

LINUX WORKSHOP

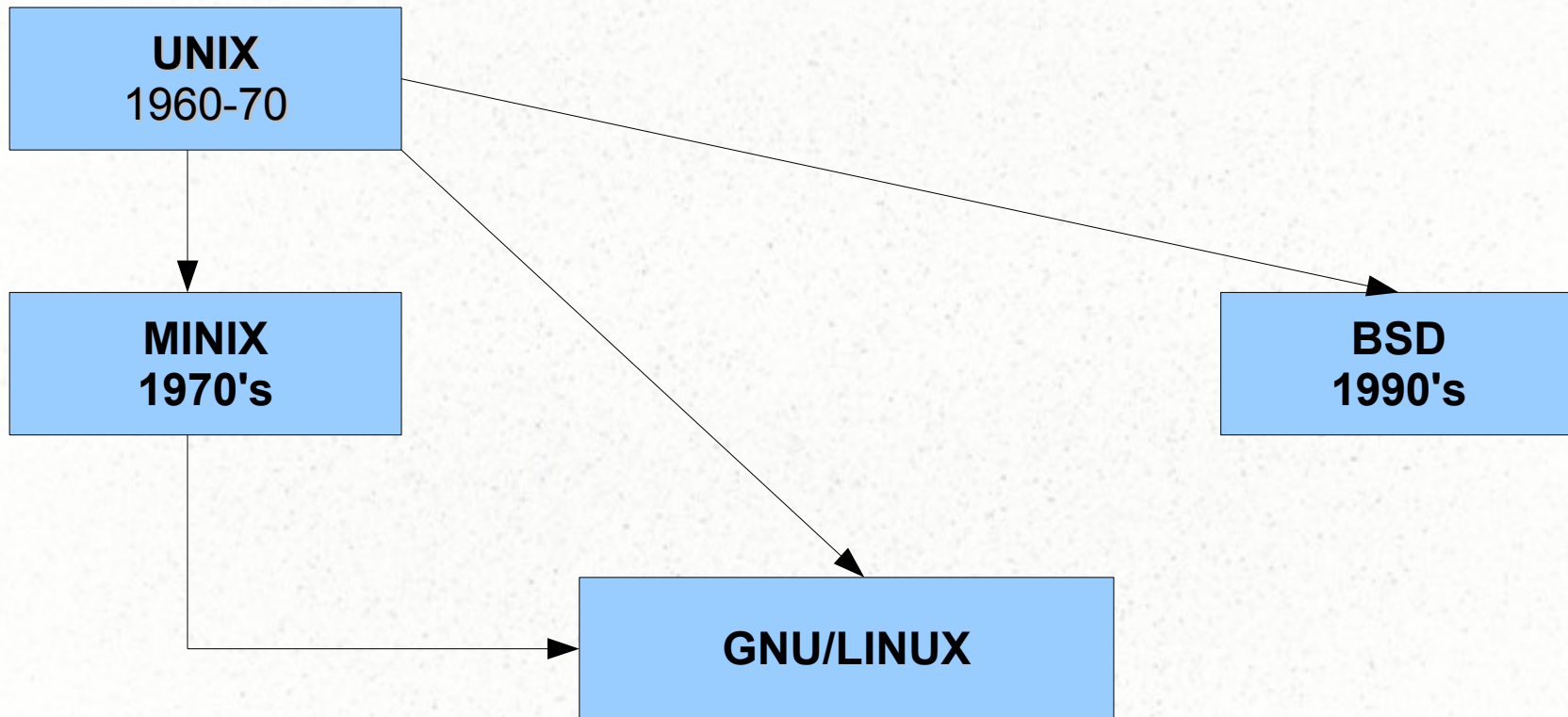
3rd - 4th October

IMS Ghaziabad

Abhishek Srivastava

<http://www.GeekAb.co.cc>

GNU/LINUX



The RAW description of evolution of LINUX Operating System

Introduction of LINUX

- In 1991 **Linus Benedict Trovalds** created Linux.
- It is basically a Kernel which handles process/thread management, Synchronization, Socket-based communications (TCP/UDP/IPC), Handle Device driver management(e.g. hard disks, screens, video etc) and many more..
- Linux is **Open Source and Free** (Based on Stallman's GNU project, released under GPL)
- Community development and Internet release
- POSIX and Unix compatible

What is an Operating System ??

- To manage hardware and software resources in a system
- To ensure the system behaves in a predictable way
- To provide a stable, consistent high-level interface to the hardware

Important Terms to know:

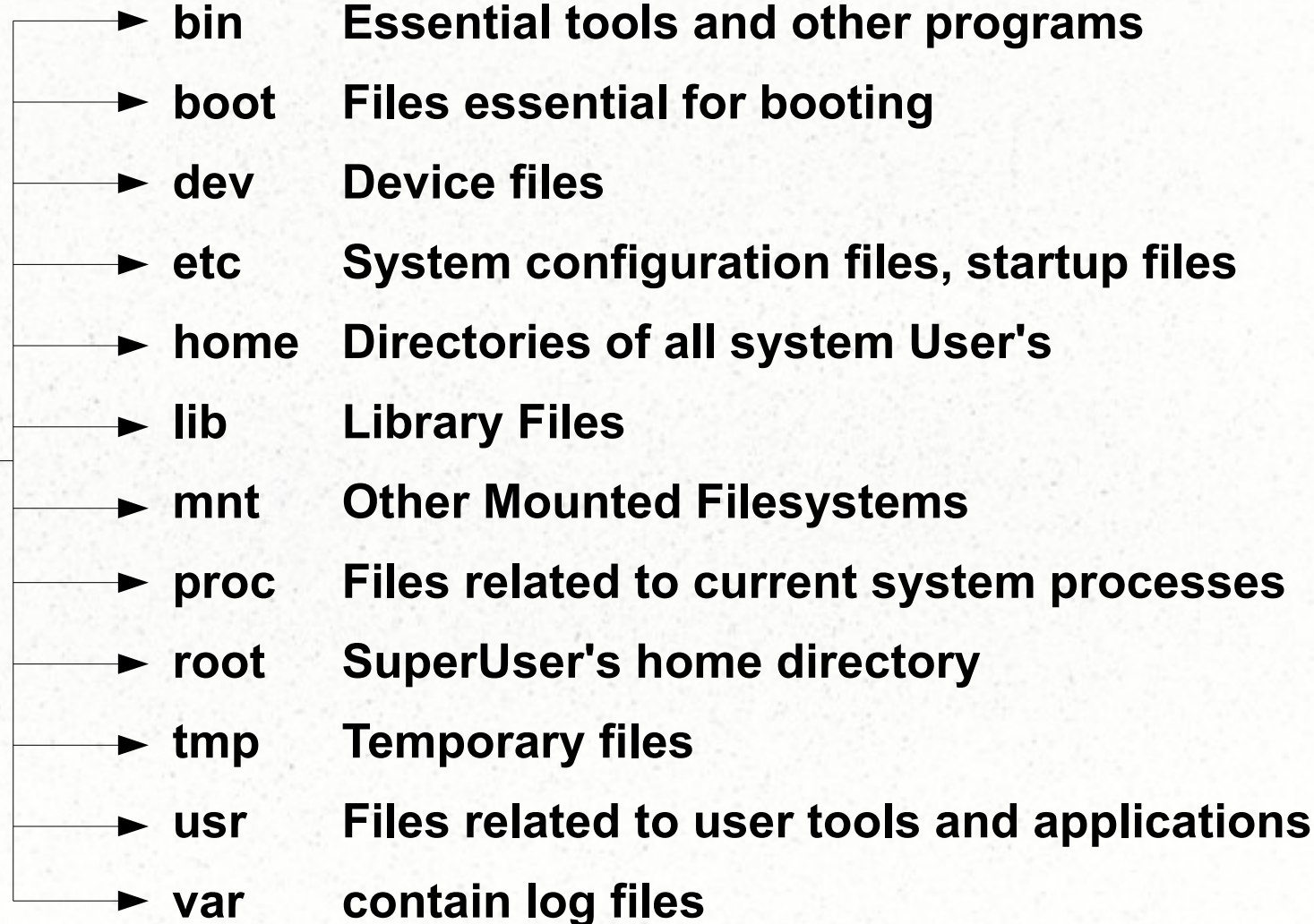
BIOS

BootLoader

GRUB

INSTALLATION

LINUX FILESYSTEM Hierararchy



▶ bin	Essential tools and other programs
▶ boot	Files essential for booting
▶ dev	Device files
▶ etc	System configuration files, startup files
▶ home	Directories of all system User's
▶ lib	Library Files
▶ mnt	Other Mounted Filesystems
▶ proc	Files related to current system processes
▶ root	SuperUser's home directory
▶ tmp	Temporary files
▶ usr	Files related to user tools and applications
▶ var	contain log files

Some System Commands

df -- Display filesystem information

Optional Argument : -h

du -- Display memory usage

Optional Argument : -a

dmesg -- Prints the messages resulting from the most recent system boot.

free -- Displays the amount of used and free system memory.

uptime -- Prints the system uptime

passwd -- to change the password

finger -- show details of a user

Handling FILE/DIRECTORY Permissions

File Permissions

Each file or directory has three permissions:

Read

Write

Execute

And permissions are given for different users , grouped into three groups:

Owner

Group (members)

Others

- ✓ Therefore we have all 3 permissions for all three groups.
- ✓ So there are total 9 permission bits associated with a file/directory.

For a Directory

- Read** to list (display) contents of dir.
- Write** to create, delete and rename a file in the dir.
- Execute** to execute an executable file inside the dir.

Note : Directories have one more permission associated, **sticky bit (t)** – It restricts user from deleting/renaming files in the directory not created by the user even if he has write permissions on that directory.

File Edit View Terminal Help

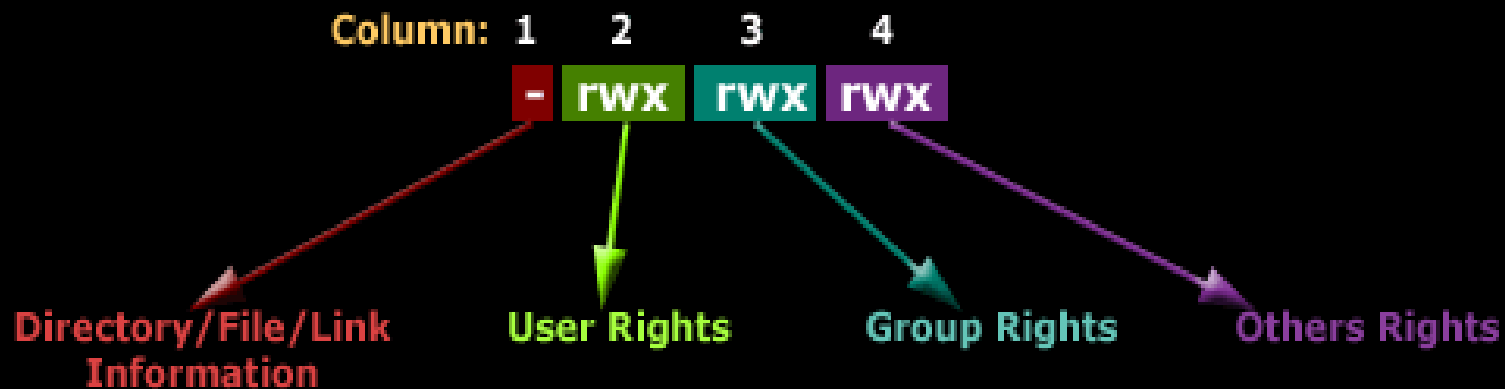
```
drwxr-xr-x  2 geekab geekab      4096 2009-08-24 12:08 C PPTS
-rw-r--r--  2 1 geekab geekab 11 12 13 1 6593 2008-02-13 20:06:08 cross.jpg
drwxr-xr-x  4 geekab geekab      4096 2009-10-01 23:53 Desktop
-rwxr-xr-x  1 geekab geekab         62 2009-10-02 09:01 dial.sh
drwxr-xr-x  2 geekab geekab      4096 2009-08-29 16:50 Documents
drwxr-xr-x 13 geekab geekab      4096 2009-09-22 20:22 EmdbL
lrwxrwxrwx  1 geekab geekab      4096 2009-08-02 26 2009-07-31 23:23 Examples -> /
usr/share/example-content
-rw-r--r--  1 geekab geekab     4096 2009-10-01 21:47 file
-rw-----  1 geekab geekab     4096 2009-09-02 01:28 irc
-rw-r--r--  1 geekab geekab     12023 2009-09-12 22:29 kishore.pls
-rw-r--r--  1 geekab geekab      1976 2009-09-30 16:00 lab manual
```

- r** - read permission
- w** - write permission
- x** - execute permission

- d** - directory

PERMISSION BITS :

Understanding The Linux File Permissions



While the first column defines a directory, file or link, the next 3 columns (2, 3, 4) define the permissions for the User, Group and Others (everyone else) groups.

chown - Changes File owner and group

Syntax :

```
chown user:group file.txt
```

```
chown -R user:group directory
```

Note : **chgrp** , is used to change the group permissions only.

One very important command in linux is **man** , shows manual of any command

Used as : `man command_name`

chmod : change file mode bits

Syntax :

```
chmod [OPTION] MODE[,MODE] file/directory  
chmod [OPTION] OCTAL-MODE file/directory
```

MODE : [u / g / o / a] [+ / - / =] [rwxXt]

u : user

g : group

o : others

a : all

r : read permission

w : write permission

x : execute permission

X : execute only if it is a directory or already has execute permission for some user

t : restricted deletion flag or sticky bit

OCTAL-MODE :

As we have three different permissions, Read (r), Write (w) and Execute (x). Each of these permissions is assigned a number as follows:

r (read) - 4

w (write) - 2

x (execute) - 1

Basic LINUX Commands

pwd -- print name of current/working directory

ls -- list directory contents

Optional Arguments :

ls -l

ls -a

ls -R

Note: file name starting with . are termed as hidden files.

Syntax: ls [options] [files/directories]

mkdir -- make directories

rmdir -- remove empty directories

Optional Arguments :

```
mkdir -p /home/geekab/dir1/dir2
```

```
rmdir -p dir1/dir2
```

Note: rmdir works only with empty directories, and produces a error message if directory is not empty.

Syntax: mkdir [options] directory

```
rmdir [options] directory
```

cat -- concatenate files and print on the standard output

Examples :

cat file.txt -- prints content of file.txt on standard output if file.txt exists else show an error message.

cat -- copies content of standard input to standard output

Optional Arguments :

cat -E

cat -n

Syntax: cat [options] [files]

Note : touch command

mv -- move or rename files and to rename directories
cp -- copy files or directories from one place to another
rm -- remove files or directories

Examples :

```
mv file.txt file1.txt (renaming a file)
cp file.txt
```

Optional Arguments :

-i when used ask for confirmation from user.

Syntax:

```
mv currentlocation newlocation
```

sort -- sort lines of text files

Optional Arguments :

sort -r
sort -n

Syntax :

sort [options] FILES

head & tail -- show top n or bottom n lines for a file.

Syntax :

head -n file

SHELL and PROCESS

What is SHELL?

The Linux/Unix shell refers to a special program that allows you to interact with it by entering certain commands from the keyboard; the shell will execute the commands and display its output on the monitor.

Different types of SHELL's

bash	GNU Bourne-Again SHell
ksh	Kourne Shell
sh	Simple shell (command interpreter)

Process and Handling Processes in LINUX

A process is a program in execution. It consists of the executing program code, a set of resources such as open files, internal kernel data, an address space, one or more threads of execution and a data section containing global variables.

ps -- show information about a selection of the active processes

Shows information of PID , TTY , TIME , CMD

PID : process ID

TTY : terminal

TIME : time to complete the command

CMD : command executed

Examples of ps command (optional arguments):

`ps -ag` To get information about all running process.

`ps aux` To display the owner of the processes along with the processes.

`ps ax | grep process_name`

pstree -- Display a tree of all processes running.

top -- To see currently running processes and other information like memory and CPU usage with real time updates.

Note : To exit from top command press q.

kill -- Cancels a job.(Sends a signal to a process)

Syntax :

kill [-s] %pid

-s Specify the signal to send

Examples :

Searching for a file :

locate -- find files by name

find -- search for files in a directory hierarchy

Note : Difference between both is that , locate finds files using locate database (if it exists) and therefor it is much faster. Whereas find searches for a file or directory in present working directory.

Usage :

locate [options] pattern

find ~ -iname file.txt

find ~ -name file.txt

Note : -maxdepth (optional argument for find command)

Remote Access

SSH (OpenSSH SSH client (remote login program))

- ✓ Used to connect remotely(remote login) to a computer
- ✓ Provide secure encrypted communications between two untrusted hosts over an insecure network.

Syntax :

```
ssh username@server [-p port]
```

Additional Arguments :

- 4 Forces ssh to use IPv4 addresses only.
- 6 Forces ssh to use IPv6 addresses only.

Note : ssh exits with the exit status of the remote command or with 255 if an error occurred.

SCP (secure copy (remote file copy program))

- ✓ Copies files between hosts on a network.
- ✓ It uses ssh for data transfer, and uses the same authentication and provides the same security as ssh.

Syntax :

```
scp user1@server1:filepath1 user2@server2:filepath2
```

Additional Arguments :

- 4 Forces scp to use IPv4 addresses only.
- 6 Forces scp to use IPv6 addresses only.
- C Compression enable

Note : The scp utility exits 0 on success, and >0 if an error occurs.

SOFTWARE INSTALLATIONS

Using Package Manager

System --> Administration --> Package Manager

Note : **Software Sources**, are the servers from where package manager searches for the package we are going to install.

Using Source Code

- How to install (Demo)
- Dependency Packages

SYNAPTIC MANAGER

- Graphical management of software packages

Command : synaptic

aptitude

- High-level interface to the package manager

Command : aptitude

apt-get

- Package handling utility

Command : apt-get

dpkg

- Install debian/ubuntu binary packages (.deb)

Command : dpkg