DAYANANDA SAGAR UNIVERSITY

LINUX PROGRAMMING

ASSIGNMENTS-2

NAME : AKHIL VENKAT D

USN: ENG24CY0079

ROLL NO :46

SEACTION: A

1. What does the command pwd, whoami, and hostname disply?

1. pwd — Print Working Directory

- Displays: The full path of the current directory you're in.
- Example output:
- /home/username/Documents

2. whoami - Who Am I

- Displays: The username of the person currently logged into the shell.
- Example output:
- username

3. hostname - Host Name

- Displays: The name of the computer (host) on the network.
- Example output:
- my-computer-name
- 2. Write the command to create a directory named "project" inside the /home/student folder and keep three .txt file into it. Give output snapshot.



Command to create a directory named project inside /home/student:

mkdir /home/student/project



Commands to create three .txt files inside that directory:

touch /home/student/project/file1.txt touch /home/student/project/file2.txt touch /home/student/project/file3.txt



Or combine all three into a single command:

touch /home/student/project/file1.txt /home/student/project/file2.txt / home/student/project/file3.txt



Optional: Verify the files were created with ls:

ls /home/student/project

Expected Output:

file1.txt file2.txt file3.txt



Output Snapshot (Text Representation):

```
$ mkdir /home/student/project
$ touch /home/student/project/file1.txt /home/student/project/file2.txt /
home/student/project/file3.txt
$ ls /home/student/project
file1.txt file2.txt file3.txt
```

3. Explain the difference between absolute path and relative path with proper examples.

Difference Between Absolute Path and Relative Path (with Examples)



1. Absolute Path

➤ Definition:

An absolute path is the complete path from the root directory / to the target file or directory.

It always starts with / and gives the full location in the file system.

➤ Example:

Suppose there's a file located in:

/home/student/project/file1.txt

This is the absolute path to file1.txt.



Characteristics:

- Starts from the root directory /
- Works from any location in the system
- Unambiguous (always points to the same file)

2. Relative Path

➤ Definition:

A relative path specifies the location of a file relative to the current working directory. It does not start with /.

➤ Example:

If your current working directory is:

/home/student

Then you can access file1.txt using:

project/file1.txt

This is a **relative path** because it starts from your **current directory**.

You could also use:

./project/file1.txt

(. means current directory)

Or:

../student/project/file1.txt

(.. means one directory up)

4. What command will give you the already executed command traces in the terminal. Give output snapshot.

♦ Command:

history

Explanation:

- The history command displays a list of previously executed commands in your terminal session.
- Each command is listed with a **line number**, which can be used with ! to re-run a specific command.

Output Snapshot (Text Representation):

\$ history

- 1 pwd
- 2 whoami
- 3 mkdir /home/student/project
- 4 touch /home/student/project/file1.txt
- 5 touch /home/student/project/file2.txt
- 6 touch /home/student/project/file3.txt
- 7 ls /home/student/project
- 8 history

Bonus: Search in History

You can search through history using:

history I grep mkdir

Bonus: Repeat a Command from History

!4

5. Compare the working functionality of find and locate command. Which one is faster and why?

1. find Command

➤ Functionality:

- Searches files in real-time by traversing the actual file system.
- Can apply complex filters like name, type, size, modification time, permissions, etc.

➤ Syntax:

find /path/to/search -name "filename"

➤ Example:

find /home/student -name "file1.txt"

V Pros:

- Always up-to-date (real-time).
- Powerful and flexible (supports complex conditions and actions).

X Cons:

- Slower, especially on large file systems.
- Requires reading every directory and file in the search path.

2. locate Command

➤ Functionality:

- Searches a pre-built database (mlocate.db) that indexes file names across the system.
- Database is usually updated by a background job (updatedb), not in real-time.

➤ Syntax:

locate filename

➤ Example:

locate file1.txt



- Very fast much quicker than find because it searches a local database, not the disk.
- Simple to use.

X Cons:

- Might return **outdated results** if the database hasn't been updated.
- Doesn't support complex filtering like file size or permissions.

Which One is Faster?

- locate is faster than find because it searches a pre-built index, not the live file system.
- find is slower but more accurate and powerful.
- 6. Which command is used to modify file permissions in Linux? Give an example.

Command:

chmod (change mode)

What it does:

chmod is used to **change the permissions** of files and directories (read, write, execute).

Basic Syntax:

chmod [options] mode filename

Example:

Suppose you have a file named script.sh and you want to:

- Allow the owner to read, write, and execute
- Allow the group and others to only read and execute

You can run:

chmod 755 script.sh

Explanation of 755:

- 7 = read (4) + write (2) + execute (1) = 7 (for owner)
- 5 = read (4) + execute (1) = 5 (for group)
- 5 = read (4) + execute (1) = 5 (for others)

Check the permissions:

Is -I script.sh

Output:

-rwxr-xr-x 1 user user 1234 Sep 27 12:00 script.sh

Other ways to use chmod:

Symbolic notation, e.g.:

chmod u+x script.sh # adds execute permission for user (owner)
chmod g-w file.txt # removes write permission from group

7. A file has permissions -rw -r- -r- -. What does this mean? It is divided into **4 parts**:

1. First character (-) → Type of file o -: regular file o d: directory o I: symbolic link Here, it's a regular file.

- 2. Next 3 characters (rw-) \rightarrow Owner permissions
 - or: read permission 🗸
 - w : write permission
 - -: no execute permission X
- → Owner can read and write, but not execute.
 - 3. Next 3 characters (r--) \rightarrow Group permissions
 - or: read permission 🗸
 - -: no write 🗙
 - -: no execute 🗙
- → Group can **only read**.
 - 4. Last 3 characters (r--) \rightarrow Others (world) permissions
 - r : read permission
 - -: no write 🗙
 - -: no execute 🗙
- → Others can only read.

V Final Meaning:

Owner: read + write

Group: read only

Others: read only

File type: regular file

8. Explain the difference between chown and chgrp with an example.

1. chown (Change Owner)

- Used to change the owner (user) of a file/directory.
- Syntax:
- chown new_owner filename
- Example:
- Is -I file.txt
- -rw-r--r-- 1 alice staff 1234 Sep 27 19:00 file.txt

Here, **owner = alice**, group = **staff**.

If we run:

sudo chown bob file.txt

→ Now **bob** becomes the owner:

-rw-r--r-- 1 bob staff 1234 Sep 27 19:00 file.txt

2. chgrp (Change Group)

- Used to change the group ownership of a file/directory.
- Syntax:
- chgrp new_group filename
- Example:
- Is -I file.txt
- -rw-r--r-- 1 bob staff 1234 Sep 27 19:00 file.txt

If we run:

sudo chgrp developers file.txt

- → Now the group becomes **developers**:
- -rw-r--r-- 1 bob developers 1234 Sep 27 19:00 file.txt
 - 9. A file needs to be accessible by multiple users but only writable by the owner. How will you set permissions?

Permission Breakdown

- Owner: rw- → read + write
- Group: r-- → read only

Others: $r--\rightarrow$ read only

So the permission string is:

-rw-r--r--

Numeric (Octal) Form

644

Command to Set It

chmod 644 filename

This ensures:

- Owner can read/write.
- Group + others can read only.
- Only the owner can make modifications.

10. How do you check the manual page for any Linux commands?

To check the manual page:

man command_name



man Is

This will open the manual for the Is command.

Useful Tips with man:

- 1. Navigation
 - Use \uparrow and \downarrow or Space to scroll.
 - o Press q to quit the manual page.
- 2. Search inside man pages
 - o Type /keyword and press Enter to search.
 - Use n to jump to the next match.
- 3. Find available man pages for a command
- 4. man -f command

(Same as whatis command)

5. Search by keyword if you don't know the exact command

6. man -k keyword

Example:

man -k copy

 \rightarrow Shows commands related to file copying like cp, scp, rsync.