Introduction to Networking

🔹 1. What is Networking?

Networking is the practice of connecting computers, servers, and other devices to share data and resources such as files, internet, printers, etc.

🔹 2. Types of Networks

- **LAN (Local Area Network):** A small network within a home, office, or building.
- **WAN (Wide Area Network):** A large network spread over a wide geographical area like the Internet.
- **MAN (Metropolitan Area Network):** A network that spans a city or large campus.

🔹 3. Essential Network Devices

- **Router:** Connects different networks and routes data between them.
- **Switch:** Connects devices in a LAN and forwards data based on MAC addresses.
- **Modem:** Converts digital data to analog (and vice versa) for internet connectivity.
- **Access Point (AP):** Enables wireless devices to connect to a wired network.
- **Hub (outdated):** Broadcasts data to all connected devices regardless of destination.

Here are the most important cables used in networking, categorized based on their use and technology:

🔌 1. Ethernet Cables (Twisted Pair)

Used for connecting devices in LAN (Local Area Network)

▸ **Types:**

- **Cat5** (Category 5):
  - Speed: Up to 100 Mbps (Fast Ethernet)
  - Max Length: 100 meters
  - Mostly outdated
- **Cat5e** (Enhanced):
  - Speed: Up to 1 Gbps
  - Reduced crosstalk and better performance than Cat5
  - Still widely used
- **Cat6:**
  - Speed: Up to 10 Gbps (at shorter distances)
  - Better shielding and performance
  - Common in modern setups
- **Cat6a:**
  - Speed: 10 Gbps
  - Shielded, supports higher bandwidth and longer distances than Cat6
- **Cat7 / Cat8:**
  - For high-performance data centers
  - Supports 10–40 Gbps speeds
  - Expensive and less common in homes/offices

🔌 2. Fiber Optic Cables

Used for high-speed, long-distance connections (WANs, backbone networks)

▸ **Types:**

- **Single-mode Fiber (SMF):**
  - Long-distance transmission (up to 100 km or more)
  - Uses laser light
  - Thin core (~9 microns)
- **Multi-mode Fiber (MMF)::**
  - Shorter distance (up to 2 km)
  - Uses LED light
  - Thicker core (~50–62.5 microns)

🔌 3. Coaxial Cable

Used in older Ethernet standards (10Base2, 10Base5), cable TV, and broadband internet

- Has a single copper conductor at the center
- Shielded for protection against interference
- Rarely used in modern networks

🔌 4. Console Cable (Rollover Cable)

Used to connect a PC to a **Cisco router/switch console port** for CLI access

- Typically RJ45 on one end, DB9 or USB on the other
- Used for configuration and troubleshooting

🔌 5. Crossover Cable *(Legacy use)*

- Used to directly connect two computers or two switches without a router
- Now mostly replaced by **Auto-MDIX** in modern switches (auto detect)

🧠 Tip:

Always choose the cable based on **speed requirements**, **distance**, and **interference**. For typical home or office LANs, **Cat6** or **Cat6a** is a future-proof choice.

🔹 4. OSI Model

The **OSI model** (Open Systems Interconnection) is a **7-layer framework** that standardizes how different networking systems communicate with each other.

Even though it's not directly used in real-world internet communications (unlike the TCP/IP model), it's **extremely important** for understanding **networking concepts, troubleshooting, and design**.

🔷 1. Physical Layer (Layer 1)

This is the **lowest layer** of the OSI model.

What it does:

- Deals with the **physical medium** (cables, connectors, electrical signals).
- Transfers **raw bits (0s and 1s)** over the network.

Examples:

- Ethernet cables, fiber optics, hubs, network cards
- Voltages, pins, and cable types

🧠 Think: *“How do bits physically move from one place to another?”*

🔷 2. Data Link Layer (Layer 2)

What it does:

- Creates **frames** from raw bits.
- Adds **MAC addresses** to identify source and destination within a local network.
- Handles **error detection**, **collision detection**, and **flow control**.

Divided into two sublayers:

- **MAC (Media Access Control):** Controls how devices access the medium.
- **LLC (Logical Link Control):** Manages frame synchronization and error control.

Devices that work here: Switches, Bridges

🧠 Think: *“How does my PC talk to another PC on the same LAN?”*

🔷 3. Network Layer (Layer 3)

What it does:

- Manages **routing** and **IP addressing**.
- Determines the best path for data to travel across networks.

Key Functions:

- Logical addressing (IP address)
- Packet forwarding and routing

Devices that work here: Routers

🧠 Think: *“How does my data find the way from one city to another?”*

🔷 4. Transport Layer (Layer 4)

What it does:

- Ensures **complete, reliable data transfer**.
- Breaks large data into **segments** and reassembles them on the receiving side.
- Handles **error checking, retransmissions, and flow control**.

Protocols:

- **TCP** (reliable, connection-based)
- **UDP** (faster, connectionless)

🧠 Think: *“How do we ensure all the pieces of a file get to the other end correctly?”*

🔷 5. Session Layer (Layer 5)

What it does:

- Establishes, manages, and **terminates sessions** between two devices.
- Keeps track of **which data belongs to which session**, especially in multi-user or multi-tasking scenarios.

🧠 Think: *“How does my computer keep my email and YouTube session separate?”*

🔷 6. Presentation Layer (Layer 6)

What it does:

- Translates **data formats** so both the sender and receiver understand it.
- Handles **encryption, decryption, compression**, and **data encoding**.

Examples:

- Encoding formats like JPEG, MP4, PDF
- Encryption: SSL, TLS

🧠 Think: *“How do we turn raw data into readable, secure content?”*

🔷 7. Application Layer (Layer 7)

What it does:

- Closest to the **user**.
- Provides **network services** to applications (like browsers, email clients, FTP tools).
- This is where users **interact with the network**.

Examples of protocols:

- HTTP/HTTPS (web), SMTP/IMAP (email), FTP (file transfers), DNS

🧠 Think: *“How do I use the network to browse, email, or download?”*

🧠 Summary (Layer Mnemonic)

From bottom to top:
Please **D**o **N**ot **T**hrow **S**ausage **P**izza **A**way

➡ Physical, Data Link, Network, Transport, Session, Presentation, Application

---

The OSI model is a **conceptual tool**, but it's incredibly useful for:

✅ Troubleshooting network issues
✅ Understanding where things go wrong
✅ Explaining how data travels from one app to another across networks

---

🔹 4.1 TCP/IP Model

The **TCP/IP model** (Transmission Control Protocol/Internet Protocol) is a **4-layer architecture** that defines how data is sent and received across networks like the internet.

It’s more practical and widely used than the OSI model. Every device communicating over the internet uses this model.

🔹 1. Application Layer

This is the **topmost layer** of the TCP/IP model.

What it does:

- Provides services directly to users or applications (e.g., web browsers, email apps).
- It handles **everything the user sees and interacts with**.

Common Protocols:

- **HTTP/HTTPS** – for web browsing
- **FTP** – for file transfers
- **DNS** – for resolving domain names to IP addresses
- **SMTP/POP3/IMAP** – for sending/receiving emails
- **DHCP** – for assigning IP addresses

This layer **combines** the functions of the OSI model’s:

- Application Layer
- Presentation Layer
- Session Layer

🔹 2. Transport Layer

This layer is responsible for **reliable communication** between devices.

What it does:

- Ensures data is sent and received correctly.
- Breaks large data into small chunks and reassembles them.
- Manages **error detection, retransmission, and flow control**.

Key Protocols:

- **TCP (Transmission Control Protocol):** Reliable, connection-based (e.g., web, email)
- **UDP (User Datagram Protocol):** Fast, connectionless (e.g., video streaming, VoIP)

This layer is equivalent to the **Transport Layer in the OSI model**.

🔹 3. Internet Layer

This layer is all about **routing and addressing**.

What it does:

- Determines the **best path** for data to travel.
- Assigns and manages **IP addresses**.
- Ensures data packets reach the correct destination, even if they cross multiple networks.

Main Protocols:

- **IP (Internet Protocol):** Assigns and handles IP addresses.
- **ICMP:** Used for error reporting and diagnostics (used by ping).
- **ARP:** Maps IP addresses to MAC addresses within local networks.

This is the equivalent of the **Network Layer in the OSI model**.

🔹 4. Network Access Layer

(Also called the **Link Layer** or **Host-to-Network Layer**)

This is the **bottom layer** and closest to the physical hardware.

What it does:

- Controls how data is physically sent over cables or wireless.
- Defines how devices access the medium (Ethernet, Wi-Fi).
- Adds **MAC addresses** to frames and handles **framing, addressing, and error detection** at the hardware level.

Components include:

- Ethernet, Wi-Fi standards
- Network Interface Cards (NICs)
- MAC addressing
- Switches and hubs

This layer combines the **Physical** and **Data Link layers** of the OSI model.

🧠 Summary

The TCP/IP model consists of:

1. **Application Layer** – User interaction and services
2. **Transport Layer** – Reliable or fast data delivery
3. **Internet Layer** – Routing and IP addressing
4. **Network Access Layer** – Physical delivery of data

Each layer builds upon the one below it to ensure **successful and accurate data communication** across networks..

🔹 5. IP Addressing

- **IPv4 Example:** `192.168.1.1` (32-bit address)
- **IPv6 Example:** `2001:db8::1` (128-bit address)
- **Private IP Ranges:**
  - `192.168.0.0` to `192.168.255.255`
  - `10.0.0.0` to `10.255.255.255`
  - `172.16.0.0` to `172.31.255.255`

🔹 6. Subnetting (Basics)

Subnetting is the process of dividing a large network into smaller, manageable sub-networks.

Example:

- Network: `192.168.1.0/24` gives 256 IPs
- Subnet: `192.168.1.0/25` gives 128 IPs

🔹 7. MAC Address vs. IP Address

- **MAC Address:**
  - Physical address burned into network card
  - Used in LAN communication
  - Example: `00:1A:2B:3C:4D:5E`
- **IP Address:**
  - Logical address assigned by network
  - Used for routing over the internet
  - Example: `192.168.1.100`

🔹 8. DNS – Domain Name System

DNS converts human-readable domain names (like `google.com`) into IP addresses. It's like the phonebook of the internet, helping devices find each other using names instead of numbers.

🔹 9. Common Networking Protocols

- **HTTP/HTTPS:** For web browsing.
- **FTP:** Used to transfer files over a network.
- **SMTP/IMAP/POP3:** Email protocols for sending and receiving emails.
- **DHCP:** Automatically assigns IP addresses to devices.
- **DNS:** Resolves domain names to IP addresses.
- **TCP:** Ensures reliable data transfer with acknowledgment.
- **UDP:** Fast, connectionless protocol, used in streaming and VoIP.

🔹 10. Basic Troubleshooting Tools

- **Ping:** Check if a device is reachable.
- **Traceroute:** See the path data takes to reach a destination.
- **ipconfig (Windows) / ifconfig (Linux):** View IP configuration.
- **nslookup:** Test DNS resolution.

🔹 11. Basic Network Security

- Use **firewalls** to block unauthorized access.
- Keep operating systems and firmware **updated**.
- Implement **VLANs** to segment sensitive devices.
- Use strong **Wi-Fi encryption** (WPA2/WPA3).
- **Disable unused ports** and services.

🖥️ Important Networking Commands

These commands help test, troubleshoot, and diagnose network issues:

✅ Windows/Linux Commands:

- **`ping [IP/hostname]`**
  ➢ Tests connectivity to another host
  ➢ Example: `ping google.com`

- **`tracert`** / **`traceroute`**
  ➢ Traces the path packets take to reach the destination
  ➢ Windows: `tracert 8.8.8.8`
  ➢ Linux: `traceroute 8.8.8.8`

- **`ipconfig`** / **`ifconfig`** / **`ip a`**
  ➢ Displays IP address and network info
  ➢ Windows: `ipconfig`
  ➢ Linux: `ip a` or `ifconfig`

- **`nslookup [domain]`**
  ➢ Queries DNS to find IP of a domain
  ➢ Example: `nslookup facebook.com`

- **`netstat`**
  ➢ Shows active connections and listening ports
  ➢ Example: `netstat -an`

- **`arp -a`**
  ➢ Displays ARP cache (IP-to-MAC address mappings)

- `hostname`
  - ➤ Displays the system’s hostname

- `telnet [IP] [port]`
  - ➤ Tests if a port on a remote host is open
  - ➤ Example: `telnet 192.168.1.1 80`

- `pathping` **(Windows)**
  - ➤ Combines `ping` and `tracert` with detailed hop info

🌐 Important Networking Protocols

These are standard rules used for communication across networks:

- **TCP (Transmission Control Protocol)** – Reliable, ordered, connection-based communication
- **UDP (User Datagram Protocol)** – Fast, connectionless communication
- **IP (Internet Protocol)** – Handles addressing and routing
- **HTTP/HTTPS** – Web communication
- **FTP** – File Transfer
- **DNS** – Domain name resolution
- **DHCP** – Dynamic IP address assignment
- **ICMP** – Used by `ping` and `tracert` for diagnostic messages
- **SNMP** – Used for network monitoring and management
- **SSH** – Secure remote login
- **Telnet** – Unsecured remote login (rarely used today)
- **NAT (Network Address Translation)** – Maps private to public IPs
- **SMTP / IMAP / POP3** – Email sending and receiving

🔢 Common Port Numbers to Remember

Here are some **well-known port numbers** you **must** know for CCNA and troubleshooting:

Port	**Protocol / Service**	**Description**
20	FTP (Data)	File Transfer Protocol (Data)
21	FTP (Control)	File Transfer Protocol (Control)
22	SSH	Secure Shell (remote login)
23	Telnet	Remote command-line access
25	SMTP	Sends emails
53	DNS	Resolves domain names
67	DHCP (Server)	Assigns IP addresses

68	DHCP (Client)	Receives IP configuration
69	TFTP	Simple file transfer (UDP-based)
80	HTTP	Web browsing (insecure)
110	POP3	Receives emails (download)
123	NTP	Time synchronization
143	IMAP	Receives emails (server storage)
161	SNMP	Network monitoring
443	HTTPS	Secure web browsing
989/990	FTPS	Secure FTP transfer
3389	RDP	Remote Desktop Protocol

🧠 **Pro Tip:**

To remember a few of these ports easily:

- **80 = HTTP**, **443 = HTTPS**
- **21 = FTP**, **22 = SSH**, **23 = Telnet**
- **53 = DNS**, **67/68 = DHCP**

https://www.linkedin.com/in/gitesh-d-aa95a8227/