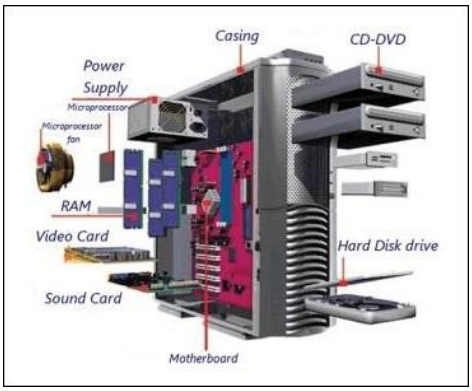
COMPUTER SCIENCE PAPER XI

Course Title: Linux Part I

Unit I Introduction to Linux



A **computer** is an electronic device that manipulates **information**, or data. It has the ability to store, retrieve, and process data. Every computer is composed of two basic components: **hardware and software**.

Hardware

The term hardware refers to mechanical device that makes up computer. Computer hardware consists of interconnected electronic devices that we can use to control computer's operation, input and output. Examples of hardware are CPU, keyboard, mouse, hard disk, etc.

Software

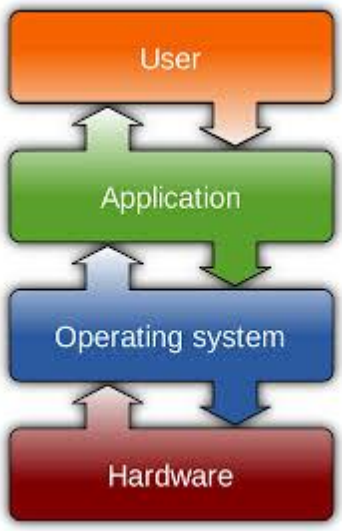
A set of instructions that drives computer to do tasks is called a program. Software instructions are programmed in a computer language, translated into machine language, and executed by computer. Software can be categorized into two types −

- System software
- Application software

System Software

System software operates directly on hardware devices of computer. It provides a platform to run an application and supports user functionality. Examples of system software include operating systems such as Windows, Linux, Unix, etc.

Application Software



Application software is designed for benefit of users to perform one or more tasks. Examples of application software include Microsoft Word, Excel, PowerPoint, Oracle, etc.

Definition of Operating System (O.S.)

An operating system is a program that acts as an interface between the user and the computer hardware and controls the execution of all kinds of programs.

Every computer must have at least one OS to run other programs. Applications like Chrome, MS Word, Games, etc needs some environment in which it will run and perform its task. It is not possible for the user to use any computer or mobile device without having an operating system.

Types of Operating system

- Multitasking
- Multi programming
- Multithreading
- Time Sharing OS
- Multiprocessing OS
- Network OS
- Mobile OS

Multitasking

As the name itself suggests, multi tasking refers to execution of multiple tasks (say processes) at a time. In the modern operating systems, we are able to play MP3 music, edit documents in Microsoft Word, surf the Google Chrome all simultaneously, this is accomplished by means of multi tasking. Most of the author use this term related with single user.

Multiprogramming

Multiprogramming is the ability of an O.S. to execute multiple programs at the same time on single processor machine. Most of the author use this term related with Multiple user.

Time-sharing Operating systems

Time-sharing operating system enables people located at a different terminal (shell or interface) to use a single computer system at the same time. The processor time (CPU) which is shared among multiple users is termed as time sharing.

Multiprocessing

In a uni-processor system, only one process executes at a time. Multiprocessing is the use of two or more CPUs (processors) within a single Computer system. The term also refers to the ability of a system to support more than one processor within a single computer system. Now since there are multiple processors available, multiple processes can be executed at a time.

Multithreading

A thread is like a small light-weight process within a process. Or we can say a collection of threads is known as a process. Threads represent a software approach to improving performance of operating system. Each thread belongs to exactly one process and no thread can exist outside a process. Each thread represents a separate flow of control. They also provide parallel execution of applications on shared memory. Most widely it is seen over the internet now days where we are using transaction processing of every type like recharges, online transfer, banking etc.

Network Operating System

Network Operating System runs on a server. It provides the capability to serve to manage data, user, groups, security, application, and other networking functions.

Mobile OS

Mobile operating systems are those OS which is especially that are designed to power smart phones, tablets. Some most famous mobile operating systems are Android and iOS, but others include BlackBerry.

Difference between Linux and windows:

S.NO	LINUX	WINDOWS
1.	Linux is a open source operating system.	While windows are not open source operating system.
2.	Linux is free of cost.	While it is costly.
3.	Download Linux and install it on as many machines as you like.	You are bound to the no. of licenses
3.	You have Online support of huge community.	You can purchase support from Microsoft itself.
4.	There is forward slash is used for Separating the directories.	While there is back slash is used for Separating the directories.
5.	Linux provides more security than windows.	While it provides less security than Linux.

History of Linux

*Linux is a community of open-source Unix like operating systems that are based on the Linux Kernel. It initially released by **Linus Torvalds** on September 17, 1991. It is a free and open-source operating system and the source code can be modified and distributed to anyone commercially or non commercially under the GNU General Public License.*

Linus Torvalds ..we can call him as the founder of Linux. He was a graduate student at University of Helsinki in Finland when he initially developed Linux kernel. In 1990's that time the available Operating Systems were DOS, Mac OS and UNIX.

Lets meet another guy named Prof. Tanenbaum who was a university Professor. He used the code of UNIX to teach his students

about Operating Systems, That time UNIX code was available on universities for study purposes. However this practice had been stopped. This left Mr.Tanenbaum without an effective tool to teach his students about the inner workings of a real Operating System. But he developed a small clone of UNIX by himself called MINIX for teaching his students. Linus Torvalds was inspired by Tanenbaum and MINIX . Then Linus made a clone of UNIX and called it LINUX. The first version of Linux was extremely minimal in nature. It named as Linux version 0.02 and released in 1991, consist of Linux kernel. Then he made a historical decision —he published his code on internet available free for everyone.

Let us discuss about GNU to continue the story…. GNU is a movement started by a programmer called Richard in 1980's. He proposed an alternative to the standard cooperate development model. In 1983 Richard came up with GNU project. It was centered on the idea Free and Open Source Software(FOSS) so that the source code of software should be available free for everyone and anyone can modify and reproduce with distribute it.

Now back to Linux. Linus and Richrard moved together and the OS started known as GNU-Linux. Linux itself licensed under GNU General Public License(GPL). The important factor about GPL is that the source code need to be remain free anyone who wants it. As a result of it anyone can download, modify and re-run Linux kernel free. As the time progressed it has been developed to such an extent that many companies replaced their OS completely to Linux in order to enhance security and strengthen it. In India Educational institutions like IIT s turned into LINUX. The reason behind the success is the transparent nature of Linux.

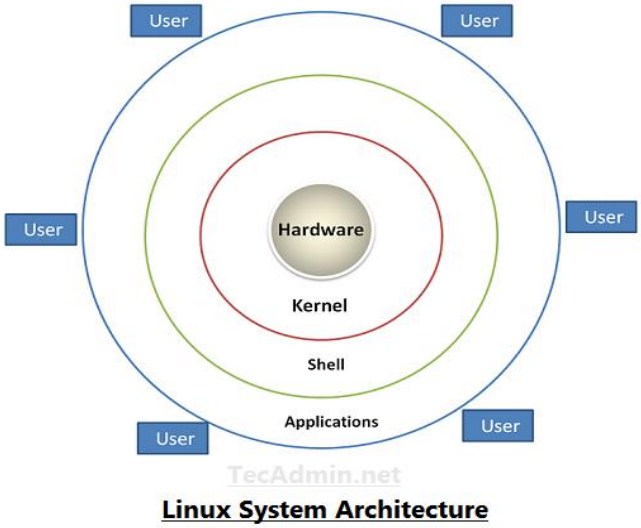
Linux System Architecture

Hardware − Hardware consists of all physical devices attached to the System. For example: Hard disk drive, RAM, Motherboard, CPU etc.

Kernel − Kernel is the core component for any (Linux) operating system which directly interacts with the hardware.

Functions of Kernel are

- Memory Management
- Processor Management
- Device Management
- Interrupt Handling
- Scheduling



Linux System Architecture

Shell − Shell is the interface which takes input from Users and sends instructions to the Kernel, Also takes the output from Kernel and send the result back to output shell.

Applications − These are the utility programms which runs on Shell. This can be any application like Your web browser, media player, text editor etc.

Operating System Services

- User Interface of Operating System
- Program execution
- File Management
- Communication
- Resource Management
- Security

User Interface of Operating System

Usually Operating system comes in types. Depending on the interface their types further subdivided.
These are:

- Command line interface
- Graphical User Interface

The command line interface (CLI) usually deals with using text commands and a technique for entering those commands.

Another type is the graphical user interface (GUI): which is a window system with a pointing device (like mouse or trackball) to point to the I/O, choose from menus driven interface and to make choices viewing from a number of lists.

Program execution

The operating system must have the capability to load a program into memory and execute that program. A process includes the complete execution context (code to execute, data to manipulate, registers, OS resources in use). Following are the major activities of an operating system with respect to program management −

- Loads a program into memory.
- Executes the program.
- Handles program's execution.
- Provides a mechanism for process synchronization.
- Provides a mechanism for process communication.

File Management

A file system is normally organized into directories for easy navigation and usage. These directories may contain files and other directions.

An Operating System does the following activities for file management

- Keeps track of information, location, uses, status etc. The collective facilities are often known as **file system**.
- Allocates the resources.
- De-allocates the resources.

Communication

In case of distributed systems which are a collection of processors that do not share memory, peripheral devices, or a clock, the operating system manages communications between all the processes. Multiple processes communicate with one another through communication lines in the network.

- Two processes often require data to be transferred between them
- Both the processes can be on one computer or on different computers, but are connected through a computer network.

Resource Management

In case of multi-user or multi-tasking environment, resources such as main memory, CPU and files storage are to be allocated to each user or job. Following are the major activities of an operating system with respect to resource management −

- The OS manages all kinds of resources using schedulers.
- CPU scheduling algorithms are used for better utilization of CPU.

Protection

Considering a computer system having multiple users and concurrent execution of multiple processes, the various processes must be protected from each other's activities.

Protection refers to a mechanism or a way to control the access of programs, processes, or users to the resources defined by a computer system. Following are the major activities of an operating system with respect to protection −

- The OS ensures that all access to system resources is controlled.
- The OS ensures that external I/O devices are protected from invalid access attempts.
- The OS provides authentication features for each user by means of passwords.

The Shell

The shell is a program that takes commands from the keyboard and gives them to the operating system(kernel) to perform. In the old days, it was the only command line interface available on a DOS, Unix. Nowadays, we have *graphical user interfaces (GUIs)* in addition to *command line interfaces (CLIs)* such as the shell. When the user logs in OS starts a shell for user.

On most Linux systems a program called bash (which stands for Bourne Again SHell, an enhanced version of the original Unix shell program, **sh**, written by Steve Bourne) acts as the shell program. Besides **bash**, there are other shell programs that can be installed in a Linux system. These include: **ksh**, **tcsh** and **zsh**.

Types of Shell:

1) The C Shell –

Denoted as csh

Bill Joy created it at the University of California at Berkeley. It incorporated features such as aliases and command history. It includes helpful programming features like built-in arithmetic and C-like expression syntax.

In C shell:

Non-root user default prompt is hostname %,

Root user default prompt is hostname #.

2) The Bourne Shell –

Denoted as sh

It was written by Steve Bourne at AT&T Bell Labs. It is the original UNIX shell. It is faster and more preferred. It lacks features for

interactive use like the ability to recall previous commands. It also lacks built-in arithmetic and logical expression handling. It is default shell for Solaris OS.

For the Bourne shell the:
Non-root user default prompt is $,
Root user default prompt is #.

3) The Korn Shell

It is denoted as ksh

It was written by David Korn at AT&T Bell Labs. It is a superset of the Bourne shell. So it supports everything in the Bourne shell. It has interactive features. It includes features like built-in arithmetic and C-like arrays, functions, and string-manipulation facilities. It is faster than C shell. It is compatible with script written for C shell.

For the Korn shell the:
Non-root user default prompt is $,
Root user default prompt is #.

GNU Bourne-Again Shell (bash**) –**

Denoted as bash

It is compatible to the Bourne shell. It includes features from Korn and Bourne shell.

For the GNU Bourne-Again shell the:
Default prompt for a non-root user is bash-$
Root user default prompt is bash-#.

Kernel

A Kernel is the central component of an Operating System. The Kernel is also said to be the heart of the Operating System. It is responsible for managing all the processes, memory, files, etc. The Kernel functions at the lowest level of the Operating System. It acts

as an interface (bridge) between the user-level application (software) and the hardware. Therefore, the communication between the software and the hardware is done via the Kernel.

The main functions that the Kernel performs are as follows:

1. Process Management
2. Memory Management
3. Device Management
4. Interrupt Handling
5. Scheduling

Process Management

The creation, execution and termination of processes keep on going inside the system whenever a system is in the ON mode. A process contains all the information about the task that needs to be done. The management of all these processes is very important for the proper functioning of the system, and it is handled by the Kernel.

Memory management

Whenever a process is created and executed, it occupies memory and when it gets terminated, it releases the memory, to use again. This task is also done by the Kernel. The kernel keeps track about which part of the memory is currently allocated and which part is available to the other processes.

Device Management

The Kernel also manages all the different devices which are connected to the system, like the Input and Output devices, etc.

Interrupt Handling

While executing the processes, there are conditions whether tasks with more priority need to be handled first. In these cases, the kernel has to interrupt in-between the execution of the current process and handle tasks with more priority which has arrived in between.

Scheduling

In a Multitasking, Multiprogramming and Time shearing system, the kernel will give every program a time and switch from process to process so quickly that it will appear to the user as if these processes were being executed simultaneously. The kernel uses Scheduling Algorithms to determine which process is running next.

Login

login is used when signing onto a system. It can also be used to switch from one user to another at any time. After these conditions have been checked, the password will be requested and checked.

This login prompt is generated by getty (get teletype) which is started by init process. getty is a program, running on system that manages physical terminal. When it detects a connection immediately shows login prompt.

init is a process having pid 1 (process id) which is responsible for the birth to all other process.

EXAMPLES

1. To log in to the system as user abc, enter the following at the login prompt

```
login: abc
```

If a password is defined, the password prompt appears. Enter your password at this prompt.

2. Attempt to login to a domain

```
$ login xyz.com
```

Would attempt to login to the xyz domain.

Ten attempts are allowed before **login** dies, but after the first three, the response starts to get very slow. Login failures are reported via

the **syslog** facility. This facility is also used to report any successful root logins.

Logout

logout command allows you to programmatically logout from your session, causes the session manager to take the requested action immediately. It is opposite to login. You are telling that you have finished using it and you no longer access to your files at this time. It is very important that you logout when you have finished using the system or leave your computer. If you have yourself logged in, then someone else may access to all your files. They may read, modify or delete them.

How to log out?

There are several ways depended on shell in use, but they achieve the same thing. The best way is to use the command

%logout [Enter]

Login:

This usually work on the C shell.

Another way,

$[ctrl+d] [Enter]

Login:

This usually work on the bourne or bash shell.

Last way, if both not work then exit command should do the job

$exit [Enter]

Login:

It work on all other than C shell

General Purpose Utilities (GPU)

banner

Syntax;

$banner text

banner command in linux is used to print the ASCII character string in large letter to standard output.

Example 1: Printing "1234567890" in large letters.

```
anshul@anshul-VirtualBox:~$ banner 1234567890
   #     #####   #####  #       #######  #####  #######  #####   #####    ###
  ##    #     # #     # #    #  #        #     # #    #  #     # #     #  #   #
 # #          #       # #    #  #        #           #   #     # #     # # #   #
   #     #####   #####  #######  #####  ######     #    #####   ###### #  #  #
   #    #           #       # #     #   #     #    #     #       # #   # #
   #    #       #     #     #  #     # #     #    #     #     # #     #  #   #
 #####  #######  #####       #   #####   #####    #     #####   #####    ###

anshul@anshul-VirtualBox:~$
```

Example 2: Printing "GeeksforGeeks" in large letters. There are two things:

- First, all the letter will be displayed in Capital letters in standard output.
- Second, only "GEEKSFORGE" will be printed as banner has a default capacity of 10 characters in a word. After that, you have to give space which is shown in further examples.
-

```
anshul@anshul-VirtualBox:~$ banner GeeksforGeeks
 #####                                                    #####
#     #  ######  ######  #    #   ####   ######   ####   #####  #     #  ######
#        #       #       #   #   #       #       #    #  #    # #        #
#  ####  #####   #####   ####     ####   #####   #    #  #    # #  ####  #####
#     #  #       #       #  #         #  #       #    #  #####  #     #  #
#     #  #       #       #   #   #    #  #       #    #  #   #  #     #  #
 #####   ######  ######  #    #   ####   #        ####   #    #  #####   ######
```

Example 3: Printing "Geeks for Geeks" on display.

cal/calender

If a user wants a quick view of calendar in Linux terminal, cal is the command for you. By default, cal command shows current month calendar as output.

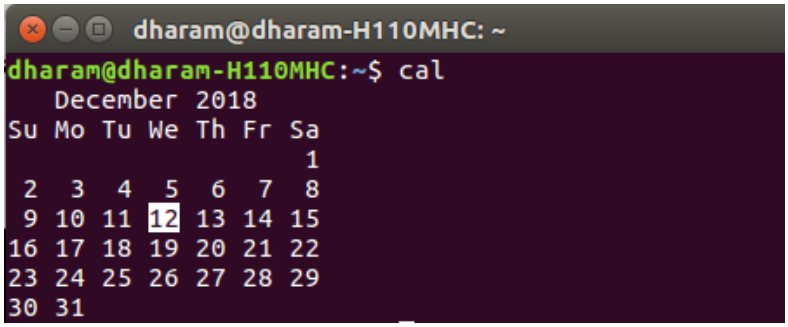
cal command is a calendar command in Linux which is used to see the calendar of a specific month or a whole year.

Syntax:

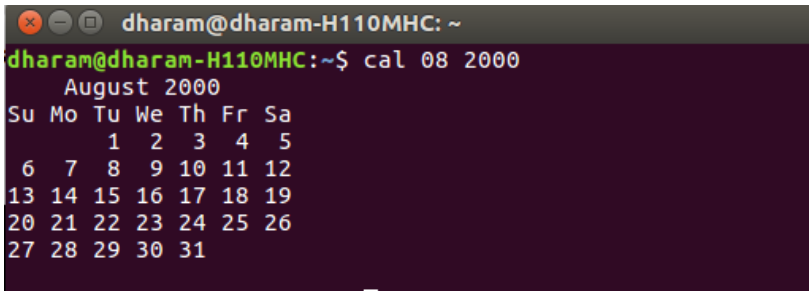
```
cal [ [ month ] year]
```

Rectangular bracket means it is optional, so if used without option, it will display a calendar of current month and year.

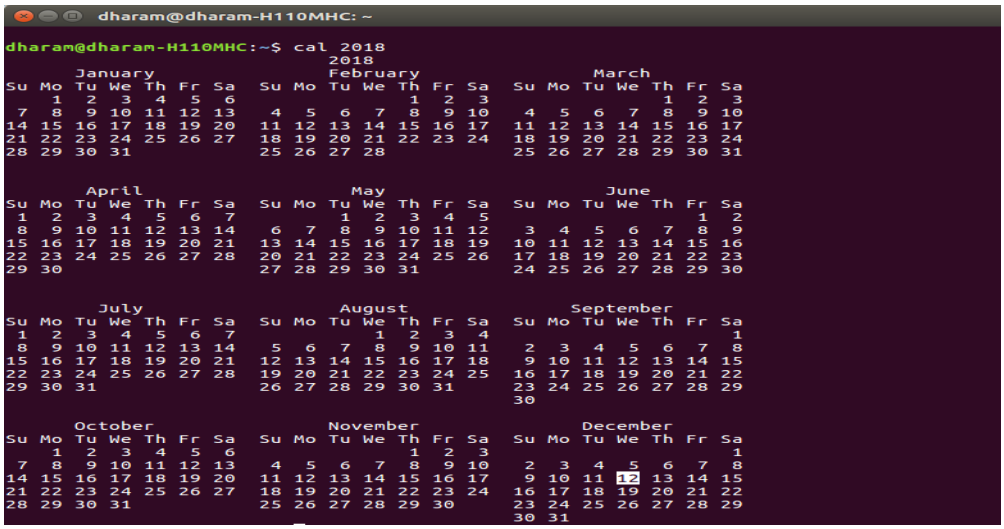
- **cal :** Shows current month calendar on the terminal



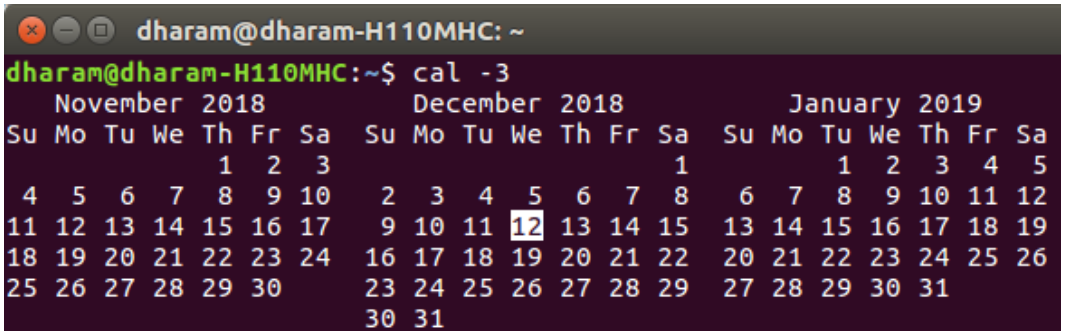
cal 08 2000 : Shows calendar of selected month and year



cal 2018 : Shows the whole calendar of the year



cal -3 : Shows calendar of previous, current and next month



date

date command is used to display the system date and time. date command is also used to set date and time of the system. By default the date command displays the date in the time zone on which unix/linux

operating system is configured. You must be the super-user (root) to change the date and time.

Syntax:

```
date [OPTION].
```

With no options, the date command displays the current date and time, including the abbreviated day name, abbreviated month name, day of the month, the time separated by colons, the time zone name, and the year.

Command:

```
$date
```

Output:

```
Tue Oct 10 22:55:01 PDT 2019
```

bc (basic calculator)

bc command is used for command line calculator. It is similar to basic calculator by using which we can do basic mathematical calculations.

When invoked bc without arguments, the cursor keeps on blinking and nothing seems to happen. Key in the following arithmetic expression and then use [ctrl+d] to quit bc:

```
$bc
12 + 5
17
[ctrl+d]
```

To do multiple calculations use the ; as delimiter.

```
$bc
12*12 ; 2^3
144
8
```

who

The who command displays the following information for each user currently logged in to the system, if no option is provided :

1. Login name of the users
2. Terminal line numbers
3. Login time of the users in to system
4. Remote host name of the user

```
hduser@mahesh-Inspiron-3543:~$ who
hduser   tty7         2018-03-18 19:08 (:0)
```

tty

Linux treats even terminals as files, it's reasonable to expect a command that tells you the filename of the terminal you are using. It's the tty command, an obvious reference to the device that has now become. The command is simple and need no arguments.

```
$tty
/dev/pts/10
```

The terminal filename is 10 residents in the pts directory. This directory in turn is under the /dev directory. You can use tty in a shell to control the behavior of the script depending on the terminal it is invoked from.

uname

uname is a command-line utility that prints basic information about the operating system running on your machine. By default, it simply displays the name of the operating system.

$uname

```
Linux
```

The output depends on system you are running

To find out the version of your operating system use option –r
$uname –r
3.11.0-14 generic

If your machine is connected to network , it must have a name. If your network is connected to the internet, then this hostname is a component of your machine domain name. The –n option tells you the hostname.
$uname –n
Rbm

passwd

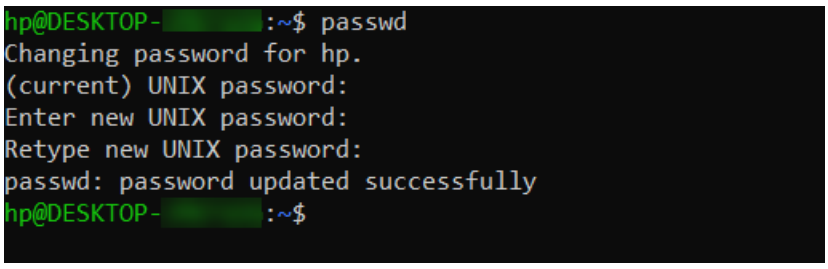
passwd command in Linux is used to change the user account passwords. The root user reserves the privilege to change the password for any user on the system, while a normal user can only change the account password for his or her own account.

Syntax:

passwd [options] [username]

Example:

Command: passwd

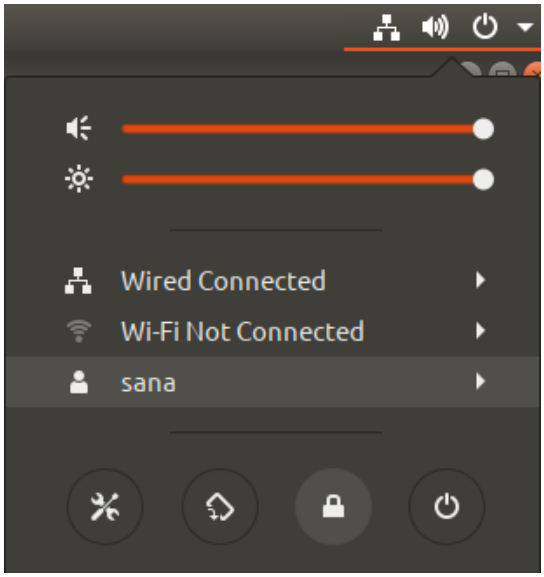


Lock

When you have to take a break from work but do not want to end your current session, locking your computer screen is the way out. You can, of course, shut down your system and start it again when you are back, but that means you have to save all your work and open all in the same state when you need to. Why option for this second option when your system allows you to temporarily lock your system so that

unauthorized people cannot access it. User can lock Ubuntu screen is through the Settings Panel that can be accessed from the top bar of your Ubuntu desktop.

Click on the downward arrow located at the top right corner of your Ubuntu screen. This is the panel that will appear:



You can lock your computer screen through the Lock icon, which by default is the second last icon located at the bottom on Ubuntu. As soon as you click on this icon, your computer screen will be locked.

Tput

The tput command can make your shell scripts pretty interesting. With tput command you can output text or warning messages in bold effect, underline effect, reverse the highlight effect, set cursor position, and much more.

$tput cup 0 0

Send the sequence to move the cursor to row 0, column 0 (the upper left corner of the screen, usually known as the "home" cursor position).

$tput clear

Echo the clear-screen sequence for the current terminal.

$tput cols

Print the number of columns for the current terminal.

clear

Sometimes, while working on the command line, you arrive at a point where there's too much text on the terminal screen, and none of that is relevant to you. So, in order to avoid distraction, you'd want to clear the terminal screen. Those new to the Linux command line may not know that there exists a dedicated command line utility that does this work for you. clear is a standard Linux computer operating system command that is used to clear the terminal screen. Also, the clear command doesn't take any argument and it is almost similar to cls command on a number of other operating systems.

Syntax: $clear

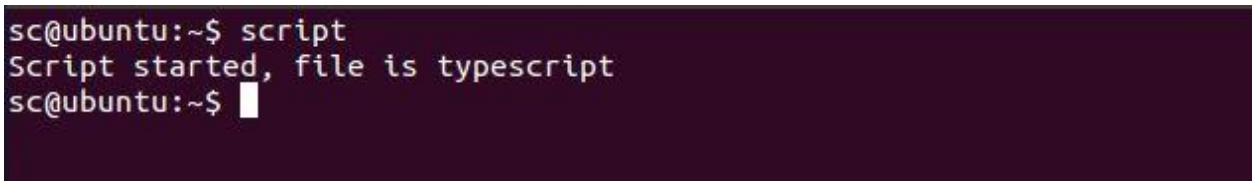
script

script command in Linux is used to make record all the terminal activities. After executing the script command it starts recording everything printed on the screen including the inputs and outputs until exit. By default, all the terminal information is saved in the file typescript , if no argument is given. script is mostly used when we want to capture the output of a command or a set of command while installing a program or assignment.

Syntax:

```
script [options] [file]
```

Example 1: To start a typescript without any argument. If no filename is given as argument, script will automatically create a file namely typescript in the home directory to save the recorded information.



In order to stop the typescript, we just need to execute *exit* command and script will stop the capturing process. Since there's no filename given as argument, the script will automatically create a file namely *typescript* in the home directory to save the recorded information.

You can also specify filename to script.

$ script assignment
Script started, file is assignment.
$

wc

wc stands for word count. As the name implies, it is mainly used for counting purpose.

- It is used to find out number of lines**,** word count**,** byte and characters count in the files specified in the file arguments.
- By default it displays four-columnar output**.**
- First column shows number of lines present in a file specified, second column shows number of words present in the file, third column shows number of characters present in file and fourth column itself is the file name which are given as argument.

Syntax:

$wc [OPTION]... [FILE]...

Let us consider two files having name **state.txt** and **capital.txt** containing 5 names of the Indian states and capitals respectively.

Let us consider two files having name **state.txt** and **capital.txt** containing 5 names of the Indian states and capitals respectively.

$ cat state.txt
Andhra Pradesh
Arunachal Pradesh
Assam
Bihar
Chhattisgarh

$ cat capital.txt
Hyderabad
Itanagar
Dispur
Patna
Raipur

Passing only one file name in the argument.

$ wc state.txt
5 7 63 state.txt
OR
$ wc capital.txt
5 5 45 capital.txt

Passing more than one file name in the argument.

$ wc state.txt capital.txt
5 7 63 state.txt
5 5 45 capital.txt
10 12 108 total

Note : When more than file name is specified in argument then command will display four-columnar output for all individual files plus one extra row displaying total number of lines, words and characters of all the files specified in argument, followed by keyword **total**.

Head

It is the complementary of Tail command. The head command, as the name implies, print the top N number of data of the given input. By default, it prints the first 10 lines of the specified files. If more than one file name is provided then data from each file is preceded by its file name.

Syntax:

head [OPTION]... [FILE]...

Let us consider file name **state.txt** contains all the names of the Indian states respectively.

$ cat state.txt
Andhra Pradesh
Arunachal Pradesh
Assam
Bihar
Chhattisgarh
Goa
Gujarat
Haryana
Himachal Pradesh
Jammu and Kashmir
Jharkhand
Karnataka
Kerala
Madhya Pradesh
Maharashtra
Manipur
Meghalaya
Mizoram
Nagaland
Odisha

Punjab
Rajasthan
Sikkim
Tamil Nadu
Telangana
Tripura
Uttar Pradesh
Uttarakhand
West Bengal

Without any option, it displays only the first 10 lines of the file specified.
Example:

$ head state.txt
Andhra Pradesh
Arunachal Pradesh
Assam
Bihar
Chhattisgarh
Goa
Gujarat
Haryana
Himachal Pradesh
Jammu and Kashmir

-n num: Prints the first 'num' lines instead of first 10 lines. **num** is mandatory to be specified in command otherwise it displays an error.

$ head -n 5 state.txt
Andhra Pradesh
Arunachal Pradesh
Assam
Bihar
Chhattisgarh

Tail

It is the complementary of head command. The tail command, as the name implies, print the last N number of data of the given input. By default it prints the last 10 lines of the specified files. If more than one file name is provided then data from each file is precede by its file name.

Syntax:

tail [OPTION]... [FILE]...

Let us consider file name state.txt contains all the names of the Indian states respectively.

$ cat state.txt
Andhra Pradesh
Arunachal Pradesh
Assam
Bihar
Chhattisgarh
Goa
Gujarat
Haryana
Himachal Pradesh
Jammu and Kashmir
Jharkhand
Karnataka
Kerala
Madhya Pradesh
Maharashtra
Manipur
Meghalaya
Mizoram
Nagaland

Odisha
Punjab
Rajasthan
Sikkim
Tamil Nadu
Telangana
Tripura
Uttar Pradesh
Uttarakhand
West Bengal

Without any option it display only the last 10 lines of the file specified. Example:

$ tail state.txt
Odisha
Punjab
Rajasthan
Sikkim
Tamil Nadu
Telangana
Tripura
Uttar Pradesh
Uttarakhand
West Bengal

-n num: Prints the last 'num' lines instead of last 10 lines. **num** is mandatory to be specified in command otherwise it displays an error. This command can also be written as without symbolizing 'n' character but '-' sign is mandatory.

$ tail -n 3 state.txt
Uttar Pradesh
Uttarakhand
West Bengal

OR

$ tail -3 state.txt
Uttar Pradesh
Uttarakhand
West Bengal