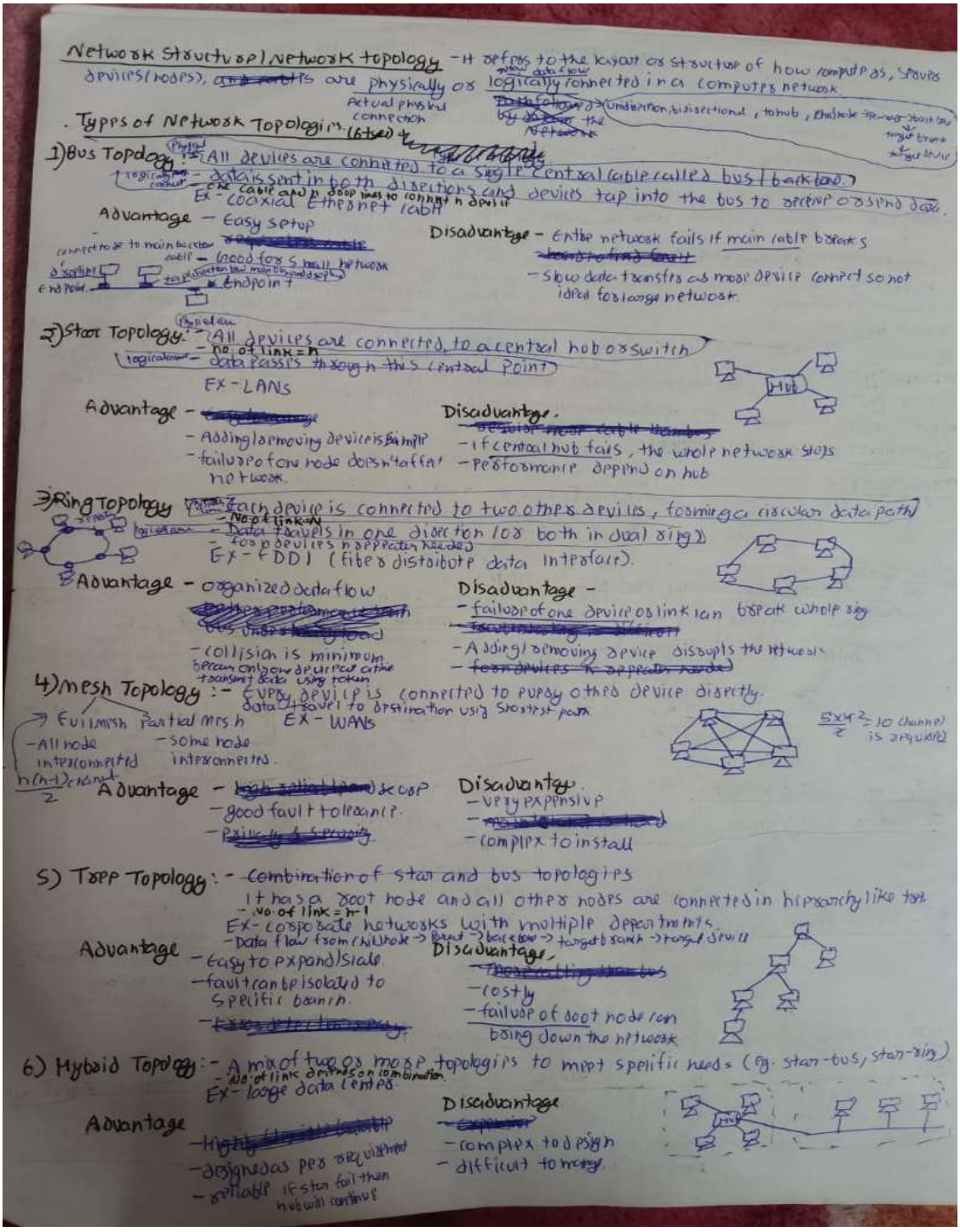
2 Broadcast Network :- One device sends data to all devices connected to a shared medium
- All receivers listen, but only the intended device processes the message.
- Supports Broadcasting (send to all), Multicasting (send to specific group), Unicasting (send to one device)
- Ex – TV or radio signals

2 Dimension of Computer Network
- Transmission Technology
- Scale/size (PAN, LAN, MAN...)

Types of Computer Networks | Categories of Computer Network | Scale.

Type	Tech	Range	Speed	Area	Ownership	Maintenance	Ex...
PAN (Personal Area Network)	Bluetooth, IR	1 - 100 m	very high	Room	Private	Very Easy	V...
LAN (Local Area Network)	Ethernet, wifi	upto 2 Km	very high	office/building	Private	Easy	I...
CAN (Campus Area Network)	Ethernet	1 - 5 Km		university/campus		Moderate	
MAN (Metropolitan Area Network)	FDDI, ATM (FDDI: fiber distributed data interface)	5 - 50 Km	high	city-wide	Private		A...
WAN (Wide Area Network)	leased line, Dialup	Above 50 Km	average low	country/continent	Private/Public	Difficult / very Difficult	

ATM = asynchronous transfer mode.



Note: Tree topology is a combination of star and bus topology. It is also known as the expanded star topology.

(PDNTSPA)

OSI Model - Open Systems Interconnection is a 7-layer framework that describe how data travels from one computer to another in a network

1. Physical layer (Hardware layer) (lowest layer) (1st layer) - sends raw bits through a physical medium (cables or wireless) and at receiver, it converts signal back into bits

functions -

- Transmission media - actual physical path the data takes - like copper cables, fiber optics or air for wifi signals
- Encoding - Turning digital 0s and 1s into physical signal like light pulses (fiber), voltage changes (copper) or radio waves (wireless)
- Bit synchronization - Aligning sender's and receiver's clocks so bits don't get out of rhythm and receiver knows exactly when one bit ends and next bit begins (without synchronization ... could be read as 10 11 or 1 011 ...)
- Topology: Shape of network layout (star, bus, ring etc) which affects how devices connect
- Transmission Mode: define direction of data flow.
  - Simplex - one way only (ex - TV broadcast)
  - Half-duplex - Two way, but one talks at a time (ex - walkie-talkie)
  - Full-duplex - Two-way, both can talk at same time (ex - phone call)

Atom, Attenuation, Delay; Ethernet (physical), Bluetooth (physical), DSL, RS-232; H/W layer

2. Data link layer (Reliable link layer) - makes raw physical link error-free and organized; ensure reliable data transfers b/w two directly connected devices over a physical link

functions -

1) Framing - Groups the raw bits into chunks called frames so the receiver knows where data start and ends. Frame contain header, payload and trailer (mac address, data, error check info)
2) MAC Addressing - uses each device's unique hardware ID to send data to right one (Add mac addr) (scope inside LAN)
3) Flow control - Ensures the sender doesn't send faster than the receiver can handle (ex - control data flow two directly connected devices) — stop and wait protocol
4) Error controls - detect errors in frames and fixes them, often by asking for a resend (in frames, checksums)
5) Access controls - who can send data ...

(PPP, ARP, Frame relay; ethernet (with mac))

3. Network layer (Addressing & Routing layer) - find best path for data b/w devices across networks

functions -

- IP addressing - (assign IP address) gives every device a logical address so data knows where to go (scope global)
- Routing - chooses the best route for the packet to reach its destination, possibly through many networks in between
- Packet forwarding - packet has header and payload (data)
- Fragmentation and reassembly at destination (breaking packet)
- Error handling and manage network congestion

(IP, ICMP, OSPF, RIP, BGP, IPsec)

4. Transport layer (End-to-End Delivery layer) of OSI - ~~makes sure the message gets completely and correctly from one app to another by adding port no.~~ reliable data transfers b/w two end system

function

- Port no - tell computer which program should receive data (ex ~~web~~ = port 80, email = port 25) (add port no.) (HTTPS port 443, HTTP ... with ...)
- Segmentation & Reassembly: splits large data into smaller chunks before sending, then reassembles them at the destination using sequence no. (segment - it contain header and payload: port & seq no, data)
- ~~Flow control and error control~~
- Connection control
  - TCP (transmission control protocol) - reliable, checks everything arrived, resends if needed
  - UDP (user datagram protocol) - faster but doesn't check for missing pieces
- Process to process delivery (using port), congestion control
- Connection establishment & release: 3-way handshake and 4-way ...
- Multiplexing & Demultiplexer

(TCP, UDP, DCCP) (controls data b/w source & destination applications on the network)

5. Session layer (Session manager) - manages and coordinates communication session b/w appⁿ (dialog is connection b/w two systems)

function

- Dialog control - decides who sends/receives and when within a session. Ex - half-duplex, full duplex
- Synchronization - inserts checkpoints so if a connection breaks, data transfer can resume from the last checkpoint
- Authentication and Authorization
- Session establish, maintain and terminating
- Session recovery - restore session using checkpoint after interruption
- Token Management - avoid collision during simultaneous ...

(RPC, NetBIOS, PPTP, SIP; SW/only ...)

6. Presentation layer (Data Translator & Security layer) - makes sure data is in a format that appⁿ can understands

function

- Translation - converts b/w different data formats (ex text encoding from ASCII to Unicode)
- Encryption - scrambles data so only the intended recipient can read it
- Compression - shrinks file size so it travels faster < lossless compression; lossy compression
- Data format - JPEG, MP3, MPEG
- Serialization - converts structure data to linear form for transmission

(SSL, TLS, JPEG, ASCII, MPEG)

7. Appⁿ layer (user interface to Network) - provide network services directly to end user ... using s/w · It don't handle data transport

services / function

- Remote Access - connect to another computer (ex - SSH, Telnet)
- File transfer - send and receive files (ex - FTP)
- Email - send and receive messages (ex - SMTP, IMAP)
- Directory services - look up addresses and resources (ex - DNS for website address) (Domain Name System)
- Web browsing - HTTP/HTTPS (ex - open ... in your browser using HTTPS ...)

(FTP 20 21, SFTP 22, SMTP 25, POP3 110, IMAP 143, DNS 53, Telnet 23, ...)

Sender

Application layer
Presentation layer
Session layer
Transport layer
Network layer
Data link layer
Physical layer

Receiver

Application layer
Presentation layer
Session layer
Transport layer
Network layer
Data link layer
Physical layer

Pdf – Portable document format

URL – Universal resource locator

MAC – media access control

[illegible]

Frame: Receiver MAC, Sender MAC, Packet: Receiver's IP address, Sender's IP address, TCP header, Port no., Data

Delay (latency) – It is the total time it takes for a data packet to go from source to destination

Sender ⟶ Receiver

Step
- Preparing msg to transmit by sender
- transmission over communication n/w
- msg received at receiver end
- Processing of msg at receiver end

Link = physical or logical path that data travels b/w two points in a network (ex – computer → switch) | Path = sequence of multiple links from source to destination (computer → switch → router → server)

Main types of Delay

1- Transmission Delay – time to push all bits of packet into the transmission medium

$$T.D = \frac{\text{Packet size (bits)}}{\text{Transmission rate (bps)}} = \frac{\text{Packet size}}{\text{Bandwidth}}$$

2- Propagation Delay – time for signal to physically travel through the medium (cable, fiber, air)

$$P.D = \frac{\text{Distance (m)}}{\text{Propagation speed (m/s)}}$$

3- Processing Delay – time a router or device takes to process the packet header, check for errors and decide where to send it (usually a few microseconds or milliseconds)

4- Queuing Delay – time packet spends waiting in a queue / buffer before being transmitted. it depends on network congestion.

Total Delay = T.D + P.D + P.D + Q.D

we know, optical fiber carries signal at 70% of speed of light

$$\therefore \text{propagation speed in fiber} = \frac{70}{100} \times 3 \times 10^8 \text{ m/s}$$

$$= 21 \times 10^7$$

$$= 2.1 \times 10^8 \text{ m/s}$$ (it is propagation speed not transmission speed)

Computer network architecture – It defines the design, structure and operational layout of network

- How devices are connected, How data flows b/w them, Protocols & technologies used

Type of Network Architecture / Type of CN on basis of architecture.

1- Peer-to-Peer (P2P) Networks – every node acts as both client and server. Each device can both request and provide service

use cases – suited for small setups like file sharing, messaging and gaming

Advantage – easy to set up, no dedicated server needed

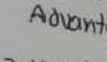
Types –
- structured P2P : nodes are organized systematically, making data search more efficient
- unstructured P2P : No specific organization, nodes can join / leave easily but searching is less efficient

2. Client-Server Networks :- client requests and server store and manage resources and ensuring security.

Use cases :- File sharing, db, emails, web hosting

Advantages : centralized control for management and security, well-suited for larger organization

3. Hybrid Networks - combines P2P and client-server features



devices can act as both client and server, depending on need

Example : IoT - sensors and actuators in devices interact dynamically

Advantages : cost effective, benefit of both.

4. Cloud-Based Architecture - on demand access to resources / services (ex- data storage, apps).

front end - user/client interface

Internet - connect client to cloud servers.

backend - cloud provider's infrastructure (~~data storage~~ db, servers)

storage - store user's data, file and backup in cloud

Advantage - flexibility, scalability.

Network architecture design - refers to structured plan for how devices, servers and services in a network are connected, communicate and operate to meet performance, security and scalability needs.

1) Flat/Two-tier Architecture - consist of Access layer and distribution layer

client interact directly with server.

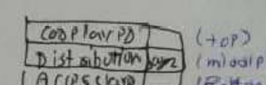
components -

firewall - control network access

router - route data b/w networks

switch - connects devices within a network

wireless access point - enable wireless connections.



Pros - cost-effective, easy to manage.

cons : limited scalability, security risk if firewall is compromised as there are maintained security policies

2) Three-Tier Architecture - consist of Access layer, distribution layer, core layer.

Access layer - connects end devices to network.

Distribution layer - divide network into subnets like sales, HR - acts as bridge b/w access and core layer.

Core layer - High speed backbone (network) - fast transfer data quickly and securely

Pros : High security and efficient traffic management

cons : complex and expensive (mainly used by large organizations)

Transmission Media - Physical or wireless medium through which data is transmitted in a network

(types)

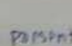
Guided media (wired) - high speed, secure, used for shorter distance

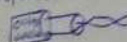
- twisted pair cable - two wires twisted to reduce interference, used in lans
- coaxial cable - High bandwidth, used in cable tv and internet connection
- optical fiber - using fibers that transmit data as light signals at very high speed and long distance

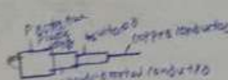
Unguided media (wireless) - less secure, used for longer distance, the signal is broadcasted through air

- Radiowaves - used in mobile communication, broadcasting
- microwaves - Point-to-point communication, satellite links
- Infrared - Short range, remote controls, wireless peripherals

Type of Twisted Pair

1. Unshielded Twisted Pair – It has 4 pair of copper wires that are present inside a plastic sheath. These wires are twisted to protect them from interference.
Usage – Telephone, LAN Network (Ethernet), DSL broadband
Advantage – Less expensive, easy to install, high speed capacity 
disadvantage – Shorter distance (signal loss due to attenuation), lower capacity than STP, more affected by external interference

2. Shielded Twisted Pair – twisted pair wires + extra shielding (copper braid/foil layer) for protection
Usage – fast Ethernet, environments with high EMI (factories, airports)
Advantage – faster, higher data rate in comparison to UTP
disadvantage – more expensive, bulky, difficult to install and handle 

Coaxial cable – It has central copper conductor, surrounded by an insulating layer, conducting shield & outermost plastic sheath 

Type of mode
1) Baseband mode (dedicated cable bandwidth) (Ex-LAN)
2) Broadband mode (cable bandwidth is split into separate ranges) (Ex-cable TV)
Advantage – Support multiple channel, less affected by EM interference, easy to install
disadvantage – expensive, must be grounded to prevent crosstalk. T-joint attack by breaking coaxial cable by hackers
Usage :- LAN, cable TV

Optical fiber cable – uses light pulses (Total internal reflection) to carry data.
Types :-
Single-mode (long distance, high speed, very thin core (8-10 µm)), Ex-ISPs)
multi-mode (short distance, thicker core (50-100 µm), Ex-LAN)
Usage :- Internet backbone, aerospace, well suited for voice, data and video transmission.
Advantage – light weight, high speed, long distance
disadvantage – high cost, difficult to install and maintain

Radio waves. (3KHz–300GHz) – can travel long distances, pass through buildings
No antenna alignment needed.
(used in analog communication systems, information like audio in radio waves)
Ex- AM | FM Radio, walkie-talkies
(Amplitude modulation, frequency modulation)

Microwaves (300MHz – 300GHz) – line of sight communication → Antennas must face each other
– used for point-to-point communication
Ex – mobile phone, satellite TV

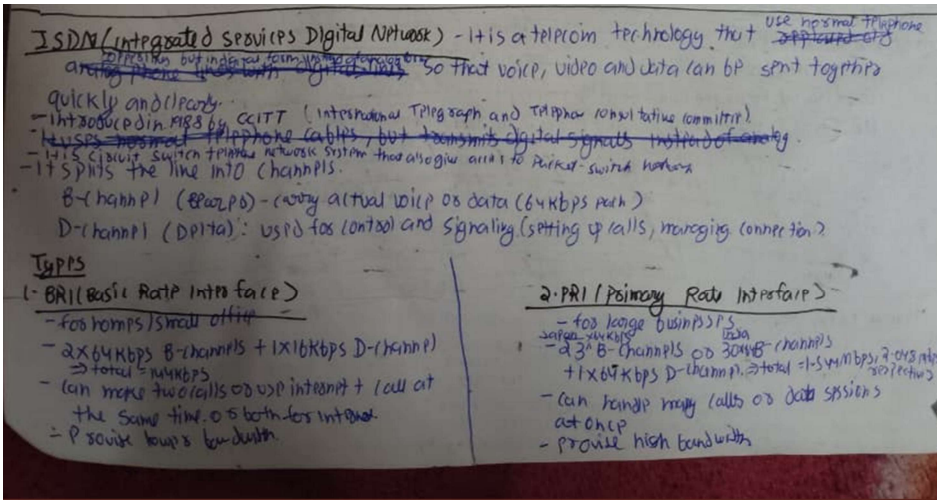
Infrared (400THz – 300GHz) – very short range, can't penetrate walls.
safe, no interference with nearby systems.
Ex – TV remote, wireless mouse/keyboard

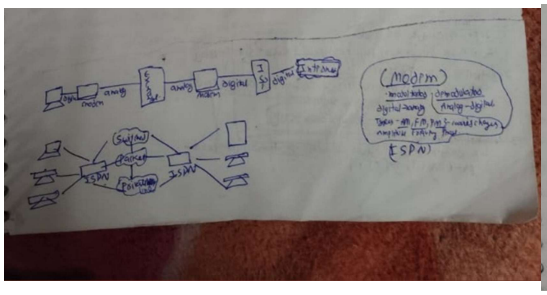
... to large messages.

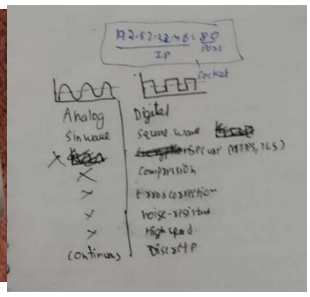
Terminal Handling – It is the process of managing the interaction b/w a user's terminal (I/O device) and the system.
It ensures that the data entered by the user is correctly understood by the system and that the output is displayed in a way the terminal can handle.

Main function

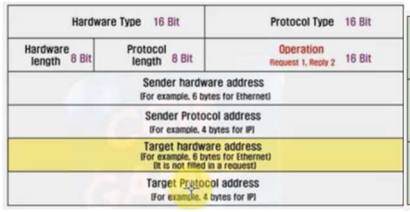
1) Terminal Identification – detect terminal type (Ex VT100) and its capabilities
2) Login session Management – display login prompts, Authenticate user and load user profile/environment
3) Input processing – Read commands or data from the terminal and handle special keys like backspace, ctrl+C
4) Output Processing – format text according to terminal capabilities, Handle cursor positioning, colors, and special characters
5) Session monitoring & control – Manage user session activity, log idle timeouts or disruption
6) Session Termination – close the session cleanly and release resources and return to login prompt

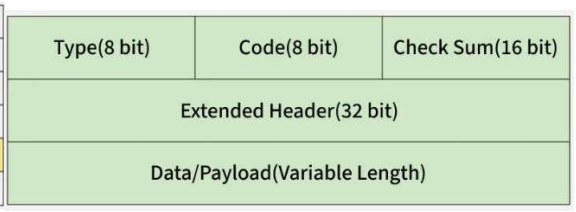




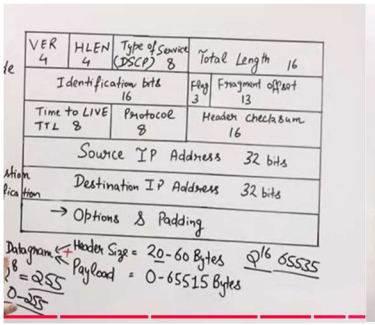


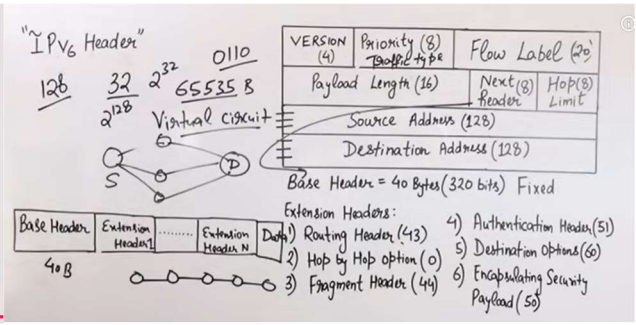
Arp header **Icmp Header**



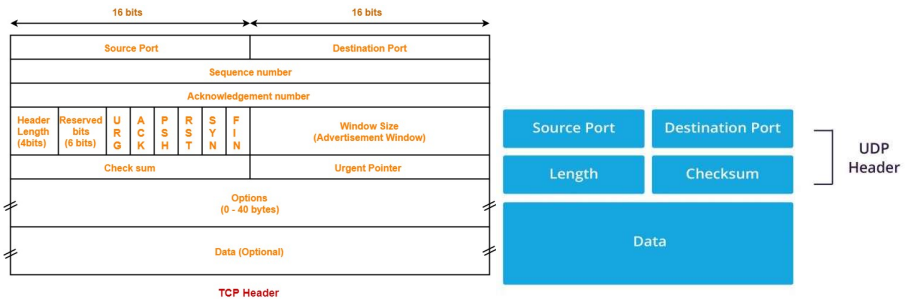


Ip4 header ipv6



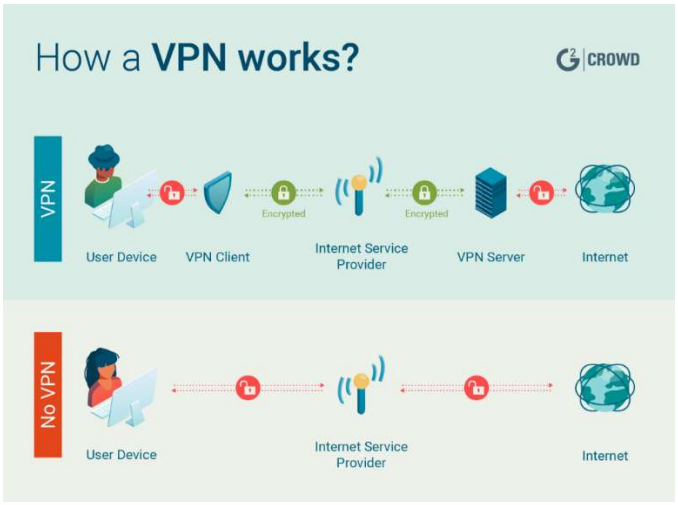


TCP header



VPN (Virtual Private Network)

A **Virtual Private Network (VPN)** is a private network built over the public internet that creates a **secure, encrypted tunnel** between networks or users. It allows users to **safely access an organization's internal network remotely**, as if they were physically present in the office.



Advantages of VPN

- Enables remote connection between offices in different geographical locations at a **lower cost than traditional WANs**.
- Provides **secure transfer of confidential data** between multiple locations.
- Protects organizational data from **unauthorized access and cyber threats**.
- **Encrypts internet traffic** and hides the user's online identity for better privacy.

Types of VPN

1. **Access VPN (Remote Access VPN)**
    - Used by **remote employees, mobile users, and telecommuters**.
    - Serves as a cost-effective alternative to dial-up or ISDN connections.
    - Allows users to securely connect to the organization's internal network from anywhere.

2. **Site-to-Site VPN (Router-to-Router VPN)**
    - Used by large organizations to **connect multiple office networks** across different locations.
    - Has two sub-types:
        - **Intranet VPN**: Connects remote offices of the same organization using shared internet infrastructure, functioning like a private WAN.
        - **Extranet VPN**: Extends limited network access to **partners, suppliers, or customers** while maintaining security through dedicated connections.

DNS (Domain Name System) – Important

- Stands for **Domain Name System**
- Introduced in **1983** by **Paul Mockapetris and Jon Postel**
- A **naming system** for internet resources (websites, servers, applications)
- Maps **domain names to IP addresses**
- Eliminates the need to remember numeric IP addresses

Example:
www.shaurya.com → 192.168.x.x

Working of DNS

1. User enters a URL like **https://www.shaurya.com** in the browser
2. Browser sends a DNS query
3. DNS translates the **domain name into an IP address**
4. Browser uses the IP address to locate and load the website

---

DNS Forwarder

- Used when a DNS server **cannot resolve a query locally**
- Forwards the request to **external DNS servers** (e.g., ISP DNS, Google DNS)
- Improves **resolution speed and efficiency**
- A DNS server with forwarder behaves differently from one without it

---

Ipconfig and Ifconfig

Ipconfig

- Stands for **Internet Protocol Configuration**
- Command used in **Microsoft Windows operating systems**
- Used to **view and configure network interface details**
- Displays information such as:
    - IP address
    - Subnet mask
    - Default gateway

Ifconfig

- Stands for **Interface Configuration**
- Command used in **Linux, macOS, and UNIX operating systems**
- Used to **view and configure network interfaces**
- Displays details like:
    - IP address
    - MAC address
    - Network status

Protocol

- A **protocol** is a set of rules used to govern all aspects of **data communication**.

Main Elements of a Protocol

- **Syntax**
    - Specifies the **structure or format** of the data
    - Defines the **order** in which data is presented
- **Semantics**
    - Specifies the **meaning** of each section of bits

    - Explains how data should be **interpreted**
- **Timing**
    - Specifies **when** data should be sent
    - Specifies **how fast** data can be sent

MAC Address and IP Address (Important)

- Both **MAC (Media Access Control) address** and **IP address** are used to **uniquely identify a device** on a network.
- The **MAC address** is assigned by the **NIC manufacturer**.
- The **IP address** is assigned by the **Internet Service Provider (ISP)** or network administrator.

Difference between MAC Address and IP Address

- **MAC address**:
    - Refers to the **physical address** of a device
    - Uniquely identifies a device on a **local network**
    - Is **permanent** and hardware-based
- **IP address**:
    - Refers to the **logical address**
    - Identifies a device's **network connection**
    - Can **change** based on the network

1. What happens when you enter google.com in a web browser? (Most Important)

Steps:

- The browser first checks its **cache**. If fresh content is available, it is displayed.
- If not, the browser checks whether the **IP address** of the URL is available in the **browser or OS cache**.
- If the IP is not found, the browser requests the **OS to perform a DNS lookup** using **UDP** to get the IP address from the DNS server.
- A **new TCP connection** is established between the browser and the server using **three-way handshake**.
- The browser sends an **HTTP request** to the server over the TCP connection.
- The web server processes the request and sends back an **HTTP response**.

- The browser receives the response and may **close or reuse the TCP connection**.
- If the response is cacheable, the browser **stores it in cache**.
- Finally, the browser **decodes and renders** the web page.

---

3. Subnet

- A subnet is a **network inside a network**
- Created using **subnetting**
- Improves:
    - Routing efficiency
    - Network security
- Reduces the time required to extract host addresses from the routing table

---

4. Network Reliability Factors

- **Downtime**: Time required to recover after failure
- **Failure Frequency**: How often the network fails
- **Catastrophe**: Unexpected events like fire or earthquake affecting the network

---

5. Criteria for an Effective Network

- **Performance**: Measured by transmit time and response time
- **Reliability**: Frequency of failures
- **Robustness**: Strength and ability to handle stress
- **Security**: Protection from unauthorized access and viruses

---

6. Node and Link

- A network is a connection of two or more computers
- The **physical medium** (fiber, coaxial cable, etc.) is called a **link**
- The connected computers are called **nodes**

---

7. Gateway and Router

- A device connected to two or more networks is called a **gateway**
- It is also known as a **router**
- Used to forward data from one network to another
- Both regulate network traffic

Difference:

- **Router**: Connects similar networks
- **Gateway**: Connects dissimilar networks

---

8. NIC (Network Interface Card) – Important

- NIC stands for **Network Interface Card**
- A hardware component used to connect a computer to a network
- Each NIC has a **unique MAC address**
- Provides wired or wireless LAN connectivity
- Commonly used in desktop computers

---

10. Private and Public IP Address

Private IP Address

- Reserved IP ranges not valid on the internet
- Used in internal networks
- Requires **NAT or proxy server** to access the internet

Public IP Address

- Assigned by the **Internet Service Provider (ISP)**
- Used for communication over the internet

---

11. RAID

- RAID stands for **Redundant Array of Inexpensive/Independent Disks**
- Used to provide **fault tolerance**
- Uses multiple hard disks

---

12. Netstat

- A **command-line utility**
- Displays information about:
    - Active connections
    - TCP/IP configuration

---

13. Ping

- A network utility used to check **connectivity**
- Can ping devices using:
    - IP address
    - Host name

---

14. Peer-to-Peer Processes

- Processes on different machines communicating at the **same layer**
- Called **peer-to-peer (P2P) processes**

---

15. Types of Transmission

- **Unicasting**: One-to-one communication
- **Anycasting**: One-to-any (used in content delivery systems)
- **Multicasting**: One-to-many (subset of nodes)
- **Broadcasting**: One-to-all
    - Used by **DHCP and ARP** in local networks