Linux – Day 03

Linux Directory Structure

The Linux directory structure, often referred to as the filesystem hierarchy, is the way files are organized on a Linux system. Here's a breakdown of the key directories and their purposes:

/ (Root directory):

The top-level directory in the filesystem hierarchy.

All other directories and files are subdirectories or files within this directory.

/bin (Binaries):

Contains essential binary executables (programs) required for system boot and maintenance.

Common commands like ls, cp, mv, etc., are stored here.

/boot (Bootloader files):

Contains files needed for booting the system, such as the Linux kernel, bootloader configuration, and initial ramdisk image.

/dev (Devices):

Contains device files that represent physical and virtual devices attached to the system, such as hard drives, terminals, printers, etc.

/etc (System configuration files):

Contains system-wide configuration files that control the behavior of various programs and services.

Configuration files for networking, user authentication, system services, etc., are typically stored here.

/home (User home directories):

Contains home directories for individual users.

Each user typically has a subdirectory here where they can store their personal files and configuration settings.

/lib and /lib64 (Libraries):

Contains shared libraries needed by the programs in /bin and /sbin during runtime.

/mnt (Mount point):

A directory used for temporarily mounting filesystems, such as external storage devices or network shares.

/opt (Optional packages):

Typically used for installing optional software packages that are not part of the core system distribution.

/proc (Process information):

A virtual filesystem that provides information about running processes and system resources.

Information is presented in a hierarchical structure of directories and files.

/root (Root user home directory):

The home directory for the root user, the system administrator.

The root user has full privileges and can access and modify any file on the system.

/sbin (System binaries):

Contains binary executables (programs) used for system administration tasks.

These executables are typically reserved for use by the system administrator.

/tmp (Temporary files):

Contains temporary files created by various programs and services.

Files in this directory are typically deleted when the system is rebooted.

/usr (User-related programs):

Contains user-related programs, libraries, documentation, and other files not needed for system boot or repair.

Subdirectories include /usr/bin for user binaries, /usr/lib for libraries, /usr/include for header files, etc.

/var (Variable data):

Contains variable data files, such as logs, spool files, and temporary files created by daemons (background services).

Subdirectories include /var/log for log files, /var/spool for print and mail queues, /var/tmp for temporary files that should persist across reboots, etc.

1. How to identify whether it is a file or a directory?

$ ls -l (or) ll

If a line is starting with -, then it is a normal file or text file

If a line is starting with d, then it is a directory

2. How to remove a file?

$ rm <FileName>

3. How to remove all the files at a time in a specific directory?

$ rm * -f

4. How to see the files based on time stamp (latest to oldest)?

$ ls -lt

5. How to see the files based on time stamp (oldest to latest)?

$ ls -lr

Note: By default, when we execute ls or ls -l or ll, the files and directories will be displayed based on the alphabetical order

6. How to see the content of a specific directory without going inside a directory?

$ ls -l <DirectoryName>

7. How to remove an empty directory?

$ rmdir <DirectoryName>

8. How to delete a directory with files inside it?

$ rm -r <DirectoryName>

9. Working with 'vi' command

By using vi command, we can create a file and also write the content inside the file.

9.1 Creation of file using 'vi' command

$ vi <FileName>

Note: If a file is not available, linux will create that file.

Step1: vi <FileName>

Step2: Go into the 'Insert' mode. Press 'i'

Step3: Write the required content

Step4: Press 'escape' to comeout of the insert mode.

Step5: Save the file and exit. :wq

Additional Commands:

:w ---> to save

:q! ---> to quit

:wq ---> Save the file and quit the file

:wq! ---> Save the file and quit the file

10. If the cursor is at starting of the line, and you want to go to the end of a line

$ shift + a

11. If the cursor is at end of the line, and you want to go to the start of a line

$ shift + i

12. How to write the content above the existing line (whereever cursor is available)?

$ shift + o

13. How to write the content below the existing line (whereever cursor is available)?

$ shift + g

14. How to see the number of lines available in a file?

$:set number

15. How to go to a specific line?

$:<LineNumber>

Note: If the line number is not available, it will take the cursor to the last line in the file.

16. How to copy a specific line?

$ yy

Note: The line gets copied based on the cursor position.

17. How to paste a copied line?

$ p

Note: The new lines will get pasted below the existing line

18. How to delete a line?

$ dd

Note: The line gets deleted based on the cursor position.

19. How to undo?

$ u

20. How to redo?

$ control + r

21. How to delete specific number of lines at a time?

$ <Number of lines you want to deleted>dd

Ex: 5dd

In the above example, 5 lines will get deleted.

22. How to copy specific number of lines at a time? (Exercise)

$ <Number of lines you want to copy>pp

Ex: 5pp

In the above example, 5 lines will get copied.

23. How to search for a specific word?

$ /<WordName>

Note: The words are case sensitive

Note: To perform the above commands (from 10th command), you should be there inside the file, but you shouldn't be in Insert mode.

GITHUB Repository URL: https://github.com/KastroVKiran/Linux-by-Kastro.git

LinkedIn URL: https://www.linkedin.com/in/kastro-kiran-493759106/