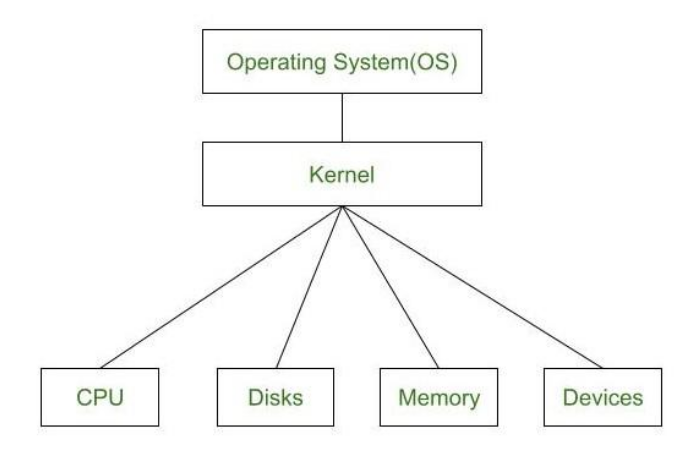
Linux

Linux is a free and open-source operating system. It is widely used in servers, supercomputers and mobile devices, as well as in embedded systems and IOT.

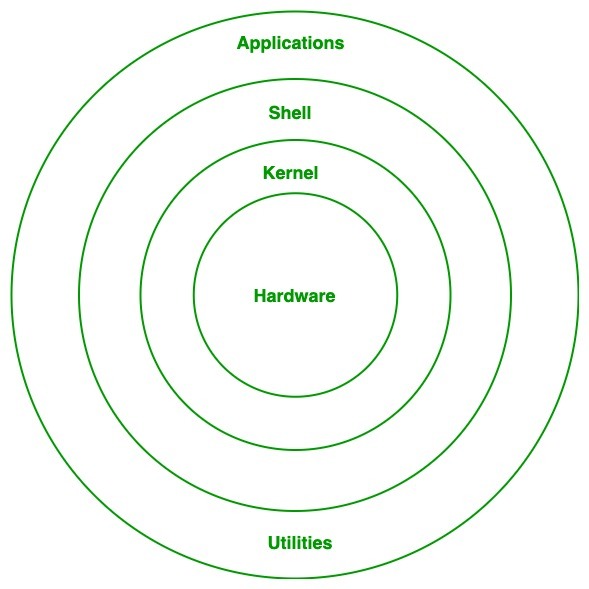
Operating System:

Operating system is a set of programs which acts as a link between the computer and the end user. And it responsible for managing computer’s resources and provide user interface for interacting with the computer.

What is Kernel?

A kernel is the core part of an operating system. It acts as a bridge between software applications and the hardware of a computer. The kernel manages system resources, such as the CPU, memory, and devices, ensuring everything works together smoothly and efficiently. It handles tasks like running programs, accessing files, and connecting to devices like printers and keyboards.

# Linux Architecture:

**Hardware** - Linux operating system contains a hardware layer that consists of several peripheral devices like CPU, HDD, and RAM.

**Kernel** − It interacts with the hardware and performs most of the tasks like memory management, task scheduling and file management.

**Shell** − When you type in a command at your terminal, the shell interprets the command and calls the program that you want.

**Commands and Utilities** − There are various commands and utilities which you can make use of in your day to day activities. There are over 250 standard commands plus numerous others provided through 3rd party software. All the commands come along with various options.

**Files and Directories** − All the data of Linux is organized into files. All files are then organized into directories. These directories are further organized into a tree-like structure called the filesystem.

**Applications:** Applications are software programs that run on the Linux operating system. They include everything from basic text editors to complex database management systems and web servers. Most Linux applications are open-source and available for free.

**Services:** Services are background processes that run on the Linux operating system. They provide essential functions such as network connectivity, security, and system monitoring.

**Device Drivers:** Device drivers are software programs that allow the Linux operating system to communicate with hardware devices such as printers, scanners, and network cards.

Advantages of Linux

1. Open source

2. Free

3. Secure

4. Stable

5. Lightweight Linux distributions:

8GB of disk space for a minimal installation.

20GB For a full installation with a GUI and a range of software packages

6. Large Community

Linux Boot Process



**The Linux boot process is divided into four main stages:**

**Stage 1: BIOS**

When you power on your computer, your computer starts BIOS and performs a POST. This is an integrity check that probes the hardware functionality of components such as hard disks, SSD, keyboard, RAM, USB ports and any other hardware. If the hardware works as expected, the boot process moves on to the next stage.

**Stage 2: The boot loader**

Once POST is complete, BIOS searches for and loads the boot loader program stored in the MBR. The MBR is a 512-byte code that is usually located at /dev/sda or /dev/hda depending on your hard drive architecture. The MBR can also be located on a live USB or DVD installation of Linux. BIOS loads and executes this MBR code.

There are three main boot loaders in Linux: LILO, GRUB and GRUB2. The GRUB2 (Grand Unified Bootloader) boot loader is the latest and primary boot loader in modern Linux distributions. The GRUB2 configuration file is located at /boot/grub2/grub2.cfg. Once BIOS locates the GRUB2 boot loader, It executes and loads it into the main memory (RAM).

**Stage 3: Linux kernel initialization**

The Linux kernel is the heart of the operating system. In your Linux system, the kernel interfaces with the hardware, controls memory management and manages processes. The boot loader loads the selected Linux kernel. The kernel self extracts from a compressed version and mounts the root file system. It then runs the /sbin/init program.

**Stage 4: systemd**

The kernel loads systemd, which is a system and service manager for Linux operating systems. systemd then runs all the other initialization processes.

**2.2 Boot process with systemd**

Once the kernel loads systemd, systemd takes over and starts the other system services that are required to bring the system up and running. This includes services such as networking service, the login manager, etc.

The boot process is parallelized in the order in which specific target units are executed. systemd uses the /etc/systemd/system/default.target file to determine the target that the Linux system should boot into. This file is a link to graphical.target which boots the graphical login manager. systemd activates all target units that are dependencies of default.target as well as recursively all dependencies of these dependencies. Once all the services are started, your system is ready to use and the login manager displays. You can now log in and start using the system.

## **Types of the Kernel**

**Monolithic Kernel:**  In this, the user and kernel services are implemented in the same memory space. Due to this, the size of the kernel increases, which in turn increases the size of the operating system. The main benefit is that the process execution is faster as there is no separate memory space for the user and Kernel.

**Microkernel:** Here, the user and kernel services are implemented in two different spaces. It has separate User Space and Kernel Space. This reduces the size of the Kernel and results in reducing the size of the operating system.

**Hybrid Kernel:** Hybrid kernels combine monolithic and microkernels. They include more services than microkernels but less than monolithic kernels. This allows them to offer some of the benefits of both kernels. It borrows speed from the monolithic kernels and modularity from microkernels.

**Nano Kernel:** Nano kernels are the smallest type of Kernel, consisting of only a few thousand lines of code. It means that the code executing in the privileged mode of the hardware is minimal. They are used primarily in embedded systems or devices with limited resources.

**Exo Kernel:** This Kernel has separate resource protection and management. It is suitable for use when performing application-specific customization. Exo kernels are designed for use in mobile devices. They are a variation of microkernels that include additional features specifically for mobile devices, such as power management and support for multiple processors.

Linux Distributions

Ubuntu: Ubuntu is one of the most popular Linux distributions. It is based on Debian and is known for its user-friendly interface and large software repository. It is often used as a desktop operating system. Fedora: Fedora is a community-driven Linux distribution sponsored by Red Hat. It is known for its fast release cycle and emphasis on cutting-edge technology. It is often used by developers and system administrators.

Debian: Debian is a stable and reliable Linux distribution that is widely used in servers and workstations. It is known for its strict adherence to free software principles and large software repository

CentOS: CentOS is a community-driven Linux distribution that is based on the Red Hat Enterprise Linux (RHEL) operating system. It is known for its stability and security features and is often used in servers.

Arch Linux: Arch Linux is a lightweight and customizable Linux distribution that is known for its simplicity and minimalism. It is often used by advanced users and developers.

openSUSE: openSUSE is a community-driven Linux distribution that is known for its user-friendly interface and extensive software repository. It is often used as a desktop operating system.

Kali Linux: Kali Linux is a Linux distribution that is designed for penetration testing and security auditing. It is often used by security professionals and ethical hackers.

Difference between Debian and RHEL

Debian and RedHat Enterprise Linux (RHEL) are two popular Linux distributions that are often compared due to their differences in package management, release cycles, and target audiences.

Debian is a community-driven Linux distribution that is known for its stability and large software repository. It uses the Advanced Package Tool (APT) for package management, which allows users to easily install, update, and remove software packages. Debian releases are known for their long development cycles, with new versions being released every two years on average.

RHEL, on the other hand, is a commercial Linux distribution that is supported by Red Hat. It is known for its stability, security, and enterprise-grade features. RHEL uses the Yellowdog Updater, Modified (YUM) for package management, which allows users to easily install and manage software packages. RHEL releases are known for their long-term support, with major versions being supported for up to 10 years.

One of the main differences between Debian and RHEL is their target audience. Debian is often used by individual users and small businesses, while RHEL is more commonly used in enterprise environments, where stability, security, and support are critical.

Another difference between Debian and RHEL is their release cycles. Debian releases are typically more stable and predictable, with a longer development cycle and more emphasis on testing and quality assurance. RHEL releases are often more focused on enterprise features and support, with shorter development cycles and more frequent updates. Overall, the choice between Debian and RHEL depends on the user's needs and preferences. Debian is a good choice for individuals and small businesses that value stability and a large software repository, while RHEL is a good choice for enterprises that require enterprise-grade features, support, and security.

Unix vs Linux

Unix and Linux are two different operating systems that share many similarities.

Unix was developed in the late 1960s and 1970s, while Linux was developed in the early 1990s.

Here are some of the main differences between Unix and Linux:

Proprietary vs. Open Source: Unix is a proprietary operating system that is owned by different vendors, such as IBM, HP, and Oracle. Linux, on the other hand, is an opensource operating system that is freely available and can be modified and distributed by anyone.

Cost: Because Unix is proprietary, it often comes with a higher cost than Linux. Linux, on the other hand, is typically free or has a low cost.

Kernel: Both Unix and Linux use a similar kernel, which is the core component of the operating system. However, the kernel used in Linux is based on the Unix kernel, but has been modified and improved by the open-source community.

Compatibility: Unix applications are typically built for specific versions of Unix and may not be compatible with other versions. Linux, on the other hand, is more compatible with different hardware and software configurations

User interface: Unix typically has a command-line interface, while Linux can be used with both a command-line interface and a graphical user interface.

Linux vs Windows

Features: Linux is known for its flexibility and customizability, while Windows is known for its ease of use and compatibility with many software applications.

Security: Linux is generally considered to be more secure than Windows due to its opensource nature and its focus on security. Windows, on the other hand, is more vulnerable to viruses and malware.

Cost: Linux is typically free or has a low cost, while Windows requires a license fee to use.

User interface: Linux offers a variety of user interfaces, including command-line interfaces and graphical user interfaces, while Windows primarily uses a graphical user interface.

Software compatibility: Windows is compatible with a wide range of software applications, while Linux may require more customization or alternative software options.

System administration: Linux requires a certain level of expertise for system administration, while Windows is generally easier to manage.

Storage: Windows uses different drivers such as C, D, E, and more having some folders to store files. But linux uses a tree structure to store and organize files. Linux file structure st arts from the root directory. (represents with /)

Files: Linux files are case sensitive; therefore we can have two files having the same name; one in uppercase and other in lower case. Comparatively, the windows files are not case sensitive. We can not have two files with the same name.

Journaling feature**:**

A journal means a record that is kept over time, like a diary. Instead of changing the structure of the file system, adding new directories, or making files bigger, the first thing the file system does is collect these changes into a series of transactions and add them to a log file. Then, these transactions can be sent to the actual file system by processes that run in the background. If there is a crash, the file system can recover immediately and start committing transactions already in the journal. However, the last entry will probably be corrupt and will be dropped. All of these things mean that the file system won't change. This kind of file system,

Types of filesystems

ext: Extended file system (1992)

ext2: Second extended file system (1993)

● Ext2 does not have journaling feature

● Maximum individual file size can be from 16GB to 2TB

● Overall ext2 file system size can be from 2TB to 32TB

ext3 : Third extended file system (2001)

● The main benefit of ext3 is that it allows journaling.

**advantages of systemd over SysVinit:**

**Parallelization and Faster Boot Times:**

systemd aggressively parallelizes service startup, leading to faster boot times. It manages services concurrently, whereas in SysVinit, achieving parallelism requires manual effort12.

Automatic dependency handling in systemd contributes to efficient startup, whereas SysVinit lacks this feature2.

Dependency Resolution:

systemd automatically resolves dependencies between services. When a service depends on another, systemd ensures the correct order of startup.

In contrast, SysVinit relies on manual configuration for dependency management.

Monitoring and Service Restart:

systemd monitors started services and can automatically restart crashed services. This self-healing behavior enhances system reliability.

SysVinit lacks built-in monitoring and automatic restart capabilities.

Cgroups and Process Tracking:

systemd organizes daemons into their own Linux control groups (cgroups). This isolation allows better resource management and tracking of processes.

SysVinit does not provide native cgroup support.

Service Management with systemctl:

systemctl is the primary tool for managing systemd services. It allows users to start, stop, enable, disable, reload, and verify services.

In SysVinit, service management relies on traditional bash scripts, whereas systemd uses .service files.

Snapshotting and State Restoration:

systemd supports snapshotting and restoring the system state, which can be useful for system maintenance and troubleshooting.

SysVinit lacks such features.

What is systemctl?

Systemctl is the primary command-line utility for interacting with systemd, the system and service manager in Linux. It serves as a pivotal tool for managing system resources and services. While systemd orchestrates the boot process and manages services of the Linux system, systemctl offers the necessary commands to communicate with systemd, facilitating the management of system services.

With systemctl, users can execute a variety of tasks such as starting, stopping, and restarting services, as well as enabling or disabling services to start automatically at boot. It also allows for checking the status of services and units, providing insights into their operational state, whether active, running, failed, or disabled.