Here’s a detailed explanation of each concept with real-life examples:

---

1. Domain Name Server (DNS) and DNS Resolution

- **What is it?**
  DNS is a system that translates human-readable domain names (like `www.google.com`) into IP addresses (like `142.250.190.14`) that computers use to identify each other on a network.
  DNS resolution is the process of converting a domain name into its corresponding IP address.
- **Why use it?**
  To simplify accessing websites and services on the internet, as domain names are easier for humans to remember than IP addresses.
- **When to use it?**
  Whenever you need to connect to a website, email server, or online service by its domain name.
- **How to use it?**
  The process is automatic. When you type a URL in a browser, your computer sends a query to a DNS server, which resolves the domain name to an IP address.
- **Where is it used?**
  It’s used in web browsing, sending emails, and virtually all internet-based services.
  Example: When you type `www.facebook.com`, the DNS server resolves the domain name to an IP address like `69.63.176.13` so your browser can load the page.

---

2. Checksum

- **What is it?**
  A checksum is a small-sized piece of data derived from a block of digital data to detect errors during data transmission.
- **Why use it?**
  To ensure data integrity by detecting errors that might occur during transmission or storage.
- **When to use it?**
  In file downloads, data transfers, and network communication where error detection is critical.
- **How to use it?**
  A checksum is calculated at the sender’s end and sent along with the data. The receiver recalculates the checksum on the received data and compares it with the original.
- **Where is it used?**
  In networking (TCP/IP headers), data storage (RAID systems), and file integrity checks.
  Example: When downloading a software file, the website may provide a checksum to verify that the file was downloaded correctly without corruption.

---

3. Session Establishment

- **What is it?**
  Session establishment is the process of setting up a communication session between two devices or applications.
- **Why use it?**
  To enable reliable communication by establishing a connection before data exchange.
- **When to use it?**
  In applications that require persistent communication, such as video calls, online gaming, or database access.
- **How to use it?**
  Protocols like TCP use a handshake mechanism (e.g., three-way handshake) to establish a session.
- **Where is it used?**
  In VoIP calls, web applications, and database connections.
  Example: When you make a Zoom call, the application establishes a session with the Zoom server before starting the call.

---

4. Hops in Networking

- **What is it?**
  A hop refers to the passage of data packets from one network device (like a router) to another on its way to the destination.
- **Why use it?**
  To efficiently route data across networks.
- **When to use it?**
  Always occurs in multi-device or multi-network data transmission.
- **How to use it?**
  Network protocols like IP automatically handle hops. Tools like `traceroute` show the number of hops a packet takes.
- **Where is it used?**
  In data transmission over large networks like the internet.
  Example: Sending an email may involve multiple hops through routers to reach the recipient's mail server.

---

5. MAC (Media Access Control) Address

- **What is it?**
  A MAC address is a unique identifier assigned to a network interface card (NIC) of a device.
- **Why use it?**
  To uniquely identify devices on a local network.
- **When to use it?**
  In LAN communication and network troubleshooting.
- **How to use it?**
  Devices use MAC addresses for data link layer communication within a network.

- **Where is it used?**
  In Wi-Fi networks, Ethernet communication, and network security filters.
  Example: A Wi-Fi router uses MAC addresses to allow or block devices from accessing the network.

---

6. Switch in Networking

- **What is it?**
  A network switch is a device that connects multiple devices within a LAN and forwards data to the intended device.
- **Why use it?**
  To create efficient communication within a local network by reducing unnecessary traffic.
- **When to use it?**
  In local area networks (LANs) to connect devices like computers, printers, and servers.
- **How to use it?**
  Connect devices to the switch using Ethernet cables.
- **Where is it used?**
  In offices, schools, and data centers.
  Example: An office network uses a switch to connect all employee computers for file sharing and internet access.

---

7. Router in Networking

- **What is it?**
  A router is a device that connects different networks and routes data between them.
- **Why use it?**
  To enable communication between devices on different networks or between a network and the internet.
- **When to use it?**
  In home or office networks to connect to the internet or other networks.
- **How to use it?**
  Configure the router with network settings and connect it to a modem and local devices.
- **Where is it used?**
  At homes, businesses, and ISPs.
  Example: A home router connects your devices to the internet via your ISP.

---

8. Gateway in Networking

- **What is it?**
  A gateway is a network node that acts as an entry or exit point to another network.

- **Why use it?**
  To enable communication between different networks or protocols.
- **When to use it?**
  When connecting two different networks or accessing external networks like the internet.
- **How to use it?**
  Gateways are usually configured on routers or servers.
- **Where is it used?**
  In enterprise networks and internet access points.
  Example: Your home router acts as a gateway to the internet.

---

9. Socket in Networking

- **What is it?**
  A socket is an endpoint for sending or receiving data between devices in a network.
- **Why use it?**
  To enable application-level communication over a network.
- **When to use it?**
  In applications like web browsers, servers, and messaging apps.
- **How to use it?**
  Applications use APIs like `socket()` in programming languages like Python or Java.
- **Where is it used?**
  In web servers, chat applications, and file transfers.
  Example: A web server listens for requests on a socket, processes them, and sends responses back to clients.

---

11. What is a Modem in Networking?

- **What is it?**
  A modem (short for Modulator-Demodulator) is a device that converts digital signals from a computer into analog signals for transmission over telephone lines, and vice versa.
- **Why use it?**
  To connect to the internet when using services like DSL or cable, as the modem bridges the digital and analog worlds.
- **When to use it?**
  When you need to establish an internet connection through ISPs that use telephone or cable infrastructure.
- **How to use it?**
  Connect the modem to the ISP’s line and your router or computer.
- **Where is it used?**
  Homes, offices, and ISPs.
  Example: Your home internet setup often includes a modem to connect to your ISP's network.

---

12. Who is a Client and Who is a Server?

- **What is it?**
  - A **client** is a device or application that requests services or resources.
  - A **server** is a device or application that provides services or resources to clients.
- **Why use it?**
  To enable resource sharing and communication in networks.
- **When to use it?**
  In client-server architectures like websites, email, or file sharing.
- **How to use it?**
  The client sends requests, and the server processes and responds.
- **Where is it used?**
  In web browsing, file downloads, and database access.
  Example: When you use Gmail, your browser (client) sends requests to Google's email servers.

---

13. ARP Mapping to MAC

- **What is it?**
  ARP (Address Resolution Protocol) maps IP addresses to MAC addresses for final delivery of messages within a local network.
- **Why use it?**
  To ensure that data packets reach the correct physical device.
- **When to use it?**
  When transmitting data within a local network.
- **How to use it?**
  ARP requests are sent to resolve IP addresses to MAC addresses.
- **Where is it used?**
  In LANs and subnet communication.
  Example: When sending data from one computer to another on the same Wi-Fi network, ARP maps the target IP address to its MAC address.

---

14. Why Structured Data?

- **What is it?**
  Structured data is organized into rows, columns, and tables, making it easily searchable and analyzable.
- **Why use it?**
  To get quick and reliable responses because the data is well-organized.
- **When to use it?**
  In databases or applications requiring fast access and processing.
- **How to use it?**
  Use relational databases like SQL for storing structured data.
- **Where is it used?**
  In banking systems, e-commerce, and business analytics.

Example: An e-commerce platform uses structured data to instantly fetch product details when searched.

---

15. Connectionless/Connection-Oriented, Stateful/Stateless

- **What is it?**
  - **Connectionless:** No dedicated connection is required (e.g., UDP).
  - **Connection-oriented:** A connection is established before data transfer (e.g., TCP).
  - **Stateless:** No memory of previous interactions (e.g., HTTP).
  - **Stateful:** Retains memory of interactions (e.g., FTP).
- **Why use it?**
  To optimize resource use and communication based on the application's needs.
- **When to use it?**
  Use stateless for lightweight tasks (e.g., API calls) and stateful for persistent communication (e.g., online gaming).
- **How to use it?**
  Choose protocols based on requirements.
- **Where is it used?**
  Connection-oriented: file transfers; Stateless: REST APIs.
  Example: Streaming a video uses TCP (connection-oriented), while DNS uses UDP (connectionless).

---

16. What are Sessions and Cookies?

- **What is it?**
  - **Session:** Temporary server-side storage of user data during a session.
  - **Cookie:** Client-side data stored in the browser for tracking and personalization.
- **Why use it?**
  To maintain user states and preferences across requests.
- **When to use it?**
  Sessions: Temporary user state; Cookies: Persistent user preferences.
- **How to use it?**
  Use server-side languages like PHP for sessions; Set cookies with HTTP headers.
- **Where is it used?**
  E-commerce websites, user authentication.
  Example: Amazon uses cookies to remember your cart items even after you close the browser.

---

17. HTTP and Session Establishment

- **What is it?**
  HTTP is stateless, so sessions are used to track multiple requests from the same user.

- **Why use it?**
  To provide a seamless user experience in web applications.
- **When to use it?**
  When managing user interactions over multiple requests.
- **How to use it?**
  Implement session IDs in cookies or URLs.
- **Where is it used?**
  Online shopping carts and user login systems.
  Example: When you log into a banking website, sessions track your activity securely.

---

18. What is VPN?

- **What is it?**
  A Virtual Private Network (VPN) creates a secure, encrypted connection over the internet.
- **Types:**
  - Remote access (home/office)
  - Site-to-site (business networks)
  - Mobile VPN (smartphones)
  - SSL VPN
  - Double VPN
- **Why use it?**
  For security, privacy, and remote access.
- **When to use it?**
  When working remotely or accessing restricted content.
- **How to use it?**
  Use VPN software or hardware.
- **Where is it used?**
  Corporate environments and personal privacy tools.
  Example: A remote employee uses a VPN to securely access the company network.

---

19. What is a Proxy and Its Types?

- **What is it?**
  A proxy acts as an intermediary between a client and a server.
- **Types:**
  - Forward proxy
  - Reverse proxy
  - Anonymous proxy
  - Transparent proxy
- **Why use it?**
  To enhance security, anonymity, and performance.
- **When to use it?**
  When accessing restricted content or managing traffic.
- **How to use it?**
  Configure the proxy in your browser or network settings.

- **Where is it used?**
  In corporate networks and content delivery systems.
  Example: A company uses a reverse proxy to manage traffic to its web servers.

---

20. What is the IP Protocol?

- **What is it?**
  The Internet Protocol (IP) is responsible for addressing and routing packets between devices.
- **Why use it?**
  To enable communication between devices over the internet or local networks.
- **When to use it?**
  Always in network communication.
- **How to use it?**
  Configured automatically or manually on devices.
- **Where is it used?**
  In all internet-based services.
  Example: Sending an email uses IP to route the data to the recipient's server.

---

22. What is ICMP?

- **What is it?**
  Internet Control Message Protocol (ICMP) is a network layer protocol used to send error messages and operational information about the status of communication between devices.
- **Functions & Operations:**
  - Error reporting (e.g., unreachable host or network).
  - Diagnostics (e.g., ping).
  - Network troubleshooting (e.g., traceroute).
  - Flow control to manage data transmission.
- **Common ICMP Messages:**
  - **Echo Request/Reply**: Used in ping.
  - **Destination Unreachable**: Indicates that a destination is unreachable.
  - **Time Exceeded**: Indicates that a packet has exceeded its Time-To-Live (TTL).
  - **Redirect**: Suggests a better route for data packets.
- **Why use it?**
  To diagnose and manage network connectivity issues.
- **When to use it?**
  When troubleshooting network problems like unreachable devices or slow responses.
- **How to use it?**
  ICMP is automatically used by tools like `ping` and `traceroute`.
- **Where is it used?**
  Network monitoring, performance checks, and fault diagnosis.
  Example: If a website is down, ICMP can confirm whether the server is reachable.

---

23. What is PING?

- **What is it?**
  Ping is a utility that tests connectivity between devices using ICMP Echo Request and Echo Reply messages.
- **How it works:**
  - Sends ICMP Echo Request packets to the target device.
  - The target device responds with Echo Reply packets.
  - Measures the round-trip time and reports packet loss.
- **Why use it?**
  To check if a device is reachable and assess its response time.
- **When to use it?**
  When verifying network connectivity or troubleshooting latency issues.
- **How to use it?**
  Use the `ping` command in the command-line interface (CLI):
- `ping www.google.com`
- **Where is it used?**
  Network troubleshooting, server health checks, and latency testing.
  Example: If your Wi-Fi isn't working, you can ping your router (`ping 192.168.0.1`) to check connectivity.

---

24. What is Traceroute?

- **What is it?**
  Traceroute is a utility that maps the route packets take from the source to the destination, showing all intermediate devices (hops).
- **Why use it?**
  To diagnose network routing issues and identify where delays or failures occur.
- **When to use it?**
  When investigating slow network speeds, packet loss, or connectivity issues.
- **How to use it?**
  Run the `traceroute` (Linux/macOS) or `tracert` (Windows) command:
- `tracert www.google.com`
- **Where is it used?**
  Network diagnostics, performance monitoring, and routing analysis.
  Example: If a website is loading slowly, traceroute helps identify which router or network is causing the delay.

---

25. What are Hops in Networking?

- **What is it?**
  A hop is a step in the path that data packets take as they travel from the source to the destination. Each intermediate router or device is considered a hop.

- **Why use it?**
  To understand the journey of packets and troubleshoot network delays or failures.
- **When to use it?**
  When analyzing routing efficiency or pinpointing network issues.
- **How to use it?**
  Tools like `traceroute` or `ping` show the number of hops.
- **Where is it used?**
  In multi-router environments or large networks like the internet.
  Example: Data sent from your computer to a website may pass through multiple ISPs, with each ISP router being a hop.

Real-Life Example Combining Hops, Traceroute, and ICMP:

1. You try to access `www.example.com`, but the page doesn't load.
2. You run `ping www.example.com` to check if the server is reachable. If there's no reply, it suggests a connectivity issue.
3. You use `traceroute www.example.com` to see the route packets take. It shows all the hops (routers) and identifies where the packets stop or slow down.
4. ICMP messages like "Destination Unreachable" or "Time Exceeded" guide you to diagnose the issue, e.g., whether a router is blocking traffic or there's a misconfigured network.

Key Notes:

- **ICMP:** The protocol enabling tools like ping and traceroute.
- **Ping:** Simple connectivity check with round-trip time measurement.
- **Traceroute:** Detailed map of packet paths, showing hops and delays.
- **Hops:** The devices (routers/switches) data travels through to reach its destination.

---

27. How does TCP/IP work in 4 layers?

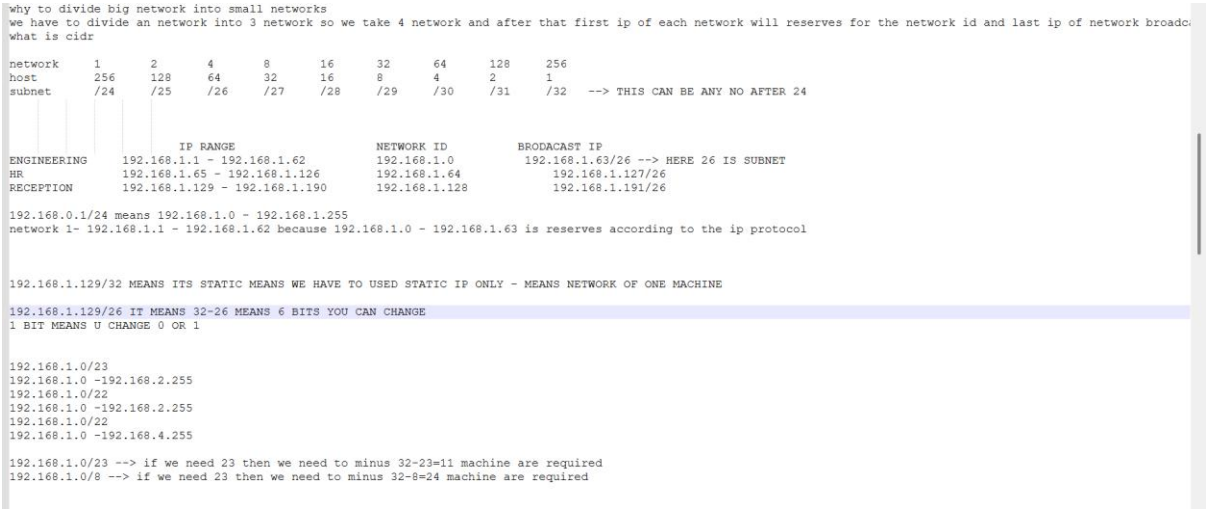
- **What is it?**
  TCP/IP (Transmission Control Protocol/Internet Protocol) is a suite of communication protocols structured in 4 layers:
  1. **Application Layer:** Handles user-facing services (e.g., HTTP, FTP, SMTP).
  2. **Transport Layer:** Ensures reliable data transmission using TCP or connectionless transfer using UDP.
  3. **Internet Layer:** Routes packets using IP addresses.
  4. **Network Access Layer (Data Link + Physical):** Handles data transmission over physical networks.
- **What are datagrams?**
  - A **datagram** is a self-contained packet of information sent across a network, primarily used in UDP for connectionless transmission.

- **Why use it?**
  TCP/IP ensures standardized communication across heterogeneous networks.
- **When to use it?**
  Always, as it forms the foundation of modern internet communication.
- **How to use it?**
  Applications and devices automatically use the TCP/IP stack.
- **Where is it used?**
  Web browsing, email, video streaming, and all internet-based communication.
  Example: Opening a website uses HTTP (Application Layer), TCP (Transport Layer), and IP (Internet Layer).

---

28. Why divide big networks into small networks?

- **What is it?**
  Subnetting divides a large network into smaller, manageable segments.
- **Why use it?**
  To improve performance, security, and efficient IP address allocation.
- **When to use it?**
  In large networks to isolate traffic or conserve IP addresses.
- **How to use it?**
  Use subnet masks (e.g., `/24`) to divide networks.
- **Where is it used?**
  Corporate networks, ISPs, and data centers.
  Example: A university divides its network into subnets for students, staff, and research labs.



```
why to divide big network into small networks
we have to divide an network into 3 network so we take 4 network and after that first ip of each network will reserves for the network id and last ip of network broadc
what is cidr

network     1       2       4       8       16      32      64      128     256
host        256     128     64      32      16      8       4       2       1
subnet      /24     /25     /26     /27     /28     /29     /30     /31     /32   --> THIS CAN BE ANY NO AFTER 24



                    IP RANGE                    NETWORK ID          BRODACAST IP
ENGINEERING     192.168.1.1 - 192.168.1.62          192.168.1.0          192.168.1.63/26 --> HERE 26 IS SUBNET
HR              192.168.1.65 - 192.168.1.126        192.168.1.64             192.168.1.127/26
RECEPTION       192.168.1.129 - 192.168.1.190       192.168.1.128            192.168.1.191/26

192.168.0.1/24 means 192.168.1.0 - 192.168.1.255
network 1- 192.168.1.1 - 192.168.1.62 because 192.168.1.0 - 192.168.1.63 is reserves according to the ip protocol



192.168.1.129/32 MEANS ITS STATIC MEANS WE HAVE TO USED STATIC IP ONLY - MEANS NETWORK OF ONE MACHINE

192.168.1.129/26 IT MEANS 32-26 MEANS 6 BITS YOU CAN CHANGE
1 BIT MEANS U CHANGE 0 OR 1


192.168.1.0/23
192.168.1.0 -192.168.2.255
192.168.1.0/22
192.168.1.0 -192.168.2.255
192.168.1.0/22
192.168.1.0 -192.168.4.255

192.168.1.0/23 --> if we need 23 then we need to minus 32-23=11 machine are required
192.168.1.0/8 --> if we need 23 then we need to minus 32-8=24 machine are required
```

```
192.168.1.0/23 --> if we need 23 then we need to minus 32-23=11 machine are required
192.168.1.0/8 --> if we need 23 then we need to minus 32-8=24 machine are required


region, vpc , availability zone, subnet we can customize the cidr
cidr theory
vpc belong to region and subnet belong to availability zone

NAT / NATTING - NETWORK ADDRESS TRANSLATOR
GATEWAY ARE USED TO CONNECT TWO DIFFERENT DEVICES

192.168.1.0 MEANS 32-24=8 MEANS 2^8=256


NAT MEANS FRAMES ADDRESS GET CHNAGE TO DEVICE ID ADDRESS
SWITCH USE ARP AND IT HAS DEFAULT ROUTE AND ITS USED WHEN ALL THE OPTIONS ARE OVER

FRAMES -> SWITCH -> NATTING -> DESTINATION MSG AND WHEN RETURNING BACK TO SOURCE THEN IT WILL DO DENATTING
```

---

29. What are datagrams?

- **What is it?**
  A datagram is an independent packet of data with source and destination information, sent without establishing a connection.
- **Why use it?**
  For fast, connectionless communication.
- **When to use it?**
  When reliability isn’t crucial (e.g., live video streaming).
- **How to use it?**
  Applications use UDP to send datagrams.
- **Where is it used?**
  DNS queries, VoIP, and online gaming.
  Example: When playing an online game, datagrams are used to send real-time updates.

---

30. What is CIDR?

- **What is it?**
  Classless Inter-Domain Routing (CIDR) is a method for IP address allocation and routing, replacing the older class-based system.
- **Why use it?**
  To efficiently allocate IP addresses and reduce waste.
- **When to use it?**
  When managing IP ranges in networks or routing traffic.
- **How to use it?**
  Use CIDR notation (e.g., `192.168.1.0/24`) to define IP blocks.
- **Where is it used?**
  ISPs, enterprise networks, and routing.
  Example: Your home Wi-Fi might use a CIDR block like `192.168.0.0/24`.

---

31. What is the Data Link Layer?

- **What is it?**
  This layer encapsulates data with source and destination MAC addresses and ensures error-free transmission.
- **Why use it?**
  For reliable communication within local networks.
- **When to use it?**
  Always in LAN communication.
- **How to use it?**
  Switches and NICs handle the Data Link Layer automatically.
- **Where is it used?**
  Ethernet, Wi-Fi, and other local network technologies.
  Example: When two devices on the same LAN communicate, their MAC addresses are used for transmission.

---

32. What is NAT?

- **What is it?**
  Network Address Translation (NAT) maps private IP addresses to a single public IP address for internet access.
- **Why use it?**
  To conserve public IP addresses and improve security.
- **When to use it?**
  In private networks accessing the internet.
- **How to use it?**
  Configure NAT on a router or firewall.
- **Where is it used?**
  Home networks, corporate networks.
  Example: Your router uses NAT to allow all your devices to share one public IP address.

---

33. Switch Uses ARP and Default Route

- **Why?**
  Switches use ARP to resolve IP to MAC addresses for forwarding packets within a LAN. A default route may be configured if no other routes exist.

---

34. Why?

Because switches operate at Layer 2 (Data Link Layer), they use ARP to resolve addresses and rely on routing tables for traffic beyond the local network.

---

35. Symmetric vs. Asymmetric Encryption

- **What is it?**
  - **Symmetric Encryption:** The same key is used for encryption and decryption.
  - **Asymmetric Encryption:** Uses a public-private key pair for encryption and decryption.
- **Why use it?**
  Symmetric for speed; Asymmetric for secure key exchange.
- **When to use it?**
  Symmetric: Encrypting large files; Asymmetric: Secure communications.
- **Where is it used?**
  Symmetric: Disk encryption; Asymmetric: SSL/TLS.
  Example: Online banking uses asymmetric encryption for secure login.

---

36. SSL/TLS

- **What is it?**
  Secure Socket Layer (SSL) and Transport Layer Security (TLS) encrypt data between a client and server.
- **Why use it?**
  To secure communication over the internet.
- **When to use it?**
  Always for sensitive data exchange.
- **How to use it?**
  Configure SSL/TLS on web servers.
- **Where is it used?**
  Websites, email servers.
  Example: HTTPS websites use SSL/TLS for encryption.

---

37. Certificate Authority (CA)

- **What is it?**
  A trusted organization that issues digital certificates.
- **Why use it?**
  To authenticate the identity of websites and secure communications.
- **When to use it?**
  For HTTPS, email security, or code signing.
- **Where is it used?**
  E-commerce websites, banking, and APIs.
  Example: Browsers trust certificates issued by a CA to ensure a website is legitimate.

---

38. Microservices

- **What is it?**
  An architectural style where applications are composed of small, independently deployable services.
- **Why use it?**
  For scalability, flexibility, and easier maintenance.
- **When to use it?**
  In modern, cloud-native applications.
- **How to use it?**
  Use APIs for communication between services.
- **Where is it used?**
  E-commerce platforms, streaming services.
  Example: Amazon uses microservices to handle inventory, payments, and recommendations independently.

---

39. Monolithic vs. SOA

- **What is it?**
  - **Monolithic:** Single, unified application.
  - **SOA (Service-Oriented Architecture):** Applications built from loosely coupled services.
- **Why use it?**
  Monolithic for simplicity; SOA for scalability.
- **When to use it?**
  Monolithic: Small projects; SOA: Large, distributed systems.
- **Where is it used?**
  SOA: Enterprise applications.
  Example: A traditional CRM may use a monolithic structure, while modern CRMs use SOA.

---

40. Firewall

- **What is it?**
  A device or software that monitors and controls network traffic based on security rules.
- **Why use it?**
  To protect networks from unauthorized access.
- **When to use it?**
  Always in networks requiring security.
- **How to use it?**
  Configure rules to allow/deny traffic.
- **Where is it used?**
  Home networks, corporate networks.
  Example: Your home router includes a firewall to block malicious traffic.

---

41. Service Discovery

- **What is it?**
  Mechanism for automatically detecting services in a network.
- **Why use it?**
  To enable dynamic scaling and discovery in microservices.
- **When to use it?**
  In distributed systems with many services.
- **How to use it?**
  Use tools like Consul or Kubernetes.
- **Where is it used?**
  Cloud-native applications.
  Example: Kubernetes uses service discovery to route traffic between containers.

---

42. Load Balancer

- **What is it?**
  A device or service that distributes traffic across multiple servers.
- **Why use it?**
  To ensure high availability and scalability.
- **When to use it?**
  For applications with high traffic.
- **How to use it?**
  Configure it to route traffic based on algorithms (e.g., round-robin).
- **Where is it used?**
  Web servers, cloud environments.
  **Example

:** Netflix uses load balancers to distribute traffic across its global server network.

---

43. What is Synchronous and Asynchronous Communication in Microservices?

- **What is it?**
  - **Synchronous Communication:** Services communicate directly and wait for a response before proceeding. Example: HTTP-based APIs.
  - **Asynchronous Communication:** Services communicate indirectly without waiting for an immediate response. Example: Message queues or event-driven systems.
- **Why use it?**
  - **Synchronous:** For real-time, tightly coupled interactions requiring immediate responses.
  - **Asynchronous:** For decoupling services, improving scalability, and handling long-running processes.
- **When to use it?**

    - **Synchronous:** When the client cannot proceed without the response.
    - **Asynchronous:** When decoupling services or handling non-blocking requests.
- **How to use it?**
    - **Synchronous:** Use REST or gRPC APIs.
    - **Asynchronous:** Use message brokers like RabbitMQ, Kafka, or AWS SNS.
- **Where is it used?**
    - **Synchronous:** E-commerce order services verifying payment in real-time.
    - **Asynchronous:** Sending order confirmation emails in the background after a successful purchase.

---

44. What is API, REST API, and API Gateway?

- **What is it?**
    - **API (Application Programming Interface):** A set of rules allowing communication between software applications.
    - **REST API (Representational State Transfer API):** A lightweight API following HTTP standards for CRUD operations.
    - **API Gateway:** A single entry point for managing, routing, and securing multiple APIs in a system.
- **Why use it?**
    - API simplifies integration between systems.
    - REST API is language-agnostic, lightweight, and scalable.
    - API Gateway centralizes API management and improves security.
- **When to use it?**
    - API: Anytime services need to communicate.
    - REST API: For web-based applications.
    - API Gateway: When managing multiple APIs in microservices.
- **How to use it?**
    - Implement REST APIs using frameworks like Flask or Spring Boot.
    - Deploy an API Gateway using tools like AWS API Gateway or Kong.
- **Where is it used?**
    - APIs power integrations like payment gateways (Stripe).
    - REST APIs are used in web applications (social media platforms).
    - API Gateways are used in microservices (Netflix's Zuul).

---

45. What is Asynchronous Publish-Subscribe, Queue Model, and Notification Model?

- **What is it?**
    - **Publish-Subscribe (Pub-Sub):** A messaging model where publishers send messages to topics, and subscribers receive them.
    - **Queue Model:** A messaging model where messages are stored in a queue and consumed by one service at a time.
    - **Notification Model:** A model that informs subscribers of specific events or changes.
- **Why use it?**

    - Pub-Sub: For broadcasting events to multiple services.
    - Queue: For ensuring messages are processed once, even during failures.
    - Notification: For alerting users or systems about changes.
- **When to use it?**
    - Pub-Sub: In real-time event-driven systems.
    - Queue: For job processing (e.g., sending emails).
    - Notification: In user-facing apps (e.g., push notifications).
- **How to use it?**
    - Use tools like Kafka (Pub-Sub), RabbitMQ (Queue), or Firebase Cloud Messaging (Notification).
- **Where is it used?**
    - Pub-Sub: Logging systems, live streaming.
    - Queue: Background job processing.
    - Notification: Mobile app updates.

---

46. What is Multi-Tenant Data?

- **What is it?**
  Multi-tenancy is an architecture where a single instance of a software application serves multiple tenants (clients), with data logically separated.
- **Why use it?**
  To optimize resource usage and reduce infrastructure costs while maintaining data isolation.
- **When to use it?**
  In SaaS (Software as a Service) platforms.
- **How to use it?**
  Design databases with tenant identifiers or create separate schemas for each tenant.
- **Where is it used?**
  Cloud services like Salesforce or Google Workspace.
  Example: A SaaS CRM serves multiple companies but keeps their data isolated.

---

47. What are Logs, Logging, Event Sinking, ELK, Splunk, and Externalized Logs?

- **What is it?**
    - **Logs:** Records of system activities.
    - **Logging:** Process of collecting and storing logs for analysis.
    - **Event Sinking:** Aggregating events for analysis or actions.
    - **ELK (Elasticsearch, Logstash, Kibana):** A logging and visualization stack.
    - **Splunk:** A log analysis and monitoring tool.
    - **Externalized Logs:** Logs stored outside the application environment.
- **Why use it?**
  To monitor system health, debug issues, and analyze trends.
- **When to use it?**
  Always in production environments to ensure system reliability.
- **How to use it?**

    - Use logging libraries (e.g., Log4j).
    - Set up ELK or Splunk for log aggregation and visualization.
    - Store logs in centralized systems (e.g., AWS CloudWatch).
- **Where is it used?**
    - ELK: Monitoring distributed systems.
    - Splunk: Analyzing application logs for errors.
    - Externalized Logs: Microservices in Kubernetes.
      Example: A bank uses ELK to monitor transaction logs and detect anomalies.

NOTES

when we search domain so it will go through ip but arp is responsible for mapping to MAC for final delivering of the message

why structured data - for getting quick or immmediate response that means you trust that it will be done

when the many req coming from same person so session establishment is there

http doesnt know this because its stateless protocol

VPN and its types remote access - office and home

hops are intermiting parties from from source its will travel too many places its hops and after it reaches destination

source -> hops1->2->hopes n-> destination

ARP has mac of all

switch doesnt know the ip or mac address

tcp has 4 layer - application , transport, network and data link

Data link layer it encapsulate the source and destination MAC address

entire packet has size and message has count of packet

packet size is in bytes

we have to divide an network into 3 network so we take 4 network and after that first ip of each network will reserves for the network id and last ip of network broadcast

192.168.1.0/23 --> if we need 23 then we need to minus 32-23=11 machine are required

192.168.1.0/8 --> if we need 23 then we need to minus 32-8=24 machine are required

vpc belong to region and subnet belong to availability zone

NAT MEANS FRAMES ADDRESS GET CHNAGE TO DEVICE ID ADDRESS

SWITCH USE ARP AND IT HAS DEFAULT ROUTE AND ITS USED WHEN ALL THE OPTIONS ARE OVER

FRAMES -> SWITCH -> NATTING -> DESTINATION MSG AND WHEN RETURNING BACK TO SOURCE THEN IT WILL DO DENATTING