

## Introduction to the concept of OPEN SOURCE SOFTWARE

IN term "open source" refers to something that can be modified because its design is publicly accessible.

The term open source s/w is originated in the context of computer software development.

Open source projects, products or initiatives are those that welcome and celebrates open exchange.

### OPEN SOURCE SOFTWARE

"open source software is s/w whose source code is available for modification or enhancement by anyone."



Source code is the part of software that most computers users don't even see.

Only the computer programmers can manipulate or change a piece of software.

Only programmers who have access to a program's source code can improve that program by adding features to it or fixing bugs that come.

But in case of open source software any body can access source code, he or she can add new features, even can fix the bugs also.

"OPEN SOURCE SOFTWARE is software with source code that anyone can inspect, modify and enhance."



## Difference between open source software and other types of s/w

Some s/w has source code, that can be modified only by the team, person or organization who created it, and maintain exclusive control over it.

This type of s/w is called proprietary or "closed source" s/w.

only the original authors of proprietary s/w can legally copy, inspect and alter that- software

And in - order to use that- software, computer user must agree ( usually by signing an agreement, which is displayed on the screen, when first time we install that- software ) that- they will not- do any harm to that- software.

for example -

- 1) Adobe Photoshop
- 2) office

→ open source s/w is different.  
Its authors makes it's source  
code available to others,  
who could copy it, learn  
from it, alter it, or  
share it.

for example -

- 1) Libreoffice
- 2) GNU Image Manipulation  
program



## Linux Distributions

Linux kernel originally created by Linus Torvalds. It did not provide the functionality of full commercial operating system.

To be really useful for the majority of organization, Linux also requires -

- 1) Networking utilities
- 2) System administration utilities
- 3) Documentation
- 4) Installation tools
- 5) Technical support information
- 6) H/w drivers
- 7) A graphical tools

Date \_\_\_\_\_  
Page \_\_\_\_\_

e) Graphical environment-

9) Personal productivity applications  
for eg. word processors  
or spreadsheets

given the way linux is developed we might wonder how many pieces of linux could be combined into a complete operating system.

→ Such a productized version of linux, which includes many of its components, installation tools, documentation is called a linux distribution.

→ A linux distribution has the linux kernel at its core, along with hundreds or thousands of additional programs that run on linux.



- A linux distribution makes it possible for non-programmers to install and use linux easily.
- A linux distribution is an operating system made as a s/w collection based on linux kernel.
- Almost all linux distributions are unix-like, the most notable exception is Android.
- Android does not include a command line interface and programs, made for typical linux distributions.
- more than six hundred linux distributions exist, over <sup>them</sup> three hundreds of those are in-active development, constantly being revised and improved.
- some popular distributions are

Fedora (Red Hat)

Open Suse (SUSE)

Ubuntu (Canonical Ltd)

Debian

slackware

mint

→ Linux works differently. The Linux operating system is not produced by a single organization.

→ Different organization and people work on different-different parts.

→ many applications like systems services, graphical programs, terminal commands



are developed independently from another. They all are open source & distributed in source code form.

what-distribution should we choose

Different Linux distributions are suited for different purpose.

~~what~~- which distribution you will select, depends on what we want to do with it and what are our personal preferences.

for example:-

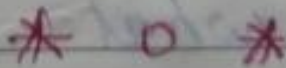
- 9) If we are a desktop user then we will like to use ubuntu or mint.

Need Name  
DATE \_\_\_\_\_  
Page \_\_\_\_\_

b) People looking for a more stable, well tested system may want to go with Debian.

→ Anyone can make their own distribution by assembling it from the source code themselves or even taking an existing distribution and modifying it.

That's why there are so many Linux distributions.





# ARCHITECTURE OF LINUX SYSTEM

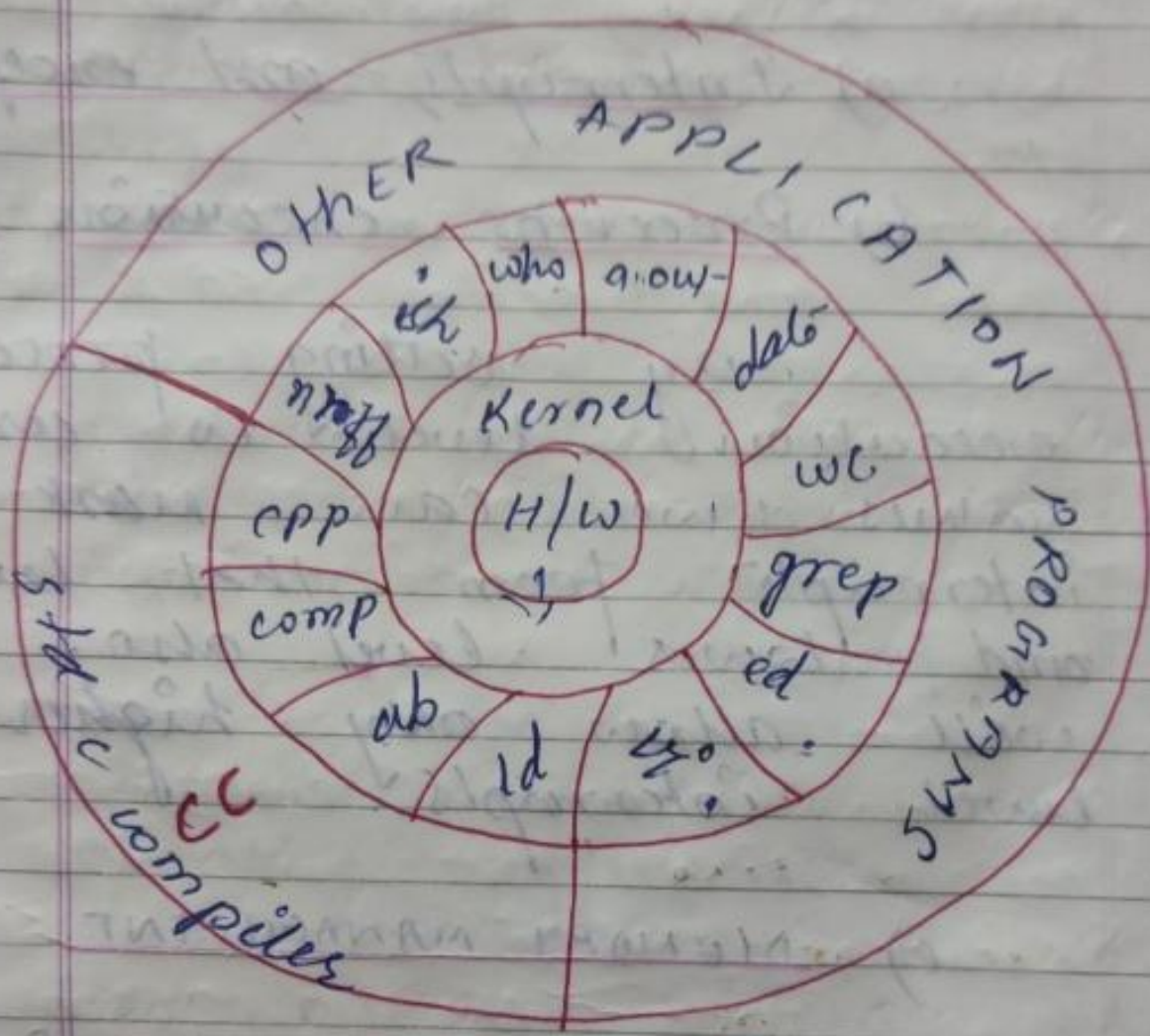


Fig: shows architecture of Linux system.

In the center of the diagram, is Hardware.

Thod Kumbhar  
DATE \_\_\_\_\_  
PAGE \_\_\_\_\_

The hardware provides basic services like -

a) Interrupts and exceptions

b) Processor execution level

By setting processor execution level to certain values we can mask-off interrupts from that level and from lower level also. It will allow only higher level interrupts.

c) MEMORY MANAGEMENT

Memory management is very important service.

The kernel permanently resides in main memory.

When a program needs to run on the machine the kernel allocates space in



Red Kameel  
Date: \_\_\_\_\_  
Page: \_\_\_\_\_

main memory, for it. physical and virtual addresses may be different.

A mapping process is required to match virtual address to physical address.

Mapping depends on the capabilities of machine h/w and the part of linux system deals with them

- These basic services are provided by H/w to the operating system.
- The o.s can interact directly with the hardware providing common services to programs.
- It separates application programs to h/w.
- Application programs can't interact with h/w directly



- with the help of o.s (or a specific part of os) (known as kernel) Application prog. interact with ~~kernel~~ H/W.
- programs such as the shell (sh) and editors (ed & vi) shown in outer layer interact with the kernel by invoking a well defined set of system calls.
- system can instruct the kernel to do various operations for the calling program and exchange data between the kernel and the program.
- Several programs shown in the figure like (cp, comp, grep, cat, wc, who) are commands.
- Privileged user programs can also exist in this layer.



for example: a.out

a.out standard name for executable file produced by C compiler.

→ Other application programs can build on top of lower level programs, as shown in outermost layer in the figure.

for eg:- CC standard 'C' compiler

→ In the figure, we have shown ~~only~~ two levels of application program, but if needed, user can extend this hierarchy to any no of levels.

→ There are about 64 system calls in which less than 32 system calls are used.

→ They have simple option, that make them easy to use but they provide lots of powers to user.

→ The set of system calls and the internal algorithms which implements system calls, form the body of the kernel.

→ The kernel provides the services upon which application program depends. The kernel defines those services.

\* o \*



A file system is a logical ~~partition~~ collection of files on a disk. A partition is a container for information and can span an entire hard drive if desired.

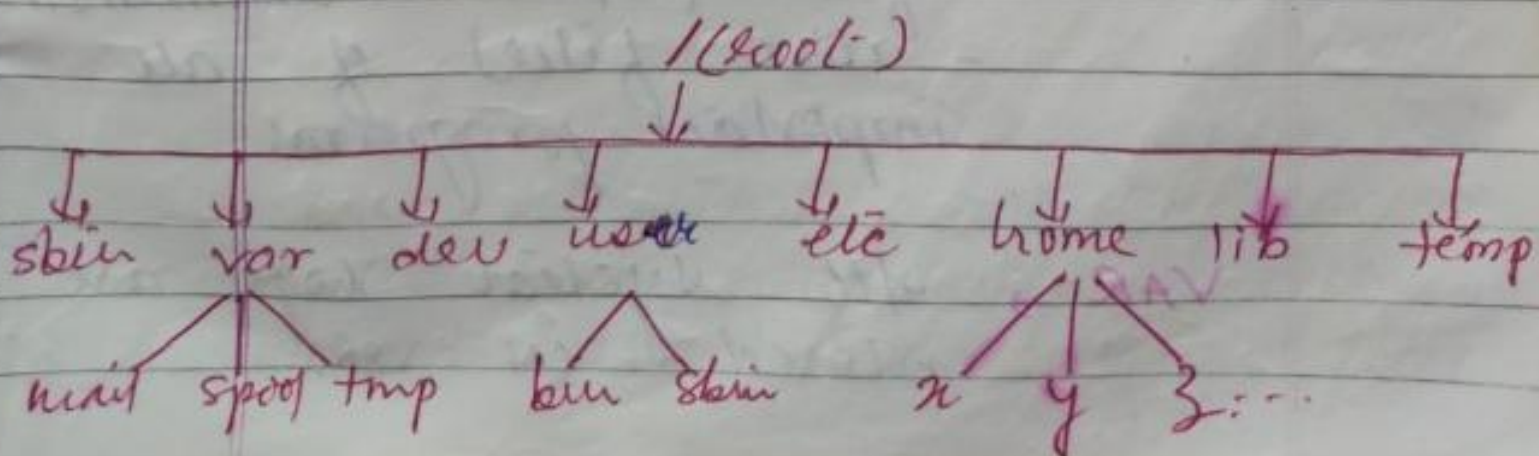
## LINUX FILE SYSTEM

Linux treats all information as a file. So files play an important role in Linux file system.

→ The Linux file system is a tree data structure, which resides on part of hard disk.

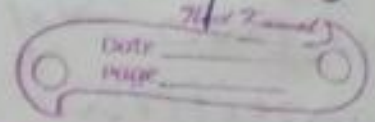
→ Almost the structure of Linux file system is similar to Unix file system.

→ The structure of Linux file system is as follows -





→ contains other files and dir



When we start the system two files 1) /root 2) /linux are read from the disk into the memory.

— details of files names are

bin → It includes all essential executables  
linux program files

sbin → It contains executable files which are specially used by super users.

home → contains sub-directories of the linux user.

etc → contains configuration files (like administrative files) of all important programs.

VAR → All system logs are stored in VAR.



uses → Use is the central location for all applications programs.

for example -

- 1) X - windows
- 2) MAN pages
- 3) documents etc.

lib → It contains all the library files which are shared.

tmp → temporary file location  
It contain all temporary files.

dev → It contain all special device files for keyboard mouse console etc.

→ Basically a file system is the method, which is used by operating system to keep track of files



Roll Number \_\_\_\_\_  
Date \_\_\_\_\_  
Page \_\_\_\_\_

on disk.  
i.e. how files will  
organized.

→ A file system is a group of files and like unix Linux chooses to have a single hierarchical file structure.

→ Everything starts with root, which is represented by '/' and then expand into sub directories instead of having drivers.

→ In windows one may put one's file almost anywhere for example - C, D or E drive

→ In windows operating system file system is called hierarchical structure, and it is managed by the program itself, not by the operating system.



- But in linux directories are stored in descending order from the root directory, according to their importance to the boot process.
- In linux everything is treated as file. For eg. application utility, data, directory or device, all are considered as a file.
- Linux file system has a tree like structure and referred as directory tree.
- The directory tree is much like a family tree where each directory has a parent, only exception is root (\) directory. Root has no parent directory.
- Different directories are required because directories



keep related files together  
 and separate from another  
 group of related files.

### Absolute Path

An absolute path starts with the root directory and follows the directory tree, branched by branch until the path to the desired directory file is completed.

for example.

/home/john/documents/a.txt



## Super Block

Super block describes the state of a file system. For example

- It describes how large it is.
- How many files it can store.
- Where to find free space on the file system.
- Some other information.

The super contains following fields —

1. The size of the file system
2. The no. of free blocks in the file system.
3. A list of free blocks available on the file system
4. Index of next free block in the free block list.

5. Size of Inode List.
6. No. of free inode in the file system.
7. A list of free inode in the file system.
8. The index of the next free inode is the free inode list.
9. Lock field for the free block and free inode list.
10. A field indicating that super block has been modified.

\* 0 \*



## INODE (INDEX - NODE)

In Linux O.S., in side the file system, the files are not accessed by their names, Because names are helpful to human not for O.S.

So the file system recognizes a file not by its name but by a no. "The no. through which the O.S. searches at the location and other attributes of file is called as an Inode."

The internal representation of a file is given by inode.

Inode contains description of disk layout of the file data and other information such as the file owner, access permission, access time, etc.



The term INODE is the contraction of the term index node.

Every file has single inode but it can have many names. But all names map into single inode. Each name is like a link.

→ Inode exist in a static form on disk, and kernel reads them into incose inode to manipulate them.

→ The disk inode contains following fields -

a) File owner identifier

→ It identifies who is owner of the file.

→ ownership is basically



Red Kandy  
Date: \_\_\_\_  
Page: \_\_\_\_

divided into two types - one is individual owner and the other is group owner. It also defines set of users who have access right to a file.

b) File TYPE:- Defines type of a file i.e. what type of file it is -

File can be of many types like -

- a) Regular file
- b) Directory file
- c) character file
- d) block special file
- e) pipe (PIFO)

c) File Access Permission:-

With the help of file access permission the system protects files

according to three classes.

- a) owner
- b) group owner
- c) other user.

Each class has right to read the file.

---

## PROCESS IN LINUX (INTRODUCTION)

A process is simply an instance of a running program. A process is said to be born when the program starts execution, and remains alive as long as the program is active.

The process is identified by the process-id (PID), and every ~~start~~ process has a parent whose PID is also available.

The login shell is the only



Roll Number \_\_\_\_\_  
Date \_\_\_\_\_  
Page \_\_\_\_\_

user process that keeps running as long as the user is logged in. Its PID is stored in the parameter \$\$.

[A] The sh process when we login to a system, a process is immediately set up by the kernel.

This is tech a command which may be sh. Any command which we type at the command prompt is input to the shell process (sh). And this process remains alive until we log out.

\$ echo \$\$

It will give process no. of the current shell.

The PID of your login shell does not change as long as we are logged in.

When we log out- and logged-in again, the login shell will be assigned a different PID.





## History of Linux system

Linux torvalds, we can call him founder of linux. He was a graduate student at university of Helsinki in Finland when he initiated developed linux kernel.

The story took place in 1990's. In that time the available O.S were DOS, MACOS and UNIX.

Let's meet another guy named prof. Tanenbaum who was a university prof. Tanenbaum used the code of UNIX to teach his student about operating system.

That time UNIX code was available at universities for study purposes. However this practice was stopped soon and so Mr. Tanenbaum developed a small clone of UNIX by



himself called MINIX for teaching his students.

Linus Torvalds was inspired by Thompson 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 on Oct-5, 1991.

It consist of linux kernel, and 3 utilities —

1) bash - a command line interface.

2) update - a utility to flushing file stream

3) gcc - a C++ compiler

After this Linus made a



Red Kaur  
Date - \_\_\_\_\_  
Page - \_\_\_\_\_

historical decision - he published his code on internet available free for everyone.

That was a wonderful deviation by which the co-operative sw development-model had been completely broken.

All us discussed about GNU to continue the story.

GNU is a movement started by a programmer Richard Stallman. He proposed an alternative to the standard co-operative development-model.

In 1983 Richard came up with GNU project. It was centered on the idea Free and Open source sw (FOSS) so that the source code of sw should be

Hand Journal  
Date \_\_\_\_\_  
Page \_\_\_\_\_

available free for everyone  
and anyone can modify and  
reproduce with distribution.

Now back to linux - Linus  
and Richard moved together  
and 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 for  
anyone, who wants it.

As a result of it anyone  
can download, modify and  
re-run linux kernel free.

The reason behind the success  
is the transparent nature of Linux.  
As the time progressed  
Linux is developed to such



Hand Toned  
Date: \_\_\_\_\_  
Page: \_\_\_\_\_

an extend that many compan-  
ies replaced their OS to  
Linux in order to enhance  
security and strengthen.

---

## KERNEL

The ~~core~~ kernel is the  
essential center of a comp.  
op. system, ~~that~~ ~~to~~ the  
core that provides basic  
services for all other parts  
of the operating system.

A kernel can be  
contrasted with a shell, the  
outermost part of an op. system  
that interact with user  
commands. Kernel and shell  
are terms used more frequently  
in Unix op. system than  
in IBM mainframe or  
micro soft windows system.