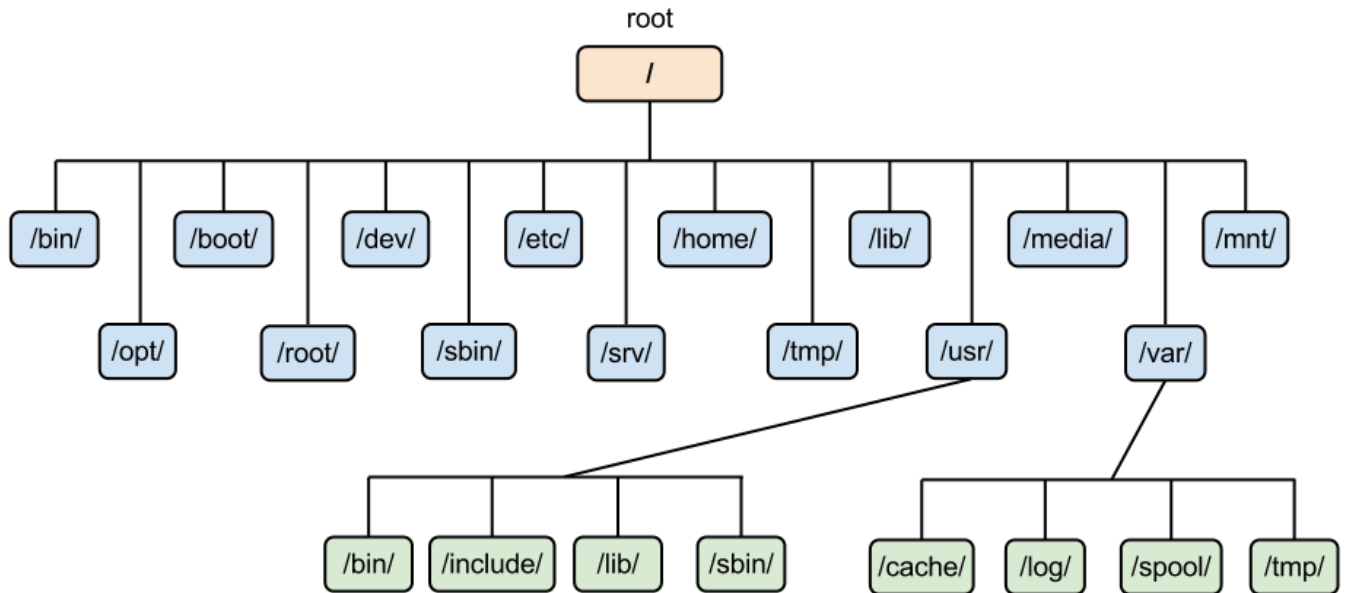


A Ride Through Application Oriented Linux Guidelines

- Linux File-system



Some useful directories are:

- **/etc** - contains most, if not all system-wide configuration files. Eg:
 - `/etc/apt/apt.conf`
- **/dev** - contains device files. Many of these are generated at boot time or even on the fly. Eg: if you plug in a new webcam or a USB pen drive into your machine, a new device entry will automatically pop up here.
- **/home** - is where you will find your users' personal directories
- **/proc** - like `/dev` is a virtual directory. It contains information about your computers, such as information about your CPU and the kernel your Linux system is running. As with `/dev`, the files and directories are generated when your computer starts, or on the fly, as your system is running and things change. It also contains the information of all running process.
- **/media** - is where external storage will be automatically mounted when you plug it in and try to access it.

Basic Commands

- ls - list directory contents
- cd - change the shell working directory
- man - an interface to view user manual of most commands
- help - Display information about built-in commands. To know more about shell built-in commands [click here](#).
- touch - change file timestamps. Can also be used to create files.
- sudo - execute a command as superuser (sudoer)
- apt - provides a high-level command line interface for the package management system. [Know More](#)
- mkdir - used to make directories.
- rmdir - used to remove directories (only if completely empty)
- rm - used to permanently delete files. **'USE WITH CAUTION'**
- mv - move (rename) files.
- cp - copy files and directories
- locate - find files by name.

Commonly used apt commands

- *sudo apt update* - used to download package information from all configured sources
- *sudo apt upgrade* - is used to install available upgrades of all packages currently installed on the system
- *sudo apt install <package1-name> [<package2-name> ...]* - install a package
- *sudo apt remove <package1-name> [<package2-name> ...]* - remove a package.

Few Footnotes

- Autocomplete on the shell (only applicable for bash shell not on POSIX shell also known as “sh”) - Write a few characters of the required text and hit a single tab, results in autocomplete of that name if found in the current working directory. If single **TAB** doesn't output anything there may be two possible reason either no file with that prefix is present or more than one file is present, for the latter case press **TAB** two times to show all possible options.
- To copy some text from the terminal, select it and use the key combination: **CTRL+SHIFT+C**. To paste copied text to terminal use the key combination: **CTRL+SHIFT+V**.
- **Never use the command `sudo rm-rf /`**
- When working in a directory which contains the file you want to access for any command you may use just the filename, otherwise, use the full name (path) of the file.

Example:- if you are currently in /home/<your username>/Desktop/
and it has filename a.txt and b.txt

Then you can access the files in this as:-

`ls -l a.txt` (two give a long description of the file)

For any other file, not in the current working directory, you can access it by full name/path like:-

`ls -l /home/<your username>/Documents/<any file present there>`

PROXY SETTINGS

- **For apt:**
 - Edit the apt.conf file present at: ***'/etc/apt/apt.conf'***
Sample apt.conf file (for MNNIT)
`Acquire::http::proxy "http://edcguest:edcguest@172.31.100.29:3128/";`
`Acquire::https::proxy "http://edcguest:edcguest@172.31.100.29:3128/";`
`Acquire::ftp::proxy "http://edcguest:edcguest@172.31.100.29:3128/";`
- To set the proxy on terminal:
 - `export http_proxy="http://edcguest:edcguest@172.31.100.29:3128/"`
`export https_proxy="http://edcguest:edcguest@172.31.100.29:3128/"`

- To set the proxy for general system shell (ie. the UI/browser):
 - Set using the UI method from settings -> network -> proxy

PS: when you change the proxy using the UI method, sometimes the “apt.conf” file gets changed and you will not be able to download and install packages. So do visit it if an error is encountered.

Some advanced topics:

- I/O redirection. Eg: `'ls -l > output-file-name'`
 - Further Reading: [Link](#)
- Read, write, Execute privileges
- [Pipelining](#). Eg: `'ls | grep Document'` (grep is a pattern matching command. For more info `'man grep'`).

Final Thoughts:

This is an overview of how to start up with Linux, this is in no sense the complete, in-depth and detailed explanation of things. Due to the specifics of the humanoid course, we kept the session very precise and tried to touch various aspects of the topic rather than detailing it. There were many things we skipped upon like that “total” in ls command because it is not used and seldom cared about data. Multiple points in the class were discussed at a fast pace due to the shortage of time so any questions are welcomed. You are open to join us again into the future classed scheduled by either the humanoid project group or the computer club where each point will be covered in more depth and no stone will be left unturned.

How Linux works and how to work upon it, requires practice, time and patience so keep sudoing and have fun with your newly installed operating system.

Don't drink and root,
Mahalo,
Abhishek Kr. Singh
Surbhit Awasthi