

A **Linux distribution (distro)** is an operating system built from the Linux kernel, along with supporting software, libraries, and tools. Linux distributions are used in everything from desktops and laptops to servers, mobile devices, and embedded systems.

🔥 Popular Linux Distributions

Here are some of the most well-known distros, each catering to specific use cases:

1. Ubuntu

- . **Base:** Debian
- . **Target users:** Beginners, general desktop users, developers
- . **Package manager:** APT (Debian-based)
- . **Desktop environment:** GNOME (by default)
- . **Use case:** Desktop, server, and cloud environments

- . **Why use it:** User-friendly, large community, great hardware support

2. Debian

- . **Base:** Independent (parent of Ubuntu)
- . **Target users:** Advanced users, developers, servers
- . **Package manager:** APT
- . **Desktop environment:** Multiple (GNOME, KDE, Xfce, etc.)
- . **Use case:** Servers, development, stable systems
- . **Why use it:** Stability, reliability, and security

3. Fedora

- . **Base:** Independent (upstream for Red Hat Enterprise Linux)
- . **Target users:** Developers, cutting-edge enthusiasts
- . **Package manager:** DNF (RPM-based)

- . **Desktop environment:** GNOME (default)

NOTE-What is GNOME (GNU Network Object Model Environment)? GNOME (GNU Network Object Model Environment) is a graphical user interface (GUI) and set of computer desktop applications for Linux operating system (OS) users. It's intended to make a Linux OS easy to use for nonprogrammers.

- . **Use case:** Development, workstations, testing the latest Linux tech
- . **Why use it:** Cutting-edge software, first to introduce new Linux features

4. Arch Linux

- . **Base:** Independent
- . **Target users:** Advanced users
- . **Package manager:** Pacman
- . **Desktop environment:** None by default (users choose)

- . **Use case:** Power users.
- . **Why use it:** control, rolling release model

5. Linux Mint

- . **Base:** Ubuntu/Debian
- . **Target users:** Beginners, Windows users switching to Linux
- . **Package manager:** APT
- . **Desktop environment:** Cinnamon (default), MATE, Xfce
- . **Use case:** Desktop systems for beginners
- . **Why use it:** User-friendly, Windows-like interface, pre-installed codecs

6. CentOS (Now Rocky Linux / AlmaLinux)

- . **Base:** Red Hat Enterprise Linux (RHEL)
- . **Target users:** Servers, enterprises
- . **Package manager:** YUM/DNF (RPM-based)

- . **Desktop environment:** Minimal (not focused on desktop use)
- . **Use case:** Enterprise servers, stable server OS
- . **Why use it:** Stability, long-term support

7. Manjaro

- . **Base:** Arch Linux
- . **Target users:** Beginners, desktop users
- . **Package manager:** Pacman
- . **Desktop environment:** XFCE, KDE Plasma, GNOME (multiple flavors)
- . **Use case:** Desktop systems with the power of Arch but with an easier setup
- . **Why use it:** Easier than Arch, rolling release model, pre-configured system

8. Red Hat Enterprise Linux (RHEL)

- . **Base:** Independent (parent of CentOS, Rocky, AlmaLinux)

- . **Target users:** Enterprises, paid support customers
- . **Package manager:** YUM/DNF (RPM-based)
- . **Desktop environment:** Not the focus (server-focused)
- . **Use case:** Enterprise, production servers, cloud environments
- . **Why use it:** Stability, enterprise-grade support, certification

9. Pop!_OS

- . **Base:** Ubuntu
- . **Target users:** Gamers, developers, hardware enthusiasts
- . **Package manager:** APT
- . **Desktop environment:** COSMIC (based on GNOME)
- . **Use case:** Desktop users, gamers, developers

- . **Why use it:** Focus on gaming, hardware optimization (from System76)

10. Kali Linux

- . **Base:** Debian
- . **Target users:** Penetration testers, security researchers
- . **Package manager:** APT
- . **Desktop environment:** XFCE (default), KDE, GNOME, etc.
- . **Use case:** Ethical hacking, cybersecurity testing
- . **Why use it:** Pre-installed with pentesting tools like Metasploit, Wireshark, etc.

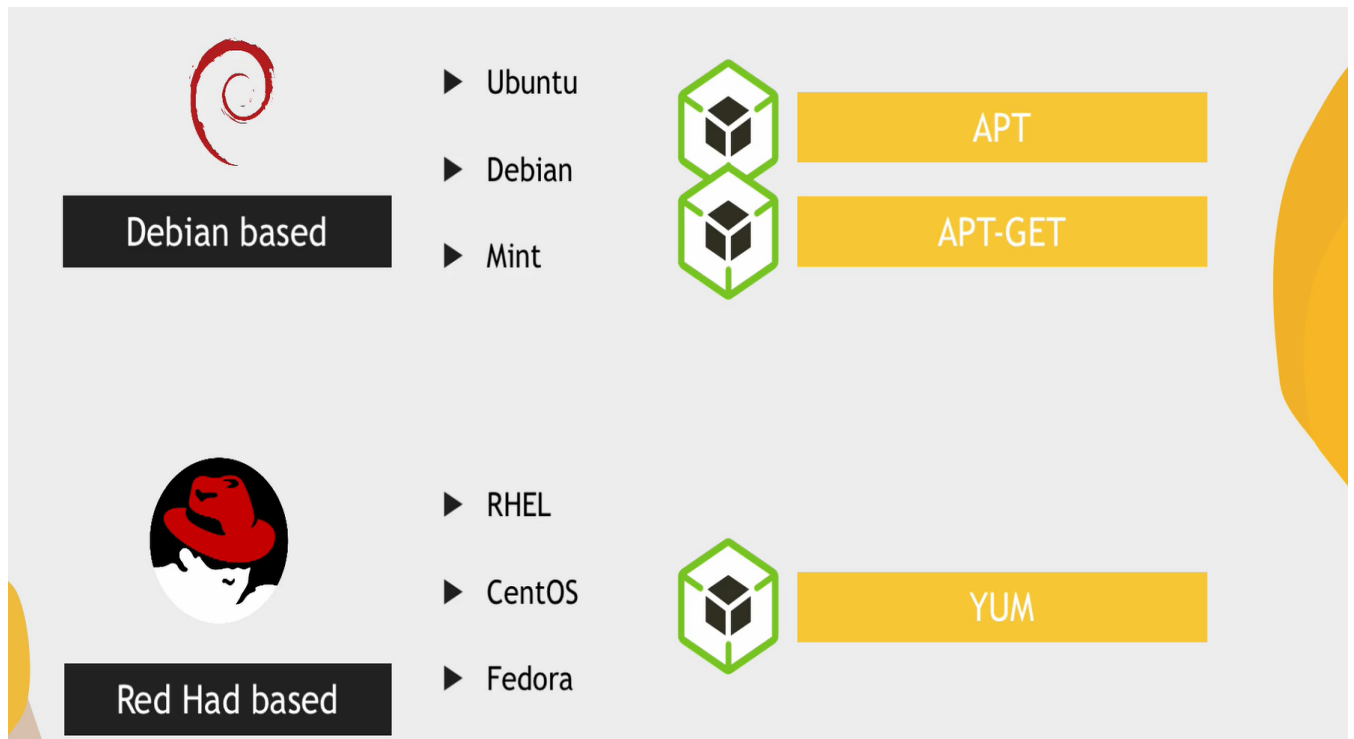
Types of Linux Distributions

1. **General-purpose:** Ubuntu, Linux Mint, Fedora, Manjaro
2. **Server-focused:** Debian, RHEL, CentOS, Rocky Linux, AlmaLinux

3. **Lightweight:** Puppy Linux, Tiny Core, Ubuntu
4. **Security-focused:** Kali Linux, Parrot OS, Qubes OS
5. **DIY-focused:** Arch Linux, Gentoo, Slackware

How to Choose a Linux Distro

- . **Beginner?** Start with **Ubuntu**, **Linux Mint**, or **Manjaro**.
- . **Server use?** Choose **Debian**, **CentOS**, **Rocky Linux**, or **RHEL**.
- . **Want Control?** Go with **Arch Linux** or **Gentoo**.
- . **Penetration Testing?** Pick **Kali Linux** or **Parrot OS**.
- . **Gaming?** Use **Pop!_OS** or **Ubuntu**.



Here is the converted text from the image:

Linux Terminology and Examples

- . Kernel
- . distribution
- . service
- . filesystem
- . command line

Kernel

. **Application ↔ Kernel ↔ Hardware**

Text:

Kernel: Interface between hardware and applications

The kernel acts as the intermediary between the hardware of the computer and the applications running on it, managing resources, memory, and system calls.

Linux Distribution Hierarchy

. **Linux Kernel**

Distro Families:

1. **Debian**
 - Ubuntu
 - Linux Mint
2. **Fedora**

- RHEL
 - CentOS
 - Oracle Linux

3. **SUSE**

- SLES
- OpenSUSE

Here is the transcription of the image:

Service Hierarchy:

. init

- **httpd**: Web Server
- **ftpd**: FTP Server
- **named**: Name Server
- **dhcpcd**: DHCP Server

Definition:

Service: Program that runs as a background process.

Examples: httpd, nfsd, ntpd, ftpd, and named.

Let me know if you'd like a detailed explanation of any of these services or how init manages background processes!

In Linux, a **file system** is the structure used to organize and manage data on storage devices, such as hard drives, SSDs, or USB drives. It defines how data is stored, accessed, and managed on a storage medium. The Linux file system is hierarchical, with a tree-like structure starting at the **root directory** (/).

Key Features of a File System in Linux:

1. **File Organization:** Files and directories are stored in a hierarchical format.
2. **Permissions:** Access to files is controlled via permissions (read, write, execute) for users, groups, and others.
3. **Mounting:** Storage devices are mounted to directories in the file system tree.
4. **Inodes:** Each file or directory has an *inode* (index node) containing metadata like size, owner, and permissions.
5. **Journaling:** Many Linux file systems use journaling to track changes, ensuring data integrity in case of system crashes.

Common File Systems in Linux

1. **Ext4 (Fourth Extended File System)**

- Default file system in many Linux distributions.
- Supports large files and volumes, journaling, and better performance.

2. **XFS**

- High-performance journaling file system for large-scale data storage.
- Ideal for servers and large file transfers.

3. **Btrfs (B-Tree File System)**

- Modern file system with advanced features like snapshots, compression, and checksums.

4. **FAT32/NTFS/exFAT**

- File systems used for compatibility with Windows and external storage devices.

5. **tmpfs**

- Temporary file system that stores data in RAM, providing fast access but non-persistence.

6. **NFS (Network File System)**

- Allows file sharing over a network.

Linux File System Hierarchy

The Linux file system starts with the root directory (/) and organizes files into standard directories. Some key directories are:

- . **/**: Root directory, the top-level directory.
- . **/bin**: Essential system binaries (e.g., ls, cp).
- . **/etc**: Configuration files.
- . **/home**: User home directories.
- . **/var**: Variable data (e.g., logs).
- . **/tmp**: Temporary files.

- . **/dev**: Device files.
- . **/proc**: Virtual file system providing system information.

File System Commands in Linux

- . **df**: Shows disk usage of file systems.
- . **mount**: Mounts a file system to a directory.
- . **umount**: Unmounts a file system.
- . **lsblk**: Lists information about block devices.
- . **fsck**: Checks and repairs file system errors.

Let me know if you'd like further details on specific file systems, commands, or how to manage Linux file systems!

In Linux, the **command line** refers to a **text-based interface** where users can type commands to interact with the operating system. This interface is typically provided by a **shell**, such as **Bash** (Bourne Again SHell), which interprets the commands entered by the user and executes them.

What is the Command Line?

- . The command line interface (CLI) allows users to perform tasks by typing commands instead of using a graphical interface (GUI).
- . The command line is accessed through a **terminal** or **terminal emulator**.
- . It is powerful, fast, and widely used for administration, scripting, automation, and development.

Key Components of the Command Line

1. **Shell:**

- The program that interprets user commands.
- Common shells include:
 - **Bash** (default on most Linux distributions)
 - **Zsh** (Z Shell)
 - **sh** (Bourne Shell)
 - **csch** (C Shell).

2. **Terminal:**

- The window or interface where the shell runs.
- Examples of terminal emulators:
 - GNOME Terminal, Konsole, xterm, PuTTY.

3. **Prompt:**

- The command prompt is where you type commands.

- Typically looks like:
- `user@hostname:~$`
 - **user**: Current user logged in.
 - **hostname**: Name of the system.
 - **~**: Current directory (e.g., **~** means home directory).
 - **\$**: Symbol indicating a standard user. A root user has **#**.

4. **Commands:**

- Instructions entered by the user, such as `ls`, `cd`, `cp`.

Benefits of the Command Line

- **Efficiency**: Faster than GUIs for repetitive tasks.
- **Automation**: Scripts and commands can automate tasks.
- **Remote Management**: Can be accessed via tools like SSH.

- . **Low Resource Usage:** Requires fewer system resources compared to GUIs.
- . **Control:** Provides advanced system control and customization.

Basic Commands in Linux

Here are some commonly used commands:

1. **Navigation Commands:**
 - `pwd`: Displays the current directory.
 - `cd [directory]`: Changes the current directory.
 - `ls`: Lists the files and directories.
2. **File Management:**
 - `cp [source] [destination]`: Copies files or directories.
 - `mv [source] [destination]`: Moves or renames files.
 - `rm [file]`: Deletes a file.

- `mkdir [directory]`: Creates a new directory.

3. **Viewing Files:**

- `cat [file]`: Displays the content of a file.
- `less [file]`: Views the content page by page.
- `head` and `tail`: Shows the first or last lines of a file.

4. **System Information:**

- `df`: Displays disk space usage.
- `top`: Displays running processes.
- `uname -a`: Shows system information.

5. **Permissions:**

- `chmod`: Changes file permissions.
- `chown`: Changes file ownership.

the Command Line

- **Terminal Emulator**: Open it in the graphical environment.

- . **TTY:** Access a full-screen terminal using Ctrl+Alt+F1 to Ctrl+Alt+F6.
- . **Remote Access:** Use tools like **SSH** to connect to Linux servers.