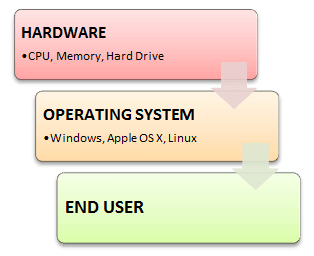
**An Operating System (OS)** acts as an interface connecting a computer user with the hardware of the computer.

**An operating system** falls under the category of system software that performs all the fundamental tasks like file management, memory management, process management, interrupts handling, device drives, networking, security manage and I/O management.



**Types of Operating system**

1. **Single user systems:**Provides a platform for only one user at a time. These are popularly associated with Desk-Top-Operating System which run on standalone systems, where no user accounts are required.

Ex. DOS

1. **Multi user systems:**

More than one user can access same system resources (Cpu, applications, memory, printers, etc) at the same time known as multiuser.

Ex. UNIX, LINUX

1. **Uni tasking**:

It allows one program at a time.

1. **Multi tasking:**

More than one program can be run at a time

1. **A uniprocessor system** is defined as a computer system that has a single central processing unit that is used to execute computer tasks.
2. **Multiprocessing system** is the use of two or more central processing units (CPUs) within a single computer system

**LINUX HISTORY**

* 1969, Dennis Ritchie and Ken Thompson developed the C language and the UNIX operating system at AT& T Bell Labs and shared the source code with others.
* 1975, AT &T started selling **Unix** commercially, about half of source code was written by others.
* Legal battle resulting two versions of **UNIX** in the Seventies: The official AT &T Unix, and the free **BSD** Unix.
* 1991, Linus Torvalds developed LINUX operating system by adding new features to the existing UNIX operating system.
* Linus Torvalds - computer science student in finland university.
* LINUX is completely free operating system.
* 1995, Linux is more and more popular on server systems
* 2000, Linux is more and more popular on embedded systems
* 2008, Linux is more and more popular on mobile devices
* 2010, Linux is more and more popular on phones

**What is Free Open Source Software (FOSS)**

A program is considered free when its license offers to all its users the following four freedoms:

* Freedom to run software for any purpose
* Freedom to study, update, delete and modify the existing source code.
* Freedom to re-distribute copies
* Freedom to distribute copies of modified versions
* These freedoms are granted both commercial and non-commercial use
* They imply the availability of source code, software can be modified and distributed to customers.

**ARCHITECTURE OF LINUX OPERATING SYSTEM:**





USER

SHELL

KERNEL

HARDWARE

**USER:**

The usernothing but an individual who uses the available hardware and software resources.

**SHELL:**

Shell is a command language interpreter that executes commands read from the standard input device(keyboard) or any file.

**Shell** is a program, which accepts user instructions and commands and passes to the kernel.

**Types of Shell:**

**Shell name Developed by Prompt Interpreter name**

Bourne Shell Stephen Bourne $ sh

Bash shell Stephen Bourne $ bash

Korn shell David Korn $ ksh

Z Shell Paul $ zsh

C Shell Bill Joy % Csh

**NOTE:**

The advanced version of Bourne shell is Bash Shell . Bash means Bourne again shell.

**Default Shell Name** **Flavour Name**

Bash Shell Linux

Bourne Shell Sco-Unix, Solaries, HP-UX

Korn Shell IBM-AIX

C Shell IRIX

**KERNEL:**

* + The kernel is the heart of the Operating System.
  + The Kernel is responsible for interacting with the hardware and output to the screen.
  + It handles the process, memory, file, device and network management for the Operating System.
  + LINUX is truly just the kernel.

**Features of UNIX/LINUX**

1. **Multi user systems:**

A Multi user operating system allows more than one user to share computer system at the same time.

1. **Multi-tasking:**

More than one program can be run at a time. The main concept of multi-tasking is maximum utilization CPU resources

1. **Open System: The UNIX is open source code i.e., any user can modified UNIX open source code according their ideas and requirements.**

**Using UNIX open source code**

SUN Micro systems + adding additional features = SUN solaries

IBM + “ = HP-UX

SANTA CURUZ + “ = SCO-UNIX

SILICON GRAPHICS + “ = IRIX

MICRO SOFT + “ = XENIX

Any operating system developed based on UNIX open source code known as flavours of UNIX.

Linux was given to GPL(General Public License) organized by GNU.

**Distributors of LINUX**

REDHAT SUSE UBUNTU PUPPY SLAKWARE CENTOS OEL

FEDORA WHITE BOX MANDRAKE

**iv. SECURITY:**

Security povides the virus proection of all the files and directories.

**UNIX/LINUX has given two levels of securities**

1. **System level security:** its controlled by system administartor
2. **file level security:** Its controlled by owner of the file

**v.** **portability:** portability means independent of hardware and processor

**vi.** **communication:** the main concept of communication facility exchangeing information or files from one user account to another account.

**Vii. Programming facility:** UNIX o.s provides shell. Shell works like a programming language. It provides commands and keywords.

**Viii. Help facility:** it is beautiful feature of UNIX/LINUX operating system. Dont know the information about given command just gothrough the help.

ex. #man <command name> or

#info <command name> or

# <command name>

**File system:** A file system is a logical collection of files on a partition or disk.

File hierarchy system:

/(slash)

root home bin sbin boot etc usr

opt dev var mnt lib tmp

**/(slash):** this is top level working directory. It is parent directory for all other directories. It is called as ROOT directory. It is represented by forward slash(/)

**root:** it is home directory for root user(super user. It is provides working environment for root user.

**home:** it is home directory for others except root

bin(binary files): it contains commands used by all users

**sbin(super user binary files):** it contains commands used by only super user(root)

**boot:** it contains system bootable files, bootloader information, kernel related information for LINUX.

**etc:** it contains all system configuration files

etc/hosts, /etc/resolv.conf

**user:** by default softwares are installed in /user directory(UNIX sharable resources)

**opt:** it is optional directory for users. It contains third pary softwares

**dev:** it contains all device files information. Similar to device manager of winows. In linux/unix every device treated as file.

**Var:** it is containing variable files information like mails, pring, log files.

**mnt:** it is default removable media working directory. It is empty by default

**lib:** it contains library files which are used by os. It is similar to dll files of windows. Library files in linux are so(shared object) files

**tmp:** it contains temporary files information

**media:** it contains all of removable media like cd-rom, pendrive

**proc:** it contains process files

it contains are not permanent, they keep changing

its also called as virtual directory.

Its file contain useful information used by o.s

like /proc/meminfo.....

/proc/cpuinfo.......

**basic commands:**

**creating files:**

**1. cat(concatenate):** it is used to createa file and display, appending the contents of a file.

$cat>filename press enter

helloworld

ctrl+d ( to save the file)

**to display the content of the file**

$cat<filename (press enter)

or

$cat filename (press enter)

**to append the data in the existing file**

$cat>>filename (press enter)

-

-

ctrl+d (to save)

**touch:** to create multiple files but all are empty

$touch file1 file2 file3 (press enter)

ex. $touch file2 file2 file3 (press enter)

ls: display the contents of a directory

$ ls (options)

**options:**  -r reverse -i inode -a hidden

-l long list -R recursively -h human readable

**mkdir:** creates a directory $

$mkdir <directory name>

$mkdir linux (press enter)

**to create multiple directories:**

$mkdir dir1 dir2 dir3...dirn

**navigation commands:**

**cd: changes the current directory**

1. **to go one level back**

**cd../.. to go two levels back**

**cd to change users home directory**

ex. $cd world (press enter)

**cp:** copies files or directories from one location to another

cp (options) source destination

-R copies recursively

-f copies forcefully

-v provides verbose outpot

ex. $cp file1 file2 (press enter) one file to another

**mv:** moves or renames files and directories

$mv (options) source destination

-v verbose

**rename the file by specifying the file name and new name of the file**

$mv messages messages.bak (press enter)

**move it to the test directory for safe keeping**

$mv messages.bak test

$ls test (press enter)

**rm:** deletes files or directories

rm (options) file

-i interactive -r recursively -f forcefully

**delete the messages.bak file**

$cd test (press enter)

$rm -i messages.bak (press enter)

**delete the test directory:**

**$rm -rf test /**(press enter)

**pwd:** present working directory

$ pwd

$ /home/user1

**echo: outputs or displays a string**

$echo “this is same sample text”> file\_example (press enter)

**to output same text to a file:**

$echo “this is same sample text”> file\_example (press enter)

$cat file\_example (press enter) (press enter) to verify

**wc: provides a word or line count**

wc options filename

-l lines -w words -c charters

$wc filename displays lines, words and charters

$wc -l filename (press enter) only lines

$wc -w filename (press enter) only words

$wc -c filename (press enter) only charters

**head: displays top 10 lines of the file**

$head sample (press enter)

$head -5 sample (press enter) top 5 lines

**tail: displays last 10 lines of the file**

$tail sample (press enter)

$tail -5 sample (press enter) last 5 lines

$tail -f sample (press enter) file is open continusly

**sort: sorts the output of a command or file**

sort options file

-r sorts in reverse order

-b ignores leading blanks

-n compares according to numerical string value

$sort example (press enter)

$sort -r example (press enter) reverse order

$sort -n example (press enter) display numeric

$sort -u example (press enter) unique lines

$sort -f example (press enter) ignores case

**sed:** (stream edition): to search and replace strings or patterns in the given file.

Sed “s/oldstring name/new string name/g” filename (press enter)

s- cutstitution

g-global occurance in every line

$sed -n “2p” filenme (press enter) to print 2nd row

$sed -n “3,5p” (press enter) to print 3rd, 4th, 5th rows

$sed '3d' (press enter) to delete 3rd row

$sed '3,5d' (press enter) to delete 3rd to 5th row

**find:** this filter is used to search the results by depending on requirements may be on name, inode, permission, user...etc.

Find <search path> <criteria> <action>

based on name

$find /-name passwd (press enter)

$find /-home -name passwd (press enter)

**SHELL SCRIPT**

**A shell script comprises following elements –**

* + Shell Keywords – if, else, break etc.
  + Shell commands – cd, ls, echo, pwd, touch etc.
  + Functions
  + Control flow – if..then..else, case and shell loops etc.

**Why do we need shell scripts**

There are many reasons to write shell scripts –

* + minizes typing of repetitive command
  + can schedule jobs to run in the system
  + System admins use shell scripting for routine backups
  + System monitoring
  + Adding new functionality to the shell etc.

**Shell types**

* + sh simple Shell(BOURNE)
  + BASH Bourne again Shell
  + KSH Korne Shell

**.** CSH C Shell

* + SSH Secure Shell

**To find current Shell type – echo $SHELL**

**TEXT EDITORS**

. vi editor

. vim editor

. nano editor

. gedit editor

**comparision**

**for integer comparision**

-eq equal to

-ne not equal to

lt less than

gt greater than

le less than or equal to

ge greater than or equalto

**For string comparision**

= equal to

~= not equal to

**For logical operator**

-a AND

-o OR

**with editor**

1. open a file with extension .sh using nano editor and press i then type any number of commands.//enter nano filename.sh on command prompt)
2. press i on editor for insert commands
3. Execute the program- press **esc+:wq**
4. output- sh filename.sh

**without editor**

1. open text editor
2. write sourcode(commands)
3. save the program with extension of .sh
4. minimize the text editor
5. execute program by sh filename.sh

**Write a program to find whether the entered number is 42 or not**

#!/bin/bash

echo -n enter the number:

read number

if [ $number -eq 42 ]

then

echo “42 is correct”

else

echo “number greater than 42”

fi

**PRESS esc+:eq**

**sh filename.sh**

#!/bin/sh // THIS IS PATH OF BOURNE SHELL