1: Networking – Models and protocols

1.1: What is Devops & Why:

It is the combination of Development and Operations. It is used to speed of the cycle without Latency.

1.2: OSI Model:

OSI – Open-Source Interconnection having 7 layers:

1. **Physical Layer:** connects Physically (Hub, Modems, Cables)
   - Bits
2. **Data Link Layer:** data transfer node to node, by making it error free.
   - When the packet came to network it will DLL responsible for delivery to host by MAC Address
3. **Network Layer:** Transfer data from one host to another located in different network
   - It also takes care of routing, noting but identifying the shortest path.
4. **Transport Layer:** In form of segments, like delivering the message completely
   - Order of the Data(Sequence)
5. **Session Layer:** Manages the connection and how long it should be last.
6. **Presentation Layer:** Encryption and formatting.
7. **Application Layer:** Create Data

1.3: Connection Oriented : Checks whether the messages are delivered and checks aby left overs and recover it.

1.4: Windowing : We can control the flow of date between 2 devices.

- First the data is divided in to chunks.
- By giving window size to chunks we can deliver without the ack of the receiver like we can send 1 , 2, 3 and wait for Ack
- If the receiver is fast then we can speed up vice versa.

1.4: Client Server Model: Client will request to the server and server will give the data.

1.5: TCP/IP: (Connection oriented)

TCP: It is the set of rules that will ensure the data is transfer, received securely via internet.(Handshake)

IP: Ensure delivered to correct address. (IP Addresses are unique while in network, like a postal code so that can deliver to the correct destination)

1.6: UDP: (Connection less)

Faster then TCP , but not reliable , not error free

1.7: Protocols (Icmp,Arp)

ICMP (Internet control message protocol): It is network layer protocol used by routes and other devices to send the error messages and status of the network.

ARP (Address Resolution protocol): It is used to map a device's IP address to its physical MAC address in a local network.

1.8: DNS: Domain name system

Helps to convert name to the Ip Address.

For exe… we don't need to remember the port address of the website we have to just know the name.

1.9: Ports: (Switches works on MAC address, while having multiple devices)

It's like door to enter and exit the data from internet.

We are having different ports for diff services.

Mainly:

HTTP: 80

HTTPS: 443
SMPT: 25

1.13: NIC: (Network Interface card):

It's a device used to connect the device to the internet.

1.14: MAC:(Media Access Control):

It's a unique identifier having for every device

1.15: Submarine Cable Maps: Mainly for transferring of the internet with high speed by using optical fiber which are under the sea.

1.17: Nodes:

A node in a network refers to any physical device or point that is capable of sending, receiving, or forwarding data.

1.18: Scaling up and down(Vertical) :

Adding more power to the device. Like when the storage is over we can add more , that is scaling up.

And scaling down is when we don't want it should reduce.

1.19: Scaling in and out(Horizontal):

Adding more devices. Like storage is over we are connecting with another devices, which is scaling out.

And we can reduce the connecting devices which is scaling in.

1.20: Modem/Routers:

Modems: It will convert digital to analog to transmit to more distances and also are not using bits anymore.

- Modems will help to connects network to internet.

Routers: It will distribute the internet to multiple devices.

1.21: Topologies:

Bus, Ring, Tree, Mesh

Bus: Simple only one network multiple devices are connected. (dis: only 1 net so it fails all will, security ADV: simple , high speed)

Ring: 2 devices we can connect to

Tree: Central node connect to the subsequent nodes.

Mesh: Directly connected to every device.

1.22: Socket/ port:

Socket = IP Address + Port Number

The combination of an IP address and a port number allows multiple applications on the same device to use different communication channels without interference.

Example: If you're using a web browser, your browser and the web server communicate using a socket. The web server might use port 80 (for HTTP) or port 443 (for HTTPS), while the browser uses a dynamically assigned port.

Port:

A **port** is a number that identifies a specific service or application on a device. It allows the operating system to differentiate between different types of communication happening on the same device.

1.23: HTTP Method

GET: The GET method is used to request data from a specified resource.

POST: creating a resource. POST requests typically include a request body, which contains the data to be submitted.

PUT: The PUT method is used to update a resource on the server.

DELETE: The DELETE method is used to delete a specified resource from the server.

1.24: peer to peer:

Simply 2 persons are directly talking without need of third person.

And Every device will act both client and server.

1.25: Service Discovery:

Its like phone book , it helps services to find and connect with each other even they change location.

you have two services, **A** and **B**. If **A** needs to talk to **B**, it uses service discovery to find out where **B** is, without needing to know its address beforehand.

Service discovery will have to know which is alive or not.

2. Day-2

- **HTTP Error**

- **Cookies**
- **More about Switch and Ip address**
- **VPN & Types**
- **Checksum**
- **Ip Packet Structure //**
- **ICMP**
- **ARP**
- **Ping**
- **TCP/IP Layer //**
- **Subnetting**
- **VPC**
- **NAT**
- **Symmetric & Asymmetric**
- **Microservices //**

HTTP Errors:

200: ok

404: Not Found

500: Internal Server Error

Cookies:

Cookies are the text files storing some required data for a point of time. So that we can connect faster. For example, google the login credentials will store for a particular point of time.

Switch and Ip address:

Switch:

Here we can connect to multiple devices.

Example Workflow:

Device A (MAC: A1) sends a packet to Device B (MAC: B1).

The switch:

Sees the source MAC (A1): Updates its table (A1 → Port 1).

Checks for destination MAC (B1) in the table: Not found.

Broadcasts the packet to all ports except the one Device A is connected to.

Device B (MAC: B1) receives the broadcasted packet and responds.

The switch:

Sees the response and learns B1 → Port 2.

Updates its MAC address table.

Future packets from A1 to B1 are sent directly to Port 2, with no broadcast.

VPN & Types:

Simply we are encrypting the data using vpn.

For Example there are 2 devices and in one device when we entered the credentials and vpn will encrypt the data then it will send to another device, in another device also there is vpn , vpn knows how to decrypt it so it will do.

types of VPN:

1. Remote access VPN

- **Purpose**: Enables individual users to securely connect to a private network remotely.
- **Use Case**: Employees working from home accessing company resources.
- **Example**: Using a VPN app to connect to a corporate network.

2.site to site VPN:

- **Purpose:** Connects two or more networks in different locations.
- **Use Case:** Linking branch offices to the main office network.

3. SSL VPN:

- **Purpose**: Provides secure remote access through a web browser without needing special software.
- **Use Case**: Securely accessing resources via HTTPS.

4. Cloud VPN:

- **Purpose**: Connects users to cloud services securely.
- **Use Case**: Organizations moving resources to cloud platforms like AWS or Azure.

5. double VPN - two VPN instead of one

Checksum:

Simply checksum is used to check whether there is any manipulation of data or changing of data is done while reaching the destination.

For Example, 1011 is the data to send and in the adding the parity we mentioned it has odd no.of 1's but after the receiving the data by the destination it is showing that 1001 which is even 1's . so we can easily find it is not the same data.

ICMP:

ICMP will used for the error detection noting but.

Icmp first send the ICMP Eco request to the device 2 and waits to ack for the ICMP Eco Reply . Then it will understand that packets to going the correct destination without any manipulation.

Ping:

Ping is the diagnostic tool which used to test the connection between 2 devices using.

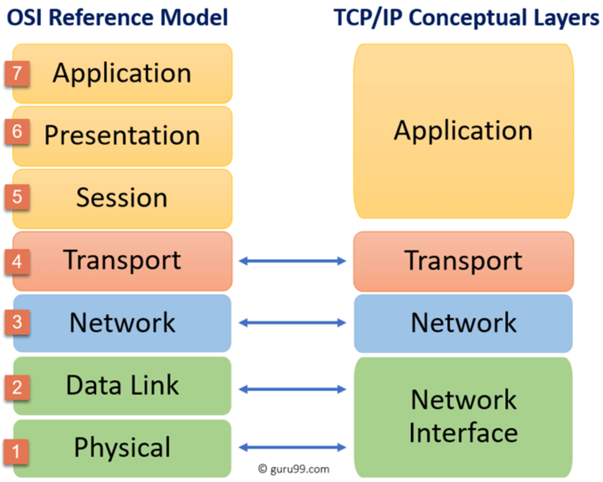
ARP:

It is used to Map the ip address to the Mac address(broadcast). Only first time it will share mac address.

YY because:

Just to connect to another machine we want ip address but **to transmit** the data we want **mac address.**

TCP/IP Model:



1. Application Layer (Layer 4 in TCP/IP)

This layer deals with application-level protocols and provides network services directly to end-users or applications. It enables communication between software applications running on different systems.

2. Transport Layer (Layer 3 in TCP/IP)

Responsible for reliable data transfer between two devices on a network, ensuring error-free and complete data delivery. This layer defines how data is packaged, transmitted, and verified between devices.

3. Internet Layer (Layer 2 in TCP/IP)

This layer is responsible for routing packets of data across different networks and ensures that data reaches its correct destination by addressing and routing it. The most important component of this layer is the IP (Internet Protocol).

4. Network Access Layer (Layer 1 in TCP/IP)

This layer is responsible for the physical transmission of data over the network hardware (such as Ethernet, Wi-Fi, etc.). It deals with the hardware addressing, data link, and physical transmission of data between devices.

Subnetting:

Subnetting is the process of dividing a larger network into smaller, more manageable sub-networks, known as subnets. This is done by borrowing bits from the host portion of an IP address to create additional network bits. Subnetting helps improve network performance, security, and address management.

NAT (Network Address Translation):

Network Address Translation (NAT) is a technique used in networking to modify the source or destination IP addresses in the header of IP packets as they pass through a router or firewall. NAT enables multiple devices on a local network to share a single public IP address for accessing external networks like the internet.

Symmetric and Asymmetric Encryption Symmetric:

symmetric encryption, the same key is used for both encryption and decryption.

Asymmetric: In asymmetric encryption, two different keys are used: a public key and a private key. The public key is used to encrypt data, while the private key is used to decrypt it.

Certificate Authority

Monolithic Architecture:

Monolithic architecture refers to a traditional model of software design where an entire application is built as a single, unified unit.

Everything is packed to one unit so that we can call it as one function.

3 .

Static Nat:

- **What it is**: In **Static NAT**, a specific **private IP address** is always mapped to a specific **public IP address**.
- **How it works**: It creates a **one-to-one** mapping between an internal device and an external IP. The mapping does not change.
- **Example**: If your internal server (with IP 192.168.1.10) always needs to be accessible from the outside, Static NAT will always map 192.168.1.10 to a specific public IP address like 203.0.113.10.

Dynamic Nat:

- **What it is**: In **Dynamic NAT**, the **private IP address** is mapped to a **public IP address** from a **pool** of available public IPs.
- **How it works**: When an internal device needs to communicate with the outside world, it is assigned an available public IP from the pool temporarily.
- **Example**: If your network has devices 192.168.1.10 and 192.168.1.20, and the public IP pool is 203.0.113.10 to 203.0.113.15, one of the private IPs will be mapped to a public IP dynamically, like 192.168.1.10 to 203.0.113.12.

NAT TABLE:

A **NAT Table** (Network Address Translation Table) is used by network devices like **routers** to map **private IP addresses** to **public IP addresses** and vice versa. It helps routers keep track of ongoing connections between devices in a private network and the internet.

Working:

1. When a device from a private network (like 192.168.1.10) wants to access the internet, the router changes its private IP to a public IP (like 203.0.113.5).

2. The router keeps a **record** of the translation in the **NAT table** to ensure that when data comes back from the internet, the router knows which private device it belongs to.
3. The router will replace the **source IP** address in the outgoing data packet with the public IP, and when a response comes back to the router, the NAT table helps the router **reverse** this mapping to send the data to the correct internal device.

Gateway:

ensure only way to come in
increase the security
microservices are not directly connected.

PAT (personal access token):
instead of giving userid and password you can share PAT.

server farm:
collection of physical server that work together to delivery service
people don't need dedicate devices
scalable and reliable.

Ipsec:

secure ip communication trough encryption and Authentication
SA define the encryption algorithm,lifetime of the connection and Authentication method

Firewall:

   - The ACK part in this message acknowledges the receipt of the client's **SYN** packet. This response tells the client that the server is ready to establish the connection.
2. **ACK (Acknowledgment)**:
   - **Client** receives the **SYN-ACK** message and responds with an **ACK** message, confirming that it received the server's ISN.
   - The client also includes its next sequence number (the ISN + 1), which signals that it is ready to begin sending data.
   - After this, the connection is established, and both the client and server can begin exchanging data.

Firewall

A Firewall is a network security device that monitors and filters incoming and outgoing network traffic based on an organization's previously established security policies. At its most basic, a firewall is essentially the barrier that sits between a private internal network and the public Internet. A firewall's main purpose is to allow non-threatening traffic in and to keep dangerous traffic out.

Stateful Firewall: A stateful firewall maintains a table of the state of each active connection, so it can track and validate whether a response is from an established connection. This helps ensure that only legitimate return traffic is allowed.

Stateless Firewall: A stateless firewall does not keep track of connection states. It filters packets based solely on pre-configured rules (such as source/destination IP address or port).

API Gateway

An **API Gateway** is a server or service that acts as an intermediary between a client (such as a web or mobile application) and multiple backend services (often microservices). It provides a unified entry point for accessing different APIs, making it easier for clients to interact with multiple services without needing to know the specifics of each one.

Personal Access Token (PAT)

PATs are used as an alternative to traditional username/password authentication, and they provide a more secure and flexible way of authenticating users or applications.

Server Farm

A **server farm** is a large collection of servers that work together to provide a high level of computing power and storage capacity for various applications, websites, and services. These servers are usually housed in a **data center** and are configured to work as a unified system, often using load balancing, redundancy, and virtualization technologies to ensure scalability, high availability, and reliability.

IPSec (Internet Protocol Security)

IPSec (Internet Protocol Security) is a suite of protocols used to secure IP communications by authenticating and encrypting each IP packet in a communication session. It is commonly used to implement **Virtual Private Networks (VPNs)** and to protect data traffic across untrusted networks, such as the internet.

IPSec operates at the **Network Layer** (Layer 3) of the OSI model and can secure communication between devices such as routers, firewalls, and gateways, as well as between hosts (e.g., computers, servers). Its primary purpose is to ensure data confidentiality, integrity, and authenticity.

Threat, Vulnerability, and Risk in Cybersecurity

Threat: A **threat** refers to any potential danger or event that could exploit a vulnerability and cause harm to a system, network, or data.

Vulnerability: A **vulnerability** is a weakness or flaw in a system, network, application, or process that can be exploited by a threat actor to gain unauthorized access or cause harm.

Risk: Risk is the potential impact of a threat exploiting a vulnerability, expressed in terms of likelihood and consequences.

Reverse Proxy

A **reverse proxy** is a server that sits between client devices and a backend server.

IPV4/IPV6

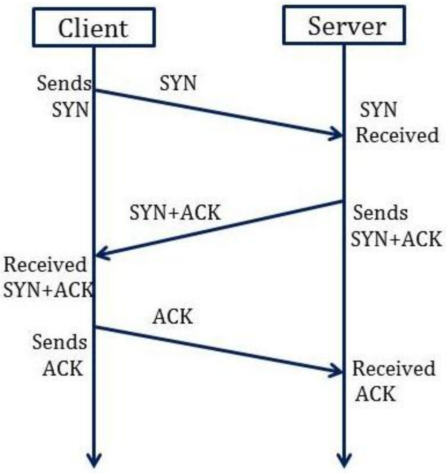
IPv4

- IPv4 is a 32-bit address.
- IPv4 is a numeric address that consists of 4 fields which are separated by dot (.).
- IPv4 has 5 different classes of IP address that includes Class A, Class B, Class C, Class D, and Class E.
- It supports manual and DHCP configuration.

IPv6

- IPv6 is a 128-bit address.
- IPv6 is an alphanumeric address that consists of 8 fields, which are separated by colon.
- IPv6 does not contain classes of IP addresses.
- It supports manual, DHCP, auto-configuration, and renumbering.

3-Way Handshake



- SYN (client to server)
- SYN-ACK (server to client)
- ACK (client to server)

DHCP (Dynamic Host Configuration Protocol)

DHCP (Dynamic Host Configuration Protocol) is a network management protocol used to dynamically assign IP addresses and other network configuration information to devices (known as **clients**) on a network.

Switch and Router

Switch

A **switch** is a network device that connects multiple devices (like computers, printers, and servers) within a local area network (**LAN**) and uses **MAC addresses** to forward data packets between devices.

Router

Definition:

A **router** is a device that connects multiple networks together, typically a local area network (**LAN**) and the internet, and forwards data packets between them using **IP addresses**.

Port forwarding

Port forwarding is a network configuration technique used to allow external devices or services to access specific services or devices within a private internal network. It works by redirecting communication requests from an external IP address (often the public IP address of a router) to a specific device or service on the internal network, using a particular port number.

Hub and switch

Hub:

A **hub** is a basic networking device that connects multiple devices in a network, enabling them to communicate with each other. It operates at **Layer 1** (Physical Layer) of the OSI model and is often referred to as a **"network hub"**.

Switch:

A **switch** is a more advanced network device that connects devices within a local network and uses **MAC addresses** to forward data only to the intended recipient. It operates at **Layer 2** (Data Link Layer) of the OSI model.

VLAN (Virtual Local Area Network)

A **VLAN** (Virtual Local Area Network) is a logical grouping of devices within a physical network, allowing them to communicate as if they are on the same local network, regardless of their actual physical location. VLANs enable network administrators to segment networks into smaller, more manageable parts, improving security, performance, and organization.

//11-01-2025

Data Centers

In a dedicated space with strong security levels, where enterprises or organizations store and share large amounts of data, is known as a data center.

Key components

- **Servers** (compute power)

  Servers are the backbone of any data center, responsible for running applications, managing data, and providing services to clients or users. These can include web servers, database servers, application servers, and storage servers.

- **Storage systems** (data management)

  Data storage systems hold and manage large volumes of data within a data center. This can include both primary data (active data) and backup or archival data.

- **Networking equipment** (communication)

  Networking equipment connects all the components of a data center, ensuring communication between servers, storage systems, and external networks. (Router, Switch, Firewall)

- **Power supply and backup systems** (reliable operation)

  Ensures that the data center operates continuously without interruption due to power failures.

- **Cooling systems** (temperature regulation)

  Servers and other IT equipment generate a lot of heat, so cooling systems are necessary to maintain an optimal operating temperature.

- **Security systems** (physical and cybersecurity)

  Protects the physical and logical infrastructure of the data center from unauthorized access, theft, and cyber threats.

Types of Data centers

On premise Data Center

- The organization owns the data center and is responsible for managing, maintaining, and upgrading the infrastructure.
- The company has complete control over how it is configured and customized to meet specific requirements.
- The data center is located within the organization's own premises.
- High initial investment.

Colocation Data Center

- A third-party data center where businesses can rent space, power, cooling, and network connectivity for their IT infrastructure.
- Lower investment
- Shared infrastructure

Cloud Data Center

- Data centers operated by cloud service providers (e.g., AWS, Google Cloud, Microsoft Azure) that host and manage virtualized resources and services.
- Pay-as-you-go model.
- High scalability

Storage Types

Direct Attached Storage (DAS):

DAS refers to storage devices that are directly attached to a single computer or server, without being connected to a network. It is typically used for personal or small-scale applications.

How it works: DAS storage is directly connected via interfaces such as USB, SATA, SAS, or Thunderbolt.

Local storage directly attached to servers.

Lack of scalability

Potential to lost data.

Device specific

Network Attached Storage (NAS):

NAS is a storage device connected to a network, allowing multiple users and devices to access data over the network. It's often used for centralized file storage and sharing.

How it works: NAS devices typically use Ethernet or Wi-Fi to connect to a local area network (LAN), and they present storage over protocols like SMB/CIFS (Windows), NFS (Linux/Unix), or AFP (Apple).

- File-based storage accessible over a network.
- Moderate scalability and performance
- Examples: Synology NAS, QNAP NAS, WD My Cloud.

Storage Area Networks (SAN):

SAN is a high-speed network that connects storage devices (such as disk arrays) with servers, enabling block-level data access. Unlike NAS, which provides file-level access, SAN provides block-level access to data, typically used for large-scale enterprise applications.

How it works: SANs often use Fibre Channel, iSCSI, or FCoE (Fibre Channel over Ethernet) to connect storage devices and servers. Data is accessed as blocks rather than files.

- High-speed network of storage devices, often used for large-scale enterprise data storage.
- Examples: EMC VMAX, NetApp FAS, Dell PowerMax

TYPES OF STORAGES

PRIMARY STORAGE

Primary storage refers to the storage that is directly accessible by the CPU and is used to store data and instructions that are actively being processed.

RAM (Random access memory)

Description: RAM is the most common type of primary storage and is used to store data and instructions that the CPU needs to access quickly while performing tasks.

Characteristics:

- Fast read and write access.
- Volatile memory (data is lost when power is turned off).
- Temporarily stores data being processed by running applications.

ROM(Read only memory)

ROM is a type of non-volatile memory used in computers and other electronic devices to store permanent data or instructions that are not meant to be altered or modified during normal operation.

Characteristics:

- Non volatile
- Read only.
- Slower access

SECONDARY STORAGE

1. **HDD (Hard Disk Drives)**
2. **SSD (Solid-State Drives)**

OPTICAL DISC

An **optical disc** is a storage medium that uses laser light to read and write data on a reflective surface. Optical discs are widely used for storing data such as software, music, videos, and backups.

RAID LEVELS

RAID 0 (Striping)

- **Configuration**: Data is split into blocks and distributed across multiple disks (at least 2).
- **Redundancy**: No redundancy—if one drive fails, all data is lost.
- **Performance**: High performance, as data is read and written in parallel to multiple drives.
- **Capacity**: Total capacity is the sum of the capacities of all disks.
- **Use Case**: Suitable for applications requiring high performance and where data loss is not critical (e.g., temporary data, non-essential files).

RAID 1 (Mirroring)

- **Configuration**: Data is duplicated (mirrored) across two or more disks.
- **Redundancy**: High redundancy—if one drive fails, the data is still available from the other drive(s).
- **Performance**: Good read performance (because the system can read from multiple disks), but write performance is similar to a single disk.
- **Capacity**: Total capacity is the size of one drive (since data is duplicated).
- **Use Case**: Suitable for situations where data integrity is critical and write performance is not as important (e.g., personal computers, critical data storage).

RAID 5 (Striping with Parity)

- **Configuration**: Data is striped across multiple disks (at least 3), with parity information distributed across all disks.
- **Redundancy**: Moderate redundancy—if one disk fails, the data can be rebuilt using the parity data from the remaining disks.
- **Performance**: Good read performance, but write performance is slower compared to RAID 0 and RAID 1 due to the overhead of parity calculations.
- **Capacity**: Total capacity is the sum of all disks minus one (because one disk is used for parity).
- **Use Case**: Suitable for applications that require a balance of redundancy, performance, and storage capacity (e.g., file servers, databases).

RAID 6 (Striping with Double Parity)

- **Configuration**: Similar to RAID 5 but with **two sets of parity data**, which are stored across different disks (requires at least 4 disks).
- **Redundancy**: High redundancy—can tolerate the failure of **two disks** simultaneously without data loss.
- **Performance**: Read performance is good, but write performance is slower than RAID 5 because of double parity calculations.
- **Capacity**: Total capacity is the sum of all disks minus two (because two disks are used for parity).

- **Use Case**: Suitable for environments where data protection is more important than write performance (e.g., critical business data storage).

RAID 10 (RAID 1+0)

- **Configuration**: Combines the features of RAID 1 and RAID 0. Data is mirrored (RAID 1) and then striped (RAID 0).
- **Redundancy**: High redundancy—can tolerate the failure of one disk per mirrored pair.
- **Performance**: High performance for both read and write operations, as data is striped (RAID 0) and mirrored (RAID 1).
- **Capacity**: Total capacity is the sum of half of the disks (since data is mirrored).
- **Use Case**: Suitable for applications that require both high performance and redundancy (e.g., databases, high-performance servers).

BACKUP AND RECOVERY

A **backup** is the process of creating a duplicate copy of data that can be restored in case the original data is lost, corrupted, or inaccessible.

TYPES OF BACKUPS

Full Backup

- **Definition**: A full backup is a complete copy of all selected data. It copies everything, including all files, folders, and system data (depending on the configuration).
- **How It Works**: Every time a full backup is performed, all data is backed up in its entirety, regardless of whether it has changed since the last backup.

Incremental Backup

- **Definition**: An incremental backup only backs up the data that has changed since the **last backup** (whether it was a full back up or the most recent incremental backup).
- **How It Works**: After an initial full backup, subsequent incremental backups only capture changes made to files since the last backup. This can be multiple times over a period.

Differential Backup

- **Definition**: A differential backup captures all the changes made since the last **full backup**. Unlike incremental backups, differential backups do not rely on previous differential backups, but only on the full backup.

Mirror Backup

- **Definition**: A mirror backup creates an exact copy (or "mirror") of the selected data. It is like a full backup but continuously synchronizes data between the source and the backup location.

BACKUP STRATEGY

The **3-2-1 backup strategy** is a widely recommended method for ensuring robust data protection and recovery. It helps mitigate the risks of data loss from various types of disasters (e.g., hardware failure, cyberattacks, accidental deletions). This strategy involves creating multiple copies of data and storing them in different locations to increase redundancy and resilience.2 Copies stores in two different media types and one in offsite.

BASIC SERVER COMPONENTS

- **MOTHER BOARD:** A **motherboard** is the central printed circuit board (PCB) in a computer that connects and allows communication between various hardware components. It serves as the backbone of the computer, providing essential connections for components like the **CPU, RAM**
- **CPU**
- **RAM**
- **NIC**
- **STORAGE DRIVE**

LOAD BALANCING

- **ROUND ROBIN: Round Robin** is one of the simplest and most used load balancing algorithms. It is a method used to distribute client requests (or traffic) across a group of servers or resources in a circular order.
- **LEAST CONNECTION: Least Connections** is a dynamic load balancing algorithm that directs incoming traffic to the server with the **fewest active connections** at the time of the request. This method is designed to distribute load based on the number of active connections each server is currently handling, aiming to prevent overloading any single server.
- **LEAST RESPONSE TIME: Least Response Time** is a dynamic load balancing algorithm that routes incoming client requests to the server with the **quickest response time** now of the request. The goal of this algorithm is to optimize user experience by sending traffic to the server that is not only least loaded but also currently capable of processing requests the fastest.
- **SOURCE IP HASHING: Source IP Hashing** is a load balancing algorithm that uses the **client's IP address** to determine which server in the pool should handle a particular request. The key idea behind this approach is to ensure that requests from the same client IP address are always directed to the same backend server, creating session persistence (also called sticky sessions).
- **WEIGHTED ROUND ROBIN: Weighted Round Robin (WRR)** is an enhancement of the traditional **Round Robin** load balancing algorithm. In **Weighted Round Robin**, each server in the pool is assigned a **weight** that reflects its capacity or performance. The load balancer distributes incoming requests across the servers, but it gives more requests to servers with higher weights, effectively allowing more powerful servers to handle more traffic.

TYPES OF LOAD BALANCER

HARDWARE LOAD BALANCER

Hardware load balancer is a physical appliance designed specifically for load balancing tasks. It is a dedicated device with specialized hardware and software to handle traffic distribution efficiently.

SOFTWARE LOAD BALANCER

A software load balancer is a software application that runs on general-purpose hardware (such as a server or virtual machine) to perform load balancing tasks. It uses algorithms and protocols to distribute traffic among multiple servers.

Types of Firewalls:

Packet Filtering Firewall:

A packet filtering firewall works by inspecting packets (chunks of data) passing through a network. It evaluates these packets based on predefined rules such as source/destination IP address, port number, and protocol. If a packet matches the set rules, it is allowed to pass through; otherwise, it is discarded.

Stateful Inspection Firewall:

Unlike packet filtering, stateful inspection firewalls keep track of the state of active connections. They evaluate packets not only based on header information but also the context of the entire session, ensuring that packets are part of an established connection.

Application-Level Gateway (Proxy Firewall):

An application-level gateway, or proxy firewall, acts as an intermediary between a user's device and the internet. It operates at the application layer and can filter traffic based on the type of application. For example, it might filter web traffic by checking HTTP requests.

Next-Generation Firewall (NGFW):

NGFWs combine traditional firewall capabilities with additional features like deep packet inspection, intrusion prevention systems (IPS), and application awareness. They are designed to detect and block modern threats, including advanced persistent threats (APTs) and malware.

Common Network Security Threats:

Malware

Malware refers to malicious software that is designed to damage, disrupt, or gain unauthorized access to systems. Common types include viruses, worms, ransomware, and spyware.

Phishing Attack:

Phishing is a social engineering attack where attackers impersonate legitimate entities, such as banks or websites, to deceive individuals into disclosing sensitive information like passwords or credit card details.

Denial of Service (DoS) Attack:

A DoS attack aims to overwhelm a system, network, or service by flooding it with traffic, making it unavailable to legitimate users.

Man-in-the-Middle (MitM) Attack:

In a MitM attack, an attacker intercepts and potentially alters the communication between two parties without their knowledge. This can lead to data theft, impersonation, or information modification.

GIT/GITHUB

--

Command	Description
git init	Initialize a local Git repository
git clone ssh://git@github.com/[username]/[repository-name].git	Create a local copy of a remote repository

Basic Snapshotting

Command	Description
git status	Check status
git add [file-name.txt]	Add a file to the staging area
git add -A	Add all new and changed files to the staging area
git commit -m "[commit message]"	Commit changes
git rm -r [file-name.txt]	Remove a file (or folder)
git remote -v	View the remote repository of the currently working file or directory

Branching & Merging

Command	Description
git branch	List branches (the asterisk denotes the current branch)
git branch -a	List all branches (local and remote)
git branch [branch name]	Create a new branch
git branch -d [branch name]	Delete a branch
git push origin --delete [branch name]	Delete a remote branch
git checkout -b [branch name]	Create a new branch and switch to it
git checkout -b [branch name] origin/[branch name]	Clone a remote branch and switch to it
git branch -m [old branch name] [new branch name]	Rename a local branch

Command	Description
git checkout [branch name]	Switch to a branch
git checkout -	Switch to the branch last checked out
git checkout -- [file-name.txt]	Discard changes to a file
git merge [branch name]	Merge a branch into the active branch
git merge [source branch] [target branch]	Merge a branch into a target branch
git stash	Stash changes in a dirty working directory
git stash clear	Remove all stashed entries
git stash pop	Apply latest stash to working directory

Sharing & Updating Projects

Command	Description
git push origin [branch name]	Push a branch to your remote repository
git push -u origin [branch name]	Push changes to remote repository (and remember the branch)
git push	Push changes to remote repository (remembered branch)
git push origin --delete [branch name]	Delete a remote branch
git pull	Update local repository to the newest commit
git pull origin [branch name]	Pull changes from remote repository
git remote add origin ssh://git@github.com/[username]/[repository-name].git	Add a remote repository
git remote set-url origin ssh://git@github.com/[username]/[repository-name].git	Set a repository's origin branch to SSH

Inspection & Comparison

Command	Description
git log	View changes
git log --summary	View changes (detailed)
git log --oneline	View changes (briefly)
git diff [source branch] [target branch]	Preview changes before merging

Case 1: for suppose we made some changes in file and save it. How to restore? How to come back?

git restore .

Case 2: if we made the changes and save and add it then. How to go back?

git restore .

case 3:

Added changes to staging area didn’t commit after this added more changes to file?

git restore –worktree.