Computer Network 💻

✅ 1. What is a Computer Network?

A **Computer Network** is when two or more computers are connected to share data, resources, or communication.

> 📦 Real Life Example: WhatsApp and Instagram work because millions of devices (phones, servers) are connected via networks.

Benefits of Networking:

- File sharing
- Internet access
- Communication (chat, email)
- Resource sharing (printers, servers)

✅ 2. Types of Networks

Type	Full Form	Range	Example
LAN	Local Area Network	Within a building or room	Home Wi-Fi
MAN	Metropolitan Area Network	City-wide	College Campus Wi-Fi
WAN	Wide Area Network	Country or world-wide	The Internet
PAN	Personal Area Network	Very short range	Bluetooth, Hotspot

---

✅ 3. Network Topologies

Topology = How devices are connected

Topology	Description	Pros	Cons
Bus	All devices in a line	Easy setup	One failure affects all
Star	All connect to a central hub	Easy to manage	Hub failure affects all
Ring	Devices in a circle	Predictable	If one breaks, all fail
Mesh	All devices connected to all others	Very reliable	Expensive
Hybrid	Mix of all	Flexible	Complex

🌐 Day 1: OSI Model – The Backbone of Networking

🤔 What is the OSI Model?

OSI stands for **Open Systems Interconnection**.

It's a **conceptual framework** that shows how data travels from one computer to another over a network. It divides the process into **7 layers**, each with a specific role.

🧱 7 Layers of the OSI Model (Top to Bottom)

Layer	Layer Name	What It Does (Simple)	Example
7	Application	Interface for the user	WhatsApp, Gmail
6	Presentation	Converts data formats (encrypt, compress)	JPEG ↔ PNG, Encryption
5	Session	Manages sessions (start, end communication)	Logging in to a server
4	Transport	Ensures reliable delivery (TCP/UDP)	Splits data into chunks
3	Network	Routes data using IP addresses	Google.com → IP
2	Data Link	Transfers frames between two directly connected devices	MAC Address
1	Physical	Actual hardware and transmission (cables, signals)	Ethernet cable, Wi-Fi

---

📦 Real-Life Analogy: Sending a Parcel

Imagine sending a parcel:

1. **Application** – You write the letter.
2. **Presentation** – You translate it (e.g., Hindi to English).
3. **Session** – You call the post office to schedule pickup.
4. **Transport** – They pack and label it.
5. **Network** – They choose the best route (road, air).
6. **Data Link** – Truck loads it and drives it locally.
7. **Physical** – The wheels move the parcel physically.

💡 Remember Like This (Mnemonic):

> **"All People Seem To Need Data Processing"**
> (A = Application, P = Presentation, S = Session, T = Transport, N = Network, D = Data Link, P = Physical)

🌐 Daily CN Lesson – Day 2: TCP/IP Model

> 📦 The real-world version of the OSI model, used by the internet today.

📘 What is the TCP/IP Model?

TCP/IP stands for **Transmission Control Protocol / Internet Protocol**.

It is the **practical model** that the internet and most modern networks use to communicate. While the OSI model is a reference framework, **TCP/IP is the actual implementation**.

🧱 Layers of the TCP/IP Model (4 Layers Only!)

Layer	OSI Equivalent	Role (Simple)	Example
4. Application	OSI Layer 5–7	Provides services to user	HTTP, FTP, SMTP
3. Transport	OSI Layer 4	Ensures reliable or fast delivery	TCP, UDP
2. Internet	OSI Layer 3	Handles addressing & routing	IP, ICMP
1. Network Access	OSI Layer 1–2	Sends bits physically	Ethernet, Wi-Fi

---

🧠 Key Differences: OSI vs TCP/IP

Feature	OSI Model	TCP/IP Model
Layers	7	4
Use	Conceptual	Practical (used in real world)
Developed By	ISO	DARPA (U.S. Dept. of Defense)
Popularity	Learning/Exams	Real-world Networking

📦 Real-Life Analogy:

Let's say you want to send a message using WhatsApp:

1. **Application Layer** → You type the message.
2. **Transport Layer** → The message is broken into packets.
3. **Internet Layer** → The address of the destination is added.
4. **Network Access Layer** → Message physically sent via Wi-Fi.

📌 Common Protocols in TCP/IP

Layer	Protocols
Application	HTTP, HTTPS, FTP, SMTP, DNS
Transport	TCP, UDP
Internet	IP (IPv4, IPv6), ICMP
Network Access	Ethernet, Wi-Fi, ARP

🌐 Daily CN Lesson – Day 3: IP Addressing + Subnetting Basics

🧠 What is an IP Address?

An **IP (Internet Protocol) address** is a **unique identifier** for every device on a network — like a phone number for your computer.

It helps:

- **Identify** a device on the internet or network
- **Send and receive** data to/from the right place

📦 Types of IP Addresses

Type	Description	Example
IPv4	32-bit address, widely used	`192.168.1.1`
IPv6	128-bit address, newer, more space	`2001:0db8:85a3:0000:0000:8a2e:0370:7334`

📌 IPv4 Format

- 32 bits (divided into 4 sections or *octets*)
- Written in **dotted decimal** form: `192.168.0.1`
- Each section (octet) ranges from **0 to 255**

🔢 Example:
`11000000.10101000.00000000.00000001` = `192.168.0.1`

🏠 Public vs Private IP Addresses

Type	Used Where?	Example
Private	Inside home or office network	`192.168.x.x`, `10.x.x.x`, `172.16.x.x`
Public	Globally on the internet	Your mobile/ISP-assigned IP

🔒 Private IPs can't be accessed directly from outside the network.

🧮 What is Subnetting?

Subnetting = Breaking a large network into smaller **sub-networks (subnets)**.

Why?

- Helps in **efficient IP allocation**
- **Improves performance** and **security**
- Reduces **broadcast traffic**

🧠 Subnet Mask

A **subnet mask** defines which part of the IP address is the **network** and which part is the **host**.

Common example:

- IP: `192.168.1.10`
- Subnet Mask: `255.255.255.0`

That means:

- `192.168.1` = Network part
- `.10` = Host (device) part

🎯 CIDR Notation

Instead of writing the whole mask:

- `192.168.1.10/24` → Means **first 24 bits** are for the network.

🌐 Daily CN Lesson – Day 4: IP Address Classes + Default Subnets + Practice

---

🧠 Why IP Address Classes?

Back in early networking days, the internet needed a way to organize IP addresses for different **sizes of networks**. That's how **IP Classes** were introduced.

IPv4 addresses are divided into **5 classes** based on their **starting bits and range**.

📊 IP Address Classes (A to E)

Class	Starting Bit(s)	Range (1st Octet)	Default Subnet Mask	Use
A	0xxxxxxx	1 – 126	255.0.0.0 (/8)	Large networks
B	10xxxxxx	128 – 191	255.255.0.0 (/16)	Medium networks
C	110xxxxx	192 – 223	255.255.255.0 (/24)	Small networks
D	1110xxxx	224 – 239	N/A	Multicast
E	1111xxxx	240 – 255	N/A	Research only (Experimental)

> Note: `127.x.x.x` is reserved for **loopback (localhost)**

🎯 Default Subnet Masks

Class	Subnet Mask	CIDR Notation	Usable Hosts
A	255.0.0.0	`/8`	~16 million
B	255.255.0.0	`/16`	~65,000
C	255.255.255.0	`/24`	254

> Usable hosts = $2^n - 2$
> (`n` = number of host bits)

🌐 Day 5: Static vs Dynamic IP + DHCP + DNS + Real-World Workflow

---

🔁 1. Static vs Dynamic IP

Type	Description	Example Use Case
Static IP	Manually assigned IP that never changes	Servers, printers
Dynamic IP	Automatically assigned by DHCP	Home Wi-Fi, mobile data

---

🧠 Key Differences

Feature	Static IP	Dynamic IP
Assigned by	Admin manually	DHCP server automatically
Changes?	Never (unless manually changed)	Yes, can change any time
Cost	Usually paid	Free
Example	`192.168.1.10`	`192.168.1.100` (changes)

> ⚠️ If you're hosting a website or game server, **static IP** is better.

⚙️ 2. What is DHCP (Dynamic Host Configuration Protocol)?

DHCP automatically assigns:

- **IP address**
- **Subnet mask**
- **Gateway**
- **DNS**

> 🧑‍💻 Without DHCP, every device must be configured **manually**.

🧭 DHCP Workflow (Real World Example):

1. **Device joins network**
   (e.g., You connect your phone to Wi-Fi)
2. Device sends a **DHCPDISCOVER**
3. DHCP server replies with **DHCPOFFER**
4. Device sends **DHCPREQUEST**
5. Server confirms with **DHCPACK**

✅ Your phone now has an IP and can access the internet!

🌍 3. What is DNS (Domain Name System)?

DNS = Phonebook of the internet

It **converts domain names to IP addresses**.

> 📦 Example:
> You type `www.google.com` → DNS says: "Its IP is `142.250.195.36` " → Browser connects to that

> server

🔗 DNS Resolution Flow (Simplified)

1. You type `www.example.com`
2. Your device asks a **DNS Resolver**
3. Resolver checks:
    - Local Cache
    - Root Server → TLD Server → Authoritative Server
4. Gets IP address → sends it to your browser
5. Browser connects to server via IP

🔁 Real-Life Network Flow Summary

```
You → Connect to Wi-Fi
→ DHCP assigns IP, DNS
→ You open google.com
→ DNS translates domain to IP
→ Browser connects via IP
→ Server sends response
→ You see Google homepage
```

🌐 Day 6: Core Protocols (TCP, UDP, HTTP, FTP, ICMP, ARP) Explained with Real Examples

---

🚦 Why Protocols?

A **protocol** is a set of rules that decide **how data is transferred** over a network. Just like people need a common language to communicate, devices need protocols.

Type of protocols:

- TCP
- UDP
- HTTP
- FTP
- ICMP
- ARP

1️⃣ TCP (Transmission Control Protocol)

- 💬 TCP is **like a phone call** – both sides connect, talk in order, and confirm delivery.
- 🔄 It uses:
  - **Three-Way Handshake** (SYN → SYN-ACK → ACK)
  - **Sequencing** (arrange data chunks)
  - **Acknowledgement** (confirm delivery)
  - **Retransmission** (if data is lost)
- **Key Port**: Depends on the application (e.g., HTTP uses TCP port 80)

Feature	Description
Type	**Connection-oriented**
Reliability	Guarantees **data delivery, order, and error checking**
Use Cases	Web (HTTP), Email, File Transfers

🧠 **Real Example:**

> When you open Gmail in a browser, the page must load completely and correctly — TCP ensures that every bit of data arrives in the correct order.

2️⃣ UDP (User Datagram Protocol)

- 🚀 UDP is **like sending a letter** – no confirmation if it arrived.
- Fast, lightweight, but doesn't guarantee:
  - Delivery
  - Order
  - Duplicate protection
- **Key Port**: Varies (e.g., DNS uses UDP port 53)

Feature	Description
Type	**Connectionless**
Reliability	No guarantee of delivery or order
Use Cases	Video streaming, VoIP, Gaming

🧠 **Real Example:**

> When you watch a YouTube video or play PUBG, missing one data packet doesn't matter — speed is more important than reliability.

3️⃣ HTTP (HyperText Transfer Protocol)

- 📄 Protocol used to **send webpages, images, text, etc.**

- Based on **TCP**
- **Stateless**: Every request is independent
- **Secure version**: HTTPS (adds encryption via SSL/TLS)

Feature	Description
Used for	Browsing websites (text, images, HTML)
Port	80 (unencrypted), 443 (HTTPS – encrypted)
Protocol Type	Application Layer

🧠 **Real Example:**

> Typing `www.google.com` and seeing the homepage — HTTP handles this text/image transfer from server to your browser.

4️⃣ FTP (File Transfer Protocol)

- Used to **upload or download files** between client and server
- Two modes:
  - **Active mode** (client opens port, server connects back)
  - **Passive mode** (server opens a port, client connects)

Requires login: via username/password or anonymously

Ports: 20 (data) and 21 (command)

Feature	Description
Used for	Uploading/downloading files
Port	20, 21
Login	Requires username/password (sometimes anonymous)

🧠 **Real Example:**

> Developers uploading code to a server, or downloading data from a remote site — FTP is commonly used here.

5️⃣ ICMP – Internet Control Message Protocol

- 📢 Used for sending **network status messages**, **not data**
- Used in diagnostic tools:
  - `ping` → Tests reachability
  - `traceroute` → Shows path to a destination

Key Idea: ICMP reports issues (like “host unreachable” or “TTL expired”)

Feature	Description
Purpose	Sends **control/error messages** (e.g., unreachable host)
Used by	`ping`, `traceroute`
Type	Network Layer

🧠 **Real Example:**

> When you use the `ping` command to test if a server is up, ICMP sends the echo request and receives the reply.

6️⃣ ARP (Address Resolution Protocol)

- 🧭 Helps your system **find the MAC address** of another device using its IP.
- Used within **LANs** (Local Area Networks)
- Maintains an **ARP table** to store resolved MACs

Example:

> Laptop wants to send data to `192.168.0.20`
> → It checks ARP table
> → If not found, it broadcasts: "Who has 192.168.0.20?"
> → The device replies with its MAC address
> → Now the laptop can send Ethernet frame directly

Feature	Description
Purpose	Maps **IP address to MAC address**
Works at	Data Link Layer
Role	Helps local delivery of packets within LAN

🧠 **Real Example:**

> When your laptop sends a packet to your printer (on the same network), it uses ARP to find the printer's MAC address first.

🌐 Day 7: MAC Address, ARP Table, NAT & IP-to-IP Translation Explained

---

🧾 1. What is a MAC Address?

MAC = Media Access Control

- It's a **unique hardware address** assigned to every device's **network interface card (NIC)**
- MACs work at the **Data Link Layer (Layer 2)** of the OSI model
- Format: `6 pairs of hexadecimal` → e.g., `AA:BB:CC:DD:EE:FF`

🧠 **Real Analogy**:
If an IP address is like your home address, your MAC address is like the **door number inside a big apartment**.

📌 MAC addresses:

- Don't change
- Are used for communication inside a **local network (LAN)**

📜 2. What is an ARP Table?

ARP = Address Resolution Protocol

- ARP **maps IP addresses to MAC addresses** within the same network
- When a device wants to send data to an IP, it asks:

> "Who has this IP? Give me their MAC!"

- The **ARP table** stores this mapping temporarily

📁 **Example of an ARP Table:**

IP Address	MAC Address
192.168.0.5	AA:BB:CC:11:22:33
192.168.0.10	44:55:66:77:88:99

🛠️ You can see it in your system using:

- Windows: `arp -a`
- Linux/Mac: `ip neigh` or `arp`

🌐 3. What is NAT (Network Address Translation)?

NAT translates **private IPs to a public IP** so multiple devices can access the internet using **one IP**.

💡 Without NAT, every device at your home would need a public IP — which are limited and costly.

📘 **Why NAT?**

- Saves public IPs
- Adds a layer of security
- Enables **many-to-one** mapping (multiple devices to one IP)

📦 **Example:**

- Your laptop (IP: `192.168.0.5`) → Router
- Router NATs it to public IP: `103.21.55.200`
- Website sees: request from `103.21.55.200`, not your private IP

🔁 Types of NAT

Type	Description
SNAT	Source NAT – changes sender's IP (e.g., local to public)
DNAT	Destination NAT – changes destination IP (e.g., port forwarding)
PAT	Port Address Translation – many-to-one using ports

🔄 IP-to-IP Translation Flow (with ARP & NAT)

```
Device: 192.168.0.5
↓
ARP: Resolve MAC of 192.168.0.1 (router)
↓
Sends data → router
↓
Router NATs: 192.168.0.5 → 103.21.55.200
↓
Sends to destination server on Internet
```

✅ Summary

Concept	Works At	What It Does
MAC Address	Layer 2 (Data Link)	Uniquely identifies a device's NIC
ARP Table	Layer 2 ↔ 3	Maps IP to MAC inside LAN
NAT	Router Level	Converts private IP to public IP
IP Translation	Network Flow	Enables communication from LAN to Internet

---

🌐 Day 8 – Network Devices (Router, Switch, Hub, Bridge, Gateway)

🔌 Why Do We Need These Devices?

These devices act like **traffic managers** for networks.
Each one has a unique job to do — from **connecting devices**, **forwarding data**, to **linking entire networks or the internet**.

📦 1. Hub – "The Dumb Distributor"

🧠 What it is:

- Basic device that simply **broadcasts** data to **all** connected devices
- Doesn't know who should receive it — **no intelligence**

🔁 How it works:

- Sends incoming data to **every port**, even if it's not needed

📦 Analogy:

> Like shouting in a room hoping the right person hears you

❌ Downsides:

- Wastes bandwidth
- Not secure (everyone sees everything)
- Rarely used today

📦 2. Switch – "The Smart Distributor"

🧠 What it is:

- More intelligent than a hub
- Can **learn MAC addresses** and **send data only to the right device**

🔁 How it works:

- Has a **MAC address table**
- When data comes in, it checks the destination MAC and forwards it only to that port

📦 Analogy:

> Like calling someone by name in a room and giving only *them* the message

✅ Why it's better:

- Faster
- Secure

- Reduces network collisions

> 📌 **Used in LANs**, almost everywhere today

📦 3. **Bridge** – "The Divider & Connector"

🧠 What it is:

- Used to **connect two different LANs** or divide one into segments
- Works at the **Data Link Layer (Layer 2)**

🔁 How it works:

- Filters traffic based on MAC addresses
- Can **reduce collisions** and segment networks logically

📦 Analogy:

> Like a **bouncer** who checks if someone belongs on one side of the club or the other

---

📦 4. **Router** – "The Internet Connector"

🧠 What it is:

- Used to **connect different networks** (LAN to WAN, home to internet)
- Works at the **Network Layer (Layer 3)**
- **Forwards packets based on IP addresses**

🔁 How it works:

- Has a **routing table**
- Decides the **best path** to send data to a remote network

📦 Analogy:

> Like Google Maps — it checks the best route for your data to reach its destination

📱 Examples:

- Home Wi-Fi routers
- Internet gateways in companies

📦 5. **Gateway** – "The Protocol Converter"

🧠 What it is:

- Connects networks **using different protocols**
- Works from **Layer 4 to Layer 7**
- Translates data between **different systems**

📦 Analogy:

> Like a **translator** between two people speaking different languages

🌐 Use case:

- Connecting a LAN using TCP/IP to a network using a different protocol

---

🧾 Summary Table

Device	Layer	Intelligence	Main Job	Use Case
Hub	Physical (L1)	❌ None	Broadcasts to all	Old LANs
Switch	Data Link (L2)	✅ MAC-based	Sends to correct device	All LANs today
Bridge	Data Link (L2)	✅ Segments LAN	Joins/splits LANs	Reduces collision
Router	Network (L3)	✅ IP-based	Connects networks	LAN ↔ Internet
Gateway	L4–L7	✅ Protocol-based	Translates protocols	Enterprise/Hybrid networks

---