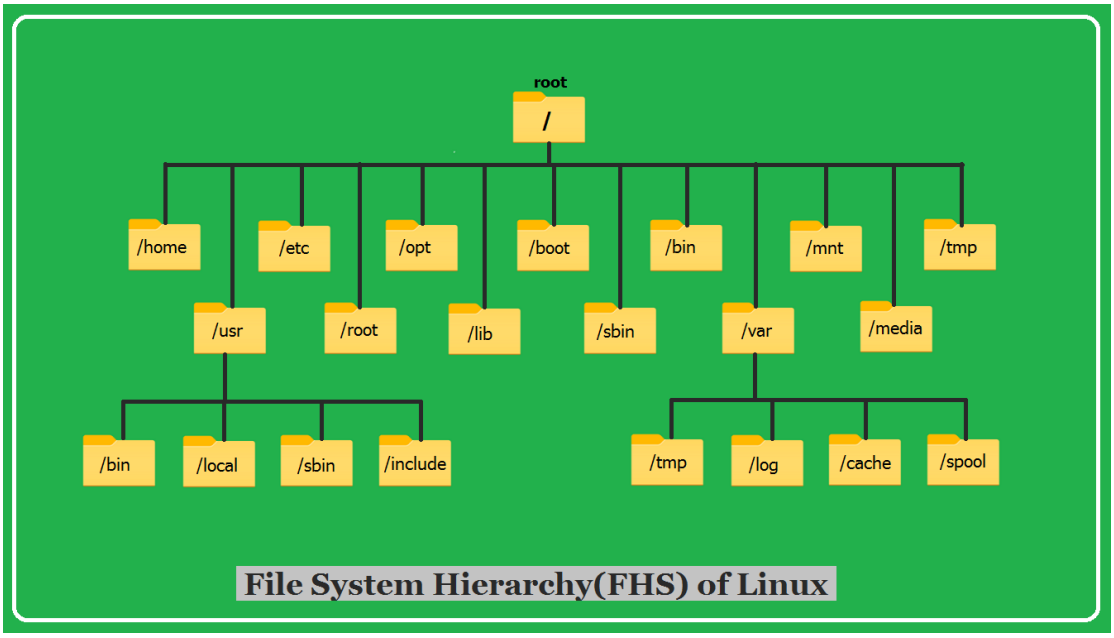
Linux Basic File System and Structure and Permission

I) Exploring Linux File System Hierarchy



i) '/'- It represent the root directory of Linux system.

- Every single file and directory start from the root directory.
- The only root user has the right to write under this directory.

ii) '/bin'- it contains essential commands and binaries need by the user.

- Commands like kill,ssh,cp,ls are located in this directory.
- Contains binary executables.
- Common linux commands you need to use in single-user modes are located under this directory.

iii) '/boot'- it contains files required for booting up the system

- Contains GRUD bootloader configuration
- Kernel initrd, vmlinux, grub files are located under /boot

iv) '/dev'-contains device configuration

- Special file that act as interface between hardware and storage
- Device files of two types and block drive and character drives.
- Example: /dev/tty1, /dev/usbmon0

v) 'etc'-This directory contains config files related to system applications ,user,service and tools.

- Examples-: /etc/resolv.conf, /etc/logrotate.conf. /etc/host

vi) '/lib'-These include dynamic libraries needed during runtime.

- Library filenames are either ld* or lib*.so.*
- Example: ld-2.11.1.so, libncurses.so.5.7

vii) '/tmp'-Directory that contains temporary files created by system and users.

- Files under this directory are deleted when the system is rebooted.

viii) '/usr'-Secondary hierarchy for read-only user data.

- contains the majority of (multi-)user utilities and applications.
- Contains binaries, libraries, documentation, and source-code for second level programs.
- /usr/bin contains binary files for user programs. If you can't find a user binary under /bin, look under /usr/bin. For example: at, awk, cc, less, scp

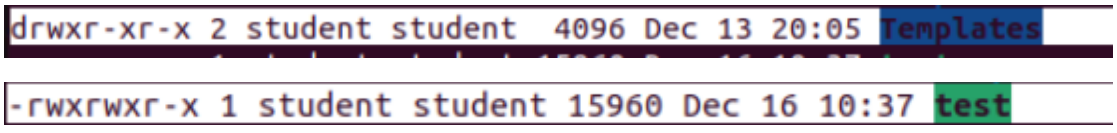
2) File Permission and Ownership

- d- Specifies directory
- r- Read permission
- w- Write or Modify files permission
- x- Execute file as a script permission
- b- Block specific device
- c- character specific device
- p- pipe
- - -> regular file
- l – symbolic link to a file

Understanding Ownership

- User (owner): The specific user who owns the file, usually the creator. This user has primary control and can often change permissions.
- Group: A collection of users who share specific access permissions to the file.
- Others: All other users on the system who are not the owner or a member of the owning group

Example



We can have read, write and execute permission for files as well directories and modify it.

Here In first example user has permission to read, write and execute, group user can execute and write followed by that others has permission only to execute.

Modifying Permission of files and directories

chown- it is a command used to modify the permission

Examples

1) (eg)
```
chmod u=rw,g=rw test.cpp
```
Giving the exact permisions
Here use user will have read,write permission and group user will also have the read write permission.

2) **Giving Permission in binary form**

- 000 000 000 – represent file permission for user, groups and others.
- First 0 for read and second for write and third to execute.
- 110 110 000 – now I have change 1 for read and write for user and groups.

(eg)

```
student@student-virtual-machine:~$ chmod 660 test.cpp
```

Modifying ownership of files and directories

1) chown (change owner): Changes the user owner and/or group owner of a file (requires sudo privileges for changing the owner).

- Example: sudo chown newuser filename
- Example: sudo chown newuser:newgroup filename //we change owner and family at same time in chown

3) chgrp (change group): Specifically changes only the group ownership of a file.
    - Example: chgrp newgroup filename