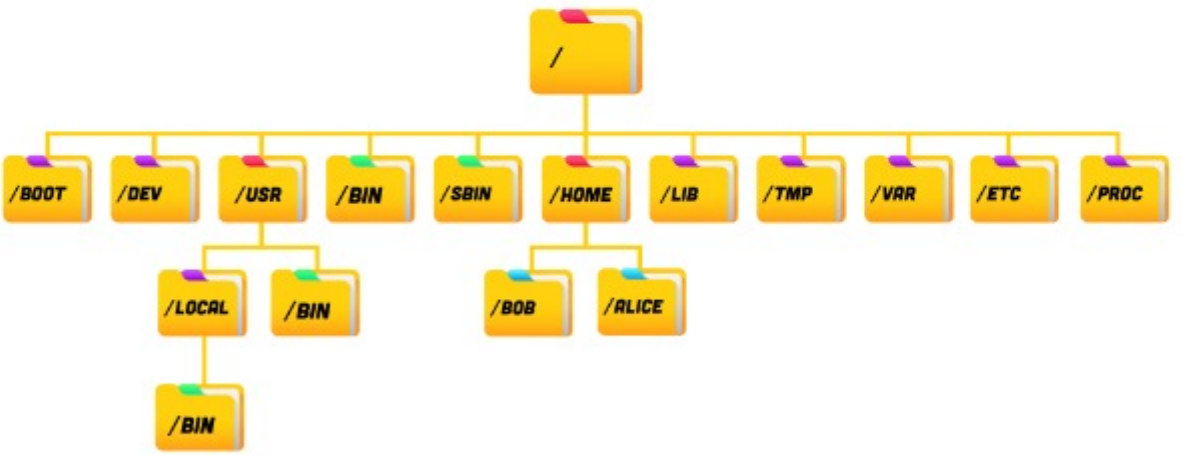
- Linux File System
- Cron Jobs
- Archiving
- Security and System Administration



Linux File System

/ (Root Directory): The top-level directory. Everything on the Linux system is located under this directory.

/bin: (binaries) Contains essential command binaries (executable programs) that are available to all users, such as ls, cp, mv.

/sbin: (system binaries) Contains essential system administration binaries, typically used by the root user, such as fdisk, reboot.

/etc: (et cetera) Contains host-specific system-wide configuration files. Almost all configuration files for services and applications reside here.

/home: Contains the personal directories for regular users. For example, a user named 'john' would have their home directory at /home/john.

/root: The home directory for the root (administrator) user. It's separate from /home to keep root's files isolated.

/usr: (Unix System Resources) Contains read-only user data, including most user utilities and applications, their libraries, and documentation. It's often split into subdirectories like /usr/bin, /usr/lib, /usr/local.

/var: (variable) Stores variable data that changes frequently during system operation, such as log files (/var/log), mail queues (/var/mail), and temporary files that might persist across reboots (/var/tmp).

/tmp: (temporary) Stores temporary files created by the system and users. Contents of this directory are often cleared on system reboot.

/dev: (devices) Contains special files that represent hardware devices (e.g., /dev/sda for the first hard drive, /dev/null).

/proc: (processes) A virtual file system that provides an interface to kernel data structures and information about running processes. It's generated dynamically and doesn't consume disk space.

/boot: Contains files required for the system to boot, including the Linux kernel and boot loader files (like GRUB).

/lib: (libraries) Contains essential shared libraries needed by the binaries in /bin and /sbin.

Archiving

Archiving is the process of **combining multiple files or directories into a single file**. This is especially useful for **backup, compression, and transfer**.

cmds:

gzip filename

zip filename

unzip filename

tar project_backup.tar.gz my_project/

sed:
unix is great os. unix is opensource. unix is free os.
learn operating system.
unix linux which one you choose.
unix is easy to learn.unix is a multiuser os.Learn unix .unix is a powerful.

sed 's/linux/windows/2' filename.txt

Filters and Redirections
command 1 | command 2
output of cmd1 as an input to cmd2
eg.
ls | sort
grep "error"
cat something.log | grep "ERROR"
filters : Filters are commands that take input, process it, and give output.
They're usually used with pipes | or in shell scripts.
grep , sort , sed, head etc
sed : **sed [OPTIONS] 'COMMAND' [INPUTFILE...]**

redirections:

- Create a file named **students.txt** and add the following names using echo and redirection (> and >>):
  - Alice
  - Bob
  - Charlie
  - Charlie
  - david
  - Eve
  - nginx
  - random

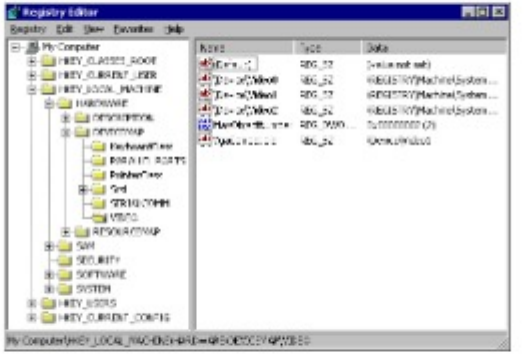
- Find all names that start with C using grep.
- Replace the name Charlie with Chris using sed (without modifying the original file).
- Save the replaced output to a new file called updated_students.txt.
- Display the contents of updated_students.txt.

- What is an OS and how does it work?
- How an OS is constructed?
- Introduction to Virtual Machine
- Tasks of an OS
- How different OSs, like Unix, Linux, Windows and MacOS
- Ubuntu Based Commands

- Package Manager - Installing Software
- Users, Groups & Permissions
- File Ownership & Permissions
- Modifying Permissions
- Filters and Redirections
- ----------
- Working with Vim Editor
- Linux File System
- Cron Jobs
- Archiving
- Security and System Administration

How Installing Software in Windows Works

- .exe/.msi file is used in windows
- .exe – Executable setup file
- .msi – Microsoft Installer Package
- installation wizard opens
- you select every option and accept and files/folders ger created
- installers modify the **Windows Registry**
- an icon for the app is created for you



Installing software in Linux

Unlike windows it uses package managers for installation which is faster and more secure

- Ubuntu/Debian: apt, dpkg
- Red Hat/Fedora: dnf, yum, rpm
- Arch: pacman
- Universal: snap, flatpak

Imp: apt yum

Function of **packages** manager :

- Search software
- Resolve dependencies
- Install, update, or remove software
- Verify package integrity

packages - software in linux comes as packages

package contains:

- Program binaries
- Configuration files
- Dependency information
- Scripts (for post-install actions)

sudo apt update

sudo apt upgrade

- sudo stands for **"superuser do"**.
  It allows a **regular user** to execute commands with **administrator (root) privileges**.
- it is required because linux is secure by design

eg **sudo** apt install python3

Security and System Administration

- linux is :
    - highly **stable, secure, and customizable**
- multi layered approach
    - user and group management
    - file permissions
        - sudo privdiedlges
        - chmod
        - Principle of Least Privilege
    - Firewalls and SSH hardening is given on a network level for security
        - linux has UFW built in uncomplicated firewall
        - firewalld
        - iptables: underlying packet filtering mevhanism that comes in built in linux
        - SSH hardening in linux

cron jobs

- Scheduled Tasks
- Cron is a daemon ; daemon is a program that runs as a background process
- used in system automation for running jobs on a schedule
- crontab cmd manages the cron job
- synatx of crontab:
    - * * * * * command_to_execute

```
* * * * * command_to_execute
| | | | |
| | | |  └── Day of the week (0–7) (Sunday=0 or 7)
| | |  └──────── Month (1–12)
| |  └────────────── Day of month (1–31)
|  └──────────────────── Hour (0–23)
 └────────────────────────── Minute (0–59)
```

backup.sh
0 5 * * * backup.sh

crontab -l to list all the cron job of the current user
crontab -r Remove all cron jobs for the user

Users, Groups & Permissions

User : A **user** is anyone who can log into the Linux system.

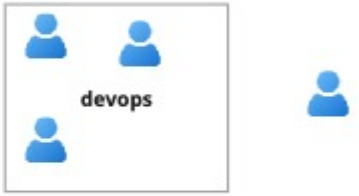
Three types of user in linux:

- root user: Superuser with full access
- normal user: Created by admin / os setup
- system user : For system services linux sometimes creates a system user eg nobody (low privilege sys user creater for procsseors to perform ceertain tasks) it is created by NFS (Network File System)

Groups: A **group** is a collection of users

eg

- you join a company as devops engineer
- another person joins in as developer
- HR , Finance, Leadership etc
-



Permissions: What a user can. can not do.

- **Read (r)**: View contents
- **Write (w)**: Modify contents
- **Execute (x)**: Run file as a program/script

ls -l filename - give you the permissions assosiated with the file

ls -l file.txt

-rw-r--r-- 1 ec2-user ec2-user 0 Jul 23 13:50 file.txt

- category of item- d for directory "-" for file
- rw- permissions assosiated with user
- r-- permissions assosiated with group
- r-- other

rwx

rw-

r--

-w-

--x

write the permission string for :

directory

user has all three rwx

group has rw

others have no permisiosn

drwxrw----

If you see **drwxrw----**, who **can access** the directory?

Only the **owner** can fully access and enter it

Group members can list, read, or modify contents

Others are completely blocked

Read r 4
Write w 2
Execute x 1

777

-rwxrwxrwx

user - 7

group - 7

other - 7

rwx- 4+2+1= 7

rw- 4+2+0= 6

r-- 4++0= 4

--- = 0

754

rwxr-xr--

755 , 644 , 700

---

755 → rwxr-xr-x

644 → rw-r--r--

700 → rwx------