

# **Certificate**

This is to certify that the practical report entitled

## **“Practical File for M.Sc. Cyber Security”**

is submitted by

**Name: Dhrumil Soni  
Enrollment No: IU2554520008  
Class: M.Sc. Cyber Security**

under the guidance of  
**Prof. Varsha Patel**

**Academic Year: 2025–2026**

# TABLE OF CONTENTS

Practical No. 1 . . . . .	1
---------------------------	---

# Practical No. 1

Date: 07/Jul/2025

## System & Process Management

### 1. Command: top

**Description:** Displays real-time information about running processes and system resource usage.

**Attributes/Options:**

- -u <username>: Show processes for a specific user
- -p <pid>: Show a specific process

**Example:** top -u root

### Screenshot

The screenshot shows a terminal window titled 'root@host: ~' displaying the output of the 'top' command. The output provides real-time statistics for various processes, including CPU usage, memory usage, and system load. Key columns include PID, User, %CPU, %MEM, and TIME+COMMAND. The terminal interface includes standard Linux navigation keys like Esc, F1-F12, and arrow keys.

Figure 1: Output of top command

### 2. Command: htop

**Description:** An interactive process viewer, enhanced version of top.

**Attributes/Options:**

- -d <delay>: Set the delay time between updates

- `-u <user>`: Show processes for a user

Example: htop -u kali

## Screenshot

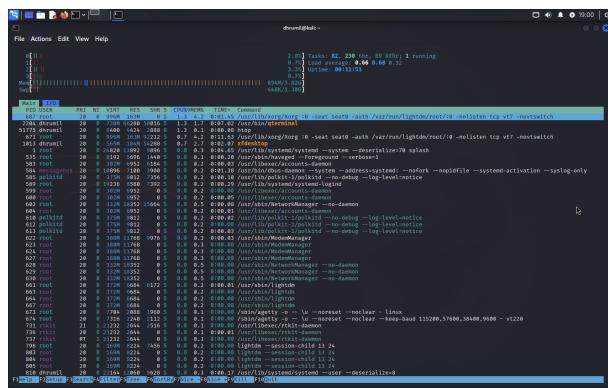


Figure 2: Output of htop command

### 3. Command: nice

**Description:** Launches a program with a custom scheduling priority.

## Attributes/Options:

- **-n <value>**: Sets the nice value (from -20 to 19)

**Example:** nice -n 10 python3 script.py

## Screenshot



Figure 3: Output of nice command

## 4. Command: renice

**Description:** Changes the priority of a running process.

**Attributes/Options:**

- **-n <value> -p <pid>**: Change priority of a process

**Example:**

## Screenshot



Figure 4: Output of renice command

## 5. Command: pidstat

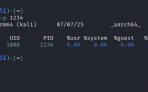
**Description:** Monitors statistics for individual Linux tasks (CPU, memory usage, etc.).

**Attributes/Options:**

- **-u**: Show CPU usage
- **-p <pid>**: Monitor specific process

**Example:** pidstat -u -p 1234

## Screenshot



```
[File Actions Edit View Help] dwon@kali: ~
```

```
---[dwon@kali: ~]---
```

```
[*] 9 processes running on 1224
```

```
(1 process, 6.13% CPU) 07/27/25 .src64a (4 CPU)
```

PTID	Name	System	Request	Waited	WCPU	CPU	Command
1224	0.00	0.10	0.00	0.00	0.00	0	gnash

```
[*] dwon@kali: ~]
```

Figure 5: Output of pidstat command

## 6. Command: strace

**Description:** Traces system calls and signals of a command, useful for debugging.

## Attributes/Options:

- **-e trace=all**: Trace all system calls
  - **-p <pid>**: Attach to running process

**Example:** strace -e trace=all ls

## Screenshot

Figure 6: Output of strace command

## 7. Command: lsof

**Description:** Lists open files and which processes opened them.

## Attributes/Options:

- **-i**: Show network files
  - **-u <user>**: Show files opened by a user

**Example:** ls of -i

# Screenshot

Figure 7: Output of lsof command

## 8. Command: watch

**Description:** Repeats a command periodically and shows fullscreen output.

## Attributes/Options:

- **-n <sec>**: Set interval in seconds
  - **-d**: Highlight differences

**Example:** `watch -n 2 date`

## Screenshot



Figure 8: Output of watch command

## 9. Command: systemctl

**Description:** Controls the systemd system and service manager.

**Attributes/Options:**

- >start—stop—status service; Manage services
- >enable—disable service; Set startup behavior

**Example:** `systemctl status ssh`

## Screenshot



Figure 9: Output of systemctl command

## 10. Command: journalctl

**Description:** Views logs collected by the systemd journal service.

**Attributes/Options:**

- **-xe**: Shows the most recent logs with extra info
  - **-u <service>**: Filter logs by service

**Example:** journalctl -u ssh

## Screenshot

Figure 10: Output of journalctl command

# File & Directory Operations

## Aim

To explore file and directory management commands used in Kali Linux.

## Theory

File and directory commands help manage the structure, synchronization, search, and comparison of files across the Linux system.

## Flow Chart

- Step 1: Choose the desired operation (sync, search, compare)
  - Step 2: Execute appropriate command with suitable options
  - Step 3: Review output to verify results

## 1. Command: rsync

**Description:** Efficiently synchronizes files/directories between two locations.

## Attributes/Options:

- **-a:** Archive mode (preserves structure/permissions)
  - **--progress:** Shows progress during transfer

**Example:** rsync -a --progress source/ destination/

## Screenshot

Figure 11: Output of rsync command

## 2. Command: scp

**Description:** Securely copies files between hosts using SSH.

## Attributes/Options:

- **-r**: Recursively copy directories
  - **-P <port>**: Specify remote SSH port

**Example:** `scp -r folder user@192.168.0.1:/home/user/`

## Screenshot



A screenshot of a terminal window titled "davids@kali:~". The window shows the command "scp -r /home/david/Desktop/\* davids@192.168.0.1:/root/" being typed. The output of the command is displayed below the command line, showing the progress of file transfers.

Figure 12: Output of scp command

## 3. Command: sftp

**Description:** Secure FTP session for transferring files over SSH.

**Attributes/Options:**

- put: Upload file
- get: Download file

**Example:** sftp user@192.168.0.1

## Screenshot



A screenshot of a terminal window titled "davids@kali:~". The window shows the command "sftp user@192.168.0.1" being typed. The output of the command is displayed below the command line, showing an error message indicating that there is no route to host.

Figure 13: Output of sftp command

## 4. Command: find

**Description:** Searches for files in a directory hierarchy.

**Attributes/Options:**

- **-name:** Search by file name
- **-type:** File type (f: file, d: directory)

**Example:** `find /home -name "script.py"`

## Screenshot



Figure 14: Output of find command

## 5. Command: locate

**Description:** Quickly finds files using a pre-built database. Faster than find but may be outdated if database isn't updated.

### Attributes/Options:

- **-i:** Ignore case in file names
- **-n <limit>:** Show only a certain number of results

**Example:** `locate passwd`

## Screenshot



```
File Actions Edit View Help
/home/dream1/Desktop/[*]
└─$ locate "script.py"
/home/dream1/script.py
└─$ locate assesse[*]
└─$ locate assesse[*]db: No such file or directory
└─$
```

Figure 15: Output of locate command

## 6. Command: stat

**Description:** Displays detailed information about a file, such as size, access times, and permissions.

**Attributes/Options:**

- **-c:** Custom output format

**Example:** stat script.py

## Screenshot



```
File Actions Edit View Help
/home/dream1/Desktop/[*]
└─$ stat script.py
  File: script.py
  Size: 0        Blocks: 0          ID Block: 0x95   regular file
Device: 1600<-->    Inode: 130097      Uid: ( 1000/ dream1)
Access: 2025-07-07 18:23:56.105801117 +0500
Modify: 2025-07-07 18:23:56.105801117 +0500
Change: 2025-07-07 18:23:56.105801117 +0500
Birth: 2025-07-07 18:23:56.105801117 +0500
  
```

Figure 16: Output of stat command

## 7. Command: diff

**Description:** Compares two text files line by line and shows the differences.

**Attributes/Options:**

- **-u:** Unified output format
- **-i:** Ignore case

**Example:** diff script.py demo.txt

## Screenshot

```
File Actions Edit View Help
/home/dvordikov
└─$ diff script.py demo.txt
script.py demo.txt differ byte 1, line 1
└─$
```

Figure 17: Output of diff command

## 8. Command: cmp

**Description:** Compares two files byte by byte and reports the first difference.

**Attributes/Options:**

- **-l:** Show all differing bytes

**Example:** cmp script.py demo.txt

## Screenshot

```
File Actions Edit View Help
/home/dvordikov
└─$ cmp script.py demo.txt
script.py demo.txt differ byte 1, line 1
└─$
```

Figure 18: Output of cmp command

## 9. Command: xargs

**Description:** Constructs command lines from input, useful for chaining commands.

**Attributes/Options:**

- **-n <number>**: Use specified arguments per command line
- **-I ‘placeholder’**: Replace a string in command

**Example:** `cat files.txt | xargs rm`

## Screenshot



Figure 19: Output of xargs command

## 10. Command: basename

**Description:** Extracts the filename from a full path, removing directory and optionally suffix.

**Attributes/Options:**

- **<file> [suffix]**: Remove optional suffix

**Example:** `basename /home/user/file.txt`

## Screenshot



Figure 20: Output of basename command

# User & Permission Management

## Aim

To explore user and permission management commands in Kali Linux for securing and administering system access.

## Theory

User and permission management in Linux is critical for controlling access, modifying users, managing permissions, and ensuring secure operations. These commands help in system administration, access restriction, and forensic auditing.

## Flow Chart

- Step 1: Identify required user or permission operation
- Step 2: Apply appropriate command
- Step 3: Validate permissions, ownership, or user info

### 1. Command: sudo

**Description:** Executes commands with superuser (root) privileges.

**Attributes/Options:**

- **-u <user>**: Run command as specified user
- **-k**: Reset timestamp

**Example:** sudo apt update

## Screenshot



Figure 21: Output of sudo command

## 2. Command: chmod

**Description:** Changes file permissions (read, write, execute).

**Attributes/Options:**

- **+x**: Add execute permission
- **777**: Full permissions (rwxrwxrwx)

**Example:** chmod 755 script.sh

## Screenshot



A screenshot of a terminal window titled "dhrumil@kali:~". The window shows the following command and its output:

```
File Actions Edit View Help
dhrumil@kali:~$ chmod 755 script.sh
dhrumil@kali:~$ ./script.sh: ./script.sh: No such file or directory
dhrumil@kali:~$
```

Figure 22: Output of chmod command

### 3. Command: chown

**Description:** Changes ownership of files or directories.

**Attributes/Options:**

- user:group: Set owner and group
- -R: Recursive change

**Example:** chown dhrumil:dhrumil file.txt

## Screenshot



A screenshot of a terminal window titled "dhrumil@kali:~". The window shows the following command and its output:

```
File Actions Edit View Help
dhrumil@kali:~$ chown dhrumil:dhrumil demo.txt
dhrumil@kali:~$
```

Figure 23: Output of chown command

### 4. Command: usermod

**Description:** Modifies a user account's settings.

**Attributes/Options:**

- **-aG <group>**: Add user to group
- **-L**: Lock user account

**Example:** `usermod -aG sudo dhrumil`

## Screenshot

A screenshot of a terminal window titled "dhrumil@kali". The window shows the command \$ usermod -aG sudo dhrumil and its output: usermod: user 'dhrumil' does not exist. At the bottom, there is a faint watermark of the Kali Linux logo.

Figure 24: Output of usermod command

## 5. Command: passwd

**Description:** Updates or sets a user account password.

**Attributes/Options:**

- **-l**: Lock a user's password
- **-e**: Expire the user's password immediately

**Example:** `passwd dhrumil`

## Screenshot



Figure 25: Output of passwd command

## 6. Command: groups

**Description:** Shows all the groups a user belongs to.

**Attributes/Options:**

- <username>: Display specific user's groups

**Example:** groups dhrumil

## Screenshot



Figure 26: Output of groups command

## 7. Command: setfacl

**Description:** Sets file Access Control Lists (ACLs) for fine-grained permissions.

**Attributes/Options:**

- **-m u:user:rw file:** Give user read/write
- **-x u:user file:** Remove user access

**Example:** `setfacl -m u:john:rw file.txt`

## Screenshot



```
droot@kali:~$ setfacl -m u:john:rw dem.txt
setfacl: Option -m: invalid argument near character '
        ^
```

Figure 27: Output of setfacl command

## 8. Command: id

**Description:** Displays user ID (UID) and group ID (GID).

**Attributes/Options:**

- <username>: Show ID for specific user

**Example:** `id dhrumil`

## Screenshot



```
droot@kali:~$ id droot
uid=1000(droot) gid=1000(droot) groups=1000(droot),4(adm),20(datout),24(cdrom),25(floppy),27(sudo),29(audio),30(dip),46(crtde),100(users),101(netdev),102(sc
sadmin),103(kmem),104(tty),105(system),106(operator),107(gresvdev),108(hammer)
```

Figure 28: Output of id command

# Networking & Security

## Aim

To demonstrate various network and security-related commands in Kali Linux useful for penetration testing and system monitoring.

## Theory

Networking and security commands help in analyzing open ports, active connections, services, network packets, firewall configurations, and downloading resources. These tools are essential in ethical hacking, forensic analysis, and network security tasks.

## Flow Chart

- Step 1: Choose the type of network/service analysis needed
- Step 2: Use the appropriate command
- Step 3: Evaluate the results for action or investigation

### 1. Command: `netstat`

**Description:** Displays network connections, routing tables, and interface stats.

**Attributes/Options:**

- `-tuln`: Show TCP/UDP listening ports numerically

**Example:** `netstat -tuln`

## Screenshot



The screenshot shows a terminal window titled 'dvorak@kali: ~'. The command 'netstat -an' was run, displaying network connections. The output includes columns for Local Address, Foreign Address, and State. One connection is listed: 'tcp 0 0 ::ffff:4950/25ff:fe87:4c7a/5e8b:546 ::\*'. The background of the terminal window features a Kali Linux logo.

Figure 29: Output of netstat command

## 2. Command: ss

**Description:** Displays socket statistics; faster and more detailed than netstat.

**Attributes/Options:**

- **-tuln:** Show TCP/UDP listening sockets

**Example:** ss -tuln

## Screenshot



The screenshot shows a terminal window titled 'dvorak@kali: ~'. The command 'ss -tuln' was run, displaying listening sockets. The output includes columns for Local Address, State, Recv-Q, Send-Q, and Peer Address/Port. One socket is listed: 'tcp 0 0 ::ffff:4950/25ff:fe87:4c7a/5e8b:546'. The background of the terminal window features a Kali Linux logo.

Figure 30: Output of ss command

## 3. Command: nmap

**Description:** Scans network hosts and services (Network Mapper).

**Attributes/Options:**

- **-sS:** TCP SYN scan
- **-p <port>:** Scan specific port

**Example:** nmap -sS 192.168.1.1

## Screenshot



```
File Actions Edit View Help
dervit@kali: ~
$ nmap -sS 192.168.1.1
[...]
Nmap scan report for green.net (192.168.1.1)
Host is up (0.0000s latency).
Not shown: 995 closed ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh
23/tcp    filtered telnet
25/tcp    open  smtp
80/tcp    open  http
443/tcp   open  https
3128/tcp  open  http-proxy
Nmap done: 1 IP address (1 host up) scanned in 1.41 seconds
dervit@kali: ~
```

Figure 31: Output of nmap command

## 4. Command: tcpdump

**Description:** Captures and analyzes network packets in real-time.

**Attributes/Options:**

- **-i <interface>:** Specify interface to capture on

**Example:** tcpdump -i eth0

## Screenshot



```
File Actions Edit View Help
dervit@kali: ~
$ tcpdump -i eth0
tcpdump: eth0: You don't have permission to perform this capture on that device
tcpdump: eth0: (eth0 is not promiscuously permitted)
dervit@kali: ~
```

Figure 32: Output of tcpdump command

## 5. Command: iptables

**Description:** Manages IPv4 packet filtering rules (firewall).

**Attributes/Options:**

- -L: List current rules
- -A INPUT -p tcp --dport 22 -j ACCEPT: Allow SSH

**Example:** `iptables -L`

### Screenshot

A screenshot of a terminal window titled "File Actions Edit View Help". The window shows the command "iptables -L" being run, which results in the error message: "iptables v1.6.13 (nf\_tables): Could not fetch rule set generation id: permission denied (you must be root)". The background of the terminal window features a Kali Linux logo.

Figure 33: Output of iptables command

## 6. Command: ufw

**Description:** Uncomplicated Firewall - user-friendly wrapper for iptables.

**Attributes/Options:**

- enable|disable: Turn firewall on or off
- allow <port>: Allow specific port

**Example:** `ufw allow 22`

## Screenshot

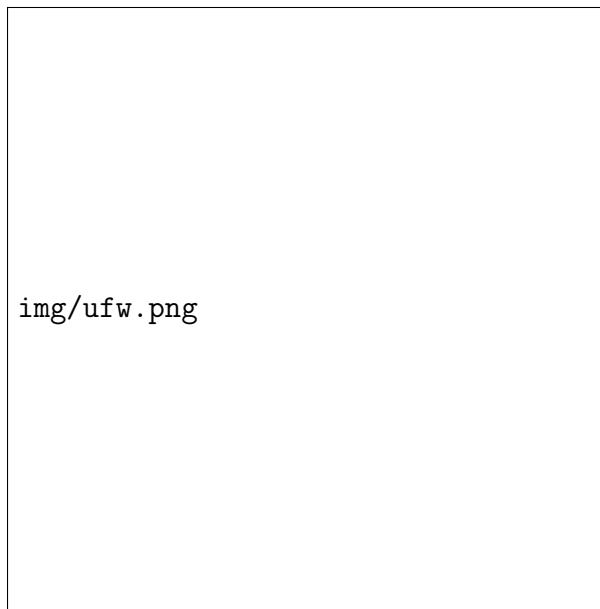


Figure 34: Output of ufw command

## 7. Command: fail2ban-client

**Description:** Bans IPs with too many failed login attempts.

**Attributes/Options:**

- **status:** Show service status
- **status sshd:** View banned IPs for SSH

**Example:** fail2ban-client status

## Screenshot



Figure 35: Output of fail2ban-client command

## 8. Command: curl

**Description:** Transfers data from or to a server using URLs.

**Attributes/Options:**

- **-O:** Save output to file
- **-I:** Show headers only

**Example:** curl -O https://example.com/file.txt

### Screenshot



The screenshot shows a terminal window titled 'curl@kali: ~' with the command 'curl -O https://example.com/file.txt' entered. The output is a large block of HTML code, which is the content of the file 'file.txt' from the specified URL. The HTML includes various meta-information, style definitions, and structural elements like headings and paragraphs.

Figure 36: Output of curl command

## 9. Command: wget

**Description:** Downloads files non-interactively via HTTP, HTTPS, FTP.

**Attributes/Options:**

- **-c:** Continue partially downloaded file

**Example:** wget https://example.com/file.txt

## Screenshot



The screenshot shows a terminal window titled 'File Actions Edit View Help' with the command 'wget' entered. The output of the command is displayed, showing the download of a file named 'file.txt' from 'http://example.com/file.txt'. The file contains the text 'Not Found'. The terminal window has a dark background with a Kali Linux logo watermark.

```
[drvenil@kali: ~] wget -O file.txt http://example.com/file.txt
--2015-07-07 23:19:18--  http://example.com/file.txt
Resolving example.com (example.com)... 96.7.138.175, 23.192.220.68
Connecting to example.com (example.com)|23.192.220.68|:443... connected.
HTTP request sent, awaiting response... 404 Not Found
2015-07-07 23:19:17 17 0K/0B [Err: Not Found]
```

Figure 37: Output of wget command

## 10. Command: dig

**Description:** Performs DNS lookups and queries.

**Attributes/Options:**

- **+short**: Shorter response
- **@dns-server**: Use specific DNS

**Example:** dig google.com +short

## Screenshot



The screenshot shows a terminal window titled 'File Actions Edit View Help' with the command 'dig' entered. The output of the command is displayed, showing a DNS lookup for the domain 'google.com' with the '+short' option. The result is '172.217.11.128'. The terminal window has a dark background with a Kali Linux logo watermark.

```
[drvenil@kali: ~] dig google.com +short
172.217.11.128
[drvenil@kali: ~]
```

Figure 38: Output of dig command

## Disk, Partition & System Info

## Aim

To explore disk usage, partitioning, and system memory commands in Kali Linux.

## Theory

Disk, partition, and system information commands allow users to monitor file system usage, manage partitions, mount drives, and inspect memory status. These tools are essential for system performance and forensic auditing.

## Flow Chart

- Step 1: Select type of system or disk analysis needed
- Step 2: Use appropriate command with options
- Step 3: Analyze space, partitions, or RAM usage

### 1. Command: df

**Description:** Displays file system disk space usage.

**Attributes/Options:**

- -h: Human-readable format

**Example:** df -h

## Screenshot



The screenshot shows a terminal window titled 'Terminal' with the command 'df -h' running. The output displays disk usage statistics for various file systems:

Filesystem	Size	Used	Avail	Use%	Mounted on
/	3938	1.3G	359M	3%	/
/dev/sda1	2.0G	4.0K	2.0G	0%	/boot
/dev/sda2	2.0G	8.0K	2.0G	0%	/var/www/html
/dev/sda3	2.0G	2.0G	0	100%	/var/www/html/elfie/elves
/dev/sda4	2.0G	8.0K	2.0G	0%	/var/log
/dev/sda5	2.0G	8.0K	2.0G	0%	/var/lib/systemd/journal/socket
/dev/sda6	2.0G	8.0K	2.0G	0%	/tmp
/dev/sda7	2.0G	152M	1.8G	8%	/var/crash
/dev/sda8	2.0G	8.0K	2.0G	0%	/run/credentials/getsbytys/service
/dev/sda9	2.0G	174K	1.9G	0%	/run/user/1000/service

Figure 39: Output of df command

## 2. Command: du

**Description:** Estimates file and directory space usage.

**Attributes/Options:**

- **-sh \***: Display total size of all files/directories in current directory

**Example:** du -sh \*

### Screenshot



The screenshot shows a terminal window titled 'du' with the command 'du -sh \*' entered. The output lists the total size of files and directories in the current directory. The terminal is running on a Kali Linux system, as indicated by the background wallpaper.

```
du -sh *
4.0G .
```

Figure 40: Output of du command

## 3. Command: mount

**Description:** Mounts a file system to a directory.

**Attributes/Options:**

- **<device> <dir>**: Mount device to directory

**Example:** mount /dev/sdb1 /mnt

## Screenshot



A screenshot of a terminal window titled "davinci@kali: ~". The window shows a single line of text: "lsblk: /mnt not mounted.". The background of the terminal is dark with a purple Kali Linux logo.

Figure 41: Output of mount command

## 4. Command: umount

**Description:** Unmounts a mounted file system.

**Attributes/Options:**

- <device|mount<sub>point</sub> >: Target device or directory **Example:** `umount /mnt`

## Screenshot



A screenshot of a terminal window titled "davinci@kali: ~". The window shows a single line of text: "umount: /mnt not mounted.". The background of the terminal is dark with a purple Kali Linux logo.

Figure 42: Output of umount command

## 5. Command: lsblk

**Description:** Lists information about block devices.

**Attributes/Options:**

- `-f`: Show file system info

**Example:** `lsblk -f`

## Screenshot



```
lsblk
NAME   FSTYPE  FSVER LABEL UUID                                     MOUNTPOINT
sda
└─sda2  vfat    0x16    7881f9c5  00000000-0000-0000-0000-000000000000  /boot/efi
sda3
└─sda5  ext4    1.0      19741547-0e7c-4bb2-a385-fc1ee5d1ee  /home
└─sda6  swap    1        c9579a00-3e41-4811-8714-992ba5485cd0  [SWAP]
```

Figure 43: Output of lsblk command

## 6. Command: blkid

**Description:** Displays block device attributes like UUID and type.

**Attributes/Options:**

- (no flags needed): Show all device information

**Example:** blkid

## Screenshot



```
blkid
/dev/sda1: UUID="19741547-0e7c-4bb2-a385-fc1ee5d1ee" TYPE="vfat" PARTUUID="00000000-0000-0000-0000-000000000000"
/dev/sda5: UUID="19279505-051c-4e48-9556-19fd4728ed4a" TYPE="ext4" PARTUUID="00000000-0000-0000-0000-000000000000"
/dev/sda6: UUID="94c0a337-4f12-488f-8506-9c972005144d" TYPE="swap" PARTUUID="00000000-0000-0000-0000-000000000000"
```

Figure 44: Output of blkid command

## 7. Command: fdisk

**Description:** A command-line utility to manage disk partitions.

**Attributes/Options:**

- -l: List all partitions

**Example:** fdisk -l

## Screenshot

```
[root@kali ~]# fdisk -l
fdisk: cannot open /dev/oda: Permission denied
[...]
Disk identifier: 242d0250-5C40-47A4-8F70-07070399C2
Device      Start  End Sectors  Size Type
/dev/vda1       2048  203731 201683 976M ext4  boot, esp
/dev/vda2     18816064 18816095  31 512B/512B filesystem
/dev/vda5     49704656 524295751 2737505  1.3G Linux swap
[...]
```

Figure 45: Output of fdisk command

## 8. Command: parted

**Description:** Advanced tool to create and manage partitions.

**Attributes/Options:**

- -l: Show partition layout

**Example:** parted -l

## Screenshot

```
[root@kali ~]# parted -l
Model: Virtio Block Device (virtblk)
Disk /dev/vda: 5120MB
Sector size (logical/physical): 512B/512B
Disk Flags:
Number  Start   End    Size   File system  Name Flags
1      2048  203731 191683  ext4          boot, esp
2      18816064 18816095   31  512B/512B
4      25.5GB  26.8GB 139MB  linux-swap(v1)  swap
[...]
```

Figure 46: Output of parted command

## 9. Command: iostat

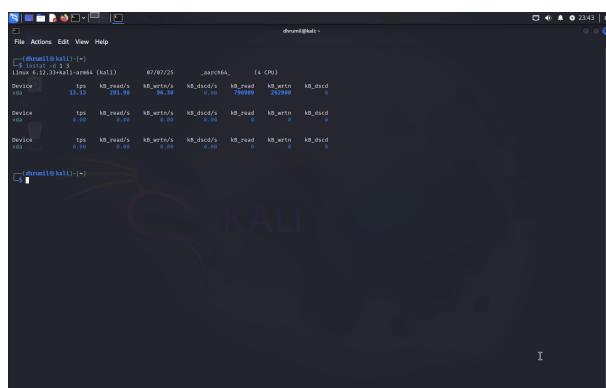
**Description:** Reports CPU and I/O statistics.

**Attributes/Options:**

- **-d:** Show only device stats

**Example:** `iostat -d 1 3`

## Screenshot



The screenshot shows a terminal window titled "dervit@kali:~" running on a Kali Linux system. The window displays the output of the iostat command with the parameters "-d 1 3". The output shows disk activity statistics over three iterations. The first iteration shows a single device (sda) with 0.00 KB read/s and 0.00 KB write/s. The second iteration shows two devices (sda and sdb) with similar activity. The third iteration shows three devices (sda, sdb, and sdc) with increased activity, particularly for sdb. The terminal window has a dark background with light-colored text and includes a watermark for "KALI" in the center.

```
[dervit@kali:~] iostat -d 1 3
Linux 4.12.30-kali1-amd64 (kali)        07/07/25      _arch64_      (4 CPU)
Device      tps   kB_read/s   kB_write/s   kB_disc/s   kB_read   kB_write   kB_disc
sda          0.00      0.00       0.00      0.00      0.00      0.00      0.00
sdb          0.00      0.00       0.00      0.00      0.00      0.00      0.00
sdc          0.00      0.00       0.00      0.00      0.00      0.00      0.00
[...]
[dervit@kali:~]
```

Figure 47: Output of iostat command

## 10. Command: free

**Description:** Displays system memory usage including RAM and swap.

**Attributes/Options:**

- **-h:** Human-readable format

**Example:** `free -h`

## Screenshot



Figure 48: Output of free command

# Miscellaneous Tools

## Aim

To demonstrate miscellaneous Linux commands used for task automation and command simplification.

## Theory

Miscellaneous commands like ‘alias’ and ‘cron’ enhance productivity by automating tasks and simplifying command execution. These tools are widely used in scripting, maintenance, and penetration testing setups.

## Flow Chart

- Step 1: Define automation or simplification goal
- Step 2: Create alias or schedule cron task
- Step 3: Verify functionality

### 1. Command: alias

**Description:** Creates a shortcut for another command. Useful to reduce typing or override default commands.

### Attributes/Options:

- `alias name='command'`: Define new alias
- `unalias <name>`: Remove alias

Example: `alias ll='ls -la'`

### Screenshot

A screenshot of a terminal window titled "dhrumil@kali: ~". The window shows the following text:

```
dhrumil@kali: ~$ alias ll='ls -la'
dhrumil@kali: ~$ ll
dhrumil@kali: ~$ ls
dhrumil@kali: ~$ ls
ls: alias: command not found
dhrumil@kali: ~$
```

The background of the desktop is visible, showing a Kali Linux logo.

Figure 49: Output of alias command

## 2. Command: cron/crontab

**Description:** Schedules tasks to run periodically at fixed times/dates using the cron daemon.

### Attributes/Options:

- `crontab -e`: Edit the crontab file
- `crontab -l`: List current cron jobs

Example: `0 5 * * * /home/dhrumil/backup.sh`

## Screenshot



Figure 50: Output of crontab command

## Conclusion

In this practical, we explored a wide range of Linux commands categorized into six core areas: system and process management, file and directory operations, user and permission management, networking and security, disk and system information, and miscellaneous tools.