

Linux Structure Overview

Linux is a widely used operating system known for its robustness, flexibility, and open-source nature. It is especially important in cybersecurity due to its security and stability.

What is Linux?

- An **Operating System (OS)** like Windows, macOS, iOS, or Android.
 - Manages hardware resources and facilitates communication between software and hardware.
 - Comes in many versions called **distributions (distros)** tailored to different needs.
 - Examples: Ubuntu, Debian, Fedora, RedHat, Manjaro, Parrot OS (used here).
-

History

- Originates from **Unix (1970)** by Ken Thompson & Dennis Ritchie.
 - BSD (Berkeley Software Distribution) followed but was limited by lawsuits.
 - **GNU Project (1983)** by Richard Stallman aimed to create a free Unix-like OS with the GNU GPL license.
 - Linux kernel started as a personal project by Linus Torvalds in 1991.
 - Today: millions of lines of code, GPL licensed, over 600 distros.
 - Linux powers servers, desktops, embedded systems, and Android devices.
-

Philosophy

- Centers on **simplicity, modularity, and openness**.
- Uses small, single-purpose programs that can be chained to perform complex tasks.
- Avoids captive user interfaces; prefers shell/terminal for greater control.
- Configuration data is stored in text files.

Core Principles:

Principle	Description
Everything is a file	All configurations and resources are represented as files.
Small, single-purpose programs	Each tool does one job well and can be combined with others.
Ability to chain programs	Tools can be piped together for complex operations.
Avoid captive user interfaces	Shell-based interface gives user direct control.
Configuration data in text files	Example: <code>/etc/passwd</code> stores registered users.

Components

Component	Description
-----------	-------------

Component	Description
Bootloader	Code that starts the boot process (e.g., GRUB in Parrot OS).
OS Kernel	Core managing hardware resources and system I/O.
Daemons	Background services handling key functions (e.g., scheduling, printing).
OS Shell	Command-line interface for user interaction (e.g., Bash, Zsh).
Graphics Server	Provides graphical subsystem (X-server) for GUI applications.
Window Manager	GUI environments (e.g., GNOME, KDE) that provide desktop features.
Utilities	Applications performing specific user or system tasks.

Linux Architecture Layers

Layer	Description
Hardware	Physical components like CPU, RAM, disks.
Kernel	Manages hardware resources, virtualizes them for processes, and prevents conflicts.
Shell	CLI interface for user commands that interact with the kernel.
System Utility	Provides the OS functionalities available to users.

File System Hierarchy (FHS)

Linux organizes files in a tree structure with the root / at the top.

Path	Description
/	Root directory containing files needed to boot and mount other filesystems.
/bin	Essential command binaries.
/boot	Bootloader files, kernel executables, and boot files.
/dev	Device files representing hardware devices.
/etc	Local system and application configuration files.
/home	User home directories for personal files.
/lib	Shared libraries required at boot and for essential binaries.
/media	Mount points for external removable media (USB drives, CDs).
/mnt	Temporary mount points for filesystems.
/opt	Optional third-party application files.
/root	Home directory for the root (superuser).
/sbin	System binaries mainly for administration tasks.

Path	Description
/tmp	Temporary files storage (cleared on reboot).
/usr	Secondary hierarchy containing user utilities, applications, and libraries.
/var	Variable data like logs, mail, spool files, and databases.

Summary

Linux is a powerful and flexible operating system with a rich history rooted in Unix and GNU projects. Its modular philosophy, layered architecture, and hierarchical filesystem make it ideal for a wide range of applications—from servers to embedded systems and mobile devices. Its security, openness, and stability have made it a cornerstone in cybersecurity.

Additional Notes

- Linux distributions (distros) vary widely to meet different needs.
- Parrot OS, used here, is a Debian-based security-focused distro.
- Mastering Linux involves understanding the shell, filesystem hierarchy, and system components.