LINUX OPERATING SYSTEM

Linux is an open-source, Unix-based operating system known for its stability, security, and flexibility. It consists of a powerful kernel that manages system resources such as memory, processes, and hardware, along with user-level tools and utilities. Linux supports multi-user and multitasking environments, making it suitable for servers, desktops, cloud platforms, and embedded systems. Due to its open-source nature, Linux allows customization, transparency, and continuous improvement by the global developer community. It is widely used in enterprise systems, DevOps environments, and modern computing infrastructures for its reliability and performance.

WHAT IS LINUX

1.Linux is an open-source operating system, meaning its source code is freely available to view, modify, and distribute.

2. It is Unix-based and follows strong design principles for security, stability, and performance.

3. The Linux kernel is the core component that manages hardware resources such as CPU, memory, storage, and devices.

4. Linux supports multi-user and multitasking, allowing multiple users and processes to run at the same time.

5. It provides both Command Line Interface (CLI) and Graphical User Interface (GUI) for user interaction.

6. Linux is widely used in servers, cloud platforms, supercomputers, and embedded systems due to its reliability.

7. It is known for high security, using permissions, user roles, and access controls to protect the system.

8. Linux is highly customizable, with many distributions (distros) designed for different purposes.

9. Linux has **strong networking support**, making it ideal for servers, networking devices, and distributed systems.

10. It is **cost-effective**, as most Linux distributions are free to use and reduce dependency on licensed software.

MAIN COMPONENTS OF LINUX

- Kernel The core of Linux that manages hardware resources such as CPU, memory, devices, and processes.
- Shell The command-line interface that allows users to interact with the system by executing commands.
- System Libraries Provide standard functions used by applications to communicate with the kernel.
- System Utilities Basic tools and commands (like file management, process control, networking) used to manage the system.
- File System Organizes and stores data in a structured hierarchy (directories and files).
- User Applications Software programs such as text editors, browsers, servers, and development tools that run on Linux.

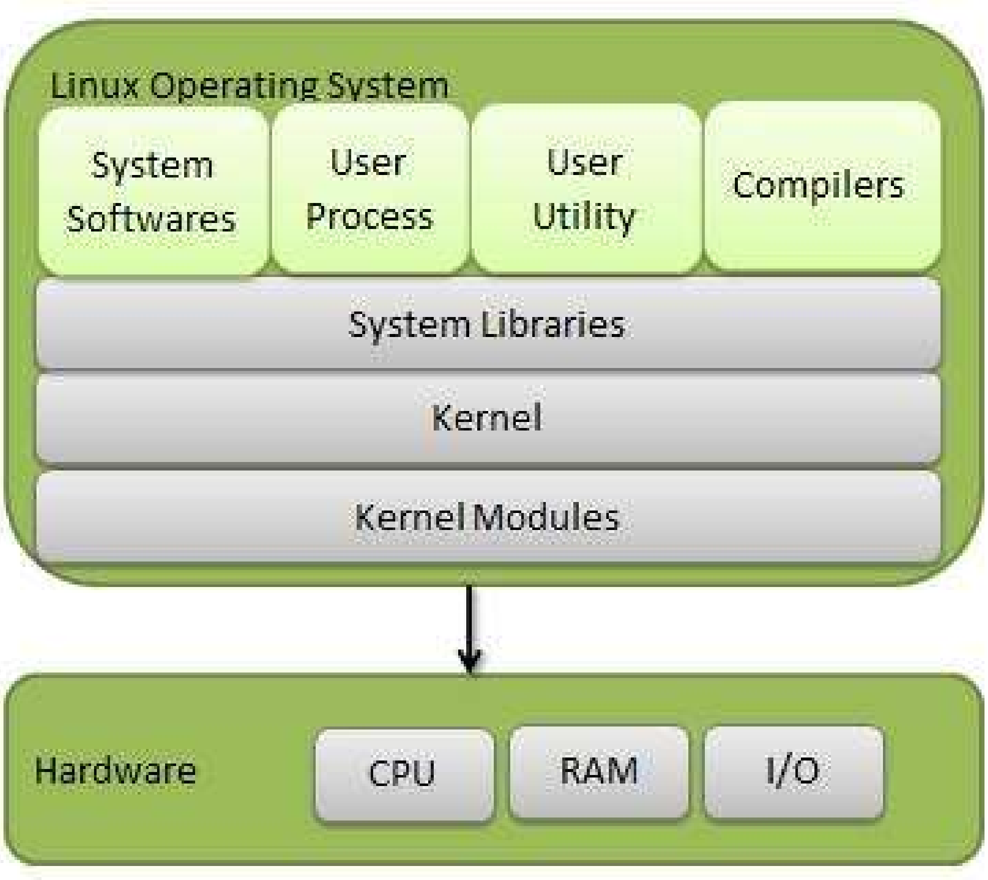

Fig:1 Linux os

ARCHITECTURE OF LINUX

The Linux OS follows a layered architecture, where each layer has a specific role and interacts with the others.

1. Hardware Layer
   This is the lowest layer and includes physical components such as CPU, memory, disk drives, and input/output devices.

2. Kernel
   The kernel is the core of Linux. It manages:

   - Process scheduling
   - Memory management
   - Device drivers
   - File system access
   - System security

   The kernel acts as a bridge between hardware and user programs.

3. System Libraries
   These libraries provide standard functions that allow applications to communicate with the kernel without directly accessing hardware.

4. Shell
   The shell is the user interface of Linux. It interprets user commands and passes them to the kernel for execution (CLI or GUI shells).

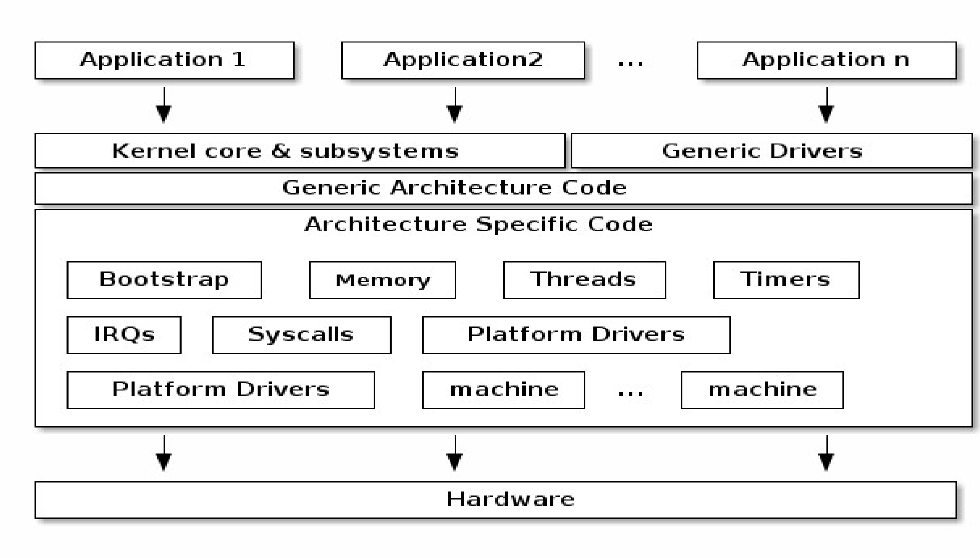

FIG:2 ARCHIETECTURE DIAGRAM OF LINUX

USES OF LINUX

1.**Server Operating System** – Widely used for web servers, database servers, and application servers due to stability and security.

2. **Cloud Computing** – Forms the backbone of cloud platforms and virtual machines.

3. **Software Development** – Preferred by developers for programming, scripting, and open-source development.

4.**Networking** – Used in routers, firewalls, and network management systems.

5. **Embedded Systems** – Powers devices like smart TVs, routers, IoT devices, and set-top boxes.

6. **Desktop Computing** – Used as a free and secure alternative to other desktop operating systems.

7. **Cybersecurity & Ethical Hacking** – Commonly used for penetration testing and security analysis.

8. **Supercomputers** – Most of the world's supercomputers run on Linux.

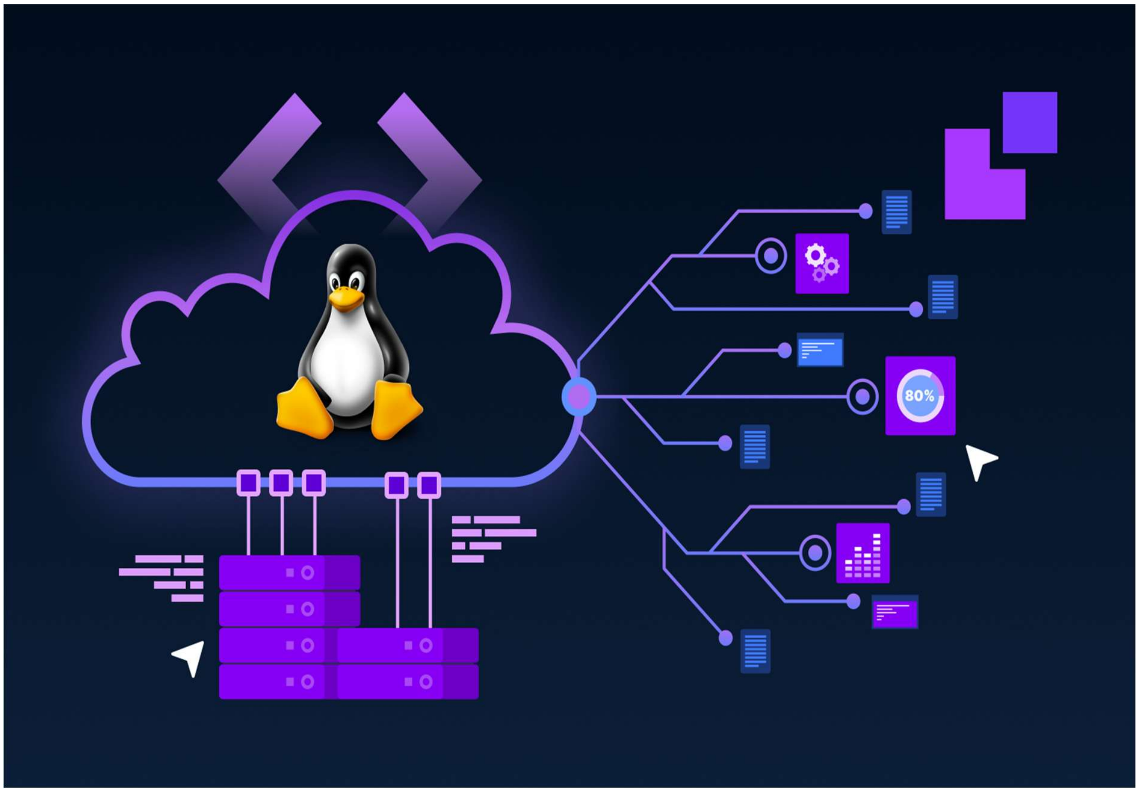

Fig:3 linux

COMMANDS FOR LINUX

SYSTEM COMMANDS:

- uname : used to get type of OS
- uname -r : used to get kernel version of our OS
- uname -a : used to get full info about OS
- clear: this command is used to clear the clear (or) you can use ctrl + l as a short cut
- uptime : used to get since how long our system is in running state
- uptime -p : this will give only time
- hostname: used to get private dns name of our system
- hostname -i : used to get private ip of our system
- hostnamectl set-hostname “swiggy” : used to change hostname
- ip addr : used to get private IP
- ip route : used to get private IP
- ifconfig : used to get private IP
- date : to get todays date
- timedatectl : used to get timezones
- timedatectl set-timezone Asia/Kolkata : used to change Timezone to IST
- who : used to see how many users have been loggin into your system
- whoami : used to see the current user.
- ps : used to see the running processors in system
- kill -9 PID : used to kill a processor
- date : Shows system date and timestamp
- date +”%d” : Prints day of the month (01-31)
- date +”%m” : Prints the month of the year 01-12
- date +”%y” : Prints only the last two digits of Year
- date +”%H” : Prints the hour 00-23
- date +”%M” : Prints the Minute of the hour 00-59

- date +”%S” : Prints the current seconds count in the minute (00-60)
- date +"%D" : Prints Date in MM/DD/YY
- date +”%F” : Prints only the Full date as YYYY-MM-DD
- date +”%A” : Prints the Day of the Week Saturday-Sunday
- date +”%B” : Prints the month between January-December

HARDWARE COMMANDS:

- **lscpu** (or) **cat /proc/cpuinfo** : Displays information about the CPU architecture
- **lsblk -a** : Lists the information about all the block devices attached to the system
- **free** (or) **cat /proc/meminfo**: Displays system memory(RAM) details in KB
- **free -m** : Displays system memory(RAM) details in MB
- **df -h** : Report file system disk space usage in human readable languages

FILE COMMANDS:

- **touch filename** : used to create a file
- **touch aws azure gcp** : used to create multiple files
- **touch linux{1..5}** : this command will create 5 files (linux1,inux2, linux3 .... linux5)
- **rm filename** : remove file with permissions (rm means remove)
- **rm -f filename** : remove file without permissions (-f means forcefully)
- **rm -f aws azure gpc** : remove multiple files without permissions
- **rm -f linux{1..5}** : this command will remove linux files from 1 to 5 without permissions
- **rm -f *** : used to delete all files
- **rm -f *.txt** : used to delete all files with .txt extension
- **rm -f a*** : this command will delete all files which are stared with “a” letter

FOLDER COMMANDS:

- **mkdir foldername** : used to create a folder
- **mkdir git maven jenkins** : used to create multiple folders
- **mkdir docker{1..5}** : used to create 5 folders at the same time
- **rmdir foldername** : used to remove empty folder
- **rmdir git maven jenkins** : will remove multiple empty folders
- **rmdir docker{1..5}** : used to remove 5 docker empty folders
- **rmdir *** : used to remove all empty folder
- **rm -rf *** : used to remove all files & folders and also it is used to remove non empty folders

LIST THE FILES:

- **ll** : used to get list of files
- **ls** : used to get list of files

ll vs ls

ll : will give the full info about files/folders

ls : it will give only file/folder names

CHANGE DIRECTORY:

- **cd foldername** : used to change directory
- **cd** : used to go to root directory
- **cd -** : used to go to previous folder
- **cd ../** : used to go to one folder back
- **cd ../../** : used to go to 2 folders back

DIRECTORY COMMANDS:

- **mkdir folder1/folder2** : this will creates folder2 inside folder1
- **ll folder1** : used to get list of files & folder which are present in folder1
- **touch folder1/aws.txt** : used to create file inside a folder
- **mkdir -p aws/azure/gcp/devops** : used to create parenting folder (folder inside the folder) automatically

COPY COMMANDS:

SYNTAX: cp source destination

command: cp file1 file2

By the above command, the data from file1 copies into file2. But the problem is it will overwrite the data which are present in file2.

To overcome this issue we will use cat command.

cat source_file (file1)>> destination_file (file2)

MOVE COMMANDS:

SYNTAX: mv source destination

COMMAND : mv file1 file2
This is also called renaming a file

ll : used to see the list of files in order (A-Z)

ll -t : used to see the list of the files based on modification/creation time

ll -r : used to see the files in reverse order (Z to A)

ll -a : used to see all files including hidden

CAT COMMANDS:

cat command is used to read the data from a file, it is also used to append the data in a file

cat filename: used to read the data from a file

cat>filename : used to overwrite the data

cat>>filename: used to append the data

NOTE: Cat command is used to append the data, but here the problem is, it is not possible to modify the data. To avoid this issue we can use editor in linux.

There are 2 types of editors in linux

1. Vim editor
2. Nano editor

VIM EDITOR: It is used to edit the files in linux machines, It has 3 modes

1. Command mode
2. Insert mode
3. Save & quit mode

To open any file in vim editor : vim filename (or) vi filename

1. **COMMAND MODE**: this is the default mode in vim editor, It is used to perform some actions like used to copy the data, delete the data and we can make undo and redo the changes as well.

gg : used to go to 1st line of the file

G : used to go to last line of the file

M : used to go to middle of the file

4gg : used to go to 4th line of the file

17gg : used to go to 17th line of the file

:23 : used to go to 23rd line of the file

:set number : used to set numbers of the file

yy : used to copy the line

4yy : used to copy 4 lines from our cursor

p : paste the copied content

10p : paste the copied content 10 times

dd : used to delete the line

5dd : used to delete 5 lines from the cursor

u : used to undo the changes

crtl + r : used to redo the changes.

/word : used to search for a word in a file

?word : used to search for a word in a file

:%s/old/new/ : used to replace

2. INSERT MODE: This mode is used to insert the data or make any modifications in a file.

- To go to insert mode : **i**
- To go back to command mode : **esc**
- To go to the ending of the line : **A**
- To go to starting of this line : **I**
- To create a new line above the cursor : **O**
- To create a new line below the cursor : **o**

DIFFERENCE B/W COMMAND MODE KEYS & INSERT MODE.

If we perform command mode keys we will be in command mode only.

If we perform insert mode keys we will go to insert mode.

3. SAVE & QUIT MODE: This is used to save the data and quit from vim editor

- To save the data - **:w**
- To quit from vim editor - **:q**
- To quit forcefully - **:q!**
- To save & quit at a time - **:wq**
- To save & quit forcefully at a time - **:wq!**

GREP (Global Regular Expression Print):

It is used to search for a word in a file

SYNTAX: grep "word" filename

- **grep "word" filename** : used to search for a word in a file
- **grep -n "word" filename** : it prints the data along with line numbers
- **grep -c "word" filename** : it prints no of occurances of a word
- **grep -i "word" filename** : used to search for a case-sensitive

USER COMMANDS:

- **cat /etc/passwd (or) getent passwd**: used to get list of users
- **useradd username** : used to add user

NOTE: whenever we added any user, then folder will gets created in /home directory

whenever we added any user, then group will also created

userdel username : used to delete user

NOTE: whenever we delete any user, then folder will not gets deleted in /home directory

whenever we delete any user, then group will gets deleted

- **userdel -r username** : used to remove the user along with folder
- **useradd -M username** : used to create a user without folder
- **su - username** : used to switch to another user
- **passwd username** : used to set a password to user

GROUP COMMANDS:

- **cat /etc/group** : used to get list of groups
- **groupadd group_name** : used to add a group
- **groupdel groupname** : used to delete group
- **usermod -a -G group user** : used to add a user in a group

PERMISSION COMMANDS:

- **chown username filename** : used to change the owner of a file
- **chgrp groupname filename** : used to change the group of a file
- **chown username:groupname filenames** : used to change user & group at a time
- **chown username:groupname folder** : used to change user & group at a time to folder
- **chown -R username:groupname folder** : used to change user & group at a time to folder and along with files which are present in inside of a folder

- **chown username:groupname folder/*** : used to change user & group to the files which are present in folder
- **chmod 777 filenames** : used to change the permissions of a file

SEARCH COMMANDS:

It is used to search for a file, To search for a file we have 2 commands

1. find command
2. locate command

FIND COMMANDS:

SYNTAX: find path -name file_name

- **find . -name file** : used to find a file in current directory
- **find /proc/ -name filename** : used to find a file in proc directory
- **find . -type d -name folder** : used to find a folder in current directory
- **find . -type f -name <file1.txt>** : used to find a file in current directory
- **find . -type f -perm 777** : Finds all the files whose permissions are 777 in the current directory
- **find . -type f ! -perm 777** : Finds all the files whose permissions are NOT 777 in the current directory
- **find / -user <username>** : Finds all the files specific user owned in / directory
- **find / -group groupname** : Finds all the files specific group owned in / directory