Proxy : It is mediator between client and server .

Forward proxy : it hide the client

Reverse proxy : It hide the server

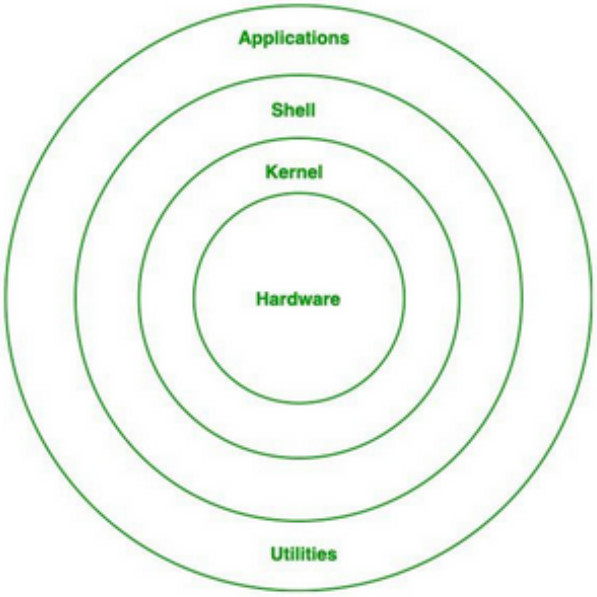
Cat /etc/ssh/sshd_config = ssh config.file

Linux

window	unix	Linux
Less secure	Highly secure	Highly secure
Easy to use	Hard to use	Hard to use
Closed source	Closed source	Open source
Dev by microsoft	AT and T bell lab	Linux torvald
Heavy hardware	Less hardware	Less hardware
GUI	CLI	CLI & GUI

Cat /etc/lsb-release or lsb_release -a = To check os version

Architecture of linux :



Linux operating system architecture

1) **Hardware :**
   - This layer represent the hardware components for eg. CPU, RAM, storage devices, etc .
2) **Kernel :**
   - It is heart of the linux OS . we can say it is the core section of the linux OS .
   - The main functions of the kernel are process management, memory management, file system management, etc .
3) **Shell :**
   - It provide env to user to interact with the kernel .
4) **System utilities or application :**
   - System utilities are the command line tools that preforms various tasks provided by user .

Booting process

1) **BIOS :**

- it is stand for basic input-output system.
- Bois is the first program that is stored only in read-only memory. It check the peripheral device and bootable device is in working condition or not.
- And it hardover control to the first sector of the memory ie. MBR.

2) MBR :

- it is stand for MASTER BOOT RECORD. It is 512 byte .
- 512 byte divided into three parts it gives 446 byte to boot loader , 64 byte to partition table and 2 byte is use for error checking.

3) GRUB :

- stands for Grand Unified Bootloader.
- If you have multiple kernel images installed on your system, you can choose which one to be executed .
- The default boot loader is GRUB2.

4) Kernel :

- Define above in arch .

5) Systemd or init :

- After the kernel is loaded, the first process that starts is either **init** or **systemd** .
- This process is the parent of all other processes and is responsible for managing them. It is assigned **Process ID 1 (PID 1)** .

6) Run lavel :

- When the Linux system is booting up you will see several services starting up like “starting sendmail ..... OK,” These are run-level programs and this are executed from the run-level directory .

- 0: System halt (shutdown)
- 1: Single-user mode (maintenance mode)
- 3: Multi-user mode without GUI
- 5: Multi-user mode with GUI
- 6: Reboot

I/O redirection

> (overwrite the content) : if you want to save the output of the command and replace all the existing content of the file .

Command : ls > pritam.txt

>> (append the content) : if you want to save the output of the command and without replacing all the existing content of the file .

Command : ls >> pritam.txt

2> (error) : if you want to transfer the error of the particular command

Command : ls 2> pritam.txt

&> (error & output) : if you want to transfer the error and output of the particular command .

Command : ls &> pritam.txt

There are three types of file permissions in Linux:

- **Read:** Users open and read files with this permission. (4)

- **Write:** Users can open and modify the files.(2)
- **Execute:** Users can run the file.(1)

File system hierarchy standards (FHS) :

1) **Bin :** normal command such as cd , mkdir , touch .
2) **Sbin :** system administer command such as useradd , adduser , groupadd , etc
3) **Root :** home directory of root user .
4) **Etc :** conf . file
5) **Proc :** provides information about running processes .
6) **Tmp :** it stored the temporary file .
7) **Mnt :** used as a mount point .
8) **Var :** `/var` is a directory in Linux or Unix-like operating systems where system and application log files and variable data are stored .

User administrator

There are three types of users :

1. **Local user :**
   - **Limited Permissions**: Local users cannot perform administrative tasks without **sudo** or root access.
   - Local user id : 1000 - 65000

2. **Root user (super user) :**
   - The root user (also known as the superuser) has full control over the system.
   - The root user always has **UID 0** .

3. **System user :**
    - **System users** are special users created for running services.
    - UID range (1–999) .

/etc/shadow

The /etc/shadow file in Linux is an important system file that stores encrypted passwords and additional information related to user accounts.

1) **Username :** it stored the user name
2) **Encrypted password :**
    - This field contains the **hashed (encrypted) version** of the user's password.
    - If no password is set for the account, this field may be empty or contain special characters like ! or *.
3) **Last password changed :** it indicate the number of days from 1 jan 1970 when the password was changed .
4) **Minimum days :** The minimum number of days a user must wait for changing their password.
5) **Maximum days :** max. Number of days your password is valid .
6) **Warning days :** it give warning your password is getting expired .
7) **Inactive days :** after expired your password for how many days your account will be inactive .
8) **Expired :** This field shows the number of days when the account will expire.
9) **Future use :** for future use .

Commands :

1) **chage -m 2 <username> : Minimum days**
2) **chage -M 2 <username> : max days**
3) **chage -I 2 <username> : Inactive days**

4) **Chage -E “17 may 2065” <username> : Expired**

/etc/passwd

The **/etc/passwd** file in Linux is an essential configuration file that stores information about all user accounts on the system.

username: The name of the user, which is used for logging into the system.

password: This field does not contain the actual password. Instead, it usually contains `x` or `*`, indicating that the password is stored in the more secure `/etc/shadow` file.

UID (User ID): Each user has a unique numeric ID. The root user has a UID of `0`, and local users typically have UIDs of `1000` or higher.

GID (Group ID): This represents the primary group ID of the user. Every user belongs to a group.

full_name: The full name or description of the user. This is optional.

home_directory: The home directory of the user where they can store their personal files (e.g., `/home/username`).

shell: The default shell for the user, which runs when the user logs in (e.g., `/bin/bash`, `/bin/sh`).

Commands :

Changing user name : sudo usermod -l <new_username> <old_username>

Changing password : sudo passwd <user_name>

Changing UID : sudo usermod -u 1002 pratik

Changing GID : sudo usermod -g new_gid username

Changing full_name : sudo usermod -c “<comment>” username

Changing home dir : sudo usermod -d /home/pritam <user_name>

Changing shell : sudo usermod -s /bin/sh <user_name>

/etc/group

1) **Change the Group Name**: sudo groupmod -n newgroup oldgroup
2) **Change the GID**: sudo groupmod -g 1001 examplegroup
3) **To add user in group :** sudo usermod -aG <group_name> <user_name>

HARD LINK & SOFT LINK

Hard link	Soft link
Cmd : ln	Cmd : ln -s
Hard links are faster as compared to soft links .	Soft links are slower.
Can only link to files (not directories) .	Can link to both files and directories.
It shares similar inode numbers .	It shares diff. inode numbers.
File type show (-)	File type show (l)
Link count increase by 1	Does not change

What is an Inode Number?

1. **Definition:** An inode number is a unique identifier assigned to each file or directory in a file system. It helps in tracking the metadata of the file .
2. **Cmd :** ls -i <filename>

Detailed inode information:

Cmd :- stat filename

Isse aapko inode number ke saath aur bhi details milengi.

What is link count :

"Link count" is a file system concept that indicates how many references (hard links) exist for a file or directory .

Default permission for root user file system

File : 644 , directory : 755

Umask for root user : 022

Default permission for local user file system

File : 664 , directory : 775

Umask for local user : 002

Special permission in linux

1) **Suid (set user id) :** all the user get access of executable file .

   Command : chmod u+s <file_name>

2) **Sgid (set group id) :** If setgid is set on a directory, all files created in that directory will attach to the directory's group.

   Command : chmod g+s <dir_name>

3) **Sticky bit :** only owner of the file or directory will delete or rename the file or directory .

   Command : chmod o+t <dir_or_file>

Access control list

- If you want to give permission to the specific user that time we use ACL. Indication : +2
- **Command to see the ACL:** getfacl <file_or_dir>
- **Command to give the ACL :**

  setfacl -m u:<username>:<permission> <file_or_dir>

- **Command to remove the ACL:**

  Setfacl -x u:<user_name> <file_or_dir>

Archiving

Tar (tap archive) : The tar command is used for compressing and archiving files. The term "tar" stands for **"tape archive."** This command is commonly used to combine multiple files into a single archive file .

Command : tar -cvxf archive.tar file1 file2

C = compress , v = verbos = o/p show , f = Specifies the name of the archive file , x = extract

Gzip and gunzip : The `gzip` and `unzip` commands are used for compressing and decompressing files in Linux/Unix environments.

Command : **gzip** <file_name> , **gunzip** <file_name>

Scheduling

Definition : If you

want to perform particular task on certain time that time we use scheduling .

<min> < hr> <date> <month> <week day>

echo "backup.sh" | at 03:00 == only one time

LVM (logical volume management)

LVM (Logical Volume Management) is a powerful tool in Linux operating systems that allows you to manage disk space .

Components of LVM:

- **Physical Volumes (PV)**: These are your actual disk drives, such as hard disks or SSDs.
- **Volume Groups (VG)**: PVs are grouped together to form a Volume Group.
- **Logical Volumes (LV)**: These are the actual storage units that you use. You can format these volumes with file systems (like ext4, xfs, etc.)

Commands :

- **To create partition :** Fdisk /dev/sdb
- **To give file system :** mkfs .ext4 /dev/sdb1
- **blkid :** The `blkid` command in Linux is used to display information about block devices, such as hard drives, partitions, and their associated file systems.
- **df -hT :** The df -hT command in Linux is used to display information about disk space usage.
- **Create Physical Volumes**: pvcreate /dev/sdb1
- **Create a Volume Group**: vgcreate myvg /dev/sdb1
- **Create a Logical Volume**: lvcreate -n mylv -L 10G myvg

What is Swap Memory?

Swap memory is a type of virtual memory that is reserved on your hard disk. When your RAM (Random Access Memory) getting full, the system uses swap memory to provide additional memory .

Process management

- Process： running executable program .

Process Types

- **Foreground Processes:** These are processes that the user starts from the terminal and interact with directly. They occupy the terminal until they finish.
- **Background Processes:** These are processes that run independently of the terminal. You can continue using the terminal while these processes run.

Process States

A process in Linux can be in one of the following states:

- **Running (R):** The process is actively executing.
- **Sleeping (S):** The process is waiting for an event for execution (e.g., I/O completion).
- **Stopped (T):** The process has been stopped, usually by receiving a signal.
- **Zombie (Z):** process killed but still present .

Kill :

Kill all = delete

Signals

- **SIGTERM (15)**: Gracefully asks the process to terminate, giving it a chance to clean up resources .
- **SIGKILL (9)**: Immediately terminates the process without any cleanup.
- **SIGHUP (1)**: Informs the process to reload its configuration files.

fork() : Used to create a new process

exec() : Execute new process.

wait() : wait until process execution

Highest cpu utilization = ps aux --sort=-%cpu | head -n 2

Networking commands

1) **ifconfig / ip :** Display and manipulate route and network interfaces.
2) **Traceroute :** It's used to troubleshoot network connectivity.

   Eg . traceroute google.com

3) **Tracepath :** Similar to traceroute but doesn't require root privileges.
4) **Ping :** To check connectivity .

   Eg . ping google.com

5) **Netstat/ss :** used to display active network connections, routing tables, and listening ports.

   Eg . netstat -tuln

6) **Nslookup :** It's used for domain name resolution.

   Eg. nslookup google.com

7) **Route :** The route command is used to display or modify the IP routing table in Linux.

   Eg. route -n

8) **Curl / wget :** To download a file from internet.

   Eg . curl http://example.com , wget http://example.com

9) **Ip table :** it is like a firewall to us . with the help of this we can filter the incoming and outgoing packet .

Text processing commands

Sed command : This command is very useful for modifying text files through the command line, replacing text, or deleting text.

Command :

- sed -n '3p' <file_name> = it print the 3rd line
- Sed -i 's/cry/three/g' <file_name> = for substitute

AWK : `awk` is a powerful text-processing tool used in Unix-like systems to modify and analyze text files.

- `awk` ek **command-line tool** hai jo **text filtering, processing aur manipulation** ke liye use hota hai. Yeh **column-based data** ko extract, modify aur analyze karne me madad karta hai.

Command :

- awk '{ print $1 }' file.txt

Cut : This command used for to extract part of each line from the file .

Command :

- Cut -d ':' -f 1 <file_name>

Grep : If i want to search specific text or pattern in file that time we use grep command .

-i = ignore case sensitive

Find : If we want to search file or dir.

Syntax : **find <file_or_dir>**

Version control system

There are three type of version control system in github

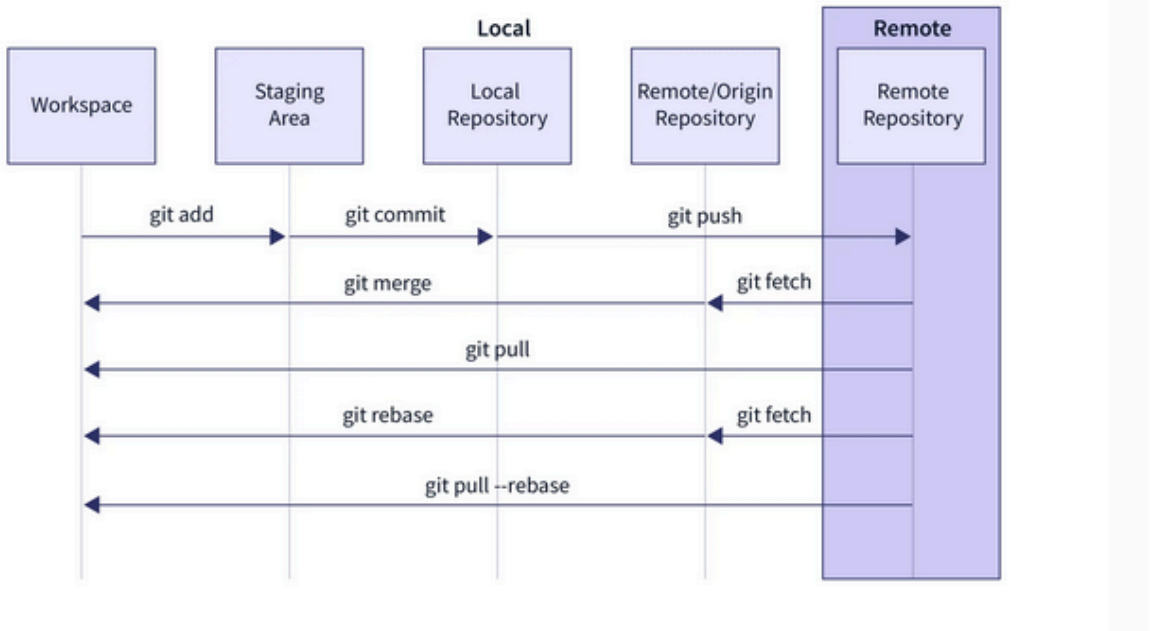
- local version control system
- centralised version control system
- distributed version control system

Definition of vcs : it is a tool which helps to track changes in code.

1) **local version control system :**
   - we stored our code in local version i.e, in our desktop
   - It does not required internet connectivity and we can not shared our data.

2) **centralised version control system :**
   - We stored our data in centralize system
   - It required internet connectivity to access our data.

3) **distributed version control system** :
   - It is the combination of both LVCS and CVCS .



1 - **Readme.md file :** it is special type of file which stored details information of our repo. (.md - markdown).

2 - Git init - it create a empty git repo.

Git init

***3* - Git clone -** To create a local copy of a repository on your machine.

“git clone <repository_url>”

4 - Git add - it move to the file to staging area or The git add command adds new or
changed files from your working directory to the Git staging area.

“Git add <file name>”

5 - Git commit - The "commit" command is used to save your changes to the local repository.

Git commit -m “<comment>”

6 - Git checkout - git-checkout - Switch branches

Git checkout <branch name” or git branch <branch_name>

7 - To create branch - git branch <branch_name>` to create a new branch. Use `git **checkout -b <branch_name>**` to create and switch to it in one step.

***8* - Git merge -** The git merge command combines the changes from one branch into another branch.

git merge —- in ideal situation we do merge at master branch

Git merge <branch name that you want to merge>

***9* - Git rebase —** It is also use for merging but the main difference is that it combine one branch into another branch in linear sequence. **This process is used when you want to create straight clean history of the commit.**

10 - Difference between merge and rebase :

The main difference between git merge and git rebase is that git merge is a way of combining changes from one branch (source branch) into another branch (target branch) where as git rebase is a way of moving the changes from one branch onto another branch.

11) - Git push – git push is used for to upload local repository content to a remote repository.

Git push origin <branch name>

12) git reset - The git reset command is used to undo the changes in your working directory .

Cmd : git reset <file_name>

git reset --hard <commit_id>

Remove commits and completely discard the changes.

git reset --soft <commit_id>

Remove commits but keep the changes in staging area.

13) git revert : It use for undo the changes from remote repo. And create a new commit.

Cmd : git revert <commit_id>.

14) git fetch : The git fetch command is used to bring the latest changes from the remote repository but it does not merge those changes into your current working branch.

15) git pull : The git command is used to bring the latest changes from the remote repository but it merge those changes into your current working branch.

16) Git cherry-pick : Git cherry-pick means you can take a specific commit from one branch and apply it to another branch without merging the entire branch .

17) Branching strategy : Branching strategy is a method used in software development to manage code changes, allowing teams to work on diff . branches

18) fork() :

19) git conflict : Git conflict tab hota hai jab tum do alag branches mein same file ko modify karte ho aur Git ko yeh decide karne mein problem hoti hai ki kaunsa change rakha jaye. Yeh generally merge ya rebase operations ke dauran hota hai jab Git ko do conflicting changes merge karne mein problem ho.

Maven

Maven is a build automation tool used primarily for Java projects. which automate everything related to building the software .

Pom.xml : It is a configuration file of maven in that we define all the setting and dependency , plugin related to the project .

Lifecycle of maven :

- **Validate :** It checks if all necessary information for the project is available or not .
- **Compile :** it convert the human readable lang in machine lavel lang i.e, 0 or 1 . also i check the is there any syntax error or not.
- **Test :** This ensures that your code functionality .
- **Package :** It create the output file in jar , war extension .

- **Verify :** It ensures that the packaged code is correct and passes various quality checks, such as static analysis .
- **Install :** The install phase is responsible for installing the packaged code into the local repository so that it can be used as a dependency in other projects .
- **Deploy :** It makes the code shareable so that other developers or projects can use it .

pom.xml

```
<project>
 <modelVersion>4.0.0</modelVersion>
 <groupId>com.mycompany.app</groupId>
 <artifactId>my-app</artifactId>
 <version>1</version>
</project>
```

3 tear architecture in aws and how we can secure

- Tier-3 architecture ka matlab hai ki application ko teen alag-alag layers me divide kiya jata hai:

1. Presentation Layer (Front-end)
2. Application Layer (Back-end / Business Logic)
3. Data Layer (Database)

1. Presentation Layer Security (Front-end) :

1. **Amazon CloudFront (CDN):** CloudFront static aur dynamic content ko cache karta hai aur DDoS protection ke liye AWS Shield ke sath integrate hota hai.
2. **Web Application Firewall (AWS WAF) :** For extra security and DDOS protection .
3. **HTTPS/SSL/TLS :** AWS Certificate Manager (ACM) se SSL/TLS certificates manage karo

HTTPS force karo taki encrypted communication ho.

2. Application Layer Security (Back-end) :

1. **Amazon API Gateway + AWS WAF :**
   - API Gateway se secure API endpoints banao.
   - WAF rules laga kar malicious traffic ko block karo.
2. **AWS IAM Roles aur Policies :**
   - IAM Roles aur Policies ka sahi istemal karo taki least privilege principle follow ho.
   - IAM Users aur Access Keys ka regular audit karo.
3. **Auto Scaling & Load Balancing :**
   - AWS Auto Scaling use karo taki sudden traffic spikes handle ho sake.
   - Application Load Balancer (ALB) / Network Load Balancer (NLB) traffic securely distribute karta hai.
4. **ECS/EKS Security (Containers Use Kar Rahe Ho Toh) :**
   - Amazon ECS (Elastic Container Service) ya EKS (Elastic Kubernetes Service) ke sath IAM Roles, Security Groups, aur Secrets Manager ka use karo.

- ECR (Elastic Container Registry) Image Scanning enable karo taaki container vulnerabilities detect ho sake.

3. Data Layer Security (Database) :

- Amazon RDS/Aurora Security Best Practices
- Private Subnet me Deploy Karo (Internet se direct access disable karo).
- Security Groups aur NACLs (Network ACLs) ka use karo taki sirf trusted applications hi access kar sake.

-