**Name – Vishal Menariya**

**TR – 1**

**Assignment – 1**

**Roll no – 22etccs170**

**7851949372**

**Points to be Covered in Linux Assignment-1**

* Explain CLI & GUI
* Login Terminals
* Explain Files & Directory
* Basic Commands
  + ls, cd, mkdir, rm, man, touch, cp, mv, less, more, head, tail, grep with options.
* File Editors
  + vi, vim, nano, gedit.
* Links o Softlink & Hardlink.
* Paths o Absolute Path & Relative Path.
* Redirections o I/O Redirection Operators.(>, >>, &> , &>>)

**Assignment-1 Questions:**

**(Note: All answers should include an attached screenshot as proof of execution)**

1. What does CLI stand for, and how does it differ from GUI?

Soln>

CLI stands for **Command Line Interface**, It is basically a **shell** which only works on **commands** or text

and the commands are already defined.

To interact with system user has to write commands in shell to produce desired output or use functionality.

While in case of GUI which stands for **Graphics user Interface**, It is basically a predefined functions which

Are shown in graphics form and user can interact with them directly without using commands.

In GUI we use buttons, images, sliders, card views etc. for better understanding or easy access to interface

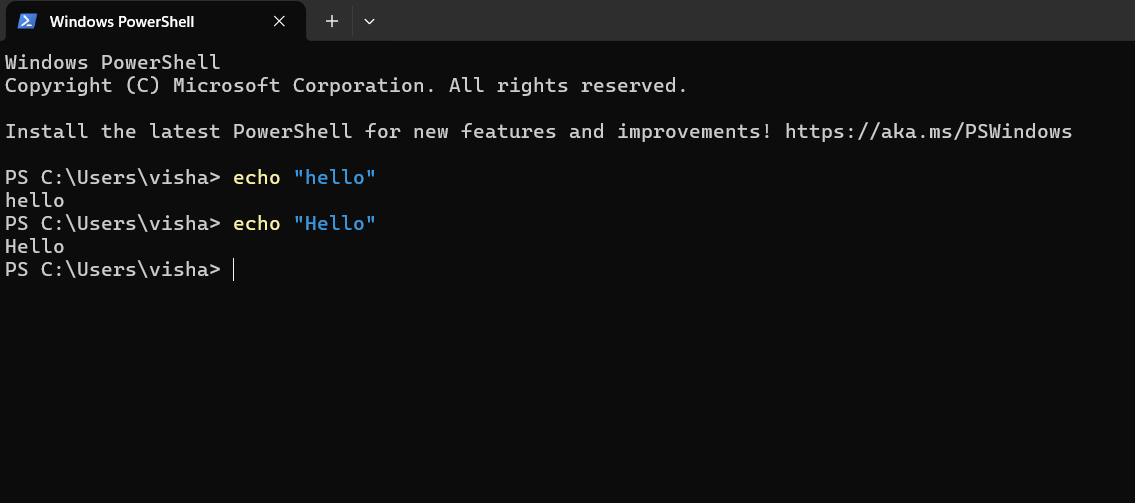
While in case of CLI we only have access to commands.

Eg. To **create files** in **CLI** we have to write commands touch /documents/file.txt

While in **GUI** we move to the destination where we want to create file using graphical interface and there with the help of keyboard shortcuts or mouse or direct button of create file we can make a file.

2. Open a terminal and execute a simple command such as **echo "Hello**". Can you perform a similar action in a GUI? What are the advantages of using a CLI compared to a GUI?

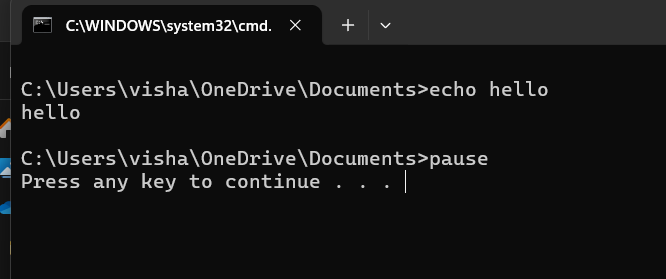
Soln>

To perform this action in GUI we have to create a batch file (.bat) and execute it.

Steps

Create a .bat file and write “echo Hello \n pause” and save it

then execute the file



Advantage of CLI over GUI

**Speed and Time** – The speed of CLI is more than GUI as it has no graphics so no graphics library has to load so it takes less speed and time while the GUI has to load the graphics first then execute functionality so it takes more time and speed is also slow.

**Resources usage** – CLI uses less resources as their tools require significantly fewer system resources than GUI.

**Functionality** – As every functionality cannot be integrated with GUI so CLI provides more functionality by using its commands and also enables precise control over operation.

**Scriptability** - Commands can be combined into scripts to perform complex tasks automatically and also Scripts ensure that processes can be replicated consistently across different systems or by other users.

**Remote Access -** Managing servers and devices over networks without GUIs.

**Security -** CLI environments lack the graphical components eg windows, buttons, and menus that could potentially have vulnerabilities. This reduces the attack surface for exploits targeting graphical libraries or GUI frameworks. Remote Access Security tools like SSH (Secure Shell) for CLI-based remote access provide encrypted communication channels, ensuring secure remote administration.

3. To convert a minimal CLI-based Linux interface into a GUI-based one, which packages are typically required? Find them

Soln>

To convert CLI – linux interface into a GUI based one we need several packeages:

1. X window system (X11) - This is the foundational layer that provides the basic framework for a GUI environment in Unix-like systems. It manages graphical display capabilities and handles input devices.
2. Window Manger/Desktop environment -

**Window Manager:** Controls the placement and appearance of windows within the X environment. Examples include Openbox, Fluxbox, and i3.

**Desktop Environment:** Offers a more comprehensive and integrated GUI experience, including a window manager, file manager, and various applications. Popular desktop environments are GNOME, KDE Plasma, XFCE, and LXDE.

1. **Display Manager (Optional):** Manages user sessions and handles graphical logins. Examples include GDM (GNOME Display Manager), SDDM (Simple Desktop Display Manager), and LightDM.

Steps:

Install the X Window System: sudo apt-get install xorg

Install a Desktop Environment: sudo apt-get install xfce4

Install a Display Manager (Optional): sudo apt-get install lightdm

After installation, you can start the GUI with: sudo systemctl set-default graphical.target

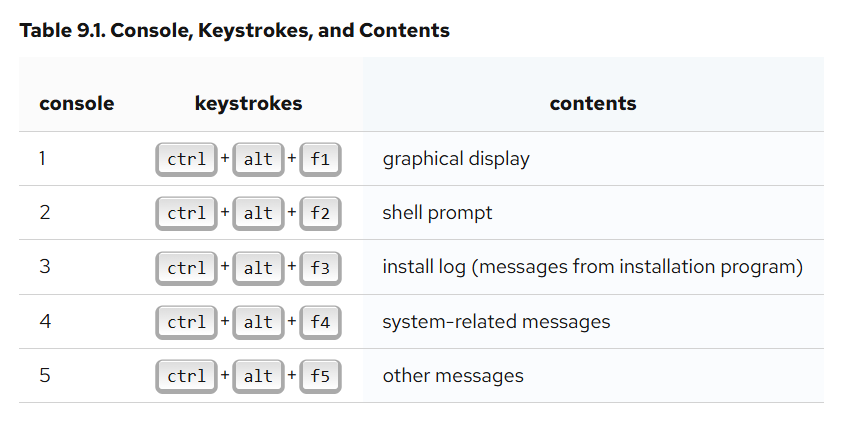
To set the system to boot into the GUI by default, adjust the system's default target: sudo systemctl set-default graphical.target

1. What are terminals in Linux? How many virtual terminals are available on your system, and which key combination is used to access them?

Soln>

Terminal:- The graphical window or application where we interact with the shell by typing commands

Generally, there are six virtual consoles/terminals in RHEL



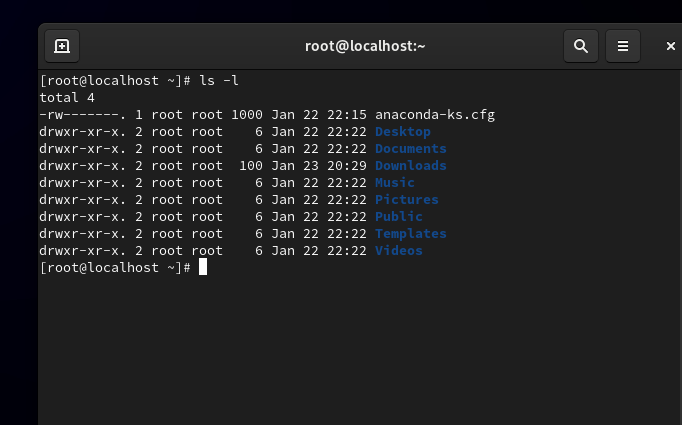
Generally, there is no reason to leave the default console (virtual console #6) for graphical installations unless you are attempting to diagnose installation problems.

1. Write the commands to check a file and a directory in a long listing format. How can you determine whether it is a file or a directory?

Soln>

To check file and a directory in a long listing format the command is:

ls -l



To identify a files/directory we have to look at first character at beginning of line

If it id “d” then it is a directory else it is file

1. Which Linux commands are used to view the content of files and directories? Write the commands.

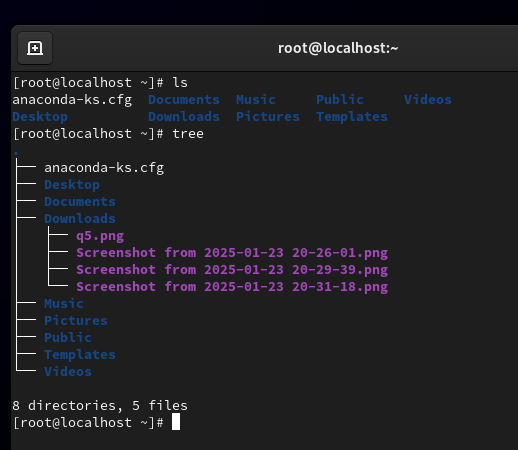
Soln>

To view content of directory

We use “ls” command with attributes eg -l to long list files and directories

We use “tree” to displays the directory structure in a tree-like format.

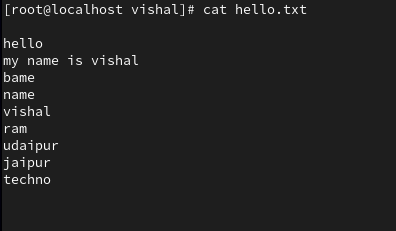
To view content of files



Commands

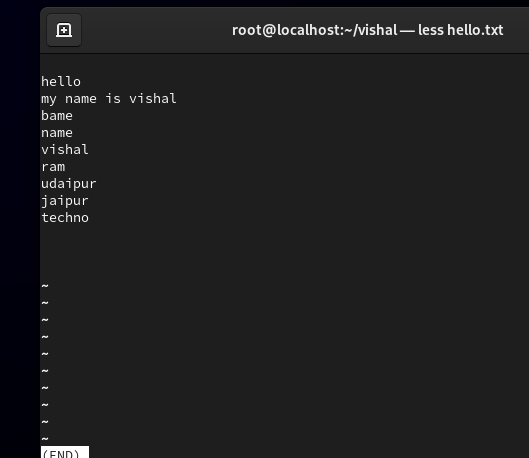
1. Cat (Concatenate) : Displays the entire content of a file.

cat filename.txt



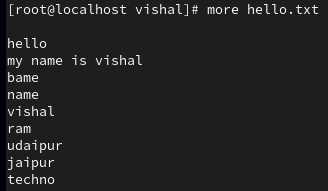
1. less : Displays the file content one page at a time.

less filename.txt



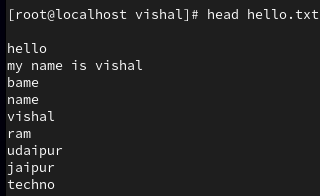
1. more : Similar to less, but with fewer navigation features.

more filename.txt



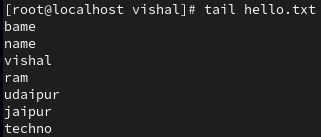
1. head : Displays the first few lines (default is 10) of a file.

head filename.txt



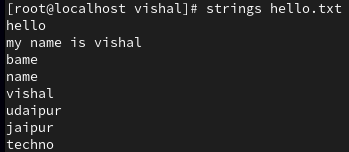
1. tail : Displays last few lines of a file( default 10

tail filename.txt



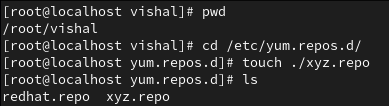
1. strings : Extracts and displays readable text strings from binary files.

strings filename



8. Change your current location to the **/etc/yum.repos.d** directory.

a. Using the relative path method, create a file named **xyz.repo** under **the /etc/yum.repos.d/** directory.



1. Using the absolute path method, create a file named **xyz.conf** under the **/etc/rsyslog.d/** directory.



1. What differences did you observe between using a relative path and an absolute path?

In case of relative path we are present in same directory where we want to create a file, and if we’re not in that directory then we have to move to that directory and using ./file\_name we can create a file.

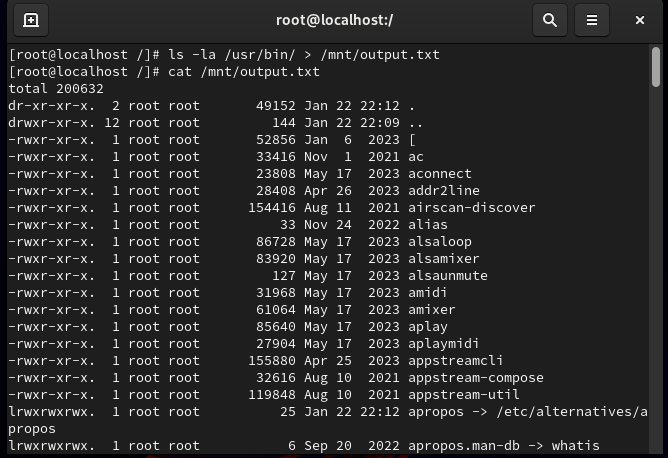
And in case of absolute path we have to specify full path starts with “/” which is main directory and move to the directory where we want to create a file.

1. List all files, including hidden ones, in the **/usr/bin/** directory with details like file permissions. Save the output to a file named **output.txt** in the **/mnt** directory. Write the command.

Soln>

Command to list all files including hidden ones and also store this output in “output.txt” file in /mnt directory using redirection.

ls -la /usr/bin/ > /mnt/output.txt



1. Create the parent directories **/Techno/Udaipur/Rajasthan/India/Asia/Earth/Solar** using one command. Then, check the full structure with details in a long listing format. Write the commands.

Soln>

Command to create parent directories which are not present and creating child one we use command:

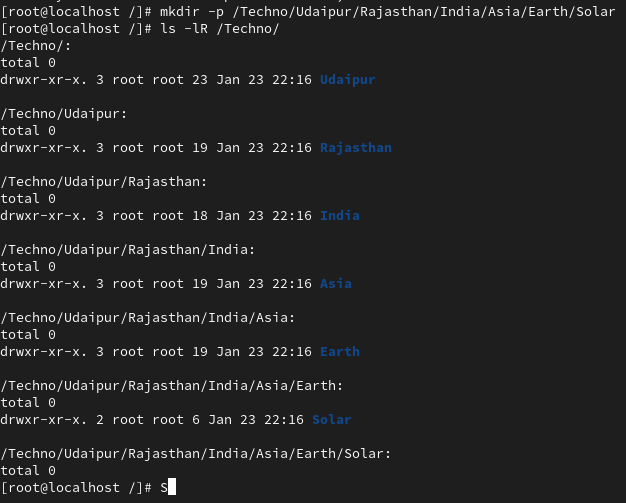
**mkdir -p /Techno/Udaipur/Rajasthan/India/Asia/Earth/Solar**

here -p Ensures that all parent directories are created if they don’t already exist.

And to check full structure we use command:

**ls -lR /Techno**

* **ls**: Lists the contents of a directory.
* **-l**: Displays details (permissions, owner, size, etc.).
* **-R**: Recursively lists all subdirectories and their contents.



1. Create a file named “**Poem”** under the **/Techno/Udaipur/Rajasthan/** directory. Write the text **"Twinkle Twinkle Little Star"** into the file and save it. Perform all actions using the absolute path method.

Soln>

To do this task in single command we have to use redirection as redirection creates file which is not present there and also writes the output.

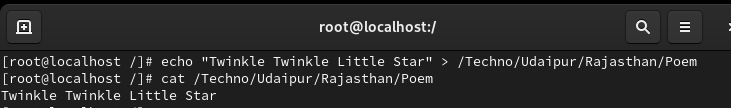
Command:

**echo "Twinkle Twinkle Little Star" > /Techno/Udaipur/Rajasthan/Poem**

**echo "Twinkle Twinkle Little Star"**: -> Prints the text Twinkle Twinkle Little Star.

**> -> used for redirection**

**/Techno/Udaipur/Rajasthan/Poem -> used to store output of command echo in Poem file**

****

1. Delete the **/Techno/Udaipur** directory, including its contents, using a single Linux command. Write the command.

Soln>

To delete these directories using single command we have to delete it recursively and forcely as it contains data and to avoid confirmation.

Commands:

**rm -rf /Techno/Udaipur**

**rm – remove command**

**-r – recursively delete directories**

**-f – forces the deletion of directories without confirmation**

**Other is path.**

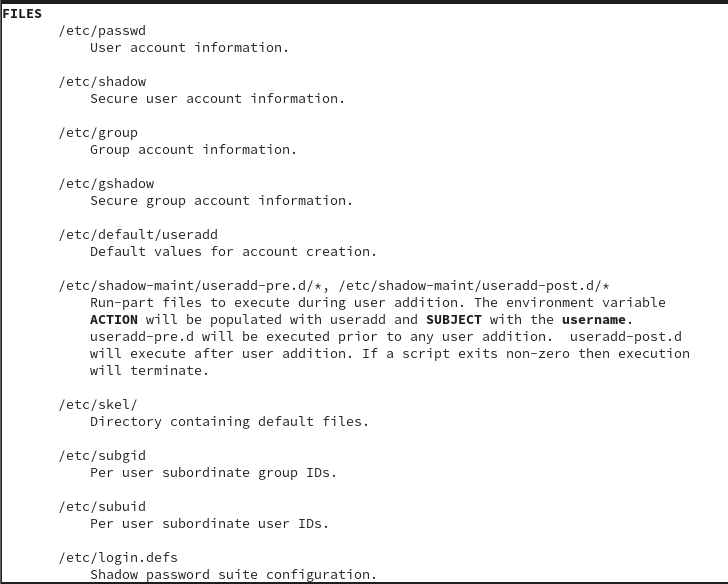


1. How can you view the manual page for the useradd command? From the manual page, identify which files are important for user administration.(Hint: Check the 'Files' Section)

Soln>

To view manual page of useradd command Shell provides us command:

**man useradd**



13. You have two files: **Fruits.txt and Vegetables.txt**, each containing related content.

a. Write content in fruits.txt using **vi.** (EX: **mango, apple, kiwi, grapes, cherry)**.

b. Write content in vegetables.txt using **nano.** (Ex: **potato, tomato, onion, chilli, garlic**).

c. Write the single command to Combine the contents of both files into a single file named **vegetarian** and display its content.

Soln>

Commands:

>Open vi editor with file name

**vi fruits.txt**

>In vi, press i to enter **Insert mode**, then type the following content:

mango

apple

kiwi

grapes

cherry

>Open vegetables.txt with nano editor:

**nano Vegetables.txt**

>In nano, type the following content:

potato

tomato

onion

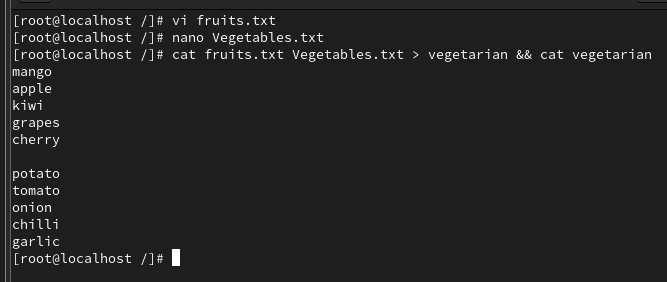
chilli

garlic

>Press **Ctrl + O** to **save** the file, then press **Enter**. To exit, press **Ctrl + X**.

**>Combine the contents of both files into a single file named vegetarian and display its content**:

**cat Fruits.txt Vegetables.txt > vegetarian && cat vegetarian**

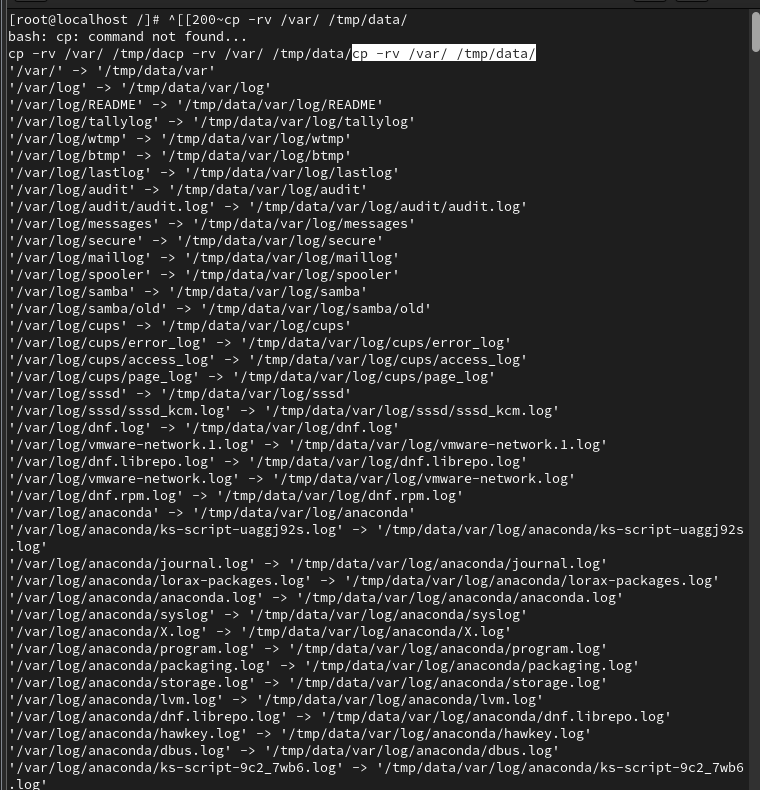
****

1. Write the command to copy all files, including related sub-files, from **/var** to a new location **/tmp/data/.** The output should be displayed during the copying process.

Soln>

Command:

**cp -rv /var/ /tmp/data/**

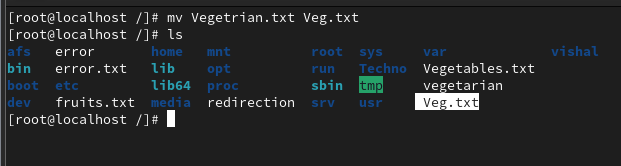


1. Rename the **file “Vegetrian.txt”** to **“Veg.txt”.** Write the command.

Soln>

To rename files we use mv (move) command as we provide new name and path so command is:

**mv Vegetrian.txt Veg.txt**



16. Open the file **“/etc/passwd** and locate the following lines using less and more:

a. Search for the text "**Root**" using the less command.

b. Search for the word "**root**" using the grep command.

c. What is the use difference between more and less commands?

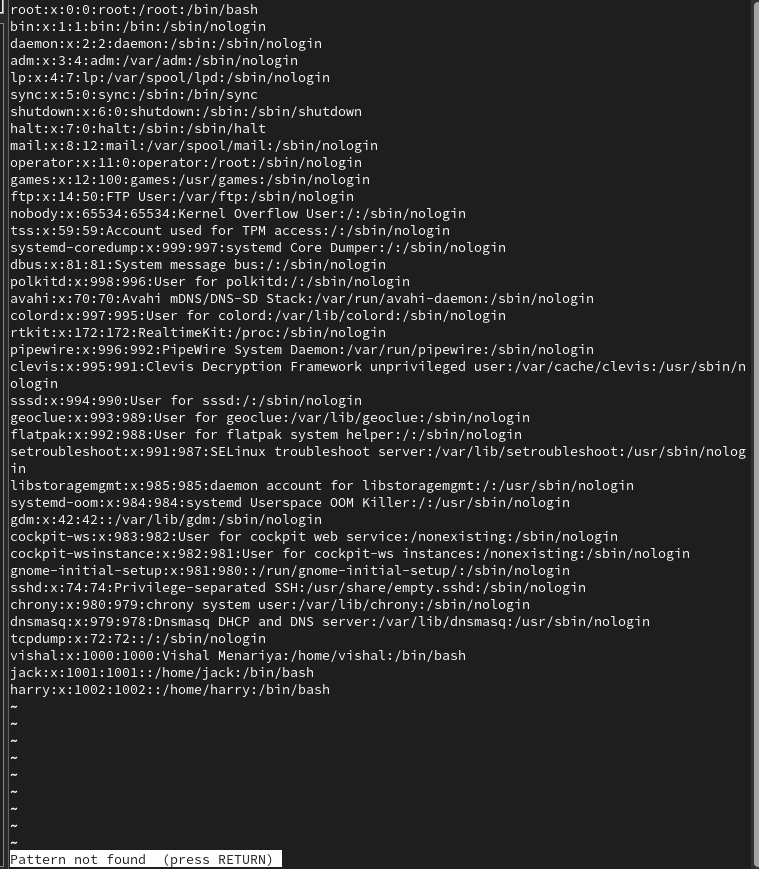
Soln>

a>Open file using less

**less /etc/passwd**

Search for text “Root”





**/Root**

Exit by pressing **q**

b>To find word root using grep command

**grep root /etc/passwd**

****

c> difference b/w more and less

In more there is only forward navigation while in case of less there is both forward and backward navigation to move on searched word.

In more there is limited search functionality while in less there is more search functionality like case sensation.

more loads half file and then acc. To requirement but less only displays one page content.

17. Perform the following tasks and write the commands to achieve them:

a. Display the **top 7th** line of the /etc/passwd file.

b. Display the **last 3** lines of the /etc/group file.

c. Display the lines **11th to 15th** from the /etc/shadow file using a pipeline.

d. Display **only the 16th** line of the /etc/passwd file.

Soln>

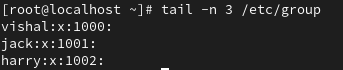
a. To display top 7th line of /etc/passwd we have to use pipeline

**head -n 7 /etc/passwd | tail -n 1**



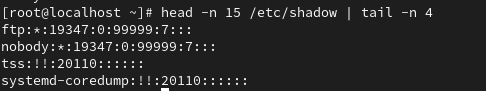
b. To display last 3 line of /etc/group file we use tail command as it directly gives no. of specified last lines of a file.

**tail -n 3 /etc/group**



c. Again we have to use pipeline as in pipeline output of one is served as input to other command

**head -n 15 /etc/shadow | tail -n 4**

****

d. using pipeline head and tail command 16 line is

**head -n 16 /etc/passwd | tail -n 1**

****

18. Perform the following tasks using the **grep** command on the **/etc/passwd** file:

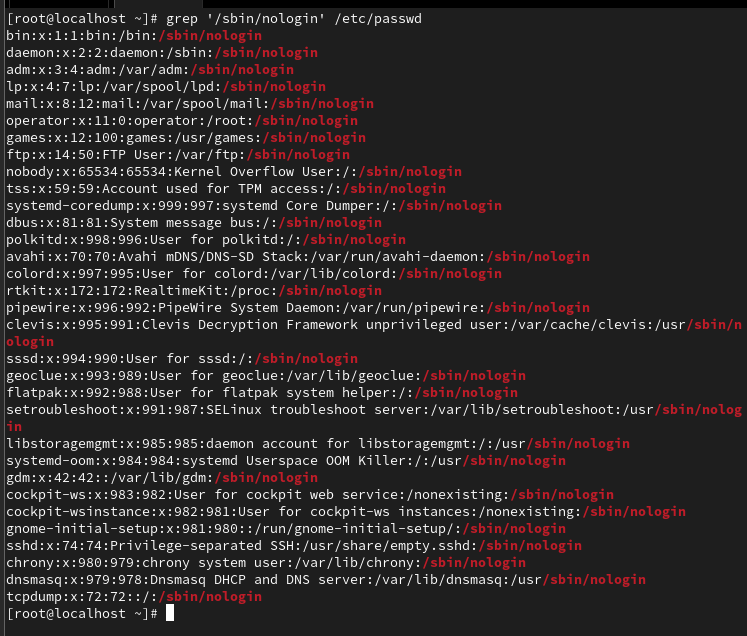
**a.** Write a command to match and display lines containing the word **/sbin/nologin.**

**b.** Write a command to match and display lines containing the multiple words (**root, sbin,** and **/home**) simultaneously, ignoring typographical case errors. Save the output for all three matches into the file **/root/test**. Without loosing data.

**Soln>**

a. To find the word /sbin/nologin command is:

**grep '/sbin/nologin' /etc/passwd**

****

b. command is:

**grep -i 'root' /etc/passwd | grep -i 'sbin' | grep -i '/home' >> /root/test**

19. Replace Text Using **sed** Linux Commands

a. Write the command to replace the word **localhost** with **localhost.localhost** in the file **/etc/hosts** without opening the file in an editor.

b. The **/var/log/audit/audit.log** file contains audit log messages, some of which include the word “**success.”** Write the command to count how many lines contain the word success.

Soln>

a. To do this command is:

**sudo sed -i 's/localhost/localhost.localhost/g' /etc/hosts**

here, sudo: Ensures you have the necessary permissions to edit the file.

sed -i: Edits the file **in place** (directly modifies the file).



b. To do this and including case sensitivity

**grep -c 'success' /var/log/audit/audit.log**

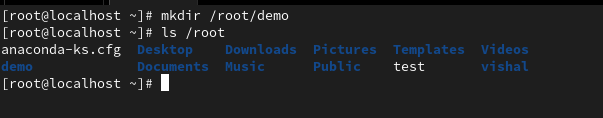
****

1. Create a directory named “**demo”** on “**/root**”.
   * 1. Create a file “**RedHat**” under the “demo” directory.
     2. Run the command “**vimtutor**” and save the output to the “RedHat” file.
     3. Now create a soft link of “RedHat” to “/etc/” location.
     4. Create a hard link of /var/log/messages to /etc/log.

Soln>

Command is:

**mkdir /root/demo**

****

a. To do this command is:

**touch /root/demo/Redhat**

****

b.To do this command is:

**vimtutor > /root/demo/RedHat**

****

c. To create softlink we use ln -s command

**ln -s /root/demo/RedHat /etc/RedHat**

****

d. To create hard link we use ln command only

**ln /var/log/messages /etc/log**

****