Thursday, 7 March 2024

Program Interaction : Linux Command Line

Terminal

- Terminal is a program that opens a window and lets you interact with the shell.
- SHELL - program that takes commands from the keyboard and gives them to the operating system to perform. On most Linux systems a program called bash(Bourne Again SHell) acts as the shell program.
- If the last characted of your shell prompt is # rather than $, you are operating as the superuser, which means that you have administrative privileges. This can be dangerous, since you are able to delete or overwrite any file on the system. Unless u absolutely need administrative privileges, do not operate as the superuser.

The File System

- Files are organized into file systems.
- Unlike Windows(which traditionally has different file systems at different anchor points C:\ ,D:\..), Linux presents a unified file system view.
- Files are stored in directories in the filesystem.
- Each process has a “current working directory”. (change it with the cd). List the files in a directory using ls command.
- Paths :
  - Absolute Paths - start with /, such as /usr, /home/nnia/flags etc
  - Relative Paths dont start with /, and are relative to the current working directory.
  - “.” - signifies “current directory”
  - “..” signifies the directory that the current directory lives in

Paths to commands - WHERE IS CAT?

- If the first word of the command has no / characters, the shell will search for it in either its builtins or a set of directories specified in the PATH environment variable.
- (interlude)ENVIRONMENT VARIABLES - set of Key/Value pairs passed into every process when it is launched.

- which - if u are curious about what program file ends up becoming your cat process after it's found using the PATH variable use which (not very obvious yet :D)

File Types:

- regular file
- d - directory
- l - symbolic link (file that transparently points to another file or directory)
- p - named pipe
- c - character device file
- b - block device file
- s - unix socket

Symbolic (AKA soft) links — special type of file that references another file. They are created ln -s (-s stands for symbolic)

Hard links - is an equally "valid" reference to the original file as the original file itself. It is a file that happens to be backed by the same data as the original. (created with ln without the -s argument)

Pipes

- Pipes facilitate a unidirectional flow of information. Two types of pipes:
  - Unnamed pipes - channels of information between processes. Most commonly used to direct data from one command to another
  - Names pipes (FIFOs) -created using mkfifo command (dk what they do :D)