

Linux : (case sensitive).

Linux is an free and open-source operating system with high security.

multi-user based O.S

what is "O.S" :-

An operating system is a software that acts as an interface between computer hardware and the user.

Every computer must have atleast one os to run.

Applications like browser, ms-office, notepad need some environment to run & perform tasks.

→ Linux is a open source model with security features and it has multi-user & multitasking capabilities, faster.

→ Linux is used for servers, embedded systems, computers, etc.

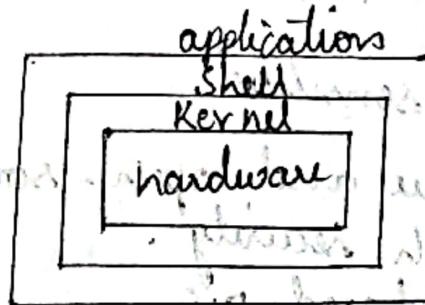
many of the users taken linux OS and modified according to their requirement and released into the market with different names called linux distribution.

Ex:- Redhat, ubuntu, Debian, centos, Amazon Linux

History of Linux:-

- On Sep 17th 1991, Linus Torvalds released the first version of linux kernel as Linux 0.01 as open source software.
- Linux Kernel is written in C
- he wrote this program specially for his PC
- first wanted to name 'Freak' later 'Linux'

Architecture



Kernel:- It is a core (or) the heart of the OS. it is the central part that manages and facilitates communication between the computer hardware and software.

Shell:- The shell is a user interface that allows user to interact with OS. It interprets commands and passes them to Kernel.

Two types:-
1. command line interface (CLI)
2. graphical user interface (GUI)

Distribution, which is a collection of software packages, operating system

By default, we are in ec2-user but if you want to perform any action we should be in root user because root has full permissions

To login as root user

→ sudo -i (sudo = super user do)

To exit from root

→ exit

In Linux, we have different types of commands

1. system 2. Hardware 3. File 4. permissions

5. user 6. search 7. Networking

1. system commands :- used to get the system info.

To get the O.S

→ uname

To get our Kernel version of OS

→ uname -r

To get full info about OS

→ uname -a

To get since how long our system is in running state

→ uptime

gives only time

→ uptime -p

To get the hostname

→ hostname

To change the hostname (it will reflect after login again to root)

→ hostnamectl set-hostname <name>

To get the private IP

→ ip addr

→ ip route

→ ifconfig

→ hostname -i

when was the last time you reboot the server

→ last reboot

To know the date (it will give date in other time zone)

→ date

To know which timezone we are in

→ timedatectl

To convert into IST (A & K in cap), Jira

→ timedatectl set-timezone Asia/Kolkata

To know who ever is accessing the server

→ who

To know, at present which user are you in

→ whoami will be at box

Processor's info, CPU utilization (Q to Quit)

→ Top

Running processes

→ PS

To remove unwanted process

→ Kill -9 <process ID>

2. Hardware Commands:

To get CPU info

→ lscpu

To get memory info (Shows in kb)

→ free -m

if you want to get memory info in MB

→ free -m

To get EBS space

→ lsblk

free space of your EBS Space

→ df -h

Session - 63. File commands :-

TO create a file

→ touch <filename>

→ regular (text file)

for multiple file creation at a time

→ touch <filename> <filename> <filename>

(or) if you want it in a serial way

→ touch <filename{1..7}> → creates 7 files

we can also create files by the extensions like

app.java index.html styles.css app.py

To remove a file

→ rm <filename>

→ will ask permission to

remove for multiple files at a time same as file creation

→ rm <filename> <filename> <filename>

without permission (forcefully)

→ rm -f <filename>

we can recover the file which is deleted in its edit tool

we can also remove a specific serial files like

→ rm -f filename{1..6}

To delete all files

→ rm -f *

TO remove files which are starting with alphabet

→ rm -f a*

samewise extensions

→ rm -f *.txt

TO create directories (folders)

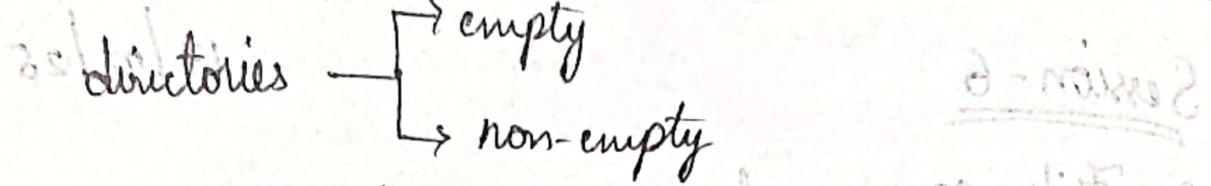
→ mkdir <foldername>

[file starts with -
folder starts with d]

For multiple directories

→ mkdir <foldername> <foldername> <foldername>

Serial way → mkdir foldername{1..4}



To remove directory

$\rightarrow \text{rmdir } <\text{filename}>$

for multiple

$\rightarrow \text{rm -dir } <\text{filename}> & <\text{filename}>$

for serial way (letter wise)

$\rightarrow \text{rmdir } \text{m} & \text{ t} & \text{ b} & \text{ rmdir } \text{ a} & \text{ (b) rmdir folder }$

we cannot remove a non-empty folder by above
like create a folder, inside this folder create a file
this is called as non-empty folder

$\rightarrow \text{mkdir } <\text{folder}>$

$\rightarrow \text{touch } \text{ folder } / \text{ file}$

$\rightarrow \text{ll } <\text{folder}>$

we can remove (non-empty) folder in recursive
forcefully

$\rightarrow \text{rm -rf } <\text{filename}>$

To remove files and folders at a time

$\rightarrow \text{rm -rf } *$

Present working directory

$\rightarrow \text{pwd}$

Change directory

$\rightarrow \text{cd } <\text{folder name}>$

To come to home / root from anywhere you are at

$\rightarrow \text{cd }$

To return back to there

$\rightarrow \text{cd ..}$

To return 1 step back $\rightarrow \text{cd ..}$

To return 2 step back $\rightarrow \text{cd ...}$

.. {1..n} 3 step back $\rightarrow \text{cd/..../..}$

if you want to remove all the files & subfolders
inside a folder called 'data' which is present in
→ rm -rf folder/*

TO copy and move
→ cp <source> <destination>
→ mv <source> <destination>
→ mv file1 file2 file3 /path1/to/1/directory/
for multiple files
→ mv file1 file2 file3 /path1/to/1/directory/

list -d lists all the directories for all the files
alphabetical order → ll (ls) -l -t
based on time → ll -t
reverse order → ll -r
hidden files → ll -a
ls -ltr (ls -l | sort -r)

To create hidden file

→ touch filename.txt and a new file will be created
we can see this file by ll -a but not by ll

mkdir -p awslazm/crp → create folder inside a folder.

Session - 7 → at 9:45 am on 18/03/25

After creating a file, we can view (this content).

→ cat <filename>
and when you give ll, you can see it means
empty.

you can insert the data by

→ cat > filename
then you can enter the data into the file

ctrl + d is used to save data at that way if you want to add additional data into the file

→ cat > filename again enter the additional data and save it by ctrl + d.

→ cat > filename it will overwrite the data in a file (new data) → cat > filename append (old + new)

Through the cat command we cannot modify the file, this will be achieved by the editors.

→ vim editor, (or) nano editor let us see how we can modify the data in VIM

There are 3 modes.

1. command mode 2. Insert mode 3. save & exit

1. command mode → it is the default mode used to perform some actions.

when you open a file which has lot of data, at first the cursor will be at top

→ vim filename

To bring the cursor from the top to the last line

→ (capital) G or (small) g

To bring the cursor from last to top line

gg (Small)

To go to specific line → n (or) Shift + f

5gg (or) 10gg (or) 9gg

To get the line numbers

Shift : set number

To go to middle lines → n (or) Shift + f

M (Capital)

if you want to copy 4th line, first have to go
that line line shift :4 & press yy prints
paste it by p. (if you give 10p → 10 lines)

To delete a line

dd (if you want to delete 10 line → 10 dd)

To copy multiple lines ex:- 4 lines to copy
give 4yy and to paste give p wherever you want

To undo all the changes u

To redo

ctrl + r

To search for a specific word → /word

To replace a word → shift: % s/word/word/g
means global

2. Insert mode :- only used to insert data

give i for insert mode

to get back to command mode give ESC

after entering data in insert mode

To go to end of the line → A (capital)

To goto starting of the line → I (capital)

To create a new line above → O (capital)

To create a new line below → o (small)

3. Save & Quit mode :-

To save change → shift:w (:w)

To quit from vim → shift:q (:q)

To save and quit → :wq

To forcefully save & quit → :wq!

To quit forcefully → :q!

Session - 8

Date: 20/3/25

~~Use Command~~

To copy and paste in the command prompt / Putty

Copy → select the line with shift+left click or

Paste → right click

In cat, To view the data along with line numbers

→ cat -n filename

To print top 10 lines in a file

→ head -n 10 filename

To print last 10 lines in a file

→ tail -n 10 filename

To print 5-17 lines in a file

→ sed -n '5,17p' filename

To print specific lines in a file

→ sed -n '5,17p' filename

To print top 5 lines in a file

→ head -n 5 filename

To print last 3 lines

→ tail -n 3 filename

To get to know that how many number of lines, words & characters

→ wc -l filename

To get only line → wc -l filename

To get only words → wc -w filename

" " " characters → wc -c filename

User commands :-

In general, a server has many users with them.

To get the list of users
→ `getent passwd` (or) `cat /etc/passwd`

for example, there are 3 user's for a system
user-1 needs to watch movies, user-2 wants to play
games and user-3 want to do project work.

If you give the username and password, then all
of the user can do their work and also others work.
To prevent this, we will create a user for user-1
by giving permission to only watch movies and
then he don't have other permissions to do.
Likewise other user's also.

In general Linux admins will do this work like
they will create a user by giving permissions to
only write into a file. not able to delete &
execute files.
→ user without homedirectory
→ `useradd -m user`

To add a user → `useradd username`
Ex: `useradd vishnu`

To get the specific user is there or not in list
→ `id vishnu`

After adding a user and do list user's by `getent passwd`
you will see like this.

`vishnu:x:1001:1001::/home/vishnu:/bin/bash`

↓ ↓
password stored |
username here | users path | users shell
↓ ↓
↓ ↓

↓ ↓
↓ ↓
↓ ↓
↓ ↓

whenever we create a user, then group will also be
Created

if you give id `visnu`

it will show you the user id, group id & group name

To delete a user & group → userdel username

→ userdel & username → delete all the users, groups & owner

After creating a user it will show no file

To get into that user → login & new user name

→ `su -> username`

you cannot see the files in other user which is created in your user. Now we edit know, if you private files also at ordinary place of

To set a password

→ passwd username (you have to set the password here)

you can directly login to a user without any password by root user.

if you are in ec-user then it will asks for the password to login.

If there are 5 users and you want give some permission to all 5 users at a time this is done by adding users to all group and then give permission

To get the list of groups

→ getent group (user and group names same)

To add group (Creating group)

→ groupadd groupname

To attach user to this group

→ usermod -aG groupname username

- To verify that this user is added to the group
→ id & username
- To delete a group with existing user add to
groupdel groupname

permissions: → Linux file system has three

if you open a file it will show you like this

-rwx-r--r-- 1 root root 1235 Mar 20 15:11 filename

→ first 3 → user permissions
→ second 3 → group permissions
→ last 3 → other's (processors)

type of file	permissions of	owner	group
file	r = 4	w = 2	x = 1
directory			
character file	x = 1		
block file	- = 0		

→ r = read

→ w = write

→ x = execute

if you want to give full permissions to a file which is for user, group & others should have full permissions

user → rwx → $4+2+1 = 7$

group → rwx → $4+2+1 = 7$

others → rwx → $4+2+1 = 7$

To change permissions to a file
→ chmod 777 (change mode full permission)

To give only read & write

→ chmod 666

To add execute to above file which x is absent

→ chmod +x filename

To change permissions for multiple files

→ chmod 741 *

or to some particular files

→ chmod 666 file1 file2

Imagine you have files inside a folder and you want to change the permission to all the files inside the folder and the folder too!

→ chmod 666 foldername -R

(*) you want to change

Now, if you want to change permissions only to files inside the folder

→ chmod 777 foldername /*

To change owner's of file

→ chown <from username> <to username> (source & destination)

To change group of file

→ chgrp <old groupname> <new groupname>

To change both at a time (user & group)

→ chown username:groupname filename

Assignments of Session - 7

- 1) let's assume that you have 2 folders } how to copy
 folder 1 has some files and folders } all the files &
 folder 2 has is an empty folder } folders from
 1 to 2

→ cp -r folder1/* folder2 (to copy hidden files
 → will also copy directories also we use -r instead of *)

- 2) let's assume that you have 5 files
 file1 has some data, remaining are empty files
 how to copy the data from one file to remaining
 n files at a time

→ cat file1 | tee -a file2 file3 file4 file5
 -a = without overwriting existing data
 tee = will writes the content simultaneously.

- 3) how to Rename a file or folder

→ mv oldfile newfile

- 4) let's assume that you have 4 folder's
 folder-1 has some files, remaining 3 folders are
 empty.

how to copy the files from folder to remaining
 folders. → echo folder{2,3} | xargs -n1 cp -v folder1/*

→ echo folder2 folder3 folder4 | xargs -n1 cp
 -v folder1/*

Assignments of Session - 8

- 1) command to change owner's of all the files
 at a time.

→ chown username *

- 2) lets assume that a folder which has 5 file.
what is the command to change the owner
to folders and files inside that folder.
→ chown user:group folder -R
- 3) like wise, what is the command to change
owner's for only files which are present in
folder
→ chown user:group folder/*

Session - 9 21/03/25

Search commands

1. grep
2. find
3. Locate

1. grep :- Global regular expression print

It is used to search for a word from a file.

Syntax :- grep "word" filename

If you want the line numbers also

→ \$ grep -n "word" filename

CASE-insensitive words (Hello & Hello are same)

→ grep -in "word" filename

To get count of that word in lines

→ grep -in -c "word" filename

To search for multiple words (c=echo)

→ grep -in -e "word" -e "word" -e "word"
filename.

To search a word in multiple files.

→ grep -in "word" file1 file2

To search for multiple words in multiple files
→ grep -in -e "word" -e "word" file file

In case example, if you give timedatectl, you will get list of information. here you only want timezone in the info then
→ timedatectl | grep -i "Time zone"
and you want only root user while entering getent
password
→ getent passwd | grep -i "root"

2 Find:
Syntax: to find a path -> name filename

for example you have 2 files, file1 has aws, devops & file2 has devops, jenkins.

find . -name devops
↓
current directory

(if you want to search it in folder)
find folder1 -name devops

for example you have a 2 folder, folder1 has aji0 folder and some file and folder2 has some files.
Now to get only the aji0 folder.

→ find . -type d -name aji0.

here we have searched based on type d = directory.

we can also search based on permissions, owner, file size, modification time.

→ find . -group groupname (to search group)

→ find . -perm

- for example
chmod 777 folder1/amazon
chmod 777 folder2/ajio
- To get the files which has full permissions
→ find . -perm 777 -type f
- To search for user
→ find . -user username -type f

3. Locate :

Before using locate, you have to update the database cause all the files are located in the linux database.

- to update the database
→ updatedb
 - then locate the filename
 - locate filename
- Every time you have to update the db whenever there is a new change like file created.

Note:- find command will fetch the data faster because we are specifying the path to search whereas locate have to go through a entire database.)

In find command, we can search by the file name, owner, permissions, type, size etc. whereas, locate will search all the files in the system, so it is faster than find command.

connecting ec2 through vs code

- open vs code & download remote ssh extension
- Ctrl + Shift + P → select open ssh configuration file
.ssh\config.

host ec2-user (0)	host aws-linux
hostname <pub ip>	hostname <pub ip>
user ubuntu	user ec2-user
identityfile c:\users\1Downloads\ pemfile	