

LINUX

Page No.

18 02 25

- * Linux → It is a Operating System
It is both CLI & GUI based Operating System
CLI = Command Line interface
GUI = Graphical User interface

* CLI → In case if I need to create a file in linux or other operating system, to create that we use commands = CLI.

* GUI → mainly used to interact with the Operating System = Ex: ATM machine, Video games

* To create the directory = mkdir folder1

* The list of files created can be seen in = ls

Ex: mkdir b1 b2

space = 2-folders are being

created

mkdir b1 b2 b3

→ No space = ',' represents a single folder

ls = b1 b2 b1,b2,b3 : folder1 folder2

* Move into the command = cd folder1

ls b2 : cd L ↴ change directory

/folder1/ ⇒ inside the folder

now = ls L ↴ Empty = b2 we created

folder but nothing inside it

/folder1/ ⇒ cd a1

↳ Created a folder inside a folder.

Now the path changes ⇒

/folder1/a1/ ⇒

* ls = nothing

→ ↗ xunzi ↘

* cd = Come back to the home directory

ubuntu@ /folder1/a1/a2 = cd

" a1/a2 gone)

↓ gone = ubuntu@

* cd .. = 1 - Step back

ubuntu@ /folder1/a1 = a2 will be gone.

If I want to go multiple step - back

* cd ../../ (2-steps back)

No space = If space = Error

3-steps back = cd ../../.. /

* cd = Take back to the directory from where I

Started

* pwd = present working directory.

* ubuntu@ : \$ pwd. → gives current location
/home/ubuntu

* ubuntu@ /folder1/a1 : \$ pwd

/home/ubuntu/folder1/a1

* ubuntu@ /folder1/a1 : \$ cd a2

/home/ubuntu/folder1/a1/a2

→ it provides where I was earlier

/home/ubuntu/folder1/a1/a2 : cd - ↓

" b3 ← /a1 ↘ ↗

To visit the last location

= /cd /sublog/

- + If I directly want to go to folder2 = $cd \text{ folder1}/\text{a1}$ or mouse
- + To know all the commands Executed = history inside

Creating files in linux

- touch = used to create empty/blank file
- echo = used to create file with a single line text
- nano = used to create a file with multiple lines
- vi = used to create a file

[Note] Difference b/w nano & vi is the way we operate them. i.e writing, Saving & exiting the content in the files.

- touch file1
- ls
- cat file1
 - ↳ used to read & display files
- $\text{echo "this is my first line" > file2}$
- ls
 - ↳ the content is stored
- $\text{cat file2} = \text{this is my first line in file2}$
- $\text{echo "this is my second line" >> file3}$
- cat file3
- $\text{echo "Overwrite my data/content" > file3}$
- cat file3

Overwrite my data/content

[Already Existing @ Overwrite if doesn't Exist
Create new file]

~~calculator~~: nano file4

↳ Creates a file = Hello my name is Vaishnavi
Engineering student
Btech ECE stream.

Control 'S' = Save

control 'X' = Exit

- Cat file4
[All the content shows here]
- Nano file4.
[It doesn't create already exists]
Now make changes in content = "Great journey"
- Control 'S': & control 'X' at here
- Cat file4

[Additional line shows] = "Great journey"

- Vi file5

↳ Editor ⇒ press 'i' = insert mode

now I can write anything

"Hello & welcome to my website"

this is my second line

press = Escape key : wq → exiting here ↴

- Cat file5

- ls

↳ b1, b2, b3, b4, b5 files file1 file2 file3 file4 file5
folder1 folder2

Copy & paste = file1 into folder1

- Cp file1 folder1

- ls

- cd folder1

- ubuntu@/folder1\$ ls

Op = a1, file1

- Exit

cp = copy paste
mv = cut paste

Page No.	
Date	

- ls
- cp folder1 folder2 = Can't Copy directly
Error - recursively doing this
- cp -r folder1 folder2
- cd folder2
- ls

- ls
 - mv file1 folder2
 - cd folder1
 - ls
- file1 \Rightarrow will be gone \Rightarrow as we have moved the folder

[J- a1]

- The file should be empty while 'mv' - format cut paste
 - mv f1 folder2 \Rightarrow no need of '-r'
 - cd folder2
 - ls
- f1 folder2
- rm -rf f2
 - rm -rf folder1
 - rmdir folder1
 - ↳ Error cannot be deleted as folder1 consists of some files or contents.
 - rm -r folder1
 - ↳ recursive deleting
 - rm -r + \Rightarrow delete all files & folder once

* Advanced Linux

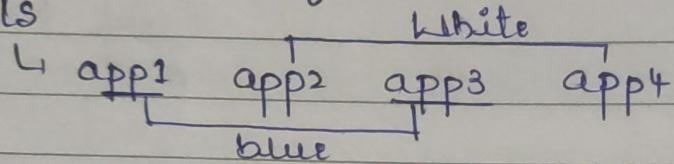
- `mkdir -p x1/x2/x3/x4`
- `ls`
- `rm -r *`

Create folder → `mkdir app1 app3`

- `touch app2 app4`

↳ Create files

- `ls`



* To identify which are files & folders

`ls -l`

drw = directory = folder

rw = read/write = files

* `[ls -a]` To see the hidden files

- `mkdir -p x1/x2/x3/x4`

- `ls -R`

↳ To see the contents inside each folder.

x1

• /x1:

x2

• /x2/x2:

x3

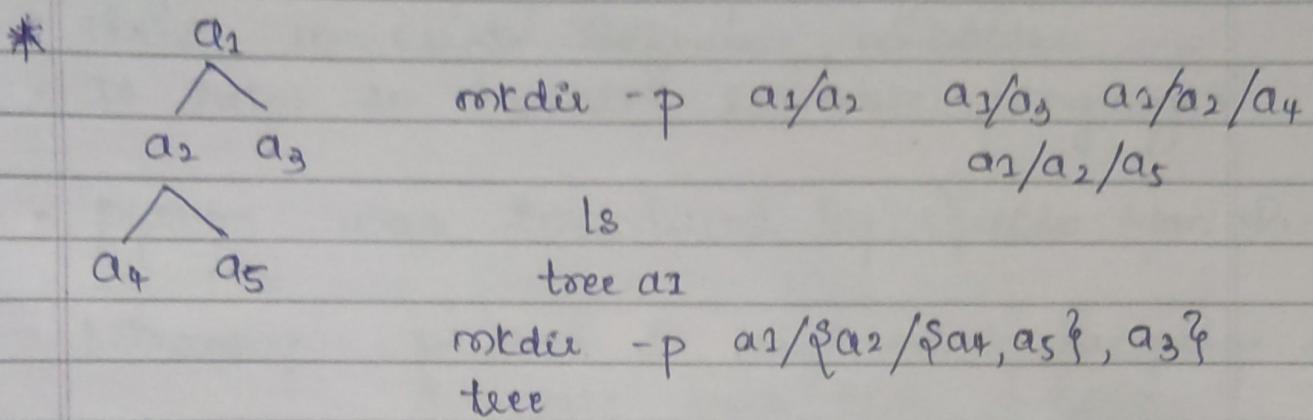
• /x1/x2/x3: x3 is a file

x4

• /x1/x2/x3/x4: x4 is a file

• /x1/x2/x3/x4: x4 is a file

- man mkdru
 - ↳ Shows all hyphen commands
- man ls
- tree x1 = works = install it first
sudo apt-get install tree -y



Linux

1. ls → list
2. pwd → Current = present working directory
3. mkdir - Creates a directory
4. cd - Command directory = to navigate b/w different folders
5. rmdir - removes empty directories from the directory list
6. cp - copy ^{file from 1} directory from to another
7. mv - rename & replace the files
8. rm - delete files
9. uname - to get basic information about OS
10. locate - find a file in the database
11. touch - create empty files
12. ln - create shortcuts to other files
13. cat - displays the content on terminal
14. ps - display the processes in terminal
15. man - Access manual for all Linux commands
16. grep - Search for a specific string in an output
17. echo - print string or text to the terminal
18. wget - download files from the internet
19. whoami - displays the current username
20. sort - Sort the file content
21. cal - View calendar in terminal
22. whereis - View exact location of any command typed after this command

23. `df` - check the details of the file system
 24. `wc` - check the lines, word count & characters in a file using different option

- `uname -r` = print kernel release information

- `free` - check the memory related detail in your system.

- `nslookup` - used to obtain information for DNS Server
`nslookup <domain name>`

- `ssh-keygen` - used to establish a secure ssh connection from your host machine to any remote Server

- It generates a public/private key pair

`ssh-keygen -t rsa`

- `curl` - it is a tool which is used to fetch data & post the data over the internet. It can use various of protocol like HTTPS, SMTP & FTP

`curl [options] [url]`

- curl -o = flag saves the data into a file on the local machine
curl -o [file-name] [url...]
- apt-get - used to manage packages in the Linux. APT stands for the Advanced packaging tool & its main used of install, update, upgrade & remove the packages
apt-get [options] command.
- du - check disk usage space
- df - to check the available disk space in system
df -h
- ifconfig - used to view the information about your network interface
ifconfig [options] [interface]
- ip - It is a modern replacement of ifconfig command. It is used to view & manage network settings. you can check ip addresses, configure network interfaces, view routing tables by this command
ip [options] OBJECT & COMMAND | help

* For Creating directories

- For creating a single directory

`mkdir GFG`

- For creating multiple directories

`mkdir GFG1 GFG2 GFG3`

- For creating directory paths (directories inside directories)

`mkdir -p /GFG/GFG1/GFG2`

- For creating a series of numbered directories

`mkdir gfg{1..3}`

* Copying & pasting files & directories

- For copying a file with verbosity & force

`cp -rvf gfg1 gfg1`

* Removing files & directories

- for removing a directory & its contents

`rm -rf gfg`

* Renaming files & directories

`mv qdq qdq-devops`

* User Management

1. Creating a User

To create a new user in Linux, you can use the `useradd`

`Sudo useradd [username]`

2. Setting a password

After creating a user, set a password

3. Deleting a User

`Sudo userdel [username]`

4. Switch to Another User

The `su` command allows you to switch to another user's account by providing the username as an argument.

To exit from the user's account & return to your original session, you can simply type 'exit'

`su [username]`

5. Rename the user

To change the username from the current name to the new name, use

`Sudo usermod -l [newname] [oldname]`

* Group Management -

A group is a collection of user accounts that is very useful to administrators for managing & applying permission to a number users.

`Sudo groupadd <groupname>` - making a new group

`Sudo groupdel <groupname>` - delete the group

`Sudo usermod -g <groupname> <username>` - Adding a user to a group

* process Management

A process management is the process of controlling & monitoring the process running on a Linux system.

1. `ps` - displays currently running processes
"ps aux"

2. `Top` - used for memory monitoring
It shows a real-time view of system processes

3. `Kill` - It is used to terminate a process using its PID.
"kill 1234"

4. `pidof` - used to get the PID of a running process by name
`pidof bash`

5. `Systemctl` - used to start/stop/restart
 Systemd - managed services
 (like docker, Nginx)
`Sudo systemctl status nginx`

* Linux File System permission

To increase the security of the file & directory. There are total 3-types of file permission are read, write, execute.

- user (u) - permissions used for the user of the file
- group (g) - permission used by the group member
- other (o) - permission used by all other users

1. `ls -ld` - used to check the permission of directory

Read - display file contents - view contents & copy the file - contents of direct

write - modify the file - modify the contents - contents of a directory

Execute - execute the file - allows use if it an executable permission of cd command - and to access the directory

+ permission with numeric & symbol	Symbol
permission type	---
No permission	- -x
Execute	- w-
Write	- wx
Execute + Write	r--
Read	r-x
Read + Execute	rw-
Read + Write	rwx
Read + Write + Execute	

2. chmod (change mode)

used to change the permission of file & directory

chmod <permission of user, group, other> &filename{}

3. chown (change owner)

used to change the owner of the file & directory

chown [owner-name] [file name]

4. cat

used to read & concatenate the text inside the files

cat <flag> &filename{}

cat -b - This flag adds number to the text line

cat -F - This flag add \$ at the end of each line

5. Grep (Global regular Expression print)

It filter searches a file for a particular pattern of characters & displays all lines that contains the pattern.

grep <flag or search-word> & file name

grep -i - delivers results for case-insensitive strings

grep -n - retrieves the corresponding strings & their respective line numbers

grep -v - provides the output of lines that do not contain the search string

6. Sort - It prints the output of a file, either alphabetically, numerically or by other specified way.

Sort filename.

7. head - used to display the first few lines of one or more text files

head -n 2 gffg

8. tail - to display the last few lines of one or more text files
tail -n 3 qfb

9. find - to search files & directories based on different criteria such as name, size, type & modification date
find [path] [expression]

10. stat - to display detailed permission & ownership metadata

stat data.txt