Foundations of Networking

~ By SAGAR BISWAS

PREFACE

Welcome to *Foundations of Networking*, your essential guide to understanding the core principles of computer networking. This guide is designed to break down complex concepts into easily digestible insights for anyone eager to explore how networks operate.

Whether you're a student new to networking, a professional seeking to deepen your understanding, or an enthusiast exploring the digital world's backbone, this resource will provide clarity and direction. Networking is the foundation of modern communication, enabling everything from social media to secure financial transactions.

With real-world examples, simplified analogies, and clear explanations, this guide aims to make networking concepts accessible, practical, and enjoyable.

♠ What is Foundations of Networking?

Networking forms the backbone of our interconnected world, enabling devices to communicate, share data, and perform tasks seamlessly. *Foundations of Networking* provides an in-depth exploration of topics like:

- ⊗ The significance of protocols (TCP, UDP) in reliable communication.
- ⊗ The importance of IP addresses, ports, and DNS in device identification and resource access.
- ⊗ Understanding the differences between IPv4 and IPv6, static vs. dynamic IPs, and public vs. private networks.
- ⊗ Exploring network models like OSI and TCP/IP for structured communication.
- ⊗ Examining Wi-Fi security, attacks, and the CIA triad (Confidentiality, Integrity, Availability).

This guide bridges theoretical knowledge and practical understanding, offering insights into how networks enable the flow of information that drives the digital era.

:- Chapter 1 -:

Covered Topics

1. What are Network Protocols?
2. Types of Protocols
3. How TCP works
4. TCP vs UDP

♠ Understanding (Ip Address, Ports, Protocols)

Example: Sending Messages Between Houses.

Imagine a neighbourhood where **House A** wants to send a message to **House B**. Here is how the process works:

A. IP Address (House Address)

Every house in the neighbourhood has a unique address, like:

- ⊗ **House A**: *XYZ/B A.G.B Colony Motijheel, Dhaka- 1000.*
- ⊗ **House B**: xy/g Kadamtala Basabo, Dhaka

Similarly, in a network, every device is identified by a unique **IP address** (for example, *192.168.1.10* for House A and *192.168.1.20* for House B). This ensures the message is sent to the correct destination.

B. Ports (Roads to Specific Houses)

The roads in the neighbourhood lead to different houses:

- ⊗ **Road 80 (Port 80):** The main road for delivering general web traffic (HTTP).
- ⊗ **Road 443 (Port 443):** A secure road used for encrypted communication (HTTPS).

When House A sends the message, it must specify which road (port) to use to reach the right destination at House B.

Ports: (How many ports do we have for communication in Networking?)

- **65,536 ports** (from 0 to 65,535) are available for communication in networking.
- Ports are divided into three ranges based on their usage and purpose.

Types of Ports:

1. Well-known Ports

- ⊗ **Range: 0 to 1,023**
- ⊗ These ports are reserved for system-level or well-established services.
- ⊗ **Examples:**
  - ¤ **HTTP**: **Port:** 80 **Uses:** Used for unencrypted web communication (e.g., websites without HTTPS).
  - ¤ **HTTPS**: **Port:** 443 **Uses:** Used for secure web communication (encrypted with SSL/TLS).

- **Other common ports:**
  - **FTP:** Port 21 (File Transfer Protocol).
  - **DNS:** Port 53 (Domain Name System).
  - **SMTP:** Port 25 (Simple Mail Transfer Protocol for email).

2. Registered Ports

- **Range: 1,024 to 49,151**
- These ports are assigned by IANA (Internet Assigned Numbers Authority) to specific applications or services upon request.
- **Examples:**
  - **Port 3306:** MySQL database.
  - **Port 3389:** Remote Desktop Protocol (RDP).
  - **Port 8080:** Alternate HTTP for web servers.

3. Dynamic/Private Ports

- **Range: 49,152 to 65,535**
- These ports are not assigned to specific services. Instead, they are used temporarily for **client-side communications** (e.g., when a web browser connects to a web server).
- **Example Scenario:**

A client device uses a dynamic port (e.g., Port 50,000) to communicate with a web server on Port 443.

Refined Example:

1. A user opens a web browser and accesses a secure website (e.g., https://example.com).
2. The client system:
   - Chooses **Port 50,000** (dynamic port) for the communication.
   - Sends a request to the server at **Port 443** (HTTPS).
3. The server responds to **Port 50,000** on the client, ensuring the communication flows smoothly.
4. After the session ends, **Port 50,000** is released for future use.

Summary Table

Port Range	Type	Purpose/Examples
0 to 1,023	Well-known ports	HTTP (80), HTTPS (443), DNS (53), FTP (21)
1,024 to 49,151	Registered ports	MySQL (3306), RDP (3389), Alternate HTTP (8080)
49,152 to 65,535	Dynamic/Private ports	Temporary client-side communication ports

C. Protocols (Traffic Rules)

Roads have traffic rules that govern how vehicles move, ensuring safety and efficiency:

- **TCP (Careful Driver):** A delivery driver who ensures every package arrives safely, resending it if necessary and getting confirmation of delivery.
- **UDP (Fast Driver):** A speedy driver who doesn't wait for confirmation and skips retries, prioritizing speed over reliability.

The protocol ensures smooth communication, like vehicles following consistent traffic rules.

Key Takeaway:

- IP Address = The unique address of a house (device).
- Port = The specific road used to reach the house.
- Protocol = The traffic rules followed to ensure successful delivery.

What are Network Protocols?

1. Set of rules
2. How data is transmitted
3. Device Communication (Purpose)

TCP/IP Model	TCP/IP Internet Protocol Suite
Application	Telent, SMTP, POP3, FTP, NTP, HTTP, SNMP, DNS, SSH, …
Transport	TCP, UDP (Core Communication Protocols)
Internet	IP, ICMP, ARP, DHCP
Network Access	Ethernet, PPP, ADSL

:- Chapter 2: TCP/IP and UDP -:

♠ **How TCP Works (Flags and Their Functions)**

TCP (Transmission Control Protocol) uses **flags** as part of its header to control communication between devices. These flags help manage and maintain a reliable connection. Here's how they work, with examples for better understanding:

1. URG (Urgent)

- **Purpose:** Indicates that the data in the packet should be processed immediately, bypassing normal queuing.
- **Example:** Imagine a situation where a priority message or command must be executed immediately, such as an emergency alert. The URG flag ensures this data is prioritized over regular traffic.

2. FIN (Finish)

- ⊗ **Purpose:** Signals the end of data transmission, indicating no more data will be sent.
- ⊗ **Example:** After completing a file transfer (e.g., via SHAREit), the FIN flag is used to gracefully close the connection between the sender and receiver.

3. RST (Reset)

- ⊗ **Purpose:** Resets a connection if an error occurs or if the communication is no longer valid.
- ⊗ **Example:** If data transmission is slow, corrupted, or encountering problems, the RST flag resets the connection to attempt a clean start. This is akin to rebooting a system to resolve errors.

4. PSH (Push)

- ⊗ **Purpose:** Instructs the receiver to process all buffered data immediately without waiting for additional packets.
- ⊗ **Example:** If a file transfer (e.g., in SHAREit) is 93% complete but the connection is about to be disrupted, the PSH flag ensures that the remaining data is sent immediately to complete the transfer.

5. ACK (Acknowledge)

- ⊗ **Purpose:** Confirms the successful receipt of a packet by the receiver.
- ⊗ **Example:** In WhatsApp, the ACK flag is akin to the indicators that show whether a message has been sent, delivered, or read by the recipient.

6. SYN (Synchronize)

- ⊗ **Purpose:** Initiates a connection between two devices. This flag is used during the **three-way handshake** at the start of a TCP connection.
- ⊗ **Example:** If you meet someone and ask for their phone number, they might hesitate if you're a stranger. However, if you introduce yourself properly, they are more likely to share their number. Similarly, the SYN flag establishes a connection by starting with a request, followed by a proper acknowledgment process.

Summary:

Flag	Function	Example
URG	Process data immediately	Emergency alert systems.
FIN	End of transmission	Closing connections after file transfer.
RST	Reset the connection	Rebooting due to slow or faulty transmission.
PSH	Send buffered data immediately	Ensuring file transfer completes before disconnection.
ACK	Acknowledge receipt of data	WhatsApp message status (sent/delivered/read).
SYN	Initiate a connection	Establishing a connection like introducing yourself.

Example: (How flag works)

Sagar : I want to talk with you Sheela on port 21, are you open? (SYN, SEQ# 10)

Arpita : Ok, let's talk Sagar. I am open on port 21.(SYN + ACK, ACK#11, SEQ#142)

Sagar : Ok, Thans Arpita. (ACK, ACK#143, SEQ#11).

..:: At this point, the connection is established.

Sagar : I am done with the data transfer. (FIN, SEQ# 50).

Arpita : Ok, I received your termination request. (ACK, ACK#51, SEQ#170)

Arpita : I have received all the data send. (FIN, SEQ#171)

Sagar : OK, Thanks Arpita. (ACK, ACK#172, SEQ#51)

..:: The connection is now gracefully terminated.

♠ Three-Way Handshake

The **three-way handshake** is a three-step process used to establish a reliable connection between two devices over a TCP/IP network. It involves the following steps:

1. **SYN (Synchronize):**
   The initiating device (client) sends a **SYN** message to the server, requesting to establish a connection.
2. **SYN-ACK (Synchronize + Acknowledge):**
   The receiving device (server) responds with a **SYN-ACK** message, acknowledging the client's request and indicating its readiness to connect.
3. **ACK (Acknowledge):**
   The initiating device (client) sends an **ACK** message back to the server, confirming the connection establishment.

At this point, the connection is established, and both devices can begin data transmission.

♠ UDP (User Datagram Protocol)

UDP is a communication protocol used for transmitting data over a network. Unlike TCP, UDP focuses on **speed** and **efficiency** rather than reliability. It is often used in applications where quick data transfer is more important than ensuring every packet is received correctly.

Key Features of UDP

1. **Connectionless Protocol:**
   - ⊗ UDP does not establish a connection before data transfer.
   - ⊗ Each packet (called a datagram) is sent independently.

2. **No Error Checking or Acknowledgments:**
    - ⊗ UDP does not guarantee the delivery of packets or verify if packets are received in the correct order.
    - ⊗ This makes it faster than TCP but less reliable.
3. **Low Overhead:**
    - ⊗ With minimal error-checking and no need for retransmissions, UDP is lightweight and efficient.
4. **Fast Data Transfer:**
    - ⊗ Ideal for applications requiring low latency, such as live video streaming or online gaming.
5. **No Congestion Control:**
    - ⊗ UDP does not adjust its data flow based on network conditions, which can result in packet loss during high traffic.

Use Cases of UDP

1. **Streaming Services:**
    - ⊗ Video streaming platforms (e.g., Netflix, YouTube) often use UDP for real-time video and audio delivery.
2. **Online Gaming:**
    - ⊗ Multiplayer games rely on UDP for real-time interaction, as speed is critical, and minor packet losses are tolerable.
3. **Voice and Video Calls:**
    - ⊗ Applications like Zoom, Skype, and VoIP use UDP for uninterrupted communication, even if some packets are dropped.
4. **DNS (Domain Name System):**
    - ⊗ DNS requests and responses are sent over UDP, as they are small and do not require reliability.

Advantages of UDP

1. **Faster Data Transmission:** No connection setup means data can be sent immediately.
2. **Low Latency:** Ideal for real-time applications.
3. **Less Overhead:** Efficient use of network resources.

Disadvantages of UDP

1. **No Reliability:** No guarantees that packets will arrive or be in the correct order.
2. **No Flow Control:** Can overwhelm the receiver if data is sent too quickly.
3. **Not Ideal for Critical Data:** Unsuitable for applications requiring data integrity, like file transfers.

Summary of UDP vs. TCP

Feature	UDP	TCP
Connection Type	Connectionless	Connection-oriented
Reliability	Unreliable	Reliable
Speed	Faster	Slower
Use Cases	Streaming, Gaming, DNS	File transfer, Web browsing

In a nutshell: UDP prioritizes speed over reliability, making it suitable for scenarios where real-time performance is critical and some packet loss is acceptable.

Note: A nutshell refers to a concise summary or the most essential part of something. The phrase "in a nutshell" is commonly used to mean "in brief" or "in a short and simple way."

TCP vs UDP

TCP (Three-Way Handshake):

Device: Sends a connection request. *(SYN)*

Server: Acknowledges the request and sends its own request. *(SYN-ACK)*

Device: Confirms the acknowledgment. *(ACK)*

Key Point: At this stage, the connection is established, and data transfer can begin.

UDP (Connectionless Communication):

Server: Sends a request or broadcast.

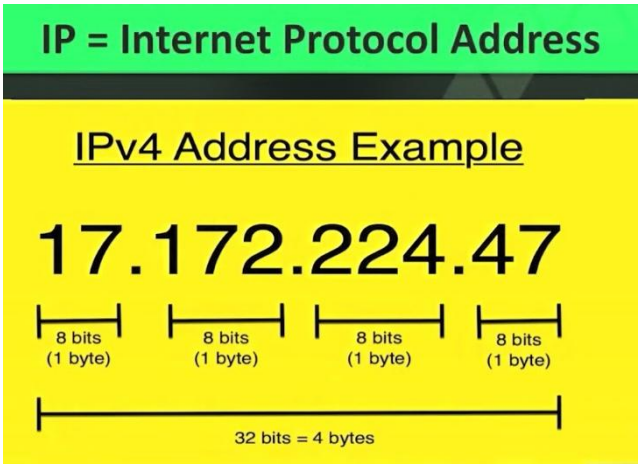
Device: Sends a response directly without waiting for acknowledgment.

Device: Sends another response if necessary, independently of the previous response.

Key Point: No handshake or connection is established. Data is sent immediately without reliability checks.

:- Chapter 3 -:

♠ **All About IP Address**



What will we learn?

1) What is IP Address?
2) IPv4 VS IPv6
3) Types of IP Address
4) What is public and private IP?
5) What is Static and Dynamic IP?

Example for Understanding:

An IP address can be compared to a **LA Financial Credit Union (LAFCU) Credit Card** with the format: **XXXX XXXX XXXX XXXX**. Similar to how each part of the credit card number carries specific information, the structure of an IP address is also segmented to convey meaningful details.

Use/Range of Credit Card:

Credit Card Analogy:

- **First Pair (XXXX):** Represents the **State** where the card was issued. *Example: California*
- **Second Pair (XXXX):** Indicates the **City**. *Example: Los Angeles*
- **Third Pair (XXXX):** Refers to the **branch's name and address** of LAFCU where the card belongs.
- **Fourth Pair (XXXX):** Uniquely identifies the **user** who owns the card.

Purpose of the Example:

This analogy demonstrates that an assigned number, such as an IP address or a credit card number, is not random. Instead, each segment carries meaningful information to identify and locate specific entities or users.

Use/Range of an IP Address:

In the world of networking:

1. **First Pair (17):** Represents the **Country**.
2. **Second Pair (172):** Identifies the **State or Region**.
3. **Third Pair (224):** Specifies the **City or ISP (Internet Service Provider)**.
4. **Fourth Pair (47):** Points to the unique **Device ID** within the network.

♠ **Tracking an IP Address Involved in a Crime (Example):**

If a crime occurs at **11:35 PM**, here's how an IP address can help:

1. **Reaching the ISP:**
   By analyzing the **third pair** of the IP address, authorities can identify the **Internet Service Provider (ISP)** and its office location.

2. **Obtaining Logs:**
   Authorities visit the ISP's office and request logs for the specific IP address at **11:35 PM**. These logs will contain details such as:
   - User information.
   - Activities performed.
   - Approximate location of the device at that time.

By combining this information, investigators can track the user or device responsible.

♠ **IP Address Limitations:**

1. **Range of Each Pair:**
   Each pair (or octet) in an IPv4 address must be a number between **0 and 255**. This range comes from the 8-bit binary representation of each segment ($2^8 = 256$ possible values).
2. **IPv4 Exhaustion:**
   Due to the limited address space in IPv4, we now use IPv6, which offers a significantly larger range of addresses.

♠ **IPv4 vs IPv6**

As of **October 2024**, there were **5.52 billion internet users worldwide**, accounting for **67.5% of the global population**. This growth in internet users highlights the importance of transitioning from IPv4 to IPv6 due to the increasing demand for IP addresses.

IPv4 (Internet Protocol Version 4):

1. **Address Size:** 32-bit number.
2. **Address Format:** Represented in **Dotted Decimal Notation** (e.g., 192.159.252.76).
3. **Prefix Notation (CIDR):** Specifies the network portion of the address (e.g., 192.149.0.0/24).
4. **Number of Addresses:** 2^{32} = ~4.29 billion addresses.

Limitations of IPv4:

Address exhaustion due to the limited number of unique addresses, leading to the development of IPv6.

IPv6 (Internet Protocol Version 6):

1. **Address Size:** 128-bit number.
2. **Address Format:**
   Represented in **Hexadecimal Notation** (e.g., 3dde:f200:0234:ab00:0123:4567:8901:abcd).
   Uses a combination of numbers (0–9) and letters (a–f).
3. **Prefix Notation (CIDR):** Specifies the starting format of the network (e.g., 3ffe:200:0234:/48).
4. **Number of Addresses: 2^{128} = ~340 undecillion addresses** (or 340 trillion, trillion, trillion).

Advantages of IPv6:

- Provides a virtually unlimited address space to accommodate the growing number of devices connected to the internet.
- Simplifies address configuration with features like auto-configuration and improved routing efficiency.

Key Differences Between IPv4 and IPv6:

Feature	IPv4	IPv6
Address Size	32-bit	128-bit
Address Format	Dotted Decimal Notation	Hexadecimal Notation
Number of Addresses	~4.29 billion	~340 undecillion
Prefix Notation	CIDR (e.g., /24)	CIDR (e.g., /48)
Usage	Older systems/networks	Modern systems and future-ready

Conclusion:

IPv4 was sufficient during the early internet days, but the rapid growth of devices and internet users made IPv6 a necessity. With its vast address space and advanced features, IPv6 is the backbone of the modern internet, ensuring scalability for decades to come.

♠ **What does Prefix Notation? in networking**

Prefix Notation is a way to represent IP addresses, where a slash (/) followed by a number indicates the network prefix length (CIDR notation)

♠ **What is prefix length?**

Prefix length in networking refers to the number of bits in an IP address that are used to identify the network portion. It's represented by a number following a slash (/) in CIDR notation. For example, in the IP address 192.168.1.100/24, the prefix length is 24, indicating that the first 24 bits (192.168.1) represent the network address.

♠ **Types of IP Addresses**

IP addresses can be categorized into the following types:

1. **Public IP Address**: Used to identify devices on a global scale over the internet.
    - ¬ **Example**: Real Name (visible to everyone).

2. **Private IP Address**: Used within a private network to identify devices locally.
    - ¬ **Example**: Nick Name (only known by close persons).

3. **Static IP Address**: An IP address that remains fixed and does not change.
    - ¬ **Example**: Permanent Address.

4. **Dynamic IP Address**: An IP address that changes periodically, often assigned by a DHCP server.
    - ¬ **Example**: Temporary Address (changes if not used for some time).

♠ Difference Between Public and Private IP Addresses

Feature	Public IP Address	Private IP Address
Usage	Useful for **WAN (Wide Area Network)**. Example: Real Name (known to everyone).	Useful for **LAN (Local Area Network)**. Example: Nickname (known to a small group).
Scope	Used to identify devices across the **internet**.	Used within a **local network** only.
Visibility	Accessible and visible globally.	Not accessible from outside the local network.
Communication Rules	Can connect with other **Public IPs** across the internet.	Can connect with other **Private IPs** in the same local network only.

♠ Rules for Public and Private IP Addresses

1. **Connectivity:**
   - ⊗ Private IPs **cannot connect** with Public IPs directly.
   - ⊗ Public IPs **cannot connect** with Private IPs directly.
2. **Range:**
   - ⊗ Private IPs can only connect with other Private IPs **within the same Local Area Network (LAN)**.
   - ⊗ Public IPs can connect with any other Public IP on the **Wide Area Network (WAN)**.

♠ Local Host

Example: Mind (no one knows what is in your mind)**)**

Local Host is only accessible within the same device for internal communication. 127.0.0.1 – same for all devices.

♠ Dynamic IP Address

- ⊗ **Definition:** A temporary IP address assigned to a device when it connects to the internet.
- ⊗ **Function:** The IP address is recycled and given to another user when the device disconnects or stops using the internet.
- ⊗ **Examples:** Commonly used for devices like **mobile phones, laptops, etc.**

♠ Static IP Address

- ⊗ **Definition:** A permanent IP address assigned to a device that does not change, even when the device is not actively connected to the internet.
- ⊗ **Function:** Ideal for devices that need consistent access, such as **servers** or **web hosting services**.

- ⊗ **Examples:** Web servers, email servers, and any critical networking equipment.

♠ **Tips:**

Check Your IP Address

You can check your IP address by visiting:
https://whatismyipaddress.com/

1. **Dynamic IP Addresses**:
   - ⊗ If you are using a mobile network (SIM), your IP address is often dynamic, meaning it may change periodically or each time you reconnect to the network.
   - ⊗ For example, turning **Airplane Mode** on and off resets your network connection, which might assign a new IP address.
2. **Idle Internet Connection**:
   - ⊗ If you do not use the internet for a while, your network provider may release your current IP address and assign a new one when you reconnect.
3. **Public vs. Private IPs**:
   - ⊗ Websites like the one mentioned show your **public IP address**, which is assigned by your Internet Service Provider (ISP) and visible on the internet.
   - ⊗ Devices on a local network (e.g., connected to a home Wi-Fi router) also have **private IP addresses**, which are only used internally.

:- Chapter 4 -:

♠ **Working of a Router**

A **router** is a device that directs data packets between networks and manages IP address allocation, ensuring communication between devices inside and outside a network. It uses the following components and concepts:

- ⊗ **Public Address:** The router's IP address visible to the internet, used for communication with external networks. Example: 82.10.250.19
- ⊗ **MAC Address:** A hardware address assigned to a device for unique identification on a local network.
- ⊗ **Private Address:** Internal IP addresses used for devices within the local network. Example: 192.168.0.102 for a laptop.
- ⊗ **Local Host:** A term used to refer to the device or system you are working on within the same network. Example: 127.0.0.1

How are IP Addresses Allotted?

1. **Manual Assignment:**
   - ⊗ IP addresses are manually configured by the user or network administrator.

2. **Automatic Assignment:**

   ⊗ IP addresses are assigned automatically using **DHCP (Dynamic Host Configuration Protocol)**.

   ⊗ **Example:** Hotspots automatically assign IP addresses to connected devices.

   ⊗ **How it Works:**

      ¤ **ARP (Address Resolution Protocol):** Assists DHCP in assigning IP addresses by mapping IP addresses to MAC addresses on the local network.

♠ **Source and Destination IP Addresses: A Practical Example**

Let's consider a laptop connected to a router and attempting to log in to Facebook.

Network Components in the Example:

⊗ **Laptop's Private IP:** 192.168.0.102

⊗ **Router's Private IP:** 192.168.0.1

⊗ **Router's Public IP:** 82.10.250.19

⊗ **Facebook Server's Public IP:** 4.4.4.4

Step-by-Step Process of Communication

1. **Laptop Sends a Request:**
   ¤ The **laptop** (192.168.0.102) sends a login request to the **router**.
2. **Router Translates the Request (NAT):**
   ¤ The **router** uses **NAT (Network Address Translation)** to switch the laptop's **private IP** (192.168.0.102) to the **router's public IP** (82.10.250.19).
   ¤ The router then sends the request to the **Facebook server**.
3. **Facebook Server Processes the Request:**
   ¤ The **Facebook server** (4.4.4.4) receives the request from the **router's public IP** (82.10.250.19) and processes it.
4. **Response Travels Back:**
   ¤ The **Facebook server** sends the response back to the **router's public IP** (82.10.250.19).
   ¤ The router converts the public IP back to the laptop's **private IP** (192.168.0.102) and delivers the response.

Summary of IP Addresses During the Process

Source IP	Destination IP
Laptop: 192.168.0.102	Router: 192.168.0.1 → Router (Public): 82.10.250.19 → Facebook Server: 4.4.4.4

Response Source IP	Response Destination IP
Facebook Server: 4.4.4.4	Router (Public): 82.10.250.19 → Router: 192.168.0.1 → Laptop: 192.168.0.102

:- Chapter 5 -:

♠ Domain name and DNS networking

What will we learn?

1. What is Domain Name?
2. What is DNS?
3. Records in DNS and there use.
4. What is Zone File?

What is a Domain Name?

A domain name is a **human-readable name for an IP address**, making it easier to remember and access. **Purpose:** Simplifies the use of public static IP addresses by renaming them to user-friendly names like google.com or facebook.com.

Why do we need a Domain name? -- To host a website.

Website components:

1. **Public Static IP Address:**
   - ⊗ Required to host the website so that any user on the internet can access it.
2. **Domain Name:**
   - ⊗ Purchased from a **domain provider** (e.g., GoDaddy, Namecheap).
   - ⊗ The provider gives a **username and password** to access the **Domain Manager**, where domain settings (called **records**) can be configured. Information updates in ZONE File.

Records in DNS (Domain Name System) help configure, map the domain name to specific functionalities.

Domain Example

- ¤ Common domain names: example.com, google.com, facebook.com.

♠ Types of Records in DNS and Their Uses

A	•IP of domain name. 192.185.141.193
CNAME	•Forwards domain and subdomain to another domain
MX	•Directs mail to email server
TXT	•Any text by Admin (spf)
NS	•Name Server of DNS entry
SOA	•Admin Info about a domain.
SRV	•Specify port for specific service
PTR	•Provides domain name in reverse-lookups

1. A Record (Address Record):

- ⊗ Maps the domain name to its IPv4 address.
- ⊗ **Example:** 192.185.141.193 → sagar1.com

2. CNAME (Canonical Name):

- ⊗ Redirects one domain or subdomain to another domain.
- ⊗ **Example:** sagar1.com → sagar.com

3. MX (Mail Exchange Record):

- ⊗ Directs emails to the specified mail server.
- ⊗ **Example:**
  - ¤ biswas@sagar.com can use Gmail as the email platform.
  - ¤ For this, gmail.com must be added in the MX record for approval.

4. TXT (Text Record):

- ⊗ Stores custom text data added by the domain admin (e.g., SPF, verification keys).
- ⊗ **Use:** Developers can use this record to write anything like SPF (Sender Policy Framework) or DKIM for email authentication.

5. NS (Name Server):

- ⊗ Specifies the name servers for the domain.
- ⊗ **Example:** ns1.sagar.com, ns2.sagar.com
- ⊗ **Note:** A domain should have **at least two-name servers** to ensure redundancy and prevent DoS (Denial of Service) attacks.

6. SOA (Start of Authority):

- ⊗ Contains administrative information about the domain, such as the admin's contact details.
- ⊗ **Use:** If the domain is not working, users can report issues using the information provided (e.g., email address, phone number).

7. SRV (Service Record):

- ⊗ Specifies ports for specific services.
- ⊗ **Use:**
  - ¤ Ensures services are running on specific ports.
  - ¤ **Caution:** Open ports can be exploited by hackers.

8. PTR (Pointer Record):

- ⊗ Maps an IP address to a domain name in reverse lookups.
- ⊗ **Purpose:**
  - ¤ Creates confusion for hackers.
  - ¤ **Example:** If a hacker queries sagar.com, the server may respond with a different domain name, like google.com.

Summary

- ⊗ **Domain Names** are essential for making IP addresses user-friendly and are configured using DNS records.
- ⊗ Proper configuration ensures functionality, security, and seamless communication over the internet.

♠ **What is DNS?**

- ⊗ **DNS** stands for **Domain Name System**.
- ⊗ It acts as the **Address Book of the Internet**, translating human-readable domain names (e.g., google.com) into machine-readable IP addresses (e.g., 142.250.190.46).
- ⊗ **Purpose:** Enables users to access websites by typing domain names instead of IP addresses.

How Does DNS Work?

- ⊗ DNS stores all data in the form of **records** within a **Zone File**.
- ⊗ The **Zone File** is a publicly accessible text file that contains the mappings of domain names to their respective IP addresses and other DNS-related records.
- ⊗ When a domain name is entered in a browser:
  1. DNS accesses the **Zone File** to find the IP address associated with the domain.
  2. It redirects the user to the appropriate server or host using this IP address.

Key Tip:

- **DNS Records** (like A, MX, TXT) can be queried publicly using tools like:
  - ¤ nslookup (command-line tool).
  - ¤ dig (advanced DNS query tool).
  - ¤ Online DNS checkers.

♠ **What is a Zone File?**

- ⊗ A **Zone File** is a **text file** used by the DNS system to manage domain-related information.
- ⊗ It contains **DNS records** that map domain names to IP addresses and other configurations.

⊗ **Key Characteristics of a Zone File:**

1. Stores information like **A records, MX records, TXT records, and NS records**.
2. Used for **IP mapping** (linking domain names to IPs).
3. Specifies the **name servers** responsible for the domain.

Summary

- **DNS** is critical for converting domain names into IP addresses, ensuring seamless browsing and communication over the internet.
- The **Zone File** is a text-based database that contains all the important records needed for DNS to function.

:- Chapter 6 -:

♠ **Network reference models -- OSI and TCP/IP**

What will we learn?

-- What is OSI Model?

-- How it Works?

-- What is TCP/IP Model?

-- OSI vs TCP/IP Model

♠ **What is the OSI Model?**

1. **Open Systems Interconnection (OSI) Model**: A conceptual framework that standardizes network communication by dividing it into seven layers.
2. **Purpose**:
   - ¤ Defines functions to enable communication between devices.
   - ¤ Ensures interoperability between different network hardware and software.
3. **7 Layers**: Each layer has a specific role.
4. **Standardizes Communication**: Provides a universal set of rules for data exchange.
5. **Process**: The communication process moves **bottom-to-up** on the sender side and **top-to-bottom** on the receiver side.

7 Layers of OSI:

⊗ **Physical Layer**: Transmits raw bit streams over physical hardware.

⊗ **Data Link Layer**: Ensures reliable data transfer and error detection.

⊗ **Network Layer**: Handles routing and addressing (e.g., IP addresses).

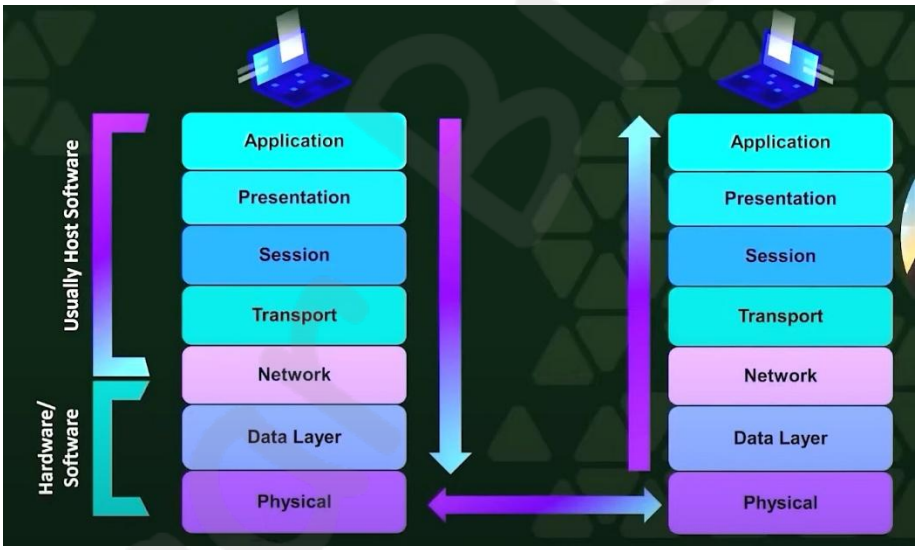
⊗ **Transport Layer**: Provides error correction, segmentation, and flow control.

- ⊗ **Session Layer**: Manages sessions and connections.
- ⊗ **Presentation Layer**: Formats, encodes, encrypts, or compresses data.
- ⊗ **Application Layer**: Interfaces directly with user applications (e.g., HTTP, SMTP).

How Does the OSI Model Work?

Data transmission follows these steps:

1. **Sender Side (Bottom-to-Up)**:
   - ¤ The data starts at the **Application Layer** and passes through all seven layers.
   - ¤ Each layer adds its headers, controls, and protocols. as part of the **encapsulation process**
2. **Receiver Side (Top-to-Bottom)**:
   - ¤ The data moves in reverse order, starting at the **Physical Layer**.
   - ¤ Each layer removes its headers, interprets the data, and passes it upwards. (a process called **decapsulation**)



♠ What is the TCP/IP Model?

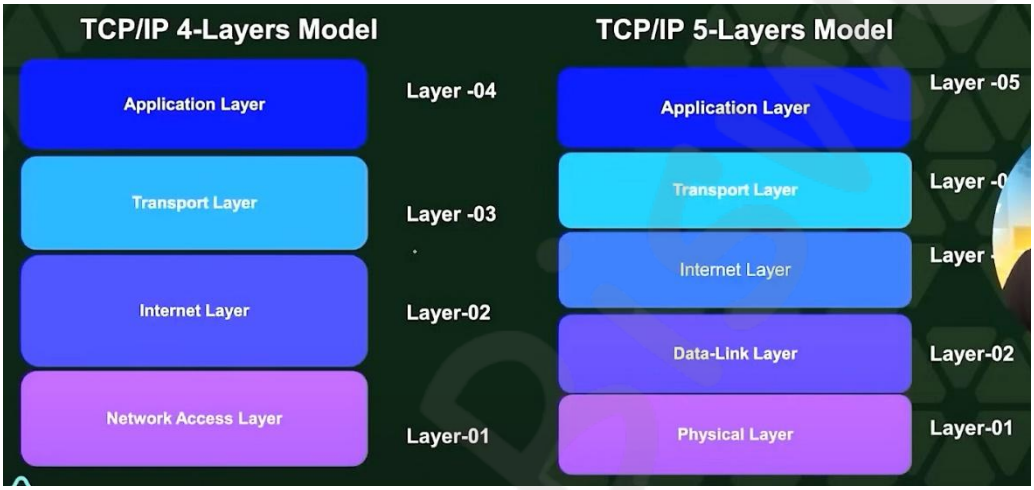
1. **Transmission Control Protocol/Internet Protocol (TCP/IP)**:
   - ⊗ A practical implementation of the OSI model for real-world use.
   - ⊗ Focuses on communication over the internet.
2. **Layers**:
   - ⊗ **4 Layers** (or **5 in some models**):
     - ¤ Application Layer
     - ¤ Transport Layer
     - ¤ Internet Layer
     - ¤ Network Access Layer (combines OSI's Data Link and Physical Layers)

3. **Purpose**:
    - ⊗ Defines protocols and standards for communication across wide area networks (WAN).
    - ⊗ Efficient for internet communication.

How does the TCP/IP Model work?

Data transmission:

1. Follows a similar process to OSI, where layers handle specific tasks.
2. Each layer in TCP/IP corresponds to one or more OSI layers.



TCP/IP vs OSI Model:

- TCP/IP is a simplified, **practical implementation** of OSI.
- OSI has **7 layers**, while TCP/IP typically has **4-5 layers**.
- TCP/IP is widely used over the internet (WAN), while OSI is mainly a theoretical model.

Key Notes:

- **Application Layer in OSI (Layer 7)** is easier to compromise and often targeted by attackers.

TCP/IP			OSI Model	Data Unit	Hardware	Protocols
Application	7		Application	Data	Gateway	S/MIME, SMTP, SNMP, HTTP, LPD, FTP, TFTP, Telnet, POP, SMB, NNTP, CDP, GOPHER, NDS, AFP, SAP, NCP, SET
	6		Presentation	Data	Gateway Redirector	No protocols. Service: TIFF, GIF, JPEG, ASCII, MPEG, MIDI, EBCDIC, POSTSCRIPT - Ensures confidentiality, Compression/Encryption
	5	SQL	Session	Data	Gateway	SOCKS, RPC, NFS, SQL, NetBIOS, RADIUS, DNS, ASP - Doesn't ensure security CORBA, DCOM, SOAP, .NET
Host-to-host	4	TCP	Transport	TCP-Segments UDP-Datagram	Gateway	TCP, UDP, SSL/TLS, SPX, SSH-2, ATP
Internet	3	IP	Network	Packets	Router Brouter	IP, IPSec, ICMP, IGMP, RIP, OSPF, BGP, IPX, SKIP, SWIPE, NAT, IGRP, EIGRP, BOOTP, DHCP, ISIS, ZIP, DDP
Network access	2	EEE 802	Data Link LLC MAC	Frames	Switch, Bridge, NIC	3thernet 802.3, Token Rin5 802.5, ATM, FDDI 802.4, Wi-Fi 802.11, PPP, L2TP, SLIP, ARP, RARP, 802.1AE MACSec, HDLC
	1		Physical	Bits	Repeater, Hub,NIC, Cables,MAU Multiplexer	ISDN, DSL, SONET, 10BASE-T, 10BASE2, 10BASE5, 100BASE-TX, 100BASE-FX, 100BASE-T, 1000BASE-T, 1000BASE-SX, EIA-x, RS-x

♠ Wi-Fi Attacks

- **Basic Tool**: **Aircrack-NG**
  Aircrack-NG is a popular suite of tools used for monitoring, attacking, testing, and cracking Wi-Fi networks. It includes features for:
  - Capturing packets
  - Cracking WEP and WPA/WPA2 passwords
  - Performing replay and deauthentication attacks

♠ CIA Triad

The **CIA Triad** is a fundamental model in cybersecurity that emphasizes three key principles:

C – Confidentiality

Ensures that sensitive information is only accessible to authorized individuals.

⊗ **Examples**:

¤ **Social Media**: Can lack confidentiality, as personal data might be exposed without proper privacy settings.

¤ **Google Drive**: Maintains confidentiality by restricting access to the owner unless shared.

¤ Using encryption (e.g., HTTPS, TLS) to protect data during transmission.

¤ Employing Multi-Factor Authentication (MFA) to prevent unauthorized access.

I – Integrity

Ensures that data is accurate and unaltered during storage, transmission, or processing.

⊗ **How is Integrity Maintained?**

¤ By using cryptographic **hashes** (e.g., MD5, SHA-256).
A hash ensures that any change to data (even a single bit) produces a completely different hash value, making tampering easily detectable.

⊗ **Example**:

¤ Verifying downloaded files using hash values provided by the source (e.g., checksums) to ensure the file hasn't been modified.

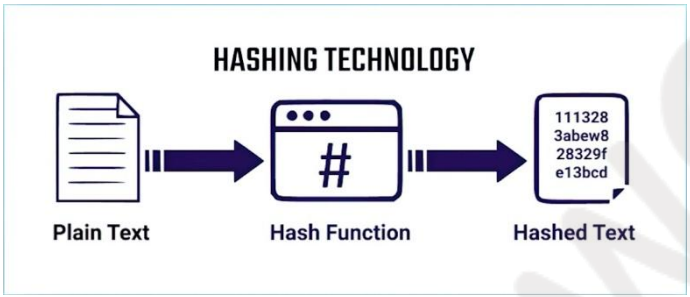
A – Availability

Ensures that data and resources are accessible to authorized users whenever needed.

⊗ **Examples**:

¤ **Bank Card Access**: Accessible 24/7 for account holders. (24/7 means 24 hours a day, 7 days a week.)

- **Social Media Accounts**: Data and features are always available as long as servers and connections are active.
- Availability can be impacted by **Denial-of-Service (DoS)** attacks, hardware failures, or natural disasters.
- Methods to ensure availability include redundancy (e.g., backup servers), load balancing, and disaster recovery plans.



ᕦ(===== END =====)ᕤ