**Computer networking**

**Definition:** A computer network is a system of interconnected devices (computers, servers, printers, etc.) that can exchange data and share resources.

**Network:**A network is a collection of computers and devices that are connected together to enable communication and data exchange.

**Nodes:**Nodes are devices that are connected to a network. These can include computers, Servers, Printers, [Routers,](https://www.geeksforgeeks.org/introduction-of-a-router/) [Switches](https://www.geeksforgeeks.org/types-of-switches-in-computer-network/), and other devices.

**Service Provider Networks:**These types of Networks give permission to take Network Capacity and Functionality on lease from the Provider. Service Provider Networks include Wireless Communications, Data Carriers, etc.

**IP Address**: An IP address is a unique numerical identifier that is assigned to every device on a network. IP addresses are used to identify devices and enable communication between them.

**DNS:**The Domain Name System (DNS) is a protocol that is used to translate human-readable domain names (such as www.google.com) into IP addresses that computers can understand.

**Firewall:**A [firewall](https://www.geeksforgeeks.org/introduction-of-firewall-in-computer-network/) is a security device that is used to monitor and control incoming and outgoing network traffic. Firewalls are used to protect networks from unauthorized access and other security threats, to ensure that data is transmitted and received correctly.

**Protocol:**A protocol is a set of rules and standards that govern how data is transmitted over a network. Examples of protocols include. TCP/IP (used for the internet) and Ethernet (used for wired LANs) etc, and encompassing various network types like LAN, WAN, MAN, PAN and WLAN

**Types of Computer Networks**

**Local Area Network (LAN):** Connects devices within a limited geographical area, such as a home, office, or school.

**Wide Area Network (WAN):** Connects devices across a large geographical area, such as different cities or countries (e.g., the internet).

**Metropolitan Area Network (MAN):** Connects devices within a metropolitan area, typically a city.

**Personal Area Network (PAN):** Connects devices within a small area, typically around a person (e.g., Bluetooth connection between a phone and headphones).

**Key Components of a Computer Network:**

**Nodes:** Devices such as computers, servers, and printers that participate in the network.

​**Transmission Media:** The physical or wireless pathways (like cables or radio waves) through which data travels between nodes. ​

**Network Interface Cards (NICs):** Hardware components that enable devices to connect to the network. ​

**Switches and Routers:** Devices that direct data to its intended destination within or between networks. ​

**Protocols:** Standardized rules governing data transmission, ensuring accurate and efficient communication.

**How Data Transmission Works in a Network**

**Data Segmentation:** Large data sets are divided into smaller units called packets. ​

**Addressing:** Each packet is assigned source and destination addresses, guiding it to the correct recipient. ​

**Transmission:** Packets are sent over the network via transmission media. ​

**Routing:** Routers and switches analyse packet addresses to determine the optimal path to the destination. ​

**Reassembly:** Upon arrival, packets are reassembled into their original form for use by the receiving device.

**Types of Enterprise Computer Networks**

**LAN:**A [Local Area Network (LAN)](https://www.geeksforgeeks.org/types-of-area-networks-lan-man-and-wan/)is a network that covers a small area, such as an office or a home. LANs are typically used to connect computers and other devices within a building or a campus.

**WAN:**A [Wide Area Network (WAN)](https://www.geeksforgeeks.org/wan-full-form/) is a network that covers a large geographic area, such as a city, country, or even the entire world. WANs are used to connect LANs together and are typically used for long-distance communication.

**Cloud Networks:**[Cloud Networks](https://www.geeksforgeeks.org/cloud-networking/) can be visualized with a Wide Area Network (WAN) as they can be hosted on public or private cloud service providers and cloud networks are available if there is a demand. Cloud Networks consist of Virtual Routers, Firewalls, etc.

Networking Devices

Network devices, also known as networking hardware, are physical components that facilitate communication and interaction between devices on a computer network. They play crucial roles in managing data traffic, ensuring security, and maintaining efficient network operations. Here's an overview of common network devices and their primary functions:

**1. Router** Routers connect multiple networks, directing data packets between them. They determine the optimal path for data transmission, enabling devices on different networks to communicate effectively. Routers also often provide features like network address translation (NAT) and firewall security. ​

**2. Switch** Switches operate within a single network, connecting devices such as computers, printers, and servers. They use MAC addresses to forward data only to the intended recipient, reducing unnecessary data traffic and enhancing network efficiency. ​

**3. Hub** Hubs are basic networking devices that connect multiple devices in a network segment. They broadcast incoming data to all ports, regardless of the destination, which can lead to inefficiencies and security concerns. Due to these limitations, hubs have largely been replaced by switches in modern networks. ​

**4. Bridge** Bridges connect two or more network segments, allowing them to function as a single network. They filter traffic based on MAC addresses, reducing collisions and improving overall network performance. ​

**5.** **Network Interface Card (NIC)** NICs are hardware components installed in devices, allowing them to connect to a network. They provide the physical interface for networking and handle data transmission and reception.

**Port**

A port can be referred to as a logical channel through which data can be sent/received to an application. Any host may have multiple applications running, and each of these applications is identified using the port number on which they are running.

A port number is a 16-bit integer; hence, we have 216 ports available which are categorized as

**Well-known/System Ports:**

Ports 0 to 1023 are reserved for well-known services and protocols, such as FTP, SSH, and HTTP.

**Registered Ports:**

Ports 1024 to 49151 are registered and assigned to specific services or applications by the Internet Assigned Numbers Authority (IANA).

**Dynamic/Private Ports:**

Ports 49152 to 65535 are dynamic or ephemeral ports, used for temporary connections and client-side communication.

**Example Ports:**

Port 20 and 21: File Transfer Protocol (FTP)

Port 22: Secure Shell (SSH)

Port 23: Telnet

Port 25: Simple Mail Transfer Protocol (SMTP)

Port 53: Domain Name System (DNS)

Port 80: Hypertext Transfer Protocol (HTTP)

Port 443: Hypertext Transfer Protocol Secure (HTTPS)

**DNS Server:**

DNS stands for **Domain Name System**. DNS is basically a server that translates web addresses or URLs (ex: www.google.com) into their corresponding IP addresses. We don’t have to remember all the IP addresses of each and every website. The command ‘**nslookup**’ gives you the IP address of the domain you are looking for. This also provides information on our DNS Server.

**ARP:** [ARP](https://www.geeksforgeeks.org/how-address-resolution-protocol-arp-works/) stands for **Address Resolution Protocol**. It is used to convert an IP address to its corresponding physical address (i.e., MAC Address). ARP is used by the Data Link Layer to identify the MAC address of the Receiver’s machine.

**RARP:** [RARP](https://www.geeksforgeeks.org/what-is-rarp/) stands for **Reverse Address Resolution Protocol**. As the name suggests, it provides the IP address of the device given a physical address as input. But RARP has become obsolete since the time DHCP has come into the picture.

The Domain Name System (DNS) is a critical component of computer networking. It converts easily recognizable domain names, such as www.example.com, into numerical IP addresses that computers use to identify each other on the network.

**How DNS Works?**

DNS works efficiently, translating user-friendly domain names into IP addresses, allowing seamless navigation on the internet. Below step by step working of DNS:

**User Input:**When a user enters a domain name in a browser, the system needs to find its [IP address](https://www.geeksforgeeks.org/what-is-an-ip-address/).

**DNS Query:** The user’s device sends a DNS query to the DNS resolver.

**Resolver Request:**The DNS resolver checks its cache for the IP address. If not found, it forwards the request to the root DNS server.

**Root DNS Server:**The root DNS server provides the address of the TLD (Top-Level Domain) server for the specific domain extension (e.g., .com).

**TLD DNS Server:** The TLD server directs the resolver to the authoritative DNS server for the actual domain.

**Authoritative DNS Server:** The authoritative DNS server knows the IP address for the domain and provides it to the resolver.

**Response to User:** The resolver stores the IP address in its cache and sends it to the user’s device.

**Access Website**: With the IP address, the user’s device can access the desired website.

**Network Security**

Ensuring the security of a network is crucial to protect data and resources from unauthorized access and attacks. Key aspects of network security include:

**Firewalls:**Devices or software that monitor and control incoming and outgoing network traffic based on security rules.

**Encryption:** The process of encoding data to prevent unauthorized access. Commonly used in [VPNs](https://www.geeksforgeeks.org/what-is-vpn-and-how-it-works/), HTTPS, and secure email.

**Intrusion Detection Systems (IDS):** Tools that monitor network traffic for suspicious activity and potential threats.

**Access Control:**Mechanisms that restrict access to network resources based on user identity and role.

**Regular Updates and Patching:** Keeping software and hardware up to date to protect against vulnerabilities.

Why Use Computer Networks?

Computer network play a important role in modern life. Here are some key benefits of computer networks:

**Fast and Easy Communication**: Networks enable all types of digital communication, like emails, messaging, file sharing, video calls, and streaming.

**More Storage Space**: Suppose if we don’t have a cloud storage then we have to store data in physical files that will consume a physical space so computer network provides a storage for storing data.

**Easier Sharing of Information**: Networks make it simpler for users and teams to share resources and information. Teams can collaborate more easily, and users get faster response from network devices.

**Better Security**: Well-designed networks are more reliable and give businesses more options for keeping data safe. They come with built-in security features like encryption and access controls to protect sensitive information from cyber threats.

Classes of IP Addresses

​IP addresses in IPv4 are categorized into five classes—A, B, C, D, and E—each serving distinct purposes and accommodating different network sizes. Here's an overview of these classes: ​

**Class A**

* **Range:** 1.0.0.0 to 126.0.0.0​
* **Default Subnet Mask:** 255.0.0.0​
* **Network/Host Allocation:** 8 bits for the network portion and 24 bits for hosts​
* **Number of Networks:** 128​
* **Hosts per Network:** 16,777,214​
* **Purpose:** Designed for very large networks​

**Class B**

* **Range:** 128.0.0.0 to 191.255.0.0​
* **Default Subnet Mask:** 255.255.0.0​
* **Network/Host Allocation:** 16 bits for the network portion and 16 bits for hosts​
* **Number of Networks:** 16,384​
* **Hosts per Network:** 65,534​
* **Purpose:** Intended for medium to large-sized networks​

**Class C**

* **Range:** 192.0.0.0 to 223.255.255.0​
* **Default Subnet Mask:** 255.255.255.0​
* **Network/Host Allocation:** 24 bits for the network portion and 8 bits for hosts​
* **Number of Networks:** 2,097,152​
* **Hosts per Network:** 254​
* **Purpose:** Suited for small networks​

**Class D**

* **Range:** 224.0.0.0 to 239.255.255.255​
* **Purpose:** Reserved for multicast groups​

**Class E**

* **Range:** 240.0.0.0 to 255.255.255.255​
* **Purpose:** Reserved for experimental and research purposes​

Understanding these IP address classes is fundamental for network design and management, ensuring efficient and organized allocation of IP addresses across various network sizes and applications.

**Classless Inter-Domain Routing (CIDR)**

​Classless Inter-Domain Routing (CIDR) is a method for allocating IP addresses and routing Internet Protocol packets that enhances the efficiency of IP address distribution and routing. CIDR allows for a more flexible and efficient allocation of IP addresses.

**Key Features of CIDR:**

* **Variable-Length Subnet Masking (VLSM):** CIDR utilizes VLSM, enabling network prefixes to have variable lengths. This flexibility allows for the allocation of IP address blocks that closely match an organization's specific needs, reducing waste associated with fixed subnet sizes. ​
* **CIDR Notation:** IP addresses in CIDR are represented with a suffix indicating the number of bits in the network prefix. For example, 192.168.1.0/24 denotes an IP address where the first 24 bits are used for the network prefix, leaving the remaining bits for host addresses. ​

**Benefits of CIDR:**

* **Efficient IP Address Allocation:** By allowing variable-length prefixes, CIDR helps in allocating IP address blocks that more precisely fit the requirements of networks, thereby conserving IP address space. ​
* **Reduced Routing Table Size:** CIDR supports route aggregation, or super netting, which consolidates multiple IP address ranges into a single routing table entry. This aggregation minimizes the size of routing tables, enhancing the speed and efficiency of routing processes. ​

**CIDR Notation Examples:**

* 192.168.0.0/16: Indicates a network with a 16-bit prefix, encompassing 65,536 IP addresses (from 192.168.0.0 to 192.168.255.255). ​
* 10.0.0.0/8: Represents a network with an 8-bit prefix, covering 16,777,216 IP addresses (from 10.0.0.0 to 10.255.255.255).​

Understanding CIDR is crucial for network design and management, as it provides a scalable and efficient method for IP address allocation and routing. For practical applications, tools like the [CIDR to IPv4 Address Range Utility Tool](https://www.ipaddressguide.com/cidr) can assist in translating CIDR notations into IP address ranges and vice versa.

**Most Used Networking Commands**

Here are some of the most popular networking commands for troubleshooting and diagnosing, configuring settings, testing connectivity, and monitoring and statistics.

**1. ping**

Syntax: ping [destination IP/domain]

Functionality: This command tests network connectivity between the device and a target host by sending ICMP Echo Request packets and receiving Echo Reply responses.

Use Case: Verifying if a remote host, such as a gateway, DNS server, or external server, is reachable across the network.

**2. Traceroute (Linux) / tracert (Windows)**

Syntax:

traceroute [destination]

tracert [destination]

Functionality: This command traces the route packets take to reach a destination by displaying each hop and its latency. It helps identify delays or failures in the network path between the source and destination.

Use Case: Diagnosing routing issues and understanding the path to a destination.

**3. ipconfig (Windows) / ifconfig (Linux/Unix/macOS)**

Syntax:

ipconfig (Windows)

ifconfig (Linux/Unix/macOS)

Functionality: This command displays network interface configuration details, including IP address, subnet masks, default gateways, and DNS servers.

Use Case: Troubleshooting network connectivity issues by viewing IP details.

**4. netstat**

Syntax:

netstat -a (all active connections)

netstat -r (routing table)

netstat -i (interface stats)

Functionality: This command provides a snapshot of active network connections, port statuses, routing tables, interface statistics, and more.

Use Case: Troubleshooting network connections, ports, and routing issues.

**5. nslookup**

Syntax: nslookup [domain]

Functionality: This command performs manual DNS queries to convert domain names into IP addresses and also works in reverse.

Use Case: Verifying DNS records and troubleshooting name resolution issues.

**6. dig (Linux/Mac)**

Syntax: dig [domain]

Functionality: This command provides more detailed DNS information than nslookup, such as the entire DNS hierarchy.

Use Case: Verifying DNS records and troubleshooting domain name resolution issues.

**7. route**

Syntax: route -n (to view routing table)

Functionality: This command is essential for managing and viewing routing tables. It is commonly used to add static routes in environments where traffic needs to be directed over specific interfaces or in multi-homed networks where multiple gateways are used.

Use Case: Configuring or inspecting network routes.

**8. arp**

Syntax: arp -a (to view the ARP table)

Functionality: This command displays and modifies the ARP (Address Resolution Protocol) table, which maps IP addresses to MAC addresses.

Use Case: Troubleshooting IP-MAC mappings in local networks.

**9. ip (Linux)**

Syntax:

ip addr (to view IP configurations)

ip route (to view routing tables)

Functionality: This is the modern replacement of ifconfig and route, used to manage IP addresses, network interfaces, and routing on Linux systems.

Use Case: Network configuration and management, including IP assignments and route settings.

**10. tcpdump**

Syntax: tcpdump -i [interface]

Functionality: This command captures network packets in real-time, providing insight into traffic flowing through the network.

Use Case: Troubleshooting network traffic issues and inspecting data packets.

**11. nmap**

Syntax: nmap [target IP]

Functionality: This is used frequently for security audits to see what services are exposed on a host. It is an excellent reconnaissance tool that helps identify potential vulnerabilities or unintentional services running on a machine.

Use Case: Network discovery and security auditing by scanning for open ports and services.

**12. ssh**

Syntax: ssh [user]@[host]

Functionality: This command securely connects to a remote system over a network, encrypting the communication between client and server, allowing secure login, command execution, and file transfer using protocols like SFTP.

Use Case: Securing remote access to devices and servers.

**13. telnet**

Syntax: telnet [host] [port]

Functionality: While replaced mainly by SSH for remote management, telnet is still useful for testing whether a specific port is open or reachable (e.g., checking if a web server responds on port 80).

Use Case: Accessing devices or systems remotely (though usually replaced by SSH).

**14. curl**

Syntax: curl [URL]

Functionality: This command is used to test and interact with web services, APIs, and other endpoints over various protocols (HTTP, FTP, etc.).

Use Case: Verifying the availability of websites, APIs, or other network services and retrieving data from URLs.

**15. hostname**

Syntax:

hostname (to view)

hostname [new-name] (to set a new hostname)

Functionality: This command displays or sets the system’s hostname. It is essential for managing network identities in environments where multiple systems or servers are deployed.

Use Case: Verifying or changing the hostname of a system.

**16. ss**

Syntax: ss -tuln (to show listening TCP/UDP ports)

Functionality: A modern replacement for netstat, this command analyzes socket-level statistics, such as open and listening ports.

Use Case: Checking for active connections and open ports.

**17. netsh**

Syntax: netsh interface ip show config

Functionality: This command-line scripting utility is used to display or modify network configurations on Windows.

Use Case: Managing networking settings like IP, firewall, wireless, and more.

**18. iwconfig (Linux)**

Syntax: iwconfig

Functionality: This command is used to configure and display wireless network interface parameters in Linux.

Use Case: Viewing or configuring wireless interfaces and settings.