



Softwarica

College of IT & E-commerce



Name: Abhishek Adhikari

Class: Foundation Group 37A

Email: adhikaryabhishek209@gmail.com

Submitted to Rikesh Maharjan



Table of Contents

Linux Fundamentals Examination	1
Setting up the terminal:.....	1
1. Basic Commands:	1
a) What does Type 2 Hypervisor do? Explain with examples?	1
b) While running “LS -lha” this command i am having an error. Why am I getting errors with proof of concept?	2
c) I don't know how to use the cut command in linux so What two commands can I use in Kali Linux to access detailed help and documentation for system commands, and how can I view examples of their usage?	2
d) Find rockyou.txt or rockyou.txt.gz file from your computer using find command?	3
e) I need to find user accounts in my shadow file where the password field is specifically! * What command can I use to extract only these entries from the shadow file?	4
f) What is the significance of 2>/dev/null file?	4
g) What will happen when we provide 4777 permission to /bin/cat file? with proof of concept?	4
h) How would you use the find command to locate all files in the /var/log directory that have a .log extension, were modified in the last 7 days, and are smaller than 50MB larger than 5KB?	5
2. File System:.....	6
a) What happened when we run “sudo rm -rf /usr/bin/*” Describe this in brief, if possible , provide screenshots?.....	6
3. Standard Redirection:.....	6
a) Display the last 7 lines of a passwd file onto the terminal and save it as noob.txt.	6
b) Redirect these lines to a new file called hacked.txt located in the /var/tmp directory.....	6
c) Verify that summary.txt contains the last 7 lines of noob.txt and ensure no extra or missing lines are present. What command can you use to confirm that hacked.txt matches the extracted lines from noob.txt?	7
4. User Management and File Permissions:.....	7
a) Create a new user named "Love_CR7" with a unique password, a home directory in /home/Love_CR7, and a default shell of /bin/bash.	7

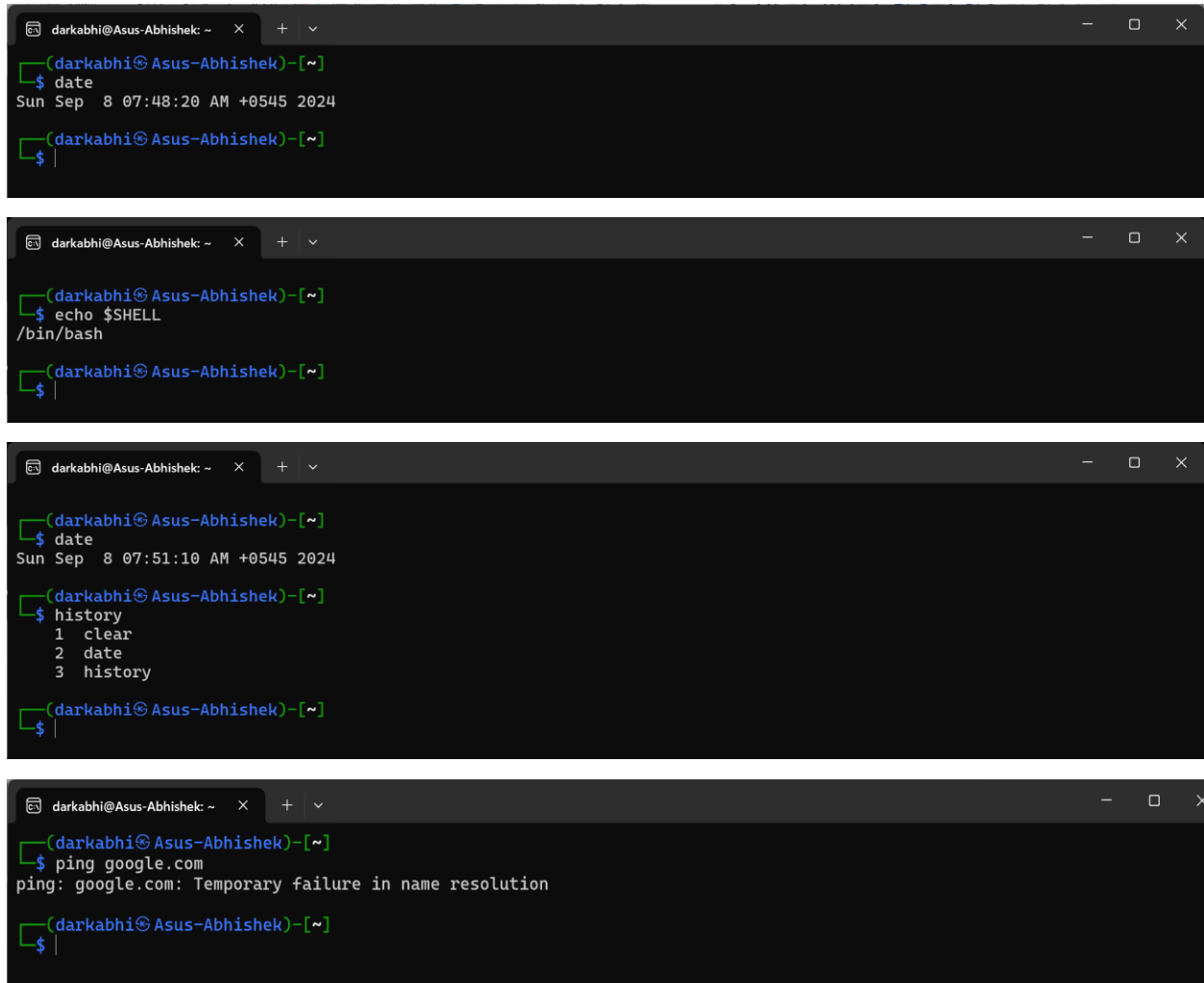
b) Add the user "Love_CR7" to the primary group "Lovers" and the secondary group "EURO_2024".	8
c) Set the uid of LOVE_CR7 user as 1337 because it was my Lucky number.	8
d) Create a file named "project_report.txt" in the home directory of "Love7" with the content "I love number 7 " and assign EURO_2024 group to this file.	8
e) Change the password for the user "Love_CR7" to a new password as "ronaldo7"	8
f) Delete the user "Love_CR7" and their home directory.	9
5. Virtualization:	9
a) What is Virtualization and why company need it also name 2 software which Provide virtualization platform.	9
b) How can we install the kali linux in Virtual box or vmware if we have (kali.ova, kali.vdi, kali.iso)?	10
6. Text Editor:	10
a) What command will you use to open the file "config.txt" in Vim?	10
b) What is the difference between normal mode and insert mode in Vim?	10
c) How can you undo the last change you made in Vim?	10
d) What command will you use to save the changes and exit Vim?	10
e) Write "I am noob" append this in hacker.txt and "i have been hacked" add this to hacker.txt file using echo command.	10
7. Create a Directory (Garuda):	11
a) Set the directory permissions so that only the owner has read, write, and execute access. For the group and others, allow read and execute access only, using numeric notation. Ensure the directory is set as a restricted deletion flag to prevent group members from deleting files they don't own.	11
b) Create three hidden files with different permissions inside the directory. (Using Character Notation)?	11
c) Set permissions for each file:	12
8. Create and Understanding about Directory:	13
a) Create a directory named testDir	13
b) Within testDir, create an empty file named testFile	13
c) Within testDir directory, use only one command to create the following empty files: testFile1, testFile2, testFile11, testFile22, testFile12, testFile21.	13

- d) Remove file(s) containing 1 in their file name without specifying their name 14
- e) Display the first 2 lines of /etc/passwd 14
- f) Create a file named passwd.txt that stores the last 2 lines of /etc/passwd 14

Linux Fundamentals Examination

Setting up the terminal:

Visual: The terminal screen shows the command.



```
darkabhi@Asus-Abhishek: ~  
$ date  
Sun Sep  8 07:48:20 AM +0545 2024  
$  
darkabhi@Asus-Abhishek: ~  
$ echo $SHELL  
/bin/bash  
$  
darkabhi@Asus-Abhishek: ~  
$ date  
Sun Sep  8 07:51:10 AM +0545 2024  
$ history  
1 clear  
2 date  
3 history  
$  
darkabhi@Asus-Abhishek: ~  
$ ping google.com  
ping: google.com: Temporary failure in name resolution  
$
```

1. Basic Commands:

a) What does Type 2 Hypervisor do? Explain with examples?

A Type 2 Hypervisor, also known as a hosted hypervisor, runs on a conventional operating system and abstracts hardware resources to create and run virtual machines. Examples include VMware and Oracle VirtualBox.



b) While running “LS -lha” this command i am having an error. Why am I getting errors with proof of concept?

The command `LS` is incorrect because the correct command is `ls`. Linux is case-sensitive, so `LS` will not be recognized as a valid command.

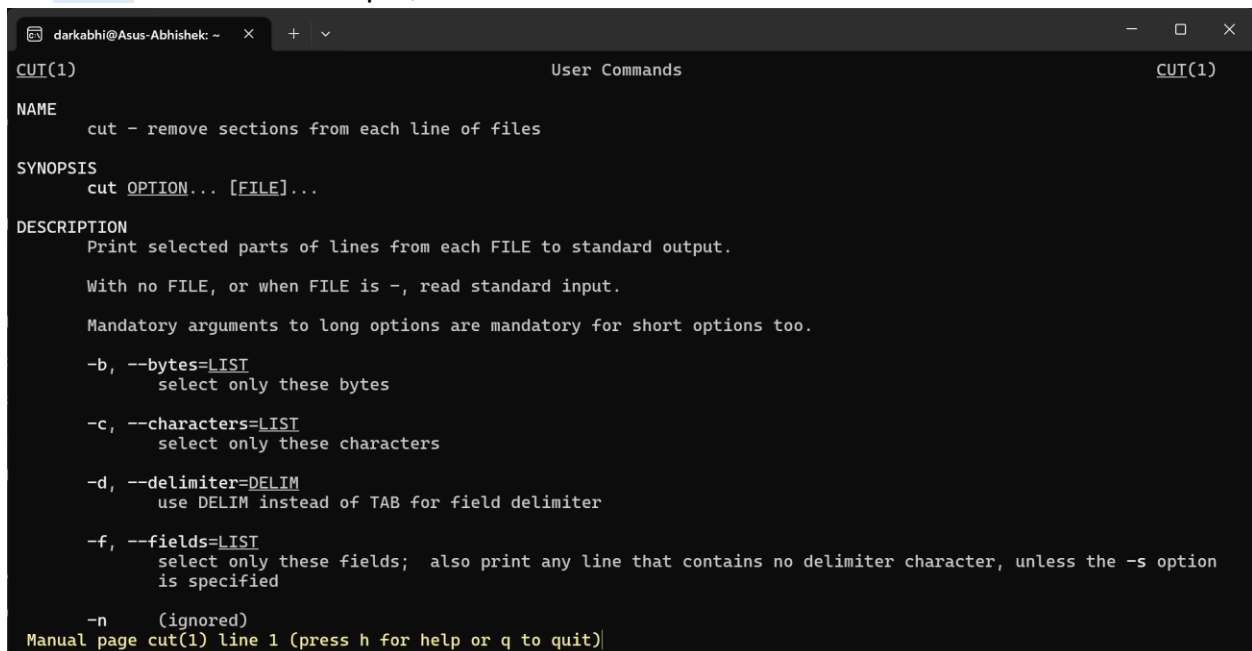
- **Proof of concept:**

A terminal window titled 'darkabhi@Asus-Abhishek: ~' shows a command prompt where the user has entered 'LS'. The terminal output is 'LS: command not found'. The prompt is now ready for the next command.

c) I don't know how to use the cut command in linux so What two commands can I use in Kali Linux to access detailed help and documentation for system commands, and how can I view examples of their usage?

You can use the following two commands to access help and documentation:

1. `man` cut – For example, `man cut` will show the manual for the cut command.

A terminal window titled 'darkabhi@Asus-Abhishek: ~' shows the output of the command 'man cut'. The output is formatted as a manual page for the 'cut' command. It includes sections for NAME, SYNOPSIS, DESCRIPTION, and a list of options: -b, -c, -d, -f, and -n. The bottom of the screen shows 'Manual page cut(1) line 1 (press h for help or q to quit)'.

2. `info` cut – For example, `info cut` provides detailed information and examples.



```
darkabhi@Asus-Abhishek: ~  
Next: paste invocation, Up: Operating on fields  
8.1 'cut': Print selected parts of lines  
=====
```

'cut' writes to standard output selected parts of each line of each input file, or standard input if no files are given or for a file name of '-'. Synopsis:

```
cut OPTION... [FILE]...
```

In the table which follows, the BYTE-LIST, CHARACTER-LIST, and FIELD-LIST are one or more numbers or ranges (two numbers separated by a dash) separated by commas. Bytes, characters, and fields are numbered starting at 1. Incomplete ranges may be given: '-M' means '1-M'; 'N-' means 'N' through end of line or last field. The list elements can be repeated, can overlap, and can be specified in any order; but the selected input is written in the same order that it is read, and is written exactly once.

The program accepts the following options. Also see [note Common options](#):

```
'-b BYTE-LIST'  
--bytes=BYTE-LIST  
Select for printing only the bytes in positions listed in  
BYTE-LIST. Tabs and backspaces are treated like any other  
character; they take up 1 byte. If an output delimiter is
```

```
-----Info: (coreutils)cut invocation, 103 lines --Top-----  
Welcome to Info version 7.1. Type H for help, h for tutorial.
```

d) Find rockyou.txt or rockyou.txt.gz file from your computer using find command?

This will search the entire file system for rockyou.txt but I use wsl and it didn't work even I fetched rockyou.txt file from github repository.

```
darkabhi@Asus-Abhishek: ~  
(darkabhi@Asus-Abhishek)~  
$ find / -name "rockyou.txt"  
find: '/root': Permission denied  
find: '/var/lib/private': Permission denied  
find: '/var/lib/apt/lists/partial': Permission denied  
find: '/var/log/private': Permission denied  
find: '/var/cache/private': Permission denied  
find: '/var/cache/ldconfig': Permission denied  
find: '/var/cache/apt/archives/partial': Permission denied  
find: '/var/spool/cron/crontabs': Permission denied  
find: File system loop detected; '/mnt/wslg/distro' is part of the same file system loop as '/'.  
find: '/mnt/c/$Recycle.Bin/S-1-5-18': Permission denied  
find: '/mnt/c/$Recycle.Bin/S-1-5-21-2264399860-4392884-3146558226-500': Permission denied  
^C  
(darkabhi@Asus-Abhishek)~  
$ sudo find / -name "rockyou.txt"  
find: File system loop detected; '/mnt/wslg/distro' is part of the same file system loop as '/'.  
find: '/mnt/c/$Recycle.Bin/S-1-5-18': Permission denied  
find: '/mnt/c/$Recycle.Bin/S-1-5-21-2264399860-4392884-3146558226-500': Permission denied  
^C  
(darkabhi@Asus-Abhishek)~  
$
```



e) I need to find user accounts in my shadow file where the password field is specifically! * What command can I use to extract only these entries from the shadow file?

This will search the /etc/shadow file for entries and greps all content with "!" then cuts the second field of the output array splitting from ":" and console into the terminal.

```
(darkabhi@Asus-Abhishek)~$ cat /etc/shadow | grep "!" | cut -f2 -d":"
!*
!*
!*
!*
(darkabhi@Asus-Abhishek)~$
```

f) What is the significance of 2 > /dev/null file?

It redirects standard error to /dev/null, which is a null file that discards all data written to it same as black hole of the universe which destroys the existence of the data.

g) What will happen when we provide 4777 permission to /bin/cat file? with proof of concept?

The permission 4777 means:

- **4 (SUID bit):** The program will run as the owner.
- **7 (rwx):** Read, write, and execute permissions for the owner, group, and others.

When setting 4777 permissions on /bin/cat, any user executing the cat command will do given command with root permissions, which is insecure or risky.

- **Proof of concept:**

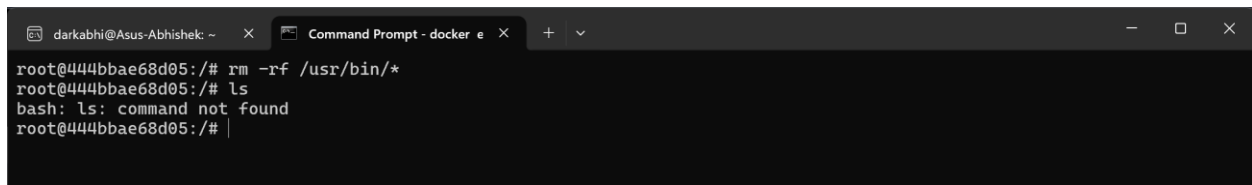
If a regular user runs the cat command, it will execute with elevated privileges.



2. File System:

a) What happened when we run “`sudo rm -rf /usr/bin/*`” Describe this in brief, **if possible**, provide screenshots?

Running this command will recursively and forcefully remove all files in `/usr/bin`, which contains essential system binaries same as C: Drive in windows. It can break the operating system, making it unbootable or unusable system crashes. Here, Docker virtual container is used to show the command as it will only crash the container which won't harm the system outside it.

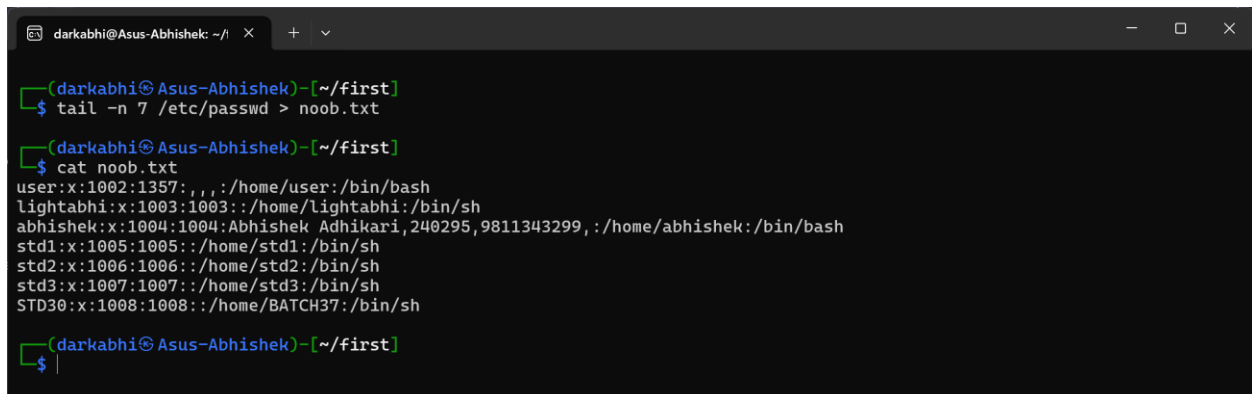


```
darkabhi@Asus-Abhishek: ~  
Command Prompt - docker e  
root@444bbae68d05:/# rm -rf /usr/bin/*  
root@444bbae68d05:/# ls  
bash: ls: command not found  
root@444bbae68d05:/#
```

3. Standard Redirection:

a) Display the last 7 lines of a `passwd` file onto the terminal and save it as `noob.txt`.

The `tail` command is used to display the context of the file from bottom and flag `-n` tells it to number of lines.



```
darkabhi@Asus-Abhishek: ~/first  
$ tail -n 7 /etc/passwd > noob.txt  
darkabhi@Asus-Abhishek: ~/first  
$ cat noob.txt  
user:x:1002:1357:,,,:/home/user:/bin/bash  
lightabhi:x:1003:1003:./home/lightabhi:/bin/sh  
abhishek:x:1004:1004:Abhishek Adhikari,240295,9811343299,./home/abhishek:/bin/bash  
std1:x:1005:1005:./home/std1:/bin/sh  
std2:x:1006:1006:./home/std2:/bin/sh  
std3:x:1007:1007:./home/std3:/bin/sh  
STD30:x:1008:1008:./home/BATCH37:/bin/sh  
darkabhi@Asus-Abhishek: ~/first  
$
```

b) Redirect these lines to a new file called `hacked.txt` located in the `/var/tmp` directory.

The `tail` command is used to display the context of the file from bottom and flag `-n` tells it to number of lines and redirecting it into the `/var/tmp/hacked.txt`.



```
darkabhi@Asus-Abhishek: ~/first
$ tail -n 7 /etc/passwd > /var/tmp/hacked.txt
darkabhi@Asus-Abhishek: ~/first
$
```

c) Verify that summary.txt contains the last 7 lines of noob.txt and ensure no extra or missing lines are present. What command can you use to confirm that hacked.txt matches the extracted lines from noob.txt?

The `diff` command is used to see the difference in 2 files. If null nothing means they are same.

```
darkabhi@Asus-Abhishek: ~/first
$ diff noob.txt /var/tmp/hacked.txt
darkabhi@Asus-Abhishek: ~/first
$
```

4. User Management and File Permissions:

a) Create a new user named "Love_CR7" with a unique password, a home directory in /home/Love_CR7, and a default shell of /bin/bash.

The `useradd` command is used to create new user and -m means make a home directory, -d means location of the home directory and -s means shell.

```
darkabhi@Asus-Abhishek: ~/first
$ sudo useradd -m -d /home/Love_CR7 -s /bin/bash Love_CR7
[sudo] password for darkabhi:
darkabhi@Asus-Abhishek: ~/first
$ sudo passwd Love_CR7
New password:
Retype new password:
passwd: password updated successfully
darkabhi@Asus-Abhishek: ~/first
$
```



b) Add the user "Love_CR7" to the primary group "Lovers" and the secondary group "EURO_2024".

The `usermod` command is used to change the group of the user, flags `-g` means primary group and `-G` means secondary group.

```
darkabhi@Asus-Abhishek: ~/first
$ sudo usermod -g Lovers -G EURO_2024 Love_CR7

darkabhi@Asus-Abhishek: ~/first
$ groups Love_CR7
Love_CR7 : Lovers EURO_2024

darkabhi@Asus-Abhishek: ~/first
$
```

c) Set the uid of LOVE_CR7 user as 1337 because it was my Lucky number.

The `usermod -u` command is used to change the uid of the group.

```
darkabhi@Asus-Abhishek: ~/first
$ sudo usermod -u 1337 Love_CR7

darkabhi@Asus-Abhishek: ~/first
$ id Love_CR7
uid=1337(Love_CR7) gid=1359(Lovers) groups=1359(Lovers),1360(EURO_2024)

darkabhi@Asus-Abhishek: ~/first
$
```

d) Create a file named "project_report.txt" in the home directory of "Love7" with the content "I love number 7" and assign EURO_2024 group to this file.

```
Love_CR7@Asus-Abhishek: ~
$ echo "I love number 7" > project_report.txt

Love_CR7@Asus-Abhishek: ~
$ cat project_report.txt
I love number 7
```

e) Change the password for the user "Love_CR7" to a new password as "ronaldo7".

The `passwd` command is used to change the password of the user.



```
Love_CR7@Asus-Abhishek: ~  
$ passwd Love_CR7  
Changing password for Love_CR7.  
Current password:  
New password:  
Retype new password:  
passwd: password updated successfully  
$
```

f) Delete the user "Love_CR7" and their home directory.

The `userdel` command is used to delete the user and `-r` means all files and folders linked with the user's home directory

```
darkabhi@Asus-Abhishek: ~  
$ sudo userdel -r Love_CR7  
userdel: group Love_CR7 not removed because it is not the primary group of user Love_CR7.  
userdel: Love_CR7 mail spool (/var/mail/Love_CR7) not found  
$ su - Love_CR7  
su: user Love_CR7 does not exist or the user entry does not contain all the required fields  
$
```

5. Virtualization:

a) What is Virtualization and why do companies need it? Also name 2 software which provide virtualization platform.

Virtualization: Virtualization refers to creating a virtual version of something, such as operating systems, servers, or storage devices. Companies need virtualization to reduce hardware costs, improve scalability, and optimize resource usage. Examples of software include VMware and VirtualBox.

Why companies need it: Companies use virtualization to reduce hardware costs, improve scalability, enhance disaster recovery options, and simplify management.

Two software platforms:

- VMware
- VirtualBox



b) How can we install the kali linux in Virtual box or vmware if we have (kali.ova, kali.vdi, kali.iso)?

- For .ova file: In VirtualBox or VMware, go to "File" > "Import Appliance" and select the .ova file.
- For .vdi file: Create a new VM in VirtualBox, then use the .vdi as the virtual disk.
- For .iso file: Create a new VM and attach the .iso file as the installation medium.

6. Text Editor:

a) What command will you use to open the file "config.txt" in Vim?

The `vim config.txt` command is use to open the file in vim.

b) What is the difference between normal mode and insert mode in Vim?

Normal mode: Used for navigating and editing text (copy, paste, delete, etc.).

Insert mode: Used for entering and modifying text.

c) How can you undo the last change you made in Vim?

Press `u` in normal mode to undo the last change.

d) What command will you use to save the changes and exit Vim?

The `:wq` command is use to save the changes and exit vim. Press `esc` and type `:wq` and then enter.

e) Write "I am noob" append this in hacker.txt and "i have been hacked" add this to hacker.txt file using echo command.

The `echo` command is use to write or append content to the file. `>` indicates write which will replace data of the file to user's input and `>>` indicates append means adding data's to the file.



```
darkabhi@Asus-Abhishek: ~  
$ echo "I am noob" >> hacker.txt  
$ echo "I have been hacked" >> hacker.txt  
$ cat hacker.txt  
I am noob  
I have been hacked  
$
```

7. Create a Directory (Garuda):

a) Set the directory permissions so that only the owner has read, write, and execute access. For the group and others, allow read and execute access only, using numeric notation. Ensure the directory is set as a restricted deletion flag to prevent group members from deleting files they don't own.

The `mkdir` command is use to make a new directory and `chmod` command is use to change the permission of the directory which indicates the permission of user, group and others. (read = 4, write = 2, execute = 1)

```
darkabhi@Asus-Abhishek: ~  
$ mkdir Garuda  
$ chmod 755 Garuda  
$ sudo chmod +t Garuda  
$
```

- 755: Owner has full permissions, while the group and others can only read and execute.
- +t: Sets the sticky bit, preventing group members from deleting files they don't own.

b) Create three hidden files with different permissions inside the directory. (Using Character Notation)?

The `touch` command is use to create the file and `.` indicates the hidden file.




```
darkabhi@Asus-Abhishek: ~  
$ touch Garuda/.Portugal.txt Garuda/.Team_CR7.txt Garuda/.public.txt  
$ ls -la Garuda/  
.. .Portugal.txt .public.txt .Team_CR7.txt  
$
```

c) Set permissions for each file:

The `chmod` command is used to change the permission of the file which refers to (read = 4, write = 2, execute = 1)

File 1: "Portugal.txt": Accessible only by the owner with read and write permissions and change the owner of the file to Love_CR7.

The `chown` command is used to change the owner of the file.

```
darkabhi@Asus-Abhishek: ~  
$ chmod 600 Garuda/.Portugal.txt  
$ sudo chown Love_CR7 Garuda/.Portugal.txt  
$ ls -l Garuda/.Portugal.txt  
-rw----- 1 Love_CR7 darkabhi 0 Sep  8 08:46 Garuda/.Portugal.txt  
$
```

File 2: "Team_CR7.txt" - Accessible only by the group with read and write permissions also change the group owner as EURO_2024.

The `chown` command is used to change the owner of the file.

```
darkabhi@Asus-Abhishek: ~  
$ chmod 600 Garuda/.Team_CR7.txt  
$ sudo chown Love_CR7:EURO_2024 Garuda/.Team_CR7.txt  
$
```

File 3: "public.txt" – Accessible to all users with read-only permissions and change the owner to Love_CR7 and group owner to EURO_2024.



```
darkabhi@Asus-Abhishek: ~  
$ sudo chmod 444 Garuda/.public.txt  
$ sudo chown Love_CR7:EURO_2024 Garuda/.public.txt  
$ ls -l Garuda/.public.txt  
-r--r--r-- 1 Love_CR7 EURO_2024 0 Sep  8 08:46 Garuda/.public.txt  
$
```

8. Create and Understanding about Directory:

a) Create a directory named testDir

The `mkdir` command is use to make a new directory.

```
darkabhi@Asus-Abhishek: ~  
$ mkdir testDir  
$ ls  
Ethical30.txt  first  Garuda  noob.txt  passwd.txt  second  testDir  
$
```

b) Within testDir, create an empty file named testFile

The `touch` command is use to make a new file.

```
darkabhi@Asus-Abhishek: ~  
$ touch testDir/testFile  
$ la testDir/  
testFile  
$
```

c) Within testDir directory, use only one command to create the following empty files: testFile1, testFile2, testFile11, testFile22, testFile12, testFile21.

The `touch` command is use to make a multiple file separated by spaces.



```
darkabhi@Asus-Abhishek: ~/testDir
$ touch testFile1 testFile2 testFile11 testFile22 testFile12 testFile21

darkabhi@Asus-Abhishek: ~/testDir
$ ls
testFile  testFile1  testFile11  testFile12  testFile2  testFile21  testFile22

darkabhi@Asus-Abhishek: ~/testDir
$
```

d) Remove file(s) containing 1 in their file name without specifying their name

The `rm` command is used to remove the file and `*` indicates all or any but in this case `1` is in the middle means anything 1 and anything then it removes the files containing 1.

```
darkabhi@Asus-Abhishek: ~/testDir
$ rm *1*

darkabhi@Asus-Abhishek: ~/testDir
$ ls
testFile  testFile2  testFile22

darkabhi@Asus-Abhishek: ~/testDir
$
```

e) Display the first 2 lines of /etc/passwd

The `head -n 2` command is used to display the file from top and flag `-n 2` means numbers of line from above.

```
darkabhi@Asus-Abhishek: ~/testDir
$ head -n 2 /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin

darkabhi@Asus-Abhishek: ~/testDir
$
```

f) Create a file named passwd.txt that stores the last 2 lines of /etc/passwd

The `tail -n 2` command is used to display the file from bottom and flag `-n 2` means numbers of line from bottom and `>` will create the file for us.



```
darkabhi@Asus-Abhishek: ~/testDir
$ tail -n 2 /etc/passwd > passwd.txt

darkabhi@Asus-Abhishek: ~/testDir
$ ls
passwd.txt  testFile  testFile2  testFile22

darkabhi@Asus-Abhishek: ~/testDir
$ cat passwd.txt
STD30:x:1008:1008:./home/BATCH37:/bin/sh
Love_CR7:x:1009:1009:./home/Love_CR7:/bin/sh

darkabhi@Asus-Abhishek: ~/testDir
$
```

