SCHOOL OF ENGINEERING AND TECHNOLOGY (SOET)

ASSINGMENT-2

COMPUTER SCIENCE FUNDAMENTAL AND CAREER PATHWAYS

TOPIC-

Basics of Linux and Open-Source Tools

SUBMITTED BY-

Medha Jain

2501010094

Btech cse core- 1ˢᵗ semester

SUBMITTED TO-

Rajesh Badrana

INTRODUCTION:

The purpose of this assignment is to explore and understand how Linux and shell scripting work in real-world computing. Linux is widely known for its stability, security, and open-source nature, making it an essential part of the tech world. Shell scripting, on the other hand, helps automate everyday system tasks and saves both time and effort. Through this assignment, we get hands-on experience with important Linux commands and learn how to create simple scripts for tasks like file management, system monitoring, and automation.

STEP-1 (LINUX INSTALLATION):

1. Download a virtual box on which we are going to run our linux iso file, for this I have used oracle virtual box .

2. Now download an iso file , for this I am using ubuntu.

3. Now after installing both ubuntu and a virtual box we have to create a virtual machine .

4. Now, for creating a virtual machine you first need to open the virtual box and click on new button present on the toolbar .

5. After that fill all the necessary requirements and click on finish.

6. Now double click on your recently made ubuntu and let that machine power up , after few seconds the screen will pop up with ubuntu running on it .

7. Our virtual machine is fully ready and we can now run commands on it's terminal.

STEP-2 (Shell Command Implementation and Documentation):

1. pwd

□ Syntax: pwd

□ Desc: Print current working directory.

□ Example: pwd → /home/medha

```
(base) nishchay@nishchay-Aspire-A315-23:~$ pwd
/home/nishchay
(base) nishchay@nishchay-Aspire-A315-23:~$ ▮
```

2. ls

□ Syntax: ls [options] [path]

□ Desc: List files.

□ Example: ls -l → shows long listing with permissions, owner, size, date.

```
(base) nishchay@nishchay-Aspire-A315-23:~$ ls
anaconda3            Documents         Pictures        Videos
cpp-lab              Downloads         Public          yolov8-eev
'def even_odd(nums).py'  'factorial= {}.py'   snap         yolov8n.onnx
'def zero(arr):.py'      Music             Templates      yolov8n.pt
 Desktop              nischay           Untitled.ipynb  yolo_webcam.py
(base) nishchay@nishchay-Aspire-A315-23:~$ ls -1
anaconda3
cpp-lab
'def even_odd(nums).py'
'def zero(arr):.py'
Desktop
Documents
Downloads
'factorial= {}.py'
Music
nischay
Pictures
Public
snap
Templates
Untitled.ipynb
Videos
yolov8-eev
yolov8n.onnx
yolov8n.pt
yolo_webcam.py
```

3. cd

□ Syntax: cd [dir]

□ Desc: Change directory.

□ Example: cd /home/medha/Documents

```
1 directory, 1 file
(base) nishchay@nishchay-Aspire-A315-23:~/Documents$ cd ~/Documents
(base) nishchay@nishchay-Aspire-A315-23:~/Documents$
```

4. tree

□ Syntax: tree [path]

□ Desc: Shows the directory structure in a tree-like (hierarchical) format.

□ Example: tree -L 2 → nested structure.

```
(base) nishchay@nishchay-Aspire-A315-23:~$ cd ~/Documents
(base) nishchay@nishchay-Aspire-A315-23:~/Documents$ tree
.
└── BigData_ExperimentList.docx

1 directory, 1 file
(base) nishchay@nishchay-Aspire-A315-23:~/Documents$
```

5. mkdir

□ Syntax: mkdir [options] dir

□ Desc: Make directory.

□ Example: mkdir project1

```
(base) nishchay@nishchay-Aspire-A315-23:~/Documents$ mkdir medhaproject1
(base) nishchay@nishchay-Aspire-A315-23:~/Documents$ cd medhaproject1
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$
```

6. touch

□ Syntax: touch filename

□ Desc: Create empty file or update timestamp.

Example: touch notes.txt

```
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$ touch file.txt
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$ ls
file.txt
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$
```

7. cp

 Syntax: cp source dest

 Desc: Copy files/directories (-r for dir).

 Example: cp file1.txt ~/backup/

```
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$ cp file.txt file1.txt
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$ ls
file1.txt  file.txt
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$
```

8. mv

 Syntax: mv source dest

 Desc: Move/rename files.

 Example: mv oldname.txt newname.txt

```
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$ mkdir folder
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$ mv file1.txt folder/
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$ ls
file.txt  folder
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$
```

9. rm

 Syntax: rm [options] file

 Desc: Remove files (-r for dirs, be careful).

 Example: rm -r tempdir

```
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$ rm file.txt
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$ ls
folder
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$
```

10. chmod

 Syntax: chmod [mode] file or chmod u+x script.sh

 Desc: Change file permissions.

 Example: chmod +x backup.sh

```
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$ chmod 755 script.sh
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$ ls
file.txt  foldername  myfolder  script.sh
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$
```

11. chown

 Syntax: sudo chown user:group file

 Desc: Change owner/group.

 Example: sudo chown medha: medha notes.txt

```
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$ sudo chown nishchay.nishc
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$ ls
file.txt  foldername  myfolder  script.sh
(base) nishchay@nishchay-Aspire-A315-23:~/Documents/medhaproject1$
```

12. cat

 Syntax: cat file

 Desc: Display file contents(Concatenate and display files).

 Example: cat README.md

```
(base) nishchay@nishchay-Aspire-A315-23:~$ cat > file.txt
my name is medha jain and this is my first time using linux ,linux is fin!!
```

13. less / more

 Syntax: less file

 Desc: View long files page by page.

```
my name is medha jain and this is my first time using linux ,linux is fin!!
.
.
.
.
.
.
.
- -
.
.
.
.
.
.
.
-
file.txt (END)
```

14. grep

 Syntax: grep 'pattern' filename

 Desc: Search for text(Global Regular Expression Print).

 Example: grep -i "error" logfile

```
(base) nishchay@nishchay-Aspire-A315-23:~$ grep "linux" file.txt
my name is medha jain and this is my first time using linux ,linux is fin!!
```

15. ps

 Syntax: ps aux

 Desc: Show running processes(Process Status).

 Example: ps aux | grep firefox

```
(base) nishchay@nishchay-Aspire-A315-23:~$ ps
    PID TTY          TIME CMD
  19414 pts/0    00:00:00 bash
  19467 pts/0    00:00:00 ps
(base) nishchay@nishchay-Aspire-A315-23:~$
```

16. top / htop

 Syntax: top

 Desc: Interactive process monitor (sudo apt install htop for htop)(Table of Processes).

 Example: run top and screenshot CPU%/MEM% lines.

```
top - 21:27:00 up 7:05,  1 user,  load average: 0.57, 0.62, 0.63
Tasks: 316 total,   1 running, 315 sleeping,   0 stopped,   0 zombie
%Cpu(s):  1.5 us,  0.5 sy,  0.0 ni, 97.9 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem :   5860.9 total,    214.7 free,   3841.1 used,   2550.7 buff/cache
MiB Swap:   4096.0 total,   3081.0 free,   1014.9 used.   2019.8 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
  19406 nishchay  20   0  706900  64560  52228 S   7.3   1.1   0:02.38 gnome-terminal-
   2328 nishchay  20   0 5416860 302784 113596 S   5.6   5.0  14:25.30 gnome-shell
   7920 nishchay  20   0 1393.8g 444416 144460 S   1.0   7.4  11:51.54 brave
    202 root     -51   0       0      0      0 S   0.7   0.0   0:45.32 irq/50-SYNA7DB5:00
  19514 nishchay  20   0   14544   5852   3676 R   0.7   0.1   0:00.11 top
   6068 nishchay  20   0   32.4g  85432  70876 S   0.3   1.4   2:12.87 brave
   7815 nishchay  20   0 1392.4g 514928 156852 S   0.3   8.6   6:34.52 brave
  14313 nishchay  20   0   32.5g  75464  63428 S   0.3   1.3   0:06.95 whatsapp-linux-
  14370 nishchay  20   0 1136.4g 500556 280032 S   0.3   8.5   3:36.39 whatsapp-linux-
  17051 root      20   0       0      0      0 I   0.3   0.0   0:00.19 kworker/3:3-events
      1 root      20   0   23764  14068   9400 S   0.0   0.2   0:18.97 systemd
      2 root      20   0       0      0      0 S   0.0   0.0   0:00.05 kthreadd
      3 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 pool_workqueue_release
      4 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-rcu_gp
      5 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-sync_wq
      6 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-kvfree_rcu_reclaim
      7 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-slub_flushwq
      8 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-netns
     11 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/0:0H-events_highpri
     13 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-mm_percpu_wq
     14 root      20   0       0      0      0 I   0.0   0.0   0:00.00 rcu_tasks_kthread
     15 root      20   0       0      0      0 I   0.0   0.0   0:00.00 rcu_tasks_rude_kthread
     16 root      20   0       0      0      0 I   0.0   0.0   0:00.00 rcu_tasks_trace_kthread
     17 root      20   0       0      0      0 S   0.0   0.0   0:00.52 ksoftirqd/0
     18 root      20   0       0      0      0 I   0.0   0.0   0:10.23 rcu_preempt
     19 root      20   0       0      0      0 S   0.0   0.0   0:00.00 rcu_exp_par_gp_kthread_worker/0
     20 root      20   0       0      0      0 S   0.0   0.0   0:00.02 rcu_exp_gp_kthread_worker
     21 root      rt   0       0      0      0 S   0.0   0.0   0:00.13 migration/0
     22 root     -51   0       0      0      0 S   0.0   0.0   0:00.00 idle_inject/0
     23 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/0
     24 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/2
     25 root     -51   0       0      0      0 S   0.0   0.0   0:00.00 idle_inject/2
     26 root      rt   0       0      0      0 S   0.0   0.0   0:00.37 migration/2
     27 root      20   0       0      0      0 S   0.0   0.0   0:00.18 ksoftirqd/2
     29 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/2:0H-events_highpri
     30 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/4
     31 root     -51   0       0      0      0 S   0.0   0.0   0:00.00 idle_inject/4
     32 root      rt   0       0      0      0 S   0.0   0.0   0:00.37 migration/4
```

17. kill / kill all

 Syntax: kill PID or killall processname

 Desc: Terminate processes.

 Example: kill 1234

```
(base) nishchay@nishchay-Aspire-A315-23:~/Documents$ kill 1234
bash: kill: (1234) - Operation not permitted
(base) nishchay@nishchay-Aspire-A315-23:~/Documents$
```

18. ping

☐ Syntax: ping -c 4 google.com

☐ Desc: Check network connectivity(Packet Internet Groper).

☐ Example: ping google.com

```
(base) nishchay@nishchay-Aspire-A315-23:~$ ping -c 4 google.com
PING google.com (172.217.24.78) 56(84) bytes of data.
64 bytes from hkg07s33-in-f14.1e100.net (172.217.24.78): icmp_seq=1 ttl=118 time=2.21 ms
64 bytes from hkg07s33-in-f14.1e100.net (172.217.24.78): icmp_seq=2 ttl=118 time=2.98 ms
64 bytes from hkg07s33-in-f14.1e100.net (172.217.24.78): icmp_seq=3 ttl=118 time=2.91 ms
64 bytes from hkg07s33-in-f14.1e100.net (172.217.24.78): icmp_seq=4 ttl=118 time=3.07 ms

--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 2.213/2.794/3.074/0.340 ms
(base) nishchay@nishchay-Aspire-A315-23:~$
```

19. ip / ip addr

☐ Syntax: ip addr (modern) or ifconfig (install net-tools)

☐ Desc: Show network interfaces and IP addresses.

☐ Example: ip , ip addr

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: enp3s0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq_codel state DOWN group default qlen 1000
    link/ether c0:18:50:a6:c5:1c brd ff:ff:ff:ff:ff:ff
3: wlp2s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether c4:03:a8:42:b5:8a brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.104/24 brd 192.168.0.255 scope global dynamic noprefixroute wlp2s0
       valid_lft 6413sec preferred_lft 6413sec
    inet6 fe80::679:4be4:4fc:bf1e/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
(base) nishchay@nishchay-Aspire-A315-23:~$
```

```
(base) nishchay@nishchay-Aspire-A315-23:~$ ip
Usage: ip [ OPTIONS ] OBJECT { COMMAND | help }
       ip [ -force ] -batch filename
where  OBJECT := { address | addrlabel | amt | fou | help | ila | ioam | l2tp |
                   link | macsec | maddress | monitor | mptcp | mroute | mrule |
                   neighbor | neighbour | netconf | netns | nexthop | ntable |
                   ntbl | route | rule | sr | tap | tcpmetrics |
                   token | tunnel | tuntap | vrf | xfrm }
       OPTIONS := { -V[ersion] | -s[tatistics] | -d[etails] | -r[esolve] |
                    -h[uman-readable] | -iec | -j[son] | -p[retty] |
                    -f[amily] { inet | inet6 | mpls | bridge | link } |
                    -4 | -6 | -M | -B | -0 |
                    -l[oops] { maximum-addr-flush-attempts } | -br[ief] |
                    -o[neline] | -t[imestamp] | -ts[hort] | -b[atch] [filename] |
                    -rc[vbuf] [size] | -n[etns] name | -N[umeric] | -a[ll] |
                    -c[olor]}
```

20. netstat(Network Statistics) / ss

☐ Syntax: ss -tuln or netstat -tuln

☐ Desc: Show listening ports and connections.

☐ Example:netstat -tuln

```
Netid  State   Recv-Q  Send-Q   Local Address:Port      Peer Address:Port   Process
udp    UNCONN  0       0        224.0.0.251:5353        0.0.0.0:*
udp    UNCONN  0       0        224.0.0.251:5353        0.0.0.0:*
udp    UNCONN  0       0        0.0.0.0:5353            0.0.0.0:*
udp    UNCONN  0       0        0.0.0.0:42706           0.0.0.0:*
udp    UNCONN  0       0        127.0.0.54:53           0.0.0.0:*
udp    UNCONN  0       0        127.0.0.53%lo:53        0.0.0.0:*
udp    UNCONN  0       0                                [::]:5353           [::]:*
udp    UNCONN  0       0                                [::]:52818          [::]:*
tcp    LISTEN  0       4096     127.0.0.1:631           0.0.0.0:*
tcp    LISTEN  0       4096     127.0.0.53%lo:53        0.0.0.0:*
tcp    LISTEN  0       4096     127.0.0.54:53           0.0.0.0:*
tcp    LISTEN  0       4096                             [::1]:631           [::]:*
(base) nishchay@nishchay-Aspire-A315-23:~$
```

```
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address      Foreign Address    State
tcp        0      0 127.0.0.1:631      0.0.0.0:*          LISTEN
tcp        0      0 127.0.0.53:53      0.0.0.0:*          LISTEN
tcp        0      0 127.0.0.54:53      0.0.0.0:*          LISTEN
tcp6       0      0 ::1:631            :::*               LISTEN
udp        0      0 224.0.0.251:5353   0.0.0.0:*
udp        0      0 224.0.0.251:5353   0.0.0.0:*
udp        0      0 0.0.0.0:5353       0.0.0.0:*
udp        0      0 0.0.0.0:42706      0.0.0.0:*
udp        0      0 127.0.0.54:53      0.0.0.0:*
udp        0      0 127.0.0.53:53      0.0.0.0:*
udp6       0      0 :::5353            :::*
udp6       0      0 :::52818           :::*
(base) nishchay@nishchay-Aspire-A315-23:~$
```

Step 3 - Shell Script Development

1. Script Name: backup.sh
   Purpose: To back up a directory with timestamp.
   Commands Used: mkdir, tar, date
   Observation: The script successfully created a compressed backup file with timestamp in the backup folder.

2. Script Name: monitor.sh
   Purpose: To log CPU and memory usage at fixed intervals.
   Commands Used: top, sleep, redirection (>>)
   Observation: The script recorded system performance continuously in a text log file.

3. Script Name: downloader.sh
   Purpose: To automatically download a file using wget.
   Commands Used: wget, mkdir, date
   Observation: The script downloaded a file and saved it with a timestamp in a predefined directory.

GIT HUB REPOSITORY LINK: **https://github.com/medhajain790-spec/csfcp**

REFLECTION:

While working on this assignment, I faced a few challenges in understanding how different Linux commands interact with the file system and how shell scripts execute step by step. Setting up the virtual environment and managing permissions also required some troubleshooting, which helped me understand the system more deeply.

Through this task, I learned how automation using shell scripting can make everyday work easier — whether it's taking backups, monitoring system performance, or downloading files automatically. These skills are extremely useful in the real world, especially for system administrators, developers, and anyone working in DevOps or cloud environments. This assignment not only improved my technical knowledge but also gave me confidence in using Linux as a powerful tool for problem-solving.