Operating System Concepts

An **operating system** (**OS**) is system software that manages computer hardware, software resources, and provides common services for computer programs.

**Process:**

In computing, a process is the instance of a computer program that is being executed by one or many threads. It contains the program code and its activity. Depending on the operating system (OS), a process may be made up of multiple threads of execution that execute instructions concurrently.

**Memory:**

The memory management function keeps track of the status of each memory location, either allocated or free. It determines how memory is allocated among competing processes, deciding which gets memory, when they receive it, and how much they are allowed. When memory is allocated it determines which memory locations will be assigned. It tracks when memory is freed or unallocated and updates the status. The memory management function keeps track of the status of each memory location, either allocated or free. It determines how memory is allocated among competing processes, deciding which gets memory, when they receive it, and how much they are allowed. When memory is allocated it determines which memory locations will be assigned. It tracks when memory is freed or unallocated and updates the status.

**File Systems:**

A file system is a process that manages how and where data on a storage disk, typically a hard disk drive (HDD), is stored, accessed and managed. It is a logical disk component that manages a disk's internal operations as it relates to a computer and is abstract to a human user.

Basic Linux Commands

**cd:** This is one of the most used commands. It is used to move around in the file system. cd takes one argument, that is the location you wish to go to. It can be an absolute address, i.e. all the way from the root, or a directory within the present working directory, or a ‘..’ for going up one directory. Simply typing cd will take you to the current user’s home directory.

**ls:** This command is used to list the contents of a directory. It can have either one argument, that is the directory whose contents you want you view, or no arguments, in which case it will display the contents of the present directory.

**pwd:** It shows the present working directory.

**mkdir:** It is used to create a new directory.

**rmdir:** It is used to remove a directory.

**touch:** It is used to create a new file.

**rm:** It is used to delete a file.

**man:** It tells information about other linux commands. It has information for most comands but not all.

**cp:** It is used to copy files.

**mv:** It is used to move files as well as rename them.

**locate:** It is used to locate a file in the computer.

Create a user and manage its permission without using sudo

Step1: switch to super user using the su command.

Step2: create a new user with useradd.

Step3: set password for this user using passwd command.

Step4: open sudoers file using visudo command.

Step5: press i to enter insert mode.

Step6: add the line “username ALL=(ALL) ALL” in the file.

Step6: press esc to enter command mode. Type “:wq” to save and exit.

Alternatively,

Add user to the sudo group, or any other group with root permissions using usermod command.

Block Storage

This kind of storage divides the file into blocks of data with each block of data having only its own address. This kind of storage is good enough for smaller storages but it becomes harder to manage when there are huge amounts of data. Scaling up is also harder with it.

Object Storage

This kind of storage treats each data item as an object. Each object is made up of data, metadata and a unique id. This kind of storage is not economical for smaller storage but its scalability and manageability makes it ideal for huge amounts of data.

Linux directory structure

The Filesystem Hierarchy Standard (FHS) defines the structure of file systems on Linux and other UNIX-like operating systems. However, Linux file systems also contain some directories that aren’t yet defined by the standard.

## **/ – The Root Directory**

Everything on your Linux system is located under the / directory, known as the root directory. You can think of the / directory as being similar to the C:\ directory on Windows – but this isn’t strictly true, as Linux doesn’t have drive letters. While another partition would be located at D:\ on Windows, this other partition would appear in another folder under/on Linux.

## **/bin – Essential User Binaries**

The /bin directory contains the essential user binaries (programs) that must be present when the system is mounted in single-user mode. Applications such as Firefox are stored in /usr/bin, while important system programs and utilities such as the bash shell are located in /bin. The /usr directory may be stored on another partition – placing these files in the /bin directory ensures the system will have these important utilities even if no other file systems are mounted. The /sbin directory is similar – it contains essential system administration binaries.

## **/boot – Static Boot Files**

The /boot directory contains the files needed to boot the system – for example, the GRUB boot loader’s files and your Linux kernels are stored here. The boot loader’s configuration files aren’t located here, though – they’re in /etc with the other configuration files.

## **/cdrom – Historical Mount Point for CD-ROMs**

The /cdrom directory isn’t part of the FHS standard, but you’ll still find it on Ubuntu and other operating systems. It’s a temporary location for CD-ROMs inserted in the system. However, the standard location for temporary media is inside the /media directory.

## **/dev – Device Files**

Linux exposes devices as files, and the /dev directory contains a number of special files that represent devices. These are not actual files as we know them, but they appear as files – for example, /dev/sda represents the first SATA drive in the system. If you wanted to partition it, you could start a partition editor and tell it to edit /dev/sda.

This directory also contains pseudo-devices, which are virtual devices that don’t actually correspond to hardware. For example, /dev/random produces random numbers. /dev/null is a special device that produces no output and automatically discards all input – when you pipe the output of a command to /dev/null, you discard it.

## **/etc – Configuration Files**

The /etc directory contains configuration files, which can generally be edited by hand in a text editor. Note that the /etc/ directory contains system-wide configuration files – user-specific configuration files are located in each user’s home directory.

## **/home – Home Folders**

The /home directory contains a home folder for each user. For example, if your user name is bob, you have a home folder located at /home/bob. This home folder contains the user’s data files and user-specific configuration files. Each user only has write access to their own home folder and must obtain elevated permissions (become the root user) to modify other files on the system.

## **/lib – Essential Shared Libraries**

The /lib directory contains libraries needed by the essential binaries in the /bin and /sbin folder. Libraries needed by the binaries in the /usr/bin folder are located in /usr/lib.

## **/lost+found – Recovered Files**

Each Linux file system has a lost+found directory. If the file system crashes, a file system check will be performed at next boot. Any corrupted files found will be placed in the lost+found directory, so you can attempt to recover as much data as possible.

## **/media – Removable Media**

The /media directory contains subdirectories where removable media devices inserted into the computer are mounted. For example, when you insert a CD into your Linux system, a directory will automatically be created inside the /media directory. You can access the contents of the CD inside this directory.

## **/mnt – Temporary Mount Points**

Historically speaking, the /mnt directory is where system administrators mounted temporary file systems while using them. For example, if you’re mounting a Windows partition to perform some file recovery operations, you might mount it at /mnt/windows. However, you can mount other file systems anywhere on the system.

## **/opt – Optional Packages**

The /opt directory contains subdirectories for optional software packages. It’s commonly used by proprietary software that doesn’t obey the standard file system hierarchy – for example, a proprietary program might dump its files in /opt/application when you install it.

## **/proc – Kernel & Process Files**

The /proc directory similar to the /dev directory because it doesn’t contain standard files. It contains special files that represent system and process information.

## **/root – Root Home Directory**

The /root directory is the home directory of the root user. Instead of being located at /home/root, it’s located at /root. This is distinct from /, which is the system root directory.

## **/run – Application State Files**

The /run directory is fairly new, and gives applications a standard place to store transient files they require like sockets and process IDs. These files can’t be stored in /tmp because files in /tmp may be deleted.

## **/sbin – System Administration Binaries**

The /sbin directory is similar to the /bin directory. It contains essential binaries that are generally intended to be run by the root user for system administration.

## **/selinux – SELinux Virtual File System**

If your Linux distribution uses SELinux for security (Fedora and Red Hat, for example), the /selinux directory contains special files used by SELinux. It’s similar to /proc. Ubuntu doesn’t use SELinux, so the presence of this folder on Ubuntu appears to be a bug.

## **/srv – Service Data**

The /srv directory contains “data for services provided by the system.” If you were using the Apache HTTP server to serve a website, you’d likely store your website’s files in a directory inside the /srv directory.

## **/tmp – Temporary Files**

Applications store temporary files in the /tmp directory. These files are generally deleted whenever your system is restarted and may be deleted at any time by utilities such as tmpwatch.

## **/usr – User Binaries & Read-Only Data**

The /usr directory contains applications and files used by users, as opposed to applications and files used by the system. For example, non-essential applications are located inside the /usr/bin directory instead of the /bin directory and non-essential system administration binaries are located in the /usr/sbin directory instead of the /sbin directory. Libraries for each are located inside the /usr/lib directory. The /usr directory also contains other directories – for example, architecture-independent files like graphics are located in /usr/share.

The /usr/local directory is where locally compiled applications install to by default – this prevents them from mucking up the rest of the system.

## **/var – Variable Data Files**

The /var directory is the writable counterpart to the /usr directory, which must be read-only in normal operation. Log files and everything else that would normally be written to /usr during normal operation are written to the /var directory. For example, you’ll find log files in /var/log.