Department of Computer Science & Information Technology

III Year,V Semester(Batch 2022-2026)

Lab Record Submission of

Linux (Lab)

Subject Code – CSIT-505

Submitted to: Submitted by:

Prof. Nidhi Nigam Manish Makode

0827CI221087

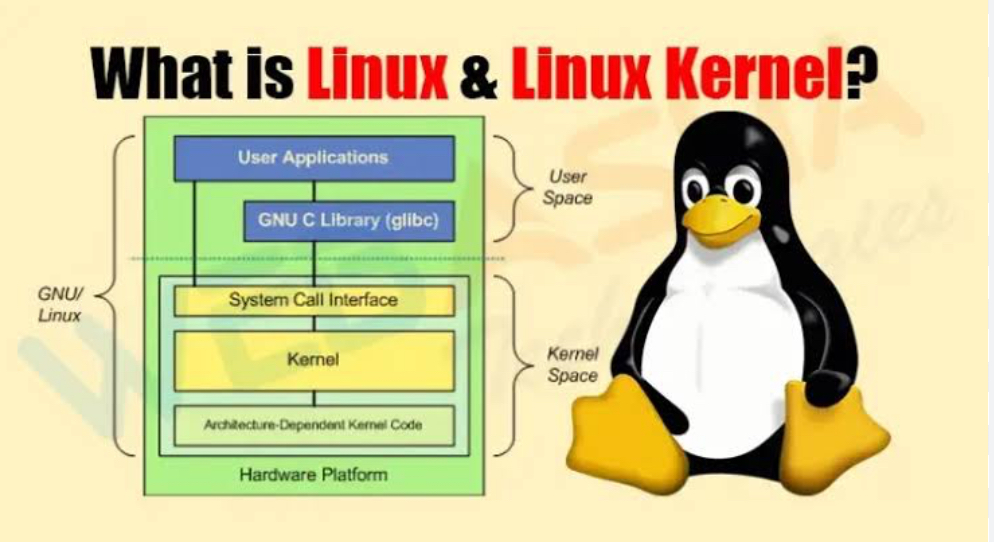
**Linux Basics**

**What is Linux?**

Linux is an open-source, Unix-like operating system kernel that was created by Linus Torvalds in 1991. It's widely used for servers, desktops, smartphones (through Android), and embedded systems due to its flexibility, security, and customization options.

Linux-based operating systems (called distributions or distros) like Ubuntu, Fedora, and Debian include the Linux kernel along with other software packages. Because it's open-source, anyone can modify and distribute Linux, making it popular among developers and enterprises alike. It supports multi-user operations, multitasking, and offers strong system stability and security features. **Key Features of Linux**

1. **Open Source**: The source code of Linux is publicly available, enabling anyone to study, modify, and distribute it. This fosters innovation and collaboration within the community.
2. **Multitasking**: Linux efficiently manages multiple processes, allowing users to run several applications simultaneously without significant performance degradation.
3. **Multiuser Capability**: Multiple users can access the system concurrently, each with their own settings and permissions. This is crucial for servers and shared environments.
4. **Portability**: Linux can run on a wide range of hardware, from personal computers to supercomputers and embedded devices, making it extremely versatile.
5. **Security**: Linux implements a robust permission system, allowing detailed control over who can access and modify files. Regular updates and a strong community contribute to its security resilience.

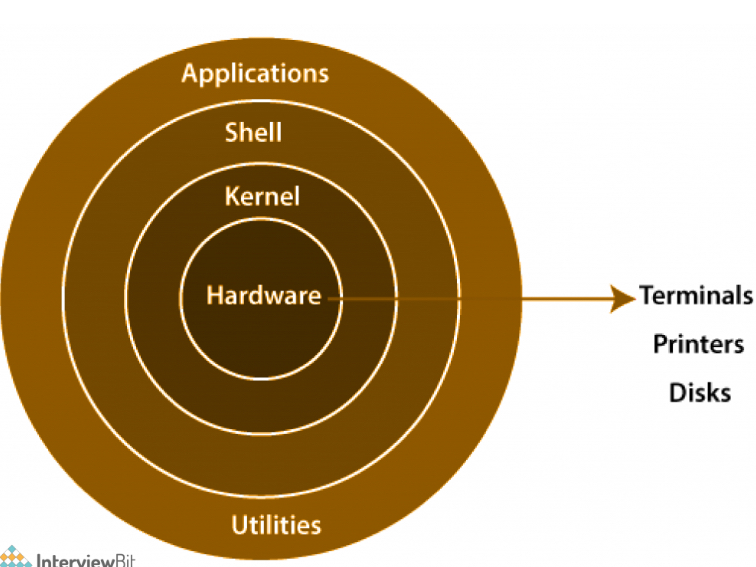


**Basic Commands**

Understanding a few essential commands can greatly enhance your productivity in Linux. Here are some key commands:

* **File and Directory Management**:
  + ls: Lists the contents of a directory, showing files and subdirectories.
  + cd [directory]: Changes the current directory to the specified one.
  + pwd: Displays the full path of the current working directory.
  + mkdir [directory]: Creates a new directory.
  + rm [file]: Removes a file; use rm -r [directory] for recursive removal.
* **File Operations**:
  + cp [source] [destination]: Copies files or directories.
  + mv [source] [destination]: Moves or renames files and directories.
  + cat [file]: Displays the content of a file in the terminal.
* **System Information**:
  + uname -a: Provides comprehensive system information, including kernel version and architecture.
  + top: Shows active processes and system resource usage in real-time.
* **Package Management**:
  + For Debian-based systems (e.g., Ubuntu): Use apt-get install [package] to install software.
  + For Red Hat-based systems (e.g., CentOS): Use yum install [package].

**Linux Architecture**

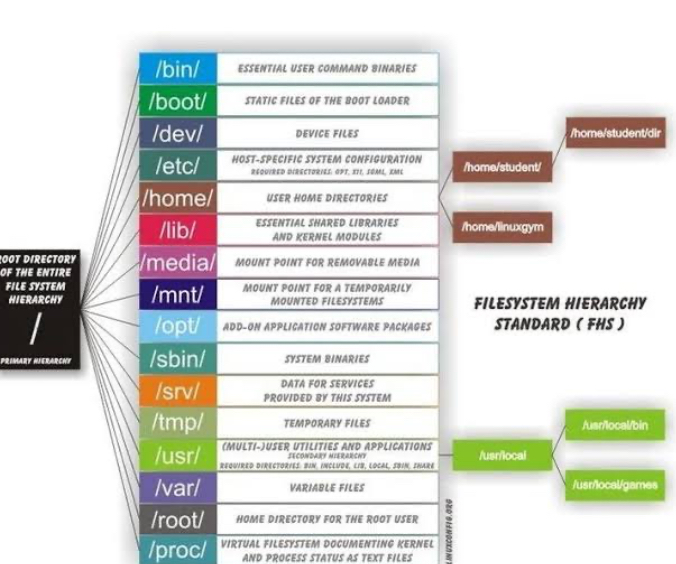


**Components of Linux**

Linux architecture can be divided into several key components:

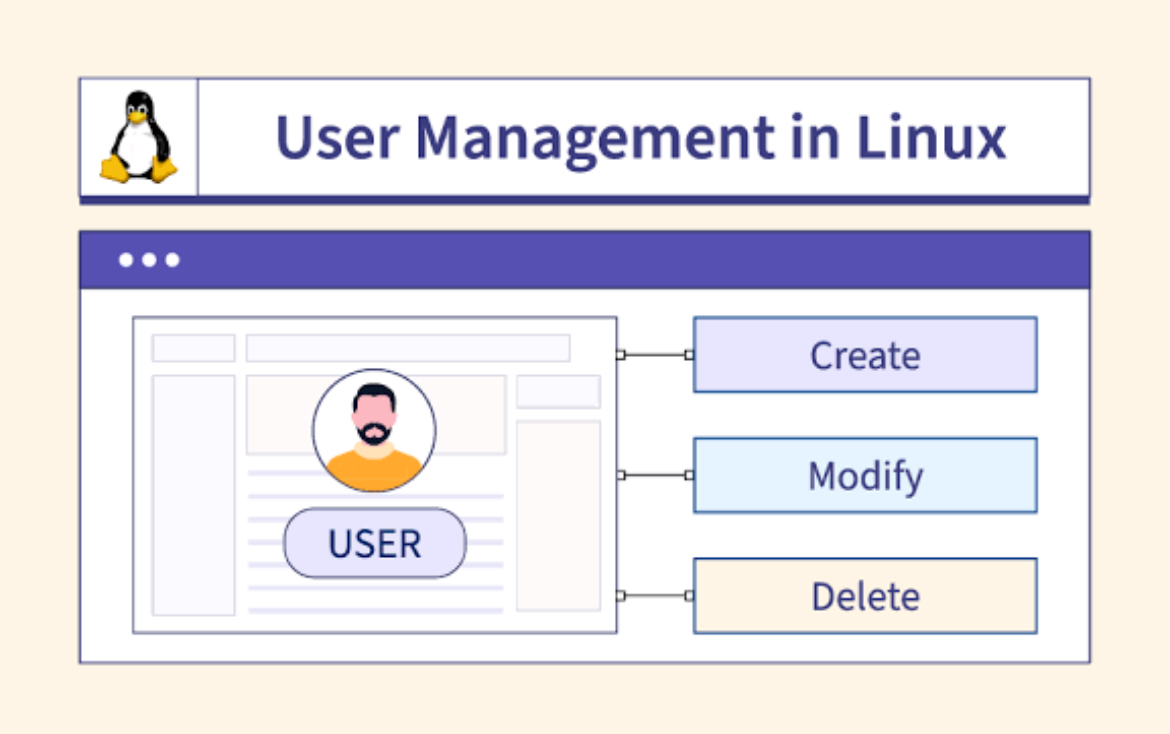
1. **Kernel**: The core of the operating system, responsible for managing hardware resources such as CPU, memory, and I/O devices. The kernel handles system calls, process .
2. management, and memory management. It operates in two modes: user mode (where applications run) and kernel mode (where the kernel runs with full access to the hardware).
3. **System Libraries**: These are essential libraries that provide a standard API for applications. The GNU C Library (glibc) is one of the most widely used libraries, facilitating communication between the software and the kernel.
4. **System Utilities**: These are basic tools that perform fundamental tasks. Common utilities include shell commands (like bash) and system monitoring tools. They provide essential functionalities for managing the system.

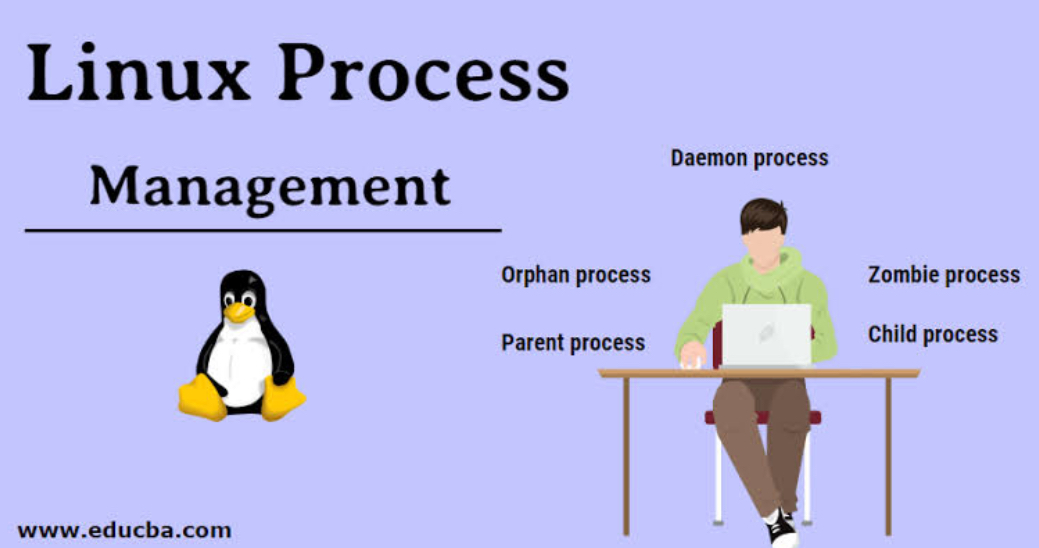
**Linux File System Hierarchy**



The Linux file system follows a standard directory structure, known as the Filesystem Hierarchy Standard (FHS). Key directories include:

* **/** (Root): The top-level directory in the Linux file system hierarchy.
* **/bin**: Contains essential command binaries (executables) required for the system's basic functioning.
* **/etc**: Holds configuration files that control the behavior of the system and applications.
* **/home**: Contains personal directories for users, where they can store their files and settings.
* **/lib**: Houses shared libraries essential for system operation, including modules used by the kernel.
* **/usr**: Contains user-related programs and data, including applications and utilities.
* **/var**: Stores variable data files, such as log files and databases, that change in size and content.



**Process Management**

Linux efficiently manages processes, which are instances of running programs. Each process has a unique Process ID (PID) and can be in one of several states:

* **Running**: Actively executing on the CPU.
* **Sleeping**: Waiting for an event (like I/O completion).
* **Stopped**: Temporarily halted, often due to user intervention.
* **Zombie**: A terminated process that still has an entry in the process table, awaiting retrieval by its parent process.

Common commands for process management include:

* ps: Displays information about currently running processes.
* top: Provides real-time monitoring of system processes and resource usage.
* kill [PID]: Terminates a process by its PID.

**User Management**

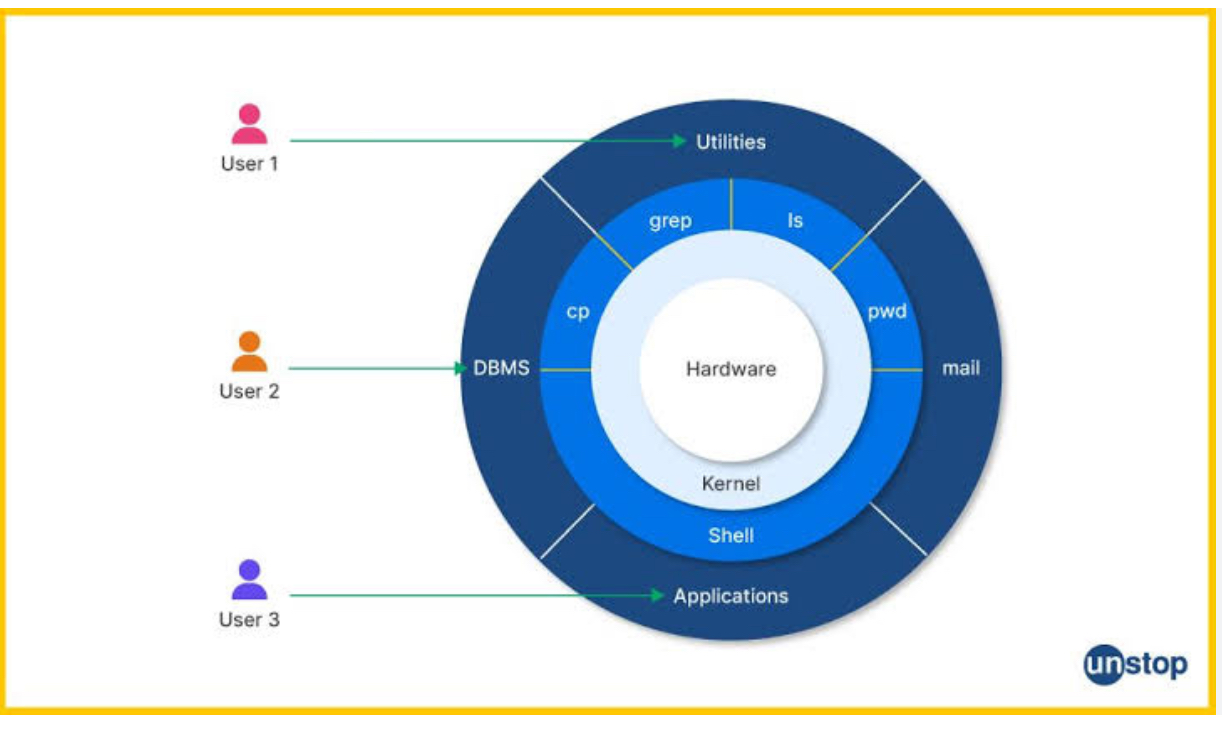
Linux employs a robust user management system, where each user has a unique User ID (UID). Users can belong to one or more groups, facilitating permission management. Key commands include:

* adduser [username]: Creates a new user.
* deluser [username]: Removes a user from the system.
* passwd [username]: Changes the password for a user.

**Shell**

The shell is a command-line interface that allows users to interact with the system. Common shells include:

* **Bash (Bourne Again Shell)**: The default shell for many Linux distributions, known for its scripting capabilities.**Zsh (Z Shell)**: Offers advanced features and better customization.
* **Fish (Friendly Interactive Shell)**: Focused on user-friendliness and interactive use.



Shell scripting allows users to automate tasks by writing scripts that execute a series of commands. Scripts typically have a .sh extension and can perform complex operations using loops, conditionals, and functions.