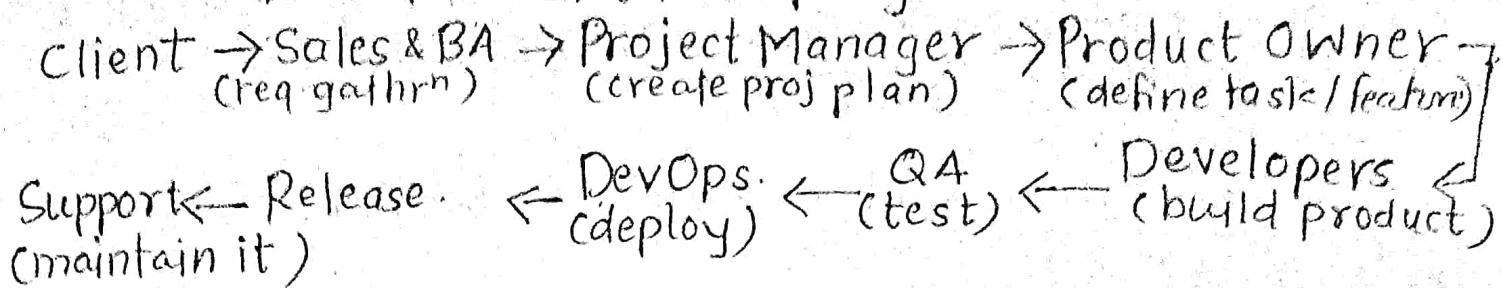


WorkFlow of IT Company.



→ Protocols & Port No.:

Protocols define the rules and format for communication b/w devices. Port no. used to identify spc. applicatn.

Most Common used Port Num:

20 - FTP Data transfer. 21 - FTP command control.

22 - SSH Secure shell 23 - Telnet - Remote login service.

25 - SMTP email routing 53 - DNS Service 80 - HTTP

110 - Post office proto(POP3) 119 - Network News Transfer Prot

123 - Network Time Proto 143 - Intent Access Protocol

161 - Simple Network Manag'n Proto. 194 : Intent Relay chat

443 - HTTPS

• Shell

Prog. that act as interface b/w user & OS. allow user to int.

→ Common Shells: ① Bash (Bourne Again Shell) - widely used.

② Sh, Zsh, Ksh, Fish : other.

Shell Scripting often used to automate task like deployment, monitoring, syst setup, CI/CD pipe.

To check the shell: Whoami :

pwd :

echo \$SHELL :

■ Servers: provide services resource or data to oth. comp.

→ Client Server: comp or devices that req for services.

→ HOST: device with unique IP add.

→ Storage Server → file server → Application Server.

→ DB Server → DNS Server → Authentication Server.

① How Communication Established (Step by step).

② Client Send Req. ③ DNS Resolve hostname.

④ TCP handshake happen. ⑤ Communication starts.

⑥ Firewall, NAT, Router.

Operating Syst: perform task.

e.g. Window, Linux, MacOS, MS-DOS, IBM Cloud.

① Linux: Free & Open Source OS. used on comp, server, kernel based core software. (Linus Torvalds)

Minix (Fcui) → Unix → Linux

→ Linux Distributions.

Desktop → Ubuntu, Linux Mint → easy to use, beginner friendly

Server → Centos, Debian → stable, used in web

DevOps → Alpine, Fedora, → lightweight, flexible

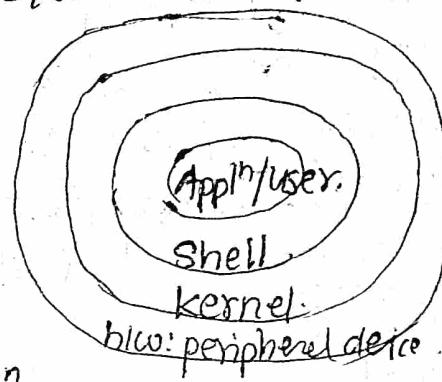
Hacking → Kali, Parrotos. → CyberSecurity ..

Mobile → Android → Built on Linux kernel

Linux is free becoz its open source licensed under GPL developed by global community.

• Kernel: establish communication b/w s/w & h/w.

Linux	Windows
Open source	closed
Command Line: Arg.	Graphical User Interface
High Secure	Less Secure
Multi user	Single user
Customizable	non-customizable
Portable	Non-Portable
Less Update	More Update
Spec. for IT professional	Normal & IT profsn.



Hypervisor: s/w that allow you to run multi VMs on single machine

① Type1 hyper: run directly on h/w eg VMware, KVM, Microsoft Hyper-V

② Type2: Run on top of host os. eg VirtualBox, VMware Workstation

• Virtualization

■ User: identity used to log in & access syst.

User has username, UID, home dir, shell, permissions

① Root User: highest privilege user in linux.

uid=0, home dir /root, full control read/write/execute

② Local User: (Regular user) non-root user created daily
uid: start from 1000+, home dir /home/username

Commands/ Basic

sudo -i start login shell as root user.

User: user is

/etc/passwd : txt file which stores user acc info on syst.
to see info. cat /etc/passwd.

Format = username:x:UID:GID:comment:homedir:shell.

cat /etc/passwd view all user with details.

grep omkar /etc/passwd: see detail of particular user

cut -d: -f1 /etc/passwd. List all username.

• When we create user: add user omkar.

It creates automatically entry in /etc/passwd,
/etc/shadow, /etc/group.

Make home directory /home/omkar.

Set default shell /bin/bash -

Shell

shell is not directory. prog that let you interact with os.

user type command → shell receive it → shell ask os to run it.

it is command-line interpreter. You see the result. ↪

translate human readable command to something that system understand.

To see our current shell - echo \$SHELL

List valid all shells - cat /etc/shells

Touch: Create empty files.

touch fname, touch file1, file2, file3

nano: simple text editor. allow to write, edit, or modify.

echo: print text to terminal or into file.

echo "Hello", echo "User:\$USER" → show curr user.

echo "Info">>file.txt write to the file.

echo "More Info">>> file.txt appends to file.

Cat view file content.

Cat file1.txt > view file, cat file1 file2 > merged.txt

Cat > newfile.txt → Merge to file
Is create new file.

Vim Editor

It's a file editor installed by def on most unix syst.

Vim Modes:

- ① Normal Mode: def mode: move, del, copy, paste.
- ② Insert : for typing text.
- ③ Visual Mode : select text for copying/cutting.
- ④ Command line for saving, quitting searching

vim filename

Navigation in Vim: h - move left, l - right, j - down, k - up
:w - save write., :q - quit, :wq or ZZ Save & quit.
:q! - quit without saving, i - insert mode, Esc - go back to

u: undo last change, Ctrl F r - redo last undo normal mode.

Searching

/word - search fw, ? word - search b/w, n - next, N - previous

Why devops Matters:

Fast delivery, Improve collabor, enhance quality & stability of
Reduce time to resolve issue

Core Principle of DevOps:

- ① CI = developer merge code changes to detect & resolve issue.
- ② CD = code changes on auto test & deploy.
- ③ Automatn
- ④ Monitoring

- Developers: write code that powers applictn. They focus on building & functionality.
- Testers: ensure applictn works as expected. Find & Report bugs.
- Devops Professionals: go b/w devlop & opn. Handle Automatn, deployment & Maintain.

Assesmt

- 1) Opern team handle post deployment bugs & updates.
- 2) Real time Monitoring in DevOps = It ensure appln & infra stabillt
- 3) Primary benefit of automation in Devops = save time by replacing manual task.
- 4) Automating work flow & maintnng syst is prim respons
- 5) Developer write code, to ktr ensur quality, Devopshandl automatn & deployment.

- 8) Planning & Design create blueprint of proj.
- 9) Key challenge in ZT is long delivery cycle due to manual proc
- 8) CI - Automating deployment of code to prod after testing
- 9) Enterprise App in use for business oprn.
- 10) DevOps aim is to collaborate & commn b/w teams.
- 11) CI Merge code changes freqn to detect & fix issue.

Bottlenecks of Devops (dmetric that slow down process)
Zeal of commun in teams, Manual testing & deployment,
delay due to siloed workflow.
Devops address this challenges by fostering collaboration,
automating repetitive task, & implement CI/CD to reduce delay.

Core Tools:

Version Control = Git, Github. Containerizn = Docker, Kubernetes.
CI/CD = Jenkins, GitLab. IaC : Terraform, Cloud Formation.
Monitoring = Prometheus, Grafana, ELK Stack.

* Real time Scenario

DevOps improve Project Oprn:
Retail Company struggled with slow s/w update & freqn down
Soln:
① Implement CI/CD pipeline to automate testing & deployment
② Migrate infra to AWS with Auto scaling
③ Introduced Docker & Kubern for containerized deployment
Outcome: Reduce deployment by 50%, Improve application availability.

Security in DevOps:

DevSecOps Integrn = Embed security at every stage CI/CD.
Tools: tools like Snyk & SonarQube for vulnerability scanning.
Access Managn: Role Based Access Control (RBAC).

Encryption:

Lives in diff sector.

① Mobile devices = Android based on Linux kernel
② Server & Data Center: handling over 70% traffic.
Popular OS Red Hat Enterprise Linux (RHEL) based
& CentOS.
③ Embedded syst & IoT: smart TV, robots.

Linux Foundation (AGL) Automotive Grade Linux based
on driving automotive industry autonomy driving tech.

Super Compt & High Computing (HPC).
more than 90% world's top 500 use Linux as OS.
Cloud Comptg: AWS, Google, Cloud, Microsoft inventory.

Career Paths for Linux:

System Administrator: Manage, Configure & troubleshoot Linux based sys.

① DevOps Eng: focus on automation.

② Linux DevOp: specialized devops app, script

④ Cloud Eng: ensure cloud based Srv.

Why AWS stand out in Cloud Market

① Market Leader: hold large share in cloud market.
wide variety.

② Pay As You Go: flexible pricing.

③ Scalability: can grow with business, from startup to enterprise.

④ Global Reach: vast network.

→ key Services in AWS

① Compute Services: EC2 allows user to run virtual servers.

② Storage Services: S3 (Simple Storage Service)

③ Networking: VPC - Virtual private cloud. private infrastructure.

④ Database: RDS, DynamoDB

Customer success by AWS.

① Netflix ② Airbnb ③ Zomato

AWS provide compute, storage, ML, AI tools while Azure focused on integrating with existing MS tech (e.g. Windows Server, SQL Server, Active Directory).

Google Cloud stands out for AI & ML offer tools like Tensorflow, AutoML & BigQuery.

Why DevOps Growing field?

① Bridging DevOp & Opns ② Fast S/wo Delivery

③ High perfum. outcome ④ Addept across industry.

Future of DevOps

Focus on Secndr: DevSecOps

Evolving tools & platform: serverless computing & edge computing, serverless.

Remote & hybrid work: DevOps professionals will continue to play critical role in enabling remote teams.

Skills & Properties in DevOps:

① Automation tools

- Proficient in Jenkins & Ansible
(Puppet or Chef)

② Cloud Platform

- Properties in AWS, Azure, Google Cloud.

Tech skills

③ Containerization

- knowledge of Docker & Kubernetes

④ Prog knowledge

- Py, Bash or Go for scripting & automation

⑤ CI/CD Pipeline (setting up CI/CD)

[*] soft Skills:

① Collaboration: strong communication to work across teams.

② Problem Solving: troubleshoot complex systems.

③ Adaptability: staying updated with evolving tools & practices.

Operating System: Mediator b/w h/w & U.I.

Provide environment for other appln to run.

two category of OSs.

→ Desktop OS: personal use. e.g. Windows, Ubuntu.

→ Server OS: handle multiuser & process.

e.g. CentOS, Windows Server 2008, Debian.

Impact of OS: ① Commn & Connectivity. ② Banking.

③ Security & Privacy (Firewall) ④ Research & Edu.

Windows: user friendly interface.

UNIX: stability & performance used in server.

LINUX: open source used in enterprise environment.

→ Versions of

① Linux: Ubuntu, Debian, Red Hat, Solaris

② UNIX: AIX, HP-UX, BSD, Tris

Open Source of Linux encourage innovation & collaboration.

Q Why organizⁿ choose server OS like RHEL or Windows over desktop OS:

bcoz Server OS provides advanced feature like resource management, scalability & security essential to handle enterprise level task.



How open source community contributes for growth of Linux
by actively collaborating, identifying bugs, introducing innovation & maintaining continuous updates;
making Linux a robust OS.

Advantages that Linux offered for DevOps. comes to prove
→ open source nature, command line efficiency, scripting capability, Integr. with Docker & kubernetes

Structure of Command Prompt

username@hostname : ~ \$
username = name of logged in user
hostname = system's name.

~ : represent curr working dir.

\$/# \$ normal user.
indicate root (administrator) sessn.

Basic Structure of Linux:

<command> - <options> <arguments>

→ commands: perform specific task.

→ Options: Modify Command's behaviour.

→ Arguments: Specify i/p, file or folder name.

Basic Commands

ttv : display terminal no.

whoami : show curr user.

clear or ctrl+L clear terminal screen.

W : display logged in user details.

cal : display curr month calendar.

cal <year> : spc. year cal.

date : display date & time.

reboot : restart syst.

poweroff : shutdown syst.

shutdown -t 0 : schedule shutdown

* View basic syst info.

host-name, host-name cat /etc/systeminfo, host-name

uname -a : OS & kernel details.

* H/W & Memory Command.

free h : display memory usage in readable form.

lsusb : list connected USB, lscpu : show CPU details.

dmidecode : display all h/w info.

Command	Function
w	shows logged-in users, processes, syst load, & uptime.
who	List logged in user with login time & terminal de
whoami	display effective user executing command.
who am i	Display original login user, even switching user

Memory Usage Info:

free -h this command display RAM & swap memory

-h represent displaying human readable form (MB/GB).

-b: in bytes, -m: in megabytes -g: in gigabytes.

dmidecode hardware info.

Linux command prompt considered more powerful than graphic bcse cmd prompt provide direct control on syst. Thru automation through scripting, adv. syst feat, merge remote sys key task that perform using Linux.

File Mngt, process mngt, user adminstr, sw installn, network config, troubleshoot.

signifn of option. It modify behaviour.

Customization:

new PS1 var used to customize command prompt. e.g.
PS1 var defines appearn of prompt.

e.g. PS1 = [\u@\h :\w]\$] customize prompt to show user name, host name & cur working dir.

Aliases & how they incr productivity.

Aliases are shortcuts for freqn used commands.

eg. alias ll='ls -la' save time by combining listing op into single cmd.

Diffrn b/w uname -a & hostname -l.

uname -a display kernel & os details.

hostname -l provide info related to hostname, OS, Kernel, architech, virtualizn details.

Scenario Based Quesn:

How to modify cmd to display curr time & date dynamically.
 update PS1 $PS1='[\ \u@ \h \w \d\ t]\ \$'$
 you want to frequently check disk usage without typing cmd
 alias diskusage = 'df -h'

Now can you list all hidden files & directions in specific folder along with permissions
 ls -la /path/to/folder.

If user accidentally delete .bashrc file now you recov or recrea
 check if backup (.bashrc.bak) exist & restore it.

Recreate from def template.

cp /etc/skel/.bashrc ~/
 source ~/.bashrc.

Navigating File Syst.

- pwd (print working dir)
- ls (list files) listing content of dir. eg ls, ls -l
- cd (change dir) navigate diff dir.
eg cd /home
- mkdir. make directory eg mkdir folder_nm
- touch create empty file
- ls -a view hidden file

• Viewing & Editing file.

- ① cat concatenate & view file. display content file.
- ② nano commandline text editor. edit file /dir from terminal
simple editing & saving.
- ③ head & tail. view beginning or end of file.
eg head -5 myfile.txt ; tail -5 myfile.txt

■ Copy, Move & Delete Files:

- * cp copy files , cp myfile.txt backup.txt
- * mv move or rename ~~myf~~ eg: mv myfile.txt newfile.txt
- * rm remove files, rm file.txt
- * rmdir. remove empty dir, rmdir emptydir

File & Directory Management

- (a) Creating file in specific directory -> touch file.txt
- (b) creating multi files at diff locn -> touch /root/file.txt
- (c) Creating multiple files at same locn with diffnames.
touch /root/Desktop/file1.txt /etc/data.mp3.
- (d) Create Multi files with seqⁿ numbering.
touch /root/{data.txt, file.txt, demo.mp3}.
- (e) touch /root/file{1..20}.txt

nano To edit file

nano /home/siddhi.txt
it open's file editor we can write in it & to print first
any lines like To print first 5 lines head -5 /home/siddhi.
To print last 2 lines tail -2 /home/siddhi.txt

how to list only dir in given folder. ls -d */

how to create multiple nested dir. mkdir -p parent/child/gran

touch create empty file or update
create empty file or overwrit
timestamp if file exist. file content if exist

how to display only hiddn file in directory ls -d .*

~~rm -rf /~~ Do not use this command it will

forcefully delete all files & directory from root.

deletes': (1) System files (2) User files (3) Configurⁿ files
(4) Kernel modules (5) Bootloader (6) everything from syst.

• Significance of Linux file system hierarchy

① Explain standardize structure make easy to find files & dirs.

② Mention compatibility. help in managing syst across distribⁿ

③ highlight the seprⁿ of user files, syst files, temp files.
for better managⁿ & security.

~ Refers to the home directory cd ~

.. Move up one directory cd ..

. Refers to cur dir. ls .

purpose of imp root-level directories.

- /bin essential binary like ls & cp.
- /sbin system binaries for administrative task.
- /etc Configur'n files.
- /home home directory for user. user specific files save.
- /var variable data like logs & cache.
- /tmp temporary files.
- /usr. user application & file
- /boot store boot loader files, include linux kernel & RAMdisk.
- /dev contain device file. represent h/w config.
- /lib & /lib64 hold essential shared libraries need by binaries in /bin & /sbin. 32 bit lib
- /media mount point for removable media like USB drive &
- /mnt temporary mount point for filesystem, commonly used for CD/DVD during sys admin.
- /opt reserved for addn s/w package installation.
- /proc virtual filesystem provide info about running process & kernel.
- /root home dir for root user.
- /run hold transient files about system start & boot.
- /sysx certain data service provide by system, like cub
- /sys virtual filesystem that expose info about device drivers.

• Vi & Vim.

vi is virtual interface. classic text editor in Unix.

Vim stand for Vi improved. advance version of Vi.

Four Modes in Vim

- ① Insert Mode
- ② Execute (Ex) Mode
- ③ Command Mode
- ④ Visual Mode

vim filename

special command in command mode.

gg move to beginning

<n>gg move to nth line

dd del curr line

<n>dd del curr word <n>dW del n word

yw copy curr line yw copy curr word

cc cut curr line & enter Insert mode
cw cut curr word & enter insert mode.

how to move cursor to 15th line 15gg

how to replace all instances of cloud, with vim in file :%s/oldword/Vim/
you accidentally did line by add how to undo u/g for global for all
dlt next word from cur cursor locn 3dW

In visual mode how to copy block of text J

which command show line number :set nu

save without quit :w open new file in current 0

Create (ExMode) allow user to execute wide commands
for file oprn, txt manipuln, confign.

Use case of Vim in DevOps:

- ① Editing Configuration files
- ② Managing IAC
- ③ Write shell scripts
- ④ Log Analysis
- ⑤ Version Control

:q! force vim to quit without saving any change made to file

:wq saves current changes to file & exist vim

* case sensitive search. :set ignorecase

* revert casesensi search. :set noignorecase

If we want to copy from 1 line to 5 & paste it to the 10 line

:1,5co10

If we want to copy from 1 line to last & paste it after last line

:1,\$co\$

To Search Specific word in file. /word

Replacing the word. :%s/oldword/newword/g

Common Flags with vim command.

vim -R file.txt open file in read only mode

vim -O — open file side by side vertically

—o — open file horizontally

—u NONE file.txt open vim without loading config

:set nu set numbers :set nonu. ~~set relativeno~~

yy copy curr line yyw copy curr word

y copy selected text dd dlt selected text c:cut selected text

p paste selected text

Change Policy

Command	Description
chage -l username	gives info about password & date of change
chage -M 30 username	set max no. of days b/w pass change to 30
chage -m 7 username	→ min 7
chage -I 10 username	account inactive after 10 days
chage -E 2025-07-1 username	Set acc. expiry date

① Modifying Existing User.

① assign customize UID to user

usermod -u 3000 jony

② Assign No login Shell (prevent user from loggin)

usermod -s /sbin/nologin steve

③ Add Comment

usermod -c "devops jony" bruce

④ Lock user password

usermod -L natasha

⑤ Unlock userpass

usermod -U natasha

② Group Management.

Create group - groupadd devops

mp File for groups :

/etc/group

grep groupname /etc/group

to check group details run getent group

avengers:x:1002:---

groupname, passphrass, GID space for user in it

/etc/gshadow

groupname, encrypted pass, group administrator, group members

grep avengers /etc/group

grep avengers /etc/gshadow

gives specific information related to the group the format is given in explanation

Types of Users in Linux

- = ① Root User: unrestricted access to entire system. can perform administrative tasks, including creating & deleting user, modify file & config file. id=0
- ② System User: created by system to manage & run specific processes or services. do not have login shell. limited privilege. id range 1 - 999
- ③ Local User: std. user created for daily tasks & oprn. use own home dir & specific perm.

Id 1000 - 65535

Add new user

adduser tom

useradd tom home dir not create by this command
to create use m.

→ set/change pass. passwd <username>

→ delete user userdel -r username

→ list all user cat /etc/passwd

④ Fmp files for user.

* /etc/passwd.

steve info steve:x:1001:1001:steveusr:/home/steve:167hBc
username password ID GID user info home dir login shell
placeholder

* /etc/shadow

⑤ Add User to group

usermod -aG avengers natasha

gpasswd -a username groupname

⑥ Remove user from group

gpasswd -d natasha avenger

⑦ Add multi user to group (this remove prev users)

gpasswd -M omkar,siddhi,sio,ohi avengers

* Remove user Assign admin to group

gpasswd -A steve avenger

Remove admin

passwd -A 'avengers'

delete grp

groupdel -f avengers

• Managing Users & Permissions:

use for protect sensitive files & resources. Control access levels for diff user & groups. ensure system operate smoothly without performance issues.

Permissions defined as who can read and execute the file.

Three category of read(r), write(w) & execute(x)

① Owner (file creator) ② Group (group of user) ③ Other

Users ① Root User. ② Regular user

③ System user. (cannot login acc. to many process)

→ Using useradd Command:

It adds new entry to /etc/passwd. stores user acc.

System setup home for that user.

options for useradd

-m create home dir. -s specify def shell.

-G assign user to supplementary group. -E set user expiration date.

→ Setting User pass.

• Managing User Groups:

group allow user to share access of file, dir & system res.

* Types of Groups:

Primary Group: assign when user created.

Supplementary group: user is part of shared access.

groupadd create new group.

usermod -G add user to supplementary group.

gpasswd manage group pass

userdel username delete user but retain their files

userdel -r username. delete acc with home dir.

Configuration file.

.bashrc used to customize command line environment

/etc/passwd keep info like user, username, user ID, home

/etc/shadow store encrypted pass

/etc/group manage grp.

/etc/sudoers. defines sudo access

View user acc details cat /etc/passwd

View group info cat /etc/group

new to troubleshoot user login issue /var/log/auth.log
adv of supplement group allows shared resource for specific tests
what happen if user not listed in /etc/sudoers & user in sudo
user denied permission to run command
function of .bashrc file in user home dir makes wide env var
now can enforce pass complexity & expiry by using usermod
system user = user created for running background service
su - without usermod switch previous user
why sudo safer than su it limits elevate permission to specific
command with option.

• Account Locking & Expiration

to lock acc we type sudo passwd -l neha
to unlock -u

{chage} command

used to view & modify password policy for user. gives control over password expiry, min days b/w change, warning period.

chage<options><parameters><username>

* Groups in Linux

com of user /etc/group /etc/gshadow

command to lock user acc sudo passwd -l neha
setting min pass length & chcr minten = 8
command to edit /etc/group sudo nano /etc/group.
unlock acc that prev lock. sudo passwd -u
pass security tip bcoz user often has access to powerful syst
primary group def group assigned to user during acc creation
new to delete group sudo groupdel but
set max no. of day b/w pass change to 60 sudo chage -M 60
command to set update pass sudo passwd rule
rename group from dev to devops sudo groupmod -n devops

chage

Manage pass aging
can change pass expiry
-l = show pass aging
-M Days : set max days.
-m days : set min days.
-w set warning
-2 = maturing after password.

passwd

change user password.
can't change expiry
-d = del pass, -e = expire in mth
-l = lock acc, -u : unlock acc
-s = show pass store

ls -l display file info

Permission	Applied to file	Applied to Direct	Letter	Number
① Read	Open file	list contents of dir	r	4
② Write	change content of file	create, delete & modify permisn	w	2
③ Execute	Run program.	change to dir	x	1

Types of files.

Regular files: normal files eg .txt.bin

Directories: similar to folders, used to organise files

Special files: created by system for specific purpose

User-defined files: eg .txt.py.sh

After running command

ls -l

drwxr-xr--	User	group	4096	Mar 07 12:00	filename
↓	User/	group	↓	month	time
↓	owner	other	link	↓	name
↓	permssn	perm	cont		

Symbolic link: act as a shortcut pointing to another file.

Diff b/w Text File & Script file.

Text File: contain plain text or code but not inherently executable.

Script File: text file with executable command, often starting with shebang to specify interpreter, require execute permission.

• Umask set default permission for files & dir.

• Root User: File permssn = 644

Dir permssn = 755

eg Umask = Full permssn - Default permssn.
022 = 777 - 755

Access Control List: grant permission to user for specific files or directory.

→ Set permission: setfacl -m u:username:rwx filename

→ view permission getfacl filename

→ remove permission: setfacl -x u:username:filename

How to Create Groups & change their permissions

We can change permission by 2 ways.

① Alphabetical Method ② Numerical Method

To change permissions we need to create file where we can change permissions.

touch /home/omkar/test.txt

because of path the owner of file is omkar.

To change permission by Numerical Method we use chmod 777 test.txt.

To check permission just use ls -l /home/omkar/test.txt

ls -l /home/omkar/test.txt.

To change permission by Alphabetical Method

chmod g-w /home/omkar/test.txt.

2 If we don't give path at time of file creation & just create file like

touch abc.txt

by default the owner & group for this file is root.

& If we want to change owner from root to user siddhi use chown siddhi abc.txt.

It will give ownership to siddhi for abc.txt file

& If we want to change group for this file then chgrp linux abc.txt.

check ls -l abc.txt.

Aspect	Hard Link (HL)	Soft Link (SL)
Purpose	Act as backup	Act as shortcut.
Filesize	Same as original file.	Diff from original file.
Mode no.	—	—
Filedeln	hard link remain valid if original file deleted.	soft link become invalid if original file deleted.
directory	hard link to directory are not possible.	soft link to directory are possible.
Link cut.	Afft link cut.	doesnt affect link cut.

→ Hard Link Command

In filename hardlink.

Soft Link File

In -s filename softlink.

Used In command (lnk name)

Links: connect b/w file name & actual data on disk
(shortcut).

We can make multiple shortcut from one file.

eg. In -s multi/dirl/dir2/dir3/myfile.txt myfile
ls command path of file name of
, cat myfile.s
 , cat myfile.txt

In -s multi/dirl/dir2/dir3/myfile.txt
ls
cat myfile.txt

This also make shortcut named as myfile.txt

① Soft link: link will remain if original file
remove or edit.

In -s myfile

If we have multiple or single shortcut & if we
delete one then it will not affect on original file.
We can use shortcut to edit it.

but mode num of shortcut & original file is diff.

② Hard Link: deleting, renaming or removing
original file will not effect link.

→ In myfile myfile-hard-link

The link we made it will work like normal file.

Actual Reason for not losing data if we
delete or rename file or shortcut is that both
shortcut & original file are having similar
mode number.

Use of SL & HL

Archiving & Compression:

Archiving Means combining multi files & directories into single file creating Archive file.

`tar -cvf archive-name.tar file1 file2 dir`
-c-create archive
-v show progress -f : specify
Extract an Archive
`tar -xvf archive-name.tar`
x-extract fw.

→ Compression: To Reduce filesize for easy storage & faster data transfer. combine with file manager.

`gzip`: Fast & efficient

`bzip2`: slow but offer better compreⁿ.

`xz`: provide high compressⁿ ratio take time.

suffix Method for compression decompression.

.gz	<code>gzip</code>	z	<code>gzip</code>	<code>gunzip</code>
.bz2	<code>bzip2</code>	j	<code>bzip2</code>	<code>unbzip2</code>
.xz	<code>xz</code>	J	<code>xz</code>	<code>unxz</code>

Combining Archiving & Compression.

`gzip` for `-cvzf archive-name.tar.gz` file1 file2

`bzip2` for `-cvjf archive-name.tar.bz2` file1 file2

`xz` for `-cvjf archive-name.tar.xz` file1 file2.

How to compress file using `gzip` & view compressⁿ ratio.

compress `gzip` file view ratio: `gzip -l file.gz`.

Use cases ① Backup Solution.

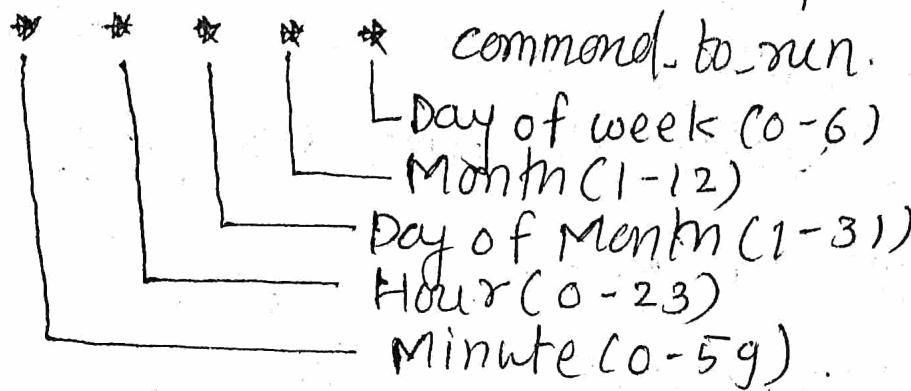
② Reduce storage cost

③ AWS Glacier: store highly euris arch

④ Automated Deploym

Crontab

Automate Repetitive task at specific intervals.



① To edit Cron jobs:

crontab -e

e.g Run script every day at 8:30 AM.

30 8 19 jun * touch ex.bat

② To view curr cronjob

crontab -l

③ To Remove cronjob

crontab -r

Some Common cron jobs:

Run backup daily at midnight. 0 0 * * * ./backup.sh

Run backup at 6AM & 6PM. 0 6,18 * * * ./backup.sh

Run Monitoring every 6 hr. 0 */6 * * * ./monitor.sh

Rm Script ~~daily~~ every 10min. */10 * * * * ./script.sh

Run backup every hour on July 20th 0 * 20 7 * ./backupsh

Steps ① open terminal

② Create shell script:

nano /home/omkar/hello.sh

③ In that write:

echo "Hello omkar I am ran at
\$ (date)" >> /home/omkar/crontab.bat

④ Give permission to execute.

chmod +x /home/omkar/hello.sh

⑤ crontab -e

⑥ Add job -

⑦ To see: cat /home/omkar/crontab.bat

Linux Utility
Used to locate, filter or search efficiently.

① Grep Utility Global Regular Expression Pattern.
grep command search for specific patt.

options

- i case sensitiv patt -I non case sensitiv patt.
- c count matching patt -n print line no. of match.
- v invert patt -m stop after count of match patt.
- o print match patt. -w print exact word.
- A print specified line after matching patt.
- B — | — before — | —
grep "pattern" filename.

eg to find apple/Apple patt in file.

cat >> apple

→ apple apple Apple apple Apple Apple

grep -i apple apple

② Uniq Utility:

remove duplicate line from sorted file.

Great for cleanup redundant data.

syn uniq filename

③ Sort Utility:

Arranges content of file ascending / descending.

syn sort [option] filename

eg sort lines alphabetically

sort file.txt

sort lines numerically

sort -n numbers.txt

④ Find Utility

used to locate files or directories based on various attr. like name, type, size or modifn date.

syn find [path] [options] [expression]

[path] where to start search.

[option] filtering on

[expression] action per form matching result.

to find all .txt files in our dir

find . -name *.txt*

Find by File type (-type)

Find : -type d → dir

find : -type f → regular file

Find by filesize (-size).

find /path -size +10M. (Files large than 10MB)

Find by Modification date (-mtime).

find /path -mtime -7 (modified in last 7 day)

Find files by name.

find /home-name "report"

Find Directories only.

find /var -type d.

Find large files Find less 3 days modified file

find / -size +50M find -mtime -3

for Search "error" in log.txt.

grep error log.txt

grep "error" log.txt

grep -e "error" log.txt

Display line number while using cat command.

cat -n file.txt

Why uniq command require sorted input.
it remove duplicate lines, but only ~~executive~~
work for consecutive duplicate.

eg sort file.txt | uniq

Now to count no. of occurrence.

sort file.txt | uniq -c

Process Management & Optimization:

→ Process simply it is running instrⁿ of program.
running command means creating process
helps OS to manage syst resource.

Types of Processes

① Foreground Process :

you interact with it directly in terminal.
start from shell & keep control on terminal until complete.
eg nano

② Background Process:

Run in background without blocking terminal.
eg wget

③ Daemon Processes:

start during system boot & continue running.
They manage syst service like networking

To Run process in foreground.
simply write name like cmd.

To Run Process in Background.
use & for running process in bg.

eg ping google.com > ping.log &

* Bringing Background process to Foreground.
use command Fg.

eg fg %1

* Terminating process

use kill command along with process ID.

eg kill 1234

Process State .

④ Running (R) : process actively executing.

⑤ sleeping (S) : waiting for event or resource.

⑥ stopped (T) : stopped temporarily.

⑦ zombie (Z) : complete but not clean up properly.

Waiting for parent process to read & exit its steps.

Suspended state : process move from memory to disk

Process is instance of program that executed.

Job Controls Basics.

Job help you to manage multi task within single terminal session.

List active jobs by jobs

To suspend/pause running jobs Ctrl + Z

Restart paused job in background bg %1

• Process & Thread Monagn:

use threads for lightweight task within process effectively ensure optimal performn.

: ps command

gives detailed info about curr running processes

syn ps [options]

-e view all syst process

Real time Monitoring with top:

display real time view syst process & their res...

shows: ① CPU usage ② Memory usage

③ process priority ④ process ZP

eg top.

List Processes sorted by memory usage.

ps aux --sort=-%mem:

↳ Start & Bg process.

(Pause process for 500 sec in bg)

sleep 500&

pgrep search process by name.

nice start process with priority.

renice change priority.

ps aux view running process.

nohup <process name>

Shift curr process to background & you can use terminal.

lsof -p <PID> List file by P2P.

pidof <prochome> find process ZP.

Advanced Management of process & Optimization

* Process Priority:

Linux assign priority to every process to decide which one get CPU time first.

Low priority val means high priority for process.

Adjusting Process priority with nice values.

Nice value determine priority level of process.

ranges from -20 (high pri) to 19 (low pri).
by default most process have nice value 0.

start process with custom nice value.

`nice -n <nice-value> <command>`

eg `nice -n 10 tar -czf backup.tar.gz /home`

tar for command with low pri

④ Change nice value of Running process (renice)

`renice <nice-value> <PID>`

eg `renice -5 1234`

increasing priority of process with PID 1234.

→→→ sending Signals to Processes.

use signal to control or communicate with process.

1) SIGTERM: politely ask process to terminate.

2) SIGKILL: forcibly stop process.

3) SIGHUP: restart process.

4) SIGSTOP/SIGCONT: pause or resumes process.

Syntax: `kill -<signalnum> <PID>`

`kill -9 1234`

④ Finding Process IDs

① by ps command.

`ps aux | grep <process_name>`

② by pgrep command.

`ps -fp $(pgrepnginx)`.

To manage process, you need process ID.

Adjusting priority of long-running Processes.
Let's say we have backup process consuming too much CPU.

Step 1) Find PID using ps or pgrep.

ps aux | grep tar.

2) Lower the priority.

renice 15 <PID>

Visualizing Process tree.

It shows hierarchy of processes - how parent & child processes related.

eg. Syntax ps tree

ps tree -P

Analysing Process Behaviour & Lifecycle.

- ① Analysing specific process.
ps -p <PID> -o pid,ppid,cmd,%cpu,%mem.
- ② Monitoring process in Real time.
top.
- ③ Tracking parent-child Relationship.
pstree.

★ Networking Fundamentals

① The OSI Model

Application Layer: provide service for user applicn.

Presentation Layer: Manages data translⁿ, encrypm & compressn.

Session Layer: establish & maintain conn.

Transport Layer: Divide data into packets.

Network Layer: select best/shortest routing path.

Data Link Layer: convert packet into frames.

Physical Layer: Converts into bit stream (binary) & send via mediums like wifi / ethernet.

② TCP/IP Model:

Application

Transport

Internet Layer.

Network Access Layer.

③ Networking Devices (device that facilitating network)

Hub: broadcast data to all connected device.

Bridge: Filter traffic based on MAC add.

Switch: Intelligent traffic managⁿ with high port count.

Router: Routs data across networks using IP.

④ Topology

- ① Bus topology
- ③ Ring topology
- ② Star topology
- ④ Mesh topology
- ⑤ Hybrid topology.

⑤ Types of Networking

LAN: Local Area = small area

MAN: Metropolitan Area = multi LAN within city

WAN: Wide Area = spans large geographic area

IPv4

32 bit

numeric dot decimal.

Reused & masked

IPv6

128 bit

alphanumeric hexdecimal.
unq add.

Network ID = starting addr of 2P range.

Broadcast ID = ending addr of 2P range.

Subnet Mask = defines internal network borders

Private 2P addr	Public 2P addr
unique to device	unique to entire network
assigned by router	assigned by respective 2P
communicate within network	communicate with internet
need router for connectn	need Modem for conn'

TCP	UPP
Transmiss'n control proto	User Datagram Proto.
Reliable data transmiss'n	Unreliable, Fast but less reliable
connec'tn oriented	connections.
Seq Sequencing.	No seq'n