Networking

Networking Basics

1. Definition of Computer Networks

A computer network is a collection of two or more interconnected devices (computers, servers, switches, routers, etc.) that communicate and share resources with each other.
These resources can be files, applications, printers, or internet access.
Communication in networks happens through transmission media such as cables, radio waves, or fiber optics.

2. Types of Networks

Networks are classified based on their coverage area and purpose:

1. **LAN (Local Area Network)**
    - Covers a small geographical area (home, office, campus).
    - High speed, low latency.
    - Example: Wi-Fi in your home.
2. **MAN (Metropolitan Area Network)**
    - Covers a city or a large campus.
    - Larger than LAN but smaller than WAN.
    - Example: Cable TV networks in a city.
3. **WAN (Wide Area Network)**
    - Covers large geographical areas (country, continent, or world).
    - Internet is the best example.
4. **WLAN (Wireless Local Area Network)**
    - A LAN that uses wireless technology (Wi-Fi) instead of cables.
5. **PAN (Personal Area Network)**
    - Very small range, usually within a few meters.
    - Example: Bluetooth devices, hotspot connection between phone and laptop.
6. **SAN (Storage Area Network)**
    - A high-speed network that provides access to storage devices.
    - Mainly used in data centers for fast storage and retrieval.
7. **CAN (Campus Area Network)**
    - Connects multiple LANs within a limited area like a university, company campus, or military base.

3. Network Topologies

The arrangement of nodes and connections in a network is called topology.

1. **Bus Topology**
    - All devices are connected to a single central cable (backbone).
    - Simple but if the backbone fails, the entire network goes down.
2. **Star Topology**
    - All devices connect to a central device (hub or switch).
    - Easy to manage, but if the central device fails, the whole network is affected.
3. **Ring Topology**
    - Devices are connected in a circular fashion.
    - Data travels in one direction until it reaches the destination.
    - A single failure can disrupt the whole network.
4. **Mesh Topology**
    - Each device is connected to every other device.
    - Highly reliable and fault-tolerant, but expensive to implement.
5. **Hybrid Topology**
    - Combination of two or more topologies (e.g., star + bus).
    - Used in large organizations for flexibility and scalability.

4. Networking Devices

1. **Hub**
    - Simple device that broadcasts data to all devices in a network.
    - Works at the Physical Layer of OSI.
    - Not secure and inefficient.
2. **Switch**
    - Smarter than a hub; forwards data only to the intended device using MAC addresses.
    - Works at the Data Link Layer.
    - Improves efficiency.
3. **Router**
    - Connects different networks and forwards data based on IP addresses.
    - Works at the Network Layer.
    - Used for internet connectivity.
4. **Bridge**
    - Connects two LAN segments and filters traffic based on MAC address.
    - Works at the Data Link Layer.
5. **Repeater**
    - Boosts and regenerates signals to extend network distance.
    - Works at the Physical Layer.
6. **Gateway**
    - Acts as a translator between different network protocols.
    - Works at multiple OSI layers.
7. **Access Point (AP)**

- Provides wireless connectivity by connecting wireless devices to a wired LAN.

8. **Modem (Modulator-Demodulator)**
    - Converts digital signals (computer) into analog signals (telephone line) and vice versa.
    - Used for internet connectivity over telephone lines.

5. Transmission Modes

Defines how data flows between sender and receiver.

1. **Simplex**
    - One-way communication only.
    - Example: Keyboard to computer.
2. **Half-duplex**
    - Two-way communication, but only one device can send data at a time.
    - Example: Walkie-talkies.
3. **Full-duplex**
    - Two-way communication, both devices can send data simultaneously.
    - Example: Telephone call.

6. Network Models

1. **OSI Model (Open Systems Interconnection)**
    - Standard model with 7 layers:
        1. Physical
        2. Data Link
        3. Network
        4. Transport
        5. Session
        6. Presentation
        7. Application
    - Helps understand how data flows in a network.
    - Each layer has specific functions.
2. **TCP/IP Model**
    - Practical model used on the internet.
    - Has 4 layers:
        1. Network Interface (Link Layer)
        2. Internet Layer
        3. Transport Layer
        4. Application Layer
    - Simpler than OSI and widely implemented.

7. Data Transmission Concepts

1. **Bandwidth**
    - Maximum amount of data that can be transmitted per second.
    - Measured in bits per second (bps).
2. **Latency**
    - Time taken for a data packet to travel from source to destination.
    - Lower latency means faster response.
3. **Throughput**
    - Actual amount of data successfully transmitted per second.
    - It is always less than or equal to bandwidth.
4. **Jitter**
    - Variation in packet arrival time.
    - High jitter affects voice and video calls.
5. **Packet**
    - A unit of data transmitted over a network.
    - Contains header (control info) and payload (actual data).
6. **Frame**
    - A packet at the Data Link Layer.
    - Contains MAC addresses, error-checking codes, and data.

Physical Layer Concepts

The **Physical Layer** is the **first layer of the OSI model**.
It deals with the **transmission of raw bits (0s and 1s)** over a physical medium such as cables, radio waves, or fiber optics.
It defines hardware elements like cables, connectors, voltages, and data rates.

1. Guided Media (Wired Transmission)

In guided media, signals travel through a physical path.

1. **Twisted Pair Cable**
    - Made of pairs of insulated copper wires twisted around each other.
    - Twisting reduces electromagnetic interference.
    - Types:
        - **UTP (Unshielded Twisted Pair):** Cheaper, less protection.
        - **STP (Shielded Twisted Pair):** Has shielding to reduce interference.
    - Used in LANs and telephone lines.
2. **Coaxial Cable**
    - Has a central copper conductor, surrounded by insulation, a metallic shield, and an outer cover.
    - Better noise resistance than twisted pair.

- Used in cable TV, broadband internet.

3. **Fiber Optic Cable**
   - Transmits data as light signals through glass or plastic fibers.
   - Extremely high bandwidth and very low signal loss.
   - Immune to electromagnetic interference.
   - Used in long-distance communication and high-speed networks.

2. Unguided Media (Wireless Transmission)

In unguided media, signals are transmitted through the air without physical cables.

1. **Radio Waves**
   - Frequency range: 3 kHz to 1 GHz.
   - Can travel long distances and penetrate buildings.
   - Used in FM radio, broadcasting, cordless phones, Wi-Fi.
2. **Microwaves**
   - Frequency range: 1 GHz to 300 GHz.
   - Require line-of-sight (cannot bend around obstacles).
   - Used in satellite communication, cellular phones, Wi-Fi, radar.
3. **Infrared**
   - Frequency range: 300 GHz to 400 THz.
   - Short-range communication, cannot penetrate walls.
   - Used in TV remote controls, short-range data transfer.
4. **Satellite Communication**
   - Uses microwaves for uplink (Earth to satellite) and downlink (satellite to Earth).
   - Provides global coverage.
   - Used in GPS, TV broadcasting, internet services.

3. Encoding & Modulation

To transmit digital data over physical media, it must be encoded or modulated.

1. **ASK (Amplitude Shift Keying)**
   - Digital data represented by changes in amplitude of the carrier wave.
   - Simple but sensitive to noise.
2. **FSK (Frequency Shift Keying)**
   - Digital data represented by changes in frequency of the carrier wave.
   - Used in modems, low-speed communication.
3. **PSK (Phase Shift Keying)**
   - Digital data represented by changes in phase of the carrier wave.
   - More noise-resistant than ASK and FSK.
4. **QAM (Quadrature Amplitude Modulation)**
   - Combines both amplitude and phase changes.
   - Provides high data rates.
   - Widely used in modern communication (e.g., 4G, 5G).

4. Multiplexing

Multiplexing allows multiple signals to share the same communication channel.

1. **FDM (Frequency Division Multiplexing)**
    - Different signals are transmitted on different frequency bands simultaneously.
    - Used in radio and TV broadcasting.
2. **TDM (Time Division Multiplexing)**
    - Each signal gets a fixed time slot on the same channel.
    - Used in digital telephony.
3. **WDM (Wavelength Division Multiplexing)**
    - A form of FDM used in fiber optics.
    - Multiple light signals (different wavelengths/colors) are transmitted through the same fiber.
    - Used in high-capacity internet backbones.
4. **CDM (Code Division Multiplexing)**
    - Each signal is assigned a unique code and all signals are transmitted simultaneously over the same frequency band.
    - Receiver separates signals using the codes.
    - Used in mobile networks (CDMA).

5. Switching

Switching determines how data is transferred across a network.

1. **Circuit Switching**
    - A dedicated communication path is established between sender and receiver before data transfer.
    - Example: Traditional telephone networks.
    - Good for continuous communication, but wastes bandwidth if no data is being sent.
2. **Packet Switching**
    - Data is divided into packets and each packet may take different routes to reach the destination.
    - Example: Internet communication.
    - Efficient use of bandwidth, but packets may arrive out of order.
3. **Message Switching**
    - Entire message is sent as a single unit and stored temporarily at intermediate devices before being forwarded.
    - Example: Old telegraph networks.
    - Not suitable for real-time communication due to delays.

Data Link Layer

The **Data Link Layer** is the **second layer of the OSI model**.
It is responsible for **node-to-node communication**, ensuring that data is transferred **error-free and in the right order** over a physical link.
It takes raw bits from the physical layer and organizes them into **frames**.

1. Framing

- **Framing** is the process of dividing a data stream into manageable units called **frames**.
- A frame typically contains:
    - **Header:** Source and destination MAC addresses, control info.
    - **Payload:** Actual data.
    - **Trailer:** Error-checking bits.
- Purpose: To make data transmission manageable and detect errors.

2. Error Detection & Correction

Errors can occur due to noise, interference, or faulty transmission.

1. **Parity Bit**
    - Adds an extra bit to data to make the total number of 1s either even (even parity) or odd (odd parity).
    - Simple, but cannot detect all errors.
2. **CRC (Cyclic Redundancy Check)**
    - A polynomial-based method to detect errors.
    - Very reliable and widely used in networks like Ethernet.
3. **Hamming Code**
    - Provides **both error detection and correction**.
    - Can correct single-bit errors and detect two-bit errors.

3. Flow Control

Ensures that the **sender does not overwhelm the receiver** with too much data.

1. **Stop-and-Wait Protocol**
    - Sender sends one frame, waits for acknowledgment before sending the next.
    - Simple but inefficient.
2. **Sliding Window Protocol**
    - Allows multiple frames to be sent before needing an acknowledgment.
    - Window size defines how many frames can be sent in advance.
    - **Go-Back-N ARQ:**

        - Sender can send multiple frames, but if one frame is lost or corrupted, all subsequent frames are resent.
    - **Selective Repeat ARQ:**
        - Only the erroneous or lost frame is resent, not the entire sequence.
        - More efficient than Go-Back-N.

4. Media Access Control (MAC) Protocols

When multiple devices share the same communication channel, MAC protocols decide **who gets to send data and when**.

1. **ALOHA**
    - Devices transmit whenever they have data.
    - If collision occurs, they wait random time and retransmit.
    - Simple but inefficient.
2. **CSMA (Carrier Sense Multiple Access)**
    - Devices listen to the channel before transmitting.
    - If channel is busy, they wait.
3. **CSMA/CD (Collision Detection)**
    - Used in wired Ethernet.
    - If collision occurs, transmission stops and retries after a random delay.
4. **CSMA/CA (Collision Avoidance)**
    - Used in wireless networks (Wi-Fi).
    - Instead of detecting collisions, devices try to avoid them by waiting before transmitting.

5. Ethernet (IEEE 802.3)

- A widely used **wired LAN standard**.
- Uses **CSMA/CD** for medium access.
- Frame structure includes MAC addresses, type/length field, data, and CRC.
- Speeds range from 10 Mbps (Ethernet) to 100 Gbps+ (Gigabit and beyond).

6. Wi-Fi (IEEE 802.11)

- A **wireless LAN standard**.
- Uses **CSMA/CA** to avoid collisions.
- Supports different frequency bands (2.4 GHz, 5 GHz, 6 GHz).
- Standards: 802.11a/b/g/n/ac/ax (Wi-Fi 6).
- Provides mobility and flexibility but has higher interference compared to wired Ethernet.

7. VLANs (Virtual LANs)

- VLANs logically divide a physical network into multiple separate networks.
- Devices in different VLANs cannot directly communicate without a router.
- Benefits:
    - Better security.
    - Reduced broadcast traffic.
    - Easier network management.

8. STP (Spanning Tree Protocol)

- Prevents **loops in Ethernet networks**.
- Automatically disables redundant paths while keeping backup paths available.
- Ensures a loop-free topology.
- Essential in networks with multiple switches.

9. Protocols at the Data Link Layer

1. **PPP (Point-to-Point Protocol)**
    - Used for direct connections between two nodes (e.g., dial-up, DSL).
    - Provides authentication (PAP, CHAP) and error detection.
2. **HDLC (High-Level Data Link Control)**
    - A bit-oriented protocol for point-to-point and multipoint links.
    - Provides error detection, flow control, and framing.
3. **Frame Relay**
    - A WAN protocol for efficient data transmission.
    - Works at the Data Link Layer.
    - Faster and more efficient than older methods but now largely replaced by MPLS and modern technologies.

Network Layer

The **Network Layer** is the **third layer of the OSI model**.
It is responsible for:

- **Logical addressing** (IP addresses)
- **Routing** (choosing the best path for data packets)
- **Packet forwarding** (moving packets from source to destination across multiple networks)

1. IPv4 & IPv6 Addressing

IPv4 (Internet Protocol Version 4)

- Uses **32-bit address** (e.g., 192.168.1.1).
- Supports about **4.3 billion unique addresses**.
- Written in **dotted decimal notation** (four numbers separated by dots, each 0–255).

Classes of IPv4

- **Class A:** 1.0.0.0 – 126.255.255.255 (Large networks)
- **Class B:** 128.0.0.0 – 191.255.255.255 (Medium networks)
- **Class C:** 192.0.0.0 – 223.255.255.255 (Small networks)
- **Class D:** 224.0.0.0 – 239.255.255.255 (Multicast)
- **Class E:** 240.0.0.0 – 255.255.255.255 (Experimental)

Subnetting

- Divides a network into smaller sub-networks.
- Helps in efficient IP address usage and better security.

Supernetting

- Combines multiple smaller networks into one larger network.
- Used mainly by ISPs to manage routing tables.

CIDR (Classless Inter-Domain Routing)

- Replaces class-based addressing.
- Uses prefix notation (e.g., 192.168.1.0/24).
- /24 means 24 bits are for network ID, remaining 8 bits for host ID.

VLSM (Variable Length Subnet Masking)

- Allows subnetting with different subnet masks within the same network.
- Helps in efficient IP address utilization.

IPv6 (Internet Protocol Version 6)

- Uses **128-bit address** (e.g., 2001:0db8:85a3::8a2e:0370:7334).
- Provides **virtually unlimited addresses**.
- Written in **hexadecimal notation** separated by colons.
- Advantages over IPv4:
    - Larger address space.
    - Built-in security (IPSec).

- Simplified header.
- Auto-configuration support.

2. IP Address Types

1. **Public IP**
   - Globally unique, used on the internet.
   - Assigned by ISPs.
2. **Private IP**
   - Used inside local networks.
   - Not routable on the internet.
   - Ranges:
     - Class A: 10.0.0.0 – 10.255.255.255
     - Class B: 172.16.0.0 – 172.31.255.255
     - Class C: 192.168.0.0 – 192.168.255.255
3. **Static IP**
   - Manually assigned, fixed address.
   - Suitable for servers, printers, routers.
4. **Dynamic IP**
   - Assigned automatically by DHCP.
   - Changes over time.
5. **APIPA (Automatic Private IP Addressing)**
   - Range: 169.254.0.1 – 169.254.255.254.
   - Automatically assigned if DHCP fails.
   - Allows local communication but not internet access.

3. Routing Concepts

- **Routing** = Process of finding the best path for data packets.

1. **Routing Table**
   - A database stored in routers containing paths to different networks.
   - Contains: destination network, next hop, metric, interface.
2. **Forwarding**
   - Actual process of sending a packet to the next hop based on the routing table.

4. Routing Protocols

1. **RIP (Routing Information Protocol)**
   - Distance-vector protocol.
   - Uses hop count as metric.
   - Max hops = 15.
   - Simple but not suitable for large networks.

2. **OSPF (Open Shortest Path First)**
    - Link-state protocol.
    - Uses Dijkstra's algorithm to calculate shortest path.
    - Supports large enterprise networks.
3. **EIGRP (Enhanced Interior Gateway Routing Protocol)**
    - Hybrid (distance-vector + link-state).
    - Cisco proprietary protocol.
    - Fast convergence and efficient.
4. **BGP (Border Gateway Protocol)**
    - Path-vector protocol.
    - Used for routing between ISPs (on the internet).
    - Ensures global connectivity.
5. **IS-IS (Intermediate System to Intermediate System)**
    - Link-state protocol similar to OSPF.
    - Used in very large service provider networks.

5. Important Protocols

1. **ARP (Address Resolution Protocol)**
    - Maps an IP address to a MAC address in a local network.
2. **RARP (Reverse ARP)**
    - Maps a MAC address to an IP address.
    - Rarely used today, replaced by DHCP.
3. **ICMP (Internet Control Message Protocol)**
    - Used for error reporting and diagnostics.
    - Example: Ping, Traceroute.
4. **IGMP (Internet Group Management Protocol)**
    - Used for managing multicast groups.
    - Example: IPTV, streaming services.

6. NAT (Network Address Translation)

- Technique to map **private IP addresses** to **public IP addresses**.
- Types:
    - **Static NAT:** One-to-one mapping.
    - **Dynamic NAT:** Many-to-many mapping (from a pool).
    - **PAT (Port Address Translation):** Many-to-one mapping (used in home routers).
- Benefits: Conserves public IP addresses, provides security.

7. MPLS (Multiprotocol Label Switching)

- High-performance data forwarding technology.
- Instead of routing based on IP, it uses **labels**.

- Faster and more efficient than traditional IP routing.
- Used in enterprise WANs, telecom networks.
- Supports QoS (Quality of Service) and VPNs.

Transport Layer

The **Transport Layer** is the **4th layer of the OSI model** (and the **host-to-host layer in TCP/IP model**).
It ensures **end-to-end communication** between applications running on different devices.

1. TCP vs UDP

TCP (Transmission Control Protocol)

- **Connection-oriented** → establishes a reliable session before sending data.
- **Reliable** → ensures delivery with acknowledgments (ACKs).
- **Sequenced** → data is reassembled in the correct order.
- **Error detection & correction** → via checksums and retransmissions.
- **Slower but reliable**.
- Example applications: Web browsing (HTTP/HTTPS), Email (SMTP, IMAP), File Transfer (FTP).

UDP (User Datagram Protocol)

- **Connectionless** → no handshake, sends data directly.
- **Unreliable** → no guarantee of delivery or order.
- **Lightweight & faster**.
- **No retransmission** or flow control.
- Example applications: Streaming, VoIP, DNS, Gaming.

Key difference:
TCP = reliability & accuracy, UDP = speed & efficiency.

2. Ports & Sockets

- **Port**: Logical endpoint of communication, helps identify specific applications.
  - **Well-known ports (0–1023)**: HTTP (80), HTTPS (443), FTP (21), SSH (22), DNS (53).
  - **Registered ports (1024–49151)**: Used by applications.
  - **Dynamic/Ephemeral ports (49152–65535)**: Temporary ports for client connections.
- **Socket** = **IP Address + Port Number** → uniquely identifies a communication channel.

Example:
192.168.1.5:443 → Device at IP 192.168.1.5 using HTTPS port.

3. Connection Establishment & Termination

TCP 3-Way Handshake (Establishment)

1. **SYN** → Client sends SYN to initiate connection.
2. **SYN-ACK** → Server acknowledges with SYN-ACK.
3. **ACK** → Client sends ACK, connection established.

TCP Connection Termination (4 steps)

1. Client sends **FIN**.
2. Server sends **ACK**.
3. Server sends **FIN**.
4. Client sends **ACK** → connection closed.

4. Flow Control & Congestion Control

Flow Control

- Ensures sender does not overwhelm receiver.
- **Stop-and-Wait** → Sender waits for ACK after each packet.
- **Sliding Window** → Multiple packets can be sent before requiring ACKs.

Congestion Control (in TCP)

- Avoids network overload.
- Techniques:
    - **AIMD (Additive Increase Multiplicative Decrease)** → Slowly increases sending rate, reduces sharply on congestion.
    - **Slow Start** → Gradually increases window size until congestion occurs.
    - **Fast Retransmit** → Retransmits packet before timeout if multiple duplicate ACKs are received.
    - **Fast Recovery** → Skips slow start after fast retransmit, resuming with a moderate window size.

5. Reliability & Sequencing

- **Sequencing** → Each segment has a sequence number so data arrives in correct order.
- **Acknowledgments (ACKs)** → Confirm receipt of packets.
- **Retransmission** → If ACK is not received, TCP resends the data.
- **Error detection** → Checksum used to detect corrupted packets.

Application Layer

The **Application Layer** is the **top layer of both the OSI and TCP/IP models**.
It provides **network services directly to end-users and applications**, enabling communication between software running on different devices.

1. DNS (Domain Name System)

- Translates **domain names** (like www.google.com) into **IP addresses** (like 142.250.183.78).
- Works as the "phonebook of the internet."
- Uses **UDP port 53** (fast lookups) and sometimes **TCP port 53**.
- Components:
    - **DNS Resolver**: Client that queries DNS.
    - **Root Servers**: Top-level servers that direct queries.
    - **TLD Servers**: Handle domains like .com, .org.
    - **Authoritative Servers**: Store actual domain-IP mappings.

2. DHCP (Dynamic Host Configuration Protocol)

- Automatically assigns **IP addresses, subnet mask, default gateway, and DNS** to devices.
- Uses **UDP ports 67 (server) and 68 (client)**.
- Process (DORA):
    1. **Discover** – Client broadcasts request.
    2. **Offer** – Server offers an IP.
    3. **Request** – Client requests chosen IP.
    4. **Acknowledge** – Server confirms assignment.

3. HTTP & HTTPS

- **HTTP (Hypertext Transfer Protocol)**: Protocol for web browsing.
    - Port **80** (insecure).
- **HTTPS (HTTP Secure)**: Encrypted HTTP using **SSL/TLS**.
    - Port **443** (secure).
- Stateless protocol → every request is independent.

4. FTP & TFTP

- **FTP (File Transfer Protocol)**
    - Transfers files between client and server.
    - Port **21** (control), **20** (data).
    - Supports authentication (username/password).
- **TFTP (Trivial File Transfer Protocol)**

    - Lightweight file transfer protocol.
    - Uses **UDP port 69**.
    - No authentication, mostly for bootstrapping devices (routers, switches).

5. Email Protocols

- **SMTP (Simple Mail Transfer Protocol)**
    - Used to **send** emails.
    - Port **25** (default), **587** (secure).
- **POP3 (Post Office Protocol v3)**
    - Used to **download emails** from server to client.
    - Port **110** (default), **995** (secure/SSL).
    - Emails usually removed from server after download.
- **IMAP (Internet Message Access Protocol)**
    - Used to **access emails** while keeping them on the server.
    - Port **143** (default), **993** (secure/SSL).
    - Supports multiple devices accessing the same mailbox.

6. Telnet & SSH

- **Telnet**
    - Remote login protocol.
    - Port **23**.
    - Sends data in plain text (insecure).
- **SSH (Secure Shell)**
    - Secure remote login with encryption.
    - Port **22**.
    - Widely used for server administration.

7. SNMP (Simple Network Management Protocol)

- Used for **monitoring and managing network devices** (routers, switches, servers, printers).
- Ports: **UDP 161** (general messages), **UDP 162** (traps/alerts).
- Versions: SNMPv1, SNMPv2, SNMPv3 (most secure with authentication & encryption).
- Components:
    - **SNMP Manager** – software monitoring system.
    - **SNMP Agent** – runs on the device being managed.
    - **MIB (Management Information Base)** – database of managed objects.

8. NTP (Network Time Protocol)

- Synchronizes **clocks of computers** over a network.

- Port **123 (UDP)**.
- Ensures logs, transactions, and security protocols run with correct time.
- Very accurate (millisecond-level precision).

Network Security

Network Security ensures **confidentiality, integrity, and availability (CIA triad)** of data as it travels across networks.

1. Firewalls, IDS, IPS

- **Firewall**
  - Filters traffic between internal and external networks.
  - Can block/allow traffic based on rules (IP, port, protocol).
  - Types:
    - Packet-filtering firewall
    - Stateful inspection firewall
    - Application-level firewall (proxy firewall)
    - Next-Generation Firewall (NGFW)
- **IDS (Intrusion Detection System)**
  - Monitors network traffic and detects suspicious activity.
  - Only **alerts**, does not block traffic.
  - Types:
    - Network-based IDS (NIDS)
    - Host-based IDS (HIDS)
- **IPS (Intrusion Prevention System)**
  - Detects and also **blocks malicious traffic** in real time.
  - Often integrated with firewalls.

2. VPN (Virtual Private Network)

- Securely connects remote users to private networks over the Internet.
- Uses encryption to ensure privacy and confidentiality.
- Types:
  - **Remote Access VPN** – for individuals working remotely.
  - **Site-to-Site VPN** – connects entire branch offices.
- Protocols: PPTP, L2TP, IPSec, OpenVPN.

3. Encryption (Symmetric & Asymmetric)

- **Symmetric Encryption**
  - Same key used for encryption & decryption.

    - Fast, but key distribution is difficult.
    - Example: AES, DES, 3DES.
- **Asymmetric Encryption**
    - Uses a **public key** (encryption) and **private key** (decryption).
    - Slower but more secure.
    - Example: RSA, ECC.

4. SSL/TLS & IPSec

- **SSL/TLS (Secure Sockets Layer / Transport Layer Security)**
    - Provides secure communication for web traffic (HTTPS).
    - Ensures data encryption, integrity, and authentication.
    - Widely used in browsers, email, VoIP.
- **IPSec (Internet Protocol Security)**
    - Secures IP communication at the **network layer**.
    - Provides encryption, authentication, and integrity.
    - Used in VPNs.

5. Authentication Protocols

- **RADIUS (Remote Authentication Dial-In User Service)**
    - Centralized authentication, authorization, and accounting (AAA).
    - Uses **UDP ports 1812/1813**.
    - Common in ISPs, corporate networks, Wi-Fi.
- **TACACS+ (Terminal Access Controller Access Control System Plus)**
    - Cisco proprietary protocol for AAA.
    - Uses **TCP port 49**.
    - More secure than RADIUS; separates authentication, authorization, and accounting.

6. Common Network Attacks

- **DoS (Denial of Service)**
    - Attacker floods a system with traffic to make it unavailable.
- **DDoS (Distributed Denial of Service)**
    - Same as DoS but from multiple compromised devices (botnet).
- **Man-in-the-Middle (MITM)**
    - Attacker intercepts communication between two parties.
- **ARP Spoofing**
    - Attacker sends fake ARP messages to link their MAC address with another device's IP.
- **DNS Spoofing**
    - Attacker provides false DNS responses, redirecting traffic to malicious sites.

- **Phishing**
    - Social engineering attack where fake emails/websites trick users into giving sensitive info.

7. Security Best Practices

- Use **firewalls, IDS/IPS, and VPNs**.
- Enable **strong encryption (TLS, IPSec, AES)**.
- Apply **patches and updates** regularly.
- Use **multi-factor authentication (MFA)**.
- Follow the **principle of least privilege**.
- Implement **network segmentation** (VLANs, DMZ).
- Train users to detect **phishing and social engineering attacks**.
- Regularly **monitor and audit logs**.

Wireless & Mobile Networking

Wireless and mobile networking allows devices to communicate **without physical cables**, using radio waves, microwaves, or infrared signals. It includes **Wi-Fi, Bluetooth, cellular networks, and mobile IP**.

1. Wi-Fi Standards (IEEE 802.11 Family)

Wi-Fi is defined by the **IEEE 802.11** standards. Different versions improve **speed, frequency, and range**:

- **802.11a** – 5 GHz, up to 54 Mbps.
- **802.11b** – 2.4 GHz, up to 11 Mbps, long range but interference-prone.
- **802.11g** – 2.4 GHz, up to 54 Mbps, backward-compatible with 802.11b.
- **802.11n** – 2.4/5 GHz, up to 600 Mbps, introduced MIMO (multiple antennas).
- **802.11ac (Wi-Fi 5)** – 5 GHz, up to several Gbps, wider channels, MU-MIMO.
- **802.11ax (Wi-Fi 6)** – 2.4/5 GHz, higher efficiency, better performance in crowded areas, supports OFDMA.

2. Short-Range Wireless Technologies

- **Bluetooth**
    - Short-range (~10 m), low-power communication.
    - Used in headsets, keyboards, file transfer, IoT.
    - Versions 4.0+ support **Bluetooth Low Energy (BLE)**.
- **ZigBee**
    - Low-power, low-data-rate wireless standard.

    - Designed for IoT, smart homes, industrial automation.
    - Operates in 2.4 GHz band.
- **NFC (Near Field Communication)**
    - Very short range (~4 cm).
    - Used in contactless payments (Google Pay, Apple Pay), access cards.

3. Cellular Networks (Mobile Generations)

- **2G** (Second Generation)
    - Digital voice communication.
    - GSM, CDMA technologies.
    - Speeds: ~64 Kbps.
- **3G** (Third Generation)
    - Supports voice + mobile data (internet).
    - UMTS, HSPA.
    - Speeds: up to 2 Mbps.
- **4G (LTE)**
    - High-speed mobile internet.
    - Supports HD video streaming, VoIP.
    - Speeds: 100 Mbps – 1 Gbps.
- **5G**
    - Latest mobile network.
    - Very high speeds (up to 10 Gbps).
    - Ultra-low latency (~1 ms).
    - Supports IoT, autonomous vehicles, AR/VR.

4. Mobile IP

- Protocol that allows a mobile device to **move between networks** while keeping the same **IP address**.
- Components:
    - **Home Agent (HA)** – tracks the mobile device's permanent IP.
    - **Foreign Agent (FA)** – provides care-of-address when device roams.
    - **Care-of-Address (CoA)** – temporary IP while roaming.
- Ensures uninterrupted communication for mobile devices.

5. Wireless Security Protocols

- **WEP (Wired Equivalent Privacy)**
    - Old, weak encryption.
    - Easily breakable → not secure.
- **WPA (Wi-Fi Protected Access)**
    - Improved over WEP, but still vulnerable.

- **WPA2**
  - Uses **AES encryption**, much stronger.
  - Widely used today.
- **WPA3**
  - Latest standard.
  - Provides stronger encryption and better protection against brute-force attacks.
  - Uses **SAE (Simultaneous Authentication of Equals)**.

Advanced Networking Concepts

Advanced networking introduces **modern approaches and technologies** to handle large-scale, flexible, and intelligent networks used in **cloud, data centers, IoT, and automation**.

1. SDN (Software-Defined Networking)

- **Definition**: A networking approach that separates the **control plane** (decision-making) from the **data plane** (packet forwarding).
- **Traditional networking**: Each device (router/switch) has its own control plane.
- **SDN approach**: Centralized **SDN Controller** manages all devices.
- **Advantages**:
  - Centralized control
  - Easier network management and automation
  - Programmable networks
- **Protocols used**: OpenFlow, NETCONF.

2. NFV (Network Functions Virtualization)

- **Definition**: Replacing dedicated hardware (like firewalls, load balancers, routers) with **virtualized software functions** running on standard servers.
- **Example**: Instead of a hardware firewall, you deploy a **virtual firewall (vFW)**.
- **Advantages**:
  - Cost savings (less hardware)
  - Scalability and flexibility
  - Faster deployment of services

3. Cloud Networking Basics

- **AWS VPC (Virtual Private Cloud)**
  - A logically isolated network within AWS.
  - Contains subnets, route tables, internet gateways, NAT gateways.
  - Used to securely deploy cloud applications.
- **Azure VNets (Virtual Networks)**

    - Equivalent of AWS VPC in Microsoft Azure.
    - Supports peering, VPN connections, and integration with on-premises networks.
- **Key concept**: Cloud providers give users control over **virtual networks** similar to on-premises networking.

4. Data Center Networking

- **Spine-Leaf Architecture**
    - **Spine switches** (core layer) connect to **Leaf switches** (access layer).
    - Every Leaf connects to every Spine → high scalability & low latency.
    - Replaces traditional 3-tier (core, distribution, access).
- **VXLAN (Virtual Extensible LAN)**
    - Overlays Ethernet frames on UDP.
    - Used to extend Layer 2 networks over Layer 3 infrastructure.
    - Helps in large-scale data centers and cloud environments.
- **EVPN (Ethernet VPN)**
    - Uses BGP for Layer 2 and Layer 3 connectivity.
    - Works with VXLAN for scalable data center interconnect.

5. Content Delivery Networks (CDNs)

- **Definition**: A globally distributed network of servers that deliver web content (like videos, images, files) closer to users.
- **Examples**: Akamai, Cloudflare, Amazon CloudFront.
- **Benefits**:
    - Faster load times
    - Reduced latency
    - High availability and redundancy
    - Handles sudden traffic spikes

6. IoT Networking Basics

- **IoT (Internet of Things)**: Devices like sensors, smart appliances, and industrial machines connected to the internet.
- **Protocols used**:
    - **MQTT** – lightweight publish/subscribe protocol.
    - **CoAP** – optimized for constrained devices.
    - **ZigBee, LoRaWAN** – for low-power wide-area IoT.
- **Challenges**: Security, scalability, low power consumption, interoperability.

7. Network Automation (Python, Ansible)

- **Network Automation**: Using scripts and tools to configure and manage networks without manual effort.
- **Python**
  - Popular for writing automation scripts.
  - Libraries: Netmiko, Paramiko, NAPALM.
  - Example: Automating router configuration, log collection.
- **Ansible**
  - Open-source automation tool.
  - Uses YAML playbooks to define tasks.
  - Agentless (uses SSH).
  - Widely used for network device configuration and cloud networking automation.

Key Points:

- **SDN & NFV** → modernize networks with software-driven control and virtualization.
- **Cloud Networking** → AWS VPC, Azure VNets for virtual networks.
- **Data Center Networking** → Spine-Leaf, VXLAN, EVPN for scalability.
- **CDNs** → optimize content delivery.
- **IoT Networking** → connects billions of devices.
- **Automation** → Python & Ansible make networking programmable and efficient.

Must-Know Networking Protocols

Protocols are **rules that define how data is transmitted and received** in a network. These are the most important ones that every student, job seeker, and professional must know.

1. ARP (Address Resolution Protocol)

- Maps **IP addresses → MAC addresses**.
- Works within a local network.
- Example: Device with IP 192.168.1.5 asks *"Who has this IP?"* → ARP reply contains MAC address.

2. IP (Internet Protocol)

- Provides **logical addressing and routing** of packets.
- Versions:
  - **IPv4** – 32-bit addressing, ~4.3 billion addresses.
  - **IPv6** – 128-bit addressing, huge address space.
- Connectionless and unreliable (relies on higher layers for reliability).

3. ICMP (Internet Control Message Protocol)

- Used for **error reporting and diagnostics**.
- Example tools: **ping, traceroute**.
- Messages: Echo request/reply, destination unreachable, time exceeded.

4. IGMP (Internet Group Management Protocol)

- Manages **multicast group memberships**.
- Allows devices to join or leave multicast groups (used in streaming, conferencing).

5. TCP (Transmission Control Protocol)

- **Connection-oriented** protocol.
- Provides **reliability, sequencing, error detection, and flow control**.
- Used by applications where accuracy is important (HTTP, FTP, Email).

6. UDP (User Datagram Protocol)

- **Connectionless, lightweight** protocol.
- **No reliability or sequencing**, but very fast.
- Used in real-time applications (VoIP, streaming, gaming, DNS).

7. Application Layer Protocols

- **HTTP/HTTPS** → Web browsing (HTTPS = secure with TLS).
- **FTP (File Transfer Protocol)** → File transfer, ports 20/21.
- **SMTP (Simple Mail Transfer Protocol)** → Sending emails, port 25/587.
- **POP3 (Post Office Protocol v3)** → Downloading emails, port 110.
- **IMAP (Internet Message Access Protocol)** → Access emails on server, port 143.
- **DNS (Domain Name System)** → Resolves domain names to IPs, port 53.
- **DHCP (Dynamic Host Configuration Protocol)** → Automatically assigns IPs, ports 67/68.
- **SNMP (Simple Network Management Protocol)** → Device monitoring & management, ports 161/162.
- **Telnet** → Remote login (insecure), port 23.
- **SSH (Secure Shell)** → Secure remote login, port 22.

8. Routing Protocols

- **RIP (Routing Information Protocol)**
    - Distance-vector protocol.
    - Uses hop count as metric (max 15 hops).
    - Simple but limited.

- **OSPF (Open Shortest Path First)**
  - Link-state protocol.
  - Uses Dijkstra's algorithm to find shortest path.
  - Scales better than RIP.
- **BGP (Border Gateway Protocol)**
  - Path-vector protocol.
  - Used between ISPs and on the internet (inter-domain routing).
  - Makes routing decisions based on policies.
- **EIGRP (Enhanced Interior Gateway Routing Protocol)**
  - Cisco proprietary (now open standard).
  - Hybrid protocol (mix of distance-vector & link-state).
  - Fast convergence.
- **IS-IS (Intermediate System to Intermediate System)**
  - Link-state protocol similar to OSPF.
  - Used in large service provider networks.

Key Points:
These protocols form the **foundation of networking knowledge**.

- ARP, IP, ICMP, IGMP → Core networking functions.
- TCP & UDP → Transport layer fundamentals.
- HTTP, FTP, SMTP, POP3, IMAP, DNS, DHCP, SNMP, Telnet, SSH → Key application protocols.
- RIP, OSPF, BGP, EIGRP, IS-IS → Essential routing protocols.

Practical & Tools

Networking is not only theory – **hands-on practice** with tools and commands is crucial for interviews and real-world jobs.

1. Simulation & Emulation Tools

- **Packet Tracer (Cisco)**
  - Beginner-friendly tool for learning networking.
  - Used for **basic router, switch, firewall configurations**.
  - Common in **CCNA training**.
- **Wireshark**
  - **Packet analyzer (sniffer)**.
  - Captures live network traffic for analysis.
  - Helps in troubleshooting (e.g., dropped packets, protocol behavior).
- **GNS3 (Graphical Network Simulator 3)**
  - Professional tool for **network emulation**.
  - Runs real router/switch IOS images.

    - Used for **CCNP, CCIE practice**.
- **NS3 (Network Simulator 3)**
    - Research-focused network simulator.
    - Used in **academic projects, IoT, wireless, 5G studies**.

2. Basic Networking Commands

- **ping** → Test connectivity between devices.
- **traceroute (Linux) / tracert (Windows)** → Shows the path packets take.
- **nslookup / dig** → DNS lookup (domain → IP).
- **ipconfig (Windows) / ifconfig (Linux)** → View or configure IP details.
- **netstat** → Displays active connections, ports, routing tables.

3. Router & Switch Configuration (Cisco IOS Basics)

- **User Modes**:
    - User EXEC (>) → Basic commands.
    - Privileged EXEC (#) → Advanced commands.
    - Global Config (config)# → Device-wide settings.
- **Common Commands**:
- enable → Enter privileged mode
- configure terminal → Enter global configuration
- hostname <name> → Set device hostname
- interface <type> <id> → Select interface (e.g., FastEthernet0/1)
- ip address <IP> <SM> → Assign IP to interface
- no shutdown → Enable the interface
- show ip interface brief → View interface status
- show running-config → View current config
- copy running-config startup-config → Save config
- **Routing Examples**:
    - router rip → Enable RIP protocol.
    - router ospf 1 → Enable OSPF.
    - network <IP> <wildcard> → Define networks.

4. Subnetting Practice & IP Address Planning

- **Subnetting**: Splitting a network into smaller sub-networks for efficient IP usage.
- **CIDR (Classless Inter-Domain Routing)** → Uses prefix length (e.g., /24, /28).
- **Example**:
    - Network: 192.168.1.0/24
    - Subnet mask: 255.255.255.0
    - Can create 4 subnets with /26:
        - 192.168.1.0 – 63
        - 192.168.1.64 – 127
        - 192.168.1.128 – 191

        - 192.168.1.192 – 255
- **Planning Guidelines**:
    - Allocate subnets per department/VLAN.
    - Leave buffer IPs for growth.
    - Use private IP ranges:
        - 10.0.0.0/8
        - 172.16.0.0/12
        - 192.168.0.0/16

Key Points:

- Tools: **Packet Tracer, Wireshark, GNS3, NS3**.
- Commands: **ping, traceroute, nslookup, ipconfig/ifconfig, netstat**.
- Cisco Basics: **Modes, show commands, IP config, routing**.
- Subnetting: **Essential for IP planning, efficient utilization**.

Interview-Oriented Topics

This section focuses on the **most frequently asked networking questions** in interviews.

1. OSI vs TCP/IP Model

- **OSI Model (7 Layers)** → Physical, Data Link, Network, Transport, Session, Presentation, Application.
- **TCP/IP Model (4 Layers)** → Network Interface, Internet, Transport, Application.

Key Differences:

- OSI is **theoretical**, TCP/IP is **practical & widely used**.
- OSI has 7 layers, TCP/IP has 4 layers.
- Protocols like **HTTP, FTP, SMTP** belong to Application Layer in both.
- OSI separates **Session & Presentation**, but TCP/IP combines them into Application.

2. Hub vs Switch vs Router

- **Hub**: Broadcasts data to all ports (Layer 1, no intelligence).
- **Switch**: Forwards data based on **MAC addresses** (Layer 2, efficient).
- **Router**: Forwards packets based on **IP addresses** (Layer 3, connects networks).

3. TCP vs UDP (with examples)

- **TCP (Transmission Control Protocol)**
    - Connection-oriented, reliable, error-checking, sequencing.

    - Used in: **HTTP/HTTPS, FTP, SMTP, Telnet**.
- **UDP (User Datagram Protocol)**
    - Connectionless, faster, no guarantee of delivery.
    - Used in: **DNS, DHCP, VoIP, video streaming, online gaming**.

4. Subnetting & Supernetting Problems

- **Subnetting** → Breaking a large network into smaller networks.
    - Example: 192.168.1.0/24 → Subnet into 4 parts of /26.
- **Supernetting (CIDR Aggregation)** → Combining multiple subnets into a larger block.
    - Example: 192.168.0.0/24 + 192.168.1.0/24 → 192.168.0.0/23.

Both are frequently tested in **interview whiteboard problems**.

5. IPv4 vs IPv6

IPv4:

- 32-bit address
- ~4.3 billion unique addresses
- Written in **dotted decimal** (e.g., 192.168.0.1)
- Requires **NAT** due to address shortage
- Security is optional (IPSec support is limited)

IPv6:

- 128-bit address
- ~3.4 × 10^38 addresses
- Written in **hexadecimal colon notation** (e.g., 2001:db8::1)
- No NAT needed, larger address pool
- **IPSec built-in** for better security

6. How DNS Works?

1. User enters a domain name (e.g., www.example.com).
2. Browser checks **local DNS cache**.
3. If not found → query sent to **Recursive DNS Resolver**.
4. Resolver queries **Root Server → TLD Server → Authoritative DNS Server**.
5. IP address returned to user.
6. Browser connects to server using that IP.

7. What happens when you type a URL in a browser?

1. **DNS resolution** → Domain name → IP address.
2. **TCP connection** → 3-way handshake with server.
3. **TLS/SSL handshake** (if HTTPS).
4. **HTTP request sent** (GET / POST).
5. **Server responds** with HTML, CSS, JS, media.

6. **Browser renders** the page.

8. Explain 3-way Handshake (TCP Connection Establishment)

1. Client → SYN → Server.
2. Server → SYN + ACK → Client.
3. Client → ACK → Server.

Connection is established after this.

9. Static vs Dynamic Routing

- **Static Routing**:
  - Routes manually configured.
  - Best for small/simple networks.
  - Low overhead, but not scalable.
- **Dynamic Routing**:
  - Uses routing protocols (RIP, OSPF, BGP, EIGRP).
  - Automatically updates routes if topology changes.
  - Scalable, but higher overhead.

10. Firewall vs IDS vs IPS

- **Firewall**: Filters traffic based on rules (ports, IPs, protocols).
- **IDS (Intrusion Detection System)**: Detects suspicious activity, alerts admin.
- **IPS (Intrusion Prevention System)**: Detects + blocks malicious traffic.

11. Load Balancing Concepts

- **Distributes network traffic** across multiple servers to improve performance & availability.
- Algorithms: **Round Robin, Least Connections, IP Hashing**.
- Ensures **fault tolerance, scalability, high availability**.
- Common tools: **HAProxy, Nginx, AWS ELB**.

12. Troubleshooting Approach in Networking

1. **Identify the problem** (e.g., user can't access internet).
2. **Check physical connections** (cables, Wi-Fi).
3. **Check IP configuration** (ipconfig / ifconfig).
4. **Ping test** (local gateway, then internet).
5. **Traceroute** (to see where packets drop).
6. **Check DNS** (nslookup).
7. **Check firewall/security rules**.
8. **Check device logs** (router/switch/server).