# My Knowledge on Linux

*Basic Command Line Notes*

Aong Cho Marma

2023

# My Knowledge On Linux

Aong Cho Marma

July 01, 2023

# Contents

# Chapter 1

# INTRODUCTION

## 1.1 Basic Linux Introduction

### 1.1.1 What is shell?

The shell is a program that takes command from the keyboard and gives them to the operating system to perform. Shell is also known as **CLI(Command Line Interface).**

### 1.1.2 Terminal

A terminal is a tool which is used to pass the shell command. It is a program that open a terminal and interact with the shell.
To open terminal

```
Ctrl + Alt + t
```

### 1.1.3 File System in Linux

Like all other operating system, the files on Linux System are also arranged in hierarchical directory structure. Which means they are organized in a tree like pattern of directories. Directories may contain files and other directories also.
The main or basic directory on Linux System is root directory. That contains all file and folders of that particular Linux Operating System.

# Chapter 2

# Basic Commands

## 2.1 Basic Linux Commands

Basic Linux Commands and their uses descriptions are discussed in this section.To know details about any command -

Command: `man command`

Example: `man ls`

### 2.1.1 `pwd` Command

*Present Working Directory.* Whenever open a terminal by default **pwd** command shows

```
/home/user_name
```

### 2.1.2 `ls` Command and related flags

*Content list of present working directory.*

- **Without flags:** It normally shows the contents of the present working directory.
  Command: `ls`
- **With directory flag:** It shows the contents of the given directory
  Command: `ls Documents/`
- **With a forward slash:** It shows the contents of the root directory
  Command: `ls /`
- **With a tilled sign:** It shows the contents of the home directory
  Command: `ls ~`
- **With (..):** It shows the content of one step backward directory
  Command: `ls ..`
- **With (../..):** It shows the content of two steps backward directory
  Command: `ls ../..`

- **With (-l):** List out the items in long format

  Command: `ls -l`

- **With (-a):** List out the items including hidden files

  Command: `ls -a`

- **With (-R):** To see directory structure.

  Command: `ls -R`

- **With (-al):** List out the items including hidden files in long format

  Command: `ls -al`

- **With (-ls):** List out the items with file size in long format.

  Command: `ls -ls`

- **With (-*.file_format):** List out the items for specific format

  Command: `ls -*.html`

  This command will list out all the .html files.

- **ls -lS > file_name.txt:** List out the items sorted in long format in a text file

  Command: `ls -lS > out.txt`

- **With ( -d */):** List out the directory only

  Command: `ls -d */`

- **man ls:** More information about `ls` command

  Command: `ls -d */`

### 2.1.3 `cd` Command (Change Directory)

**`cd`** command is used to change current working directory in Linux

- [ **`cd /`** ] **:** Change to root directory

- [ **`cd ~`** ] **:** Change to home directory

- [ **`cd ..`** ] **:** Go to parent directory of current working directory

- [ **`cd /home/userName/directoryName`** ] **:** Change directory with absolute path.

- [ **`cd directoryName/`** ] **:** Change directory with relative path.

- There are three ways to change directory when directory name contains space.

  – **`cd directory\ Name`**

  – **`cd "directory Name"`**

  – **`cd 'directory Name'`**

### 2.1.4 `mkdir` [Make Directory] Command

- ◇ [ `mkdir directoryName` ]: Simply make a directory.
- ◇ [ `mkdir "directory Name"` ]: Make a directory, name contains space.
- ◇ [ `mkdir parentDir/subDir` ]: Make a sub-directory in a parent-directory.

  **Note:** Parent directory must be exist.
- ◇ [ `mkdir -p parentDir/subDir` ]: Make a sub-directory in a parent-directory when parent-directory doesn't exist.

  **Note:** -p is short form of −−parent
- ◇ `mkdir -p parentDir/{subA,"sub B","sub C"}` Make multiple sub-directory into a parent-directory.

  **Note:** There must be no spaces between the sub-directory names. It must contains only comma. And double quote if sub-directory's name contains space.

### 2.1.5 `cat` Command

Concatenate file(s) to the standard output.

Syntax: `cat [OPTION]... [FILE]...`

- ◇ [ `cat` ]: To echo (read) program. Press `Ctrl+D` to exit the `cat` command.
- ◇ [ `cat fileName.txt` ]: It will display the contents of the file.
- ◇ [ `cat -b fileName.txt` ]: Display the contents of the file and add line number with the none blank lines.
- ◇ [ `cat -n fileName.txt` ]: Display the contents of the file and add line number with including blank lines.
- ◇ [ `cat -s fileName.txt` ]: Display the contents of the file and squeezes blank lines to single blank line.
- ◇ [ `cat -E fileName.txt` ]: Display the contents of the file and add $ sign at the end of each line.

## 2.2 Redirection in Linux

Redirection means capturing output from file, command, or program and sending it as an input to another file, command, or program. There are many options to redirection in Linux.

### 2.2.1 `cd` as Re-director

Create a file named test_1.txt and write something in that file.

Command: `cat > test_1.txt`

The command will give a space to write something. Enter `Ctrl + D` to get out from the cat command.

To append more content in test_1.txt file

Command: `cat >> test_1.txt`

Concatenate multiple files into a single file.

Command: `cat file1.txt file2.txt > file3.txt`

Here *file1.txt* and *file2.txt* will be concatenated to *file3.txt* file.

## 2.3 Remove File and Directory

`rm` command is used for removing file and `rmdir` command is used for removing directory.

### 2.3.1 `rmdir` (Remove Directory) Command

Remove directory.

Syntax: `rmdir -options directory`

◇ `rmdir dirName/` Simply remove a directory if the directory is empty.

◇ `rmdir -p a/b/c` Remove the whole directory structure.

**Note:** Use -v (verbose) option to output a diagnostic for every directory processed.

**Note:** `rmdir` command works only when the directory doesn't contain any file(s) but may contain empty directories.

### 2.3.2 `rm` (Remove) Command

Remove files and directories.

Syntax: `rm [OPTION]... [FILE]...`

◇ [ `rm fileName` ]**:** Simply remove a file.

◇ [ `rm -r dirName/` ]**:** Remove a directory.

**Note:** -r stands for −−recursive. This command will remove the directory even if the directory is not empty.

## 2.4 `cp` (Copy) Command

Copy files and directories.

Syntax: `cp [OPTION] Source Destination`

◇ [ `cp SourceFile DestinationFile` ]**:** Here simply source-file will be copied to the destination-file.

◇ [ `cp SourceFile .` ]: Here source-file will be copied in present working directory.

**Note:** Dot(.) means Present working directory.

◇ [ `cp SourceFile1 SourceFile2 DestinationFile` ]: Here content of multiple source-file will be copied to the destination-file.

◇ [ `cp -r sourceDir destinationDir` ]: Here *sourceDir* will be copied recursively to the *destinationDir*

**Note:** If the *destinationDir* does not exist then the *destinationDire* will be create and only the content of the sourceDir will be transferred. Or if the *destinationDir* exist then the whole sourceDir will be transferred.

**Note:** `-i or --interactive` Prompt before overwrite.

## 2.5 `mv` (Move) Command

Rename SOURCE to DESTINATION, or move SOURCE to DIRECTORY

Syntax: `mv [OPTION] [SOURCE] [DESTINATION]`

### 2.5.1 Rename File and Directory

◇ **Rename File:**
Rename file1.txt to file2.txt

Command: `mv file1.txt file2.txt`

◇ **Rename File:**
Rename Dir1/ to Dir2/

Command: `mv Dir1/ Dir2/`

### 2.5.2 Move file or dir to the destination dir

◇ **Move a file to the destination dir**
Move file1.txt to Dir1/

Command: `mv file1.txt Dir1/`

◇ **Move multiple file to a Dir**
Move file1.txt and file2.txt to Dir1/

Command: `mv file1.txt file2.txt Dir1/`

◇ **Move dir to a destination dir**
Move dir1/ to dir2/

Command: `mv dir1/ dir2/` **Note:** If the destination dir does not exist then it will rename the dir.

**Note:** In case of moving file or dir the duplicate file or folder will be overwritten in the destination file or folder.Use flat `-i or --interactive` flag to prompt before overwrite.

## 2.6 `less` Command

Read and search patterns conveniently in a big file.

Command: `less file-Name`

This command will show the content from the start. Press `"q"` to quite from less command.

### 2.6.1 Basic User Manual of `less` command

◇ **Navigate line by line:** Use up-arrow (↑) and down-arrow (↓)

◇ **Navigate page by page :** Use `"Space-bar"` to navigate downward and use `"b"` key to navigate upward.

◇ **Go to the end or start :** To navigate end of the file use
**texttt"Shift + g"** and
**texttt"g"** to navigate start.

### 2.6.2 Searching with `less` command

◇ **Searching from up to down :** `/word-to-search`
It will how the first matched word and to navigate the others one by one press `"n"` key on keyboard.

◇ **Searching from down to up :** `?word-to-search`
It will show the last matched word and to navigate the others one by one press `"n"` key on keyboard.

## 2.7 `touch` Command

`touch` command used to create empty file or update time-stamps to current time.

Command: `touch file1.txt`

This command will create an empty file if file1.txt does not exist. If exist then it will update the time-stamps to current time.