

Name: Gitesh Bhavsar

Reg.No: 20BCR7083

File and Directory Operations:

1. **ls**: List files and directories Syntax: **ls [options] [directory]** Example: **ls -l** (lists files and directories in long format) **ls /home/user/Documents** (lists files and directories in the specified directory)

```
(kali@kali)-[~]
└─$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos

(kali@kali)-[~]
└─$
```

2. **cd**: Change directory Syntax: **cd [directory]** Example: **cd /home/user/Documents** (changes the current directory to "/home/user/Documents")

```
(kali@kali)-[~]
└─$ cd Desktop

(kali@kali)-[~/Desktop]
└─$
```


3. **pwd**: Print working directory Syntax: **pwd** Example: **pwd** (displays the current working directory)

```
(kali@kali)-[~/Desktop]
└─$ pwd
/home/kali/Desktop
```

4. **mkdir**: Make directory Syntax: **mkdir [options] directory_name** Example: **mkdir new_directory** (creates a directory named "new_directory" in the current directory)

```
└─$ pwd
/home/kali/Desktop

└─$ mkdir new_directory
```




5. **touch**: Create an empty file Syntax: **touch [options] file_name** Example: **touch new_file.txt** (creates an empty file named "new_file.txt" in the current directory)

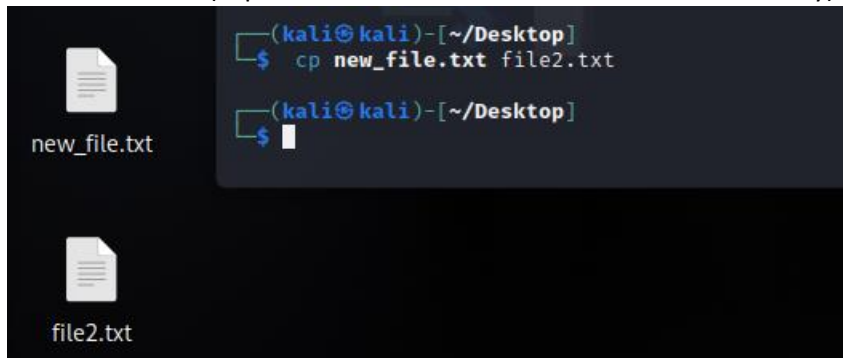
```
└─$ mkdir new_directory

(kali@kali)-[~/Desktop]
└─$ touch new_file.txt

(kali@kali)-[~/Desktop]
└─$
```

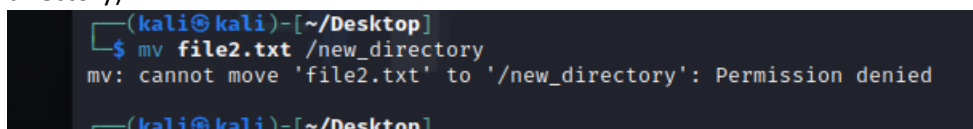


6. **cp**: Copy files and directories Syntax: **cp [options] source_file destination_file** Example: **cp file1.txt file2.txt** (copies "file1.txt" to "file2.txt" in the same directory)



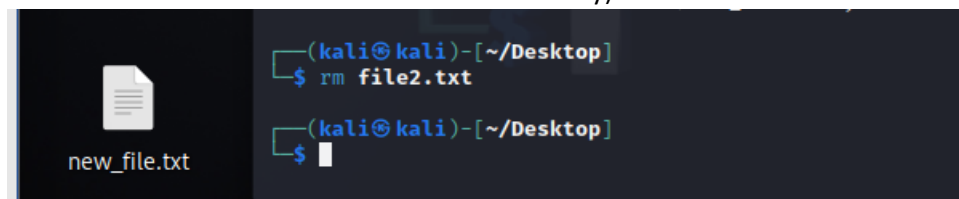
A terminal window with a dark background. On the left, there are two file icons labeled 'new_file.txt' and 'file2.txt'. The terminal shows the prompt '(kali@kali)-[~/Desktop]' followed by the command '\$ cp new_file.txt file2.txt'. The next prompt shows the user is still in the same directory.

7. **mv**: Move or rename files and directories Syntax: **mv [options] source_file destination** Example: **mv file1.txt /home/user/Documents/** (moves "file1.txt" to the "/home/user/Documents/" directory)



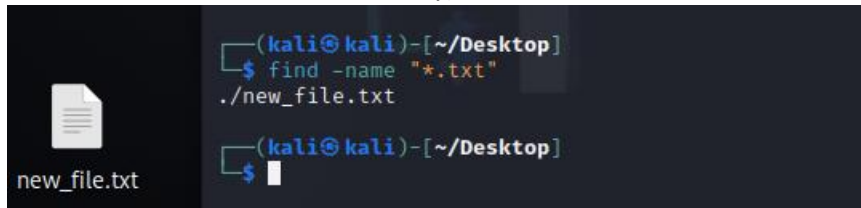
A terminal window with a dark background. The terminal shows the prompt '(kali@kali)-[~/Desktop]' followed by the command '\$ mv file2.txt /new_directory'. The output is 'mv: cannot move 'file2.txt' to '/new_directory': Permission denied'. The next prompt shows the user is still in the same directory.

8. **rm**: Remove files and directories Syntax: **rm [options] file_name** Example: **rm file.txt** (removes the file named "file.txt" from the current directory)



A terminal window with a dark background. On the left, there is a file icon labeled 'new_file.txt'. The terminal shows the prompt '(kali@kali)-[~/Desktop]' followed by the command '\$ rm file2.txt'. The next prompt shows the user is still in the same directory.

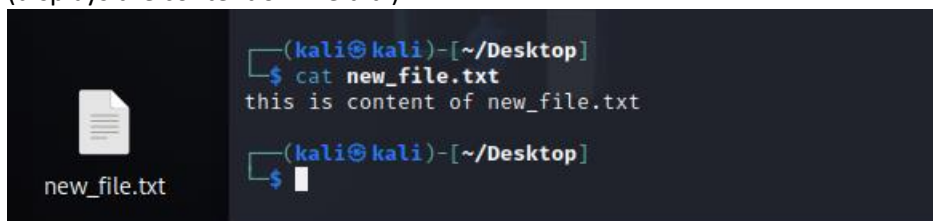
9. **find**: Search for files and directories Syntax: **find [path] [expression]** Example: **find /home/user/Documents/ -name "*.txt"** (finds all files with a .txt extension in the "/home/user/Documents/" directory)



A terminal window with a dark background. On the left, there is a file icon labeled 'new_file.txt'. The terminal shows the prompt '(kali@kali)-[~/Desktop]' followed by the command '\$ find -name "*.txt"'. The output is './new_file.txt'. The next prompt shows the user is still in the same directory.

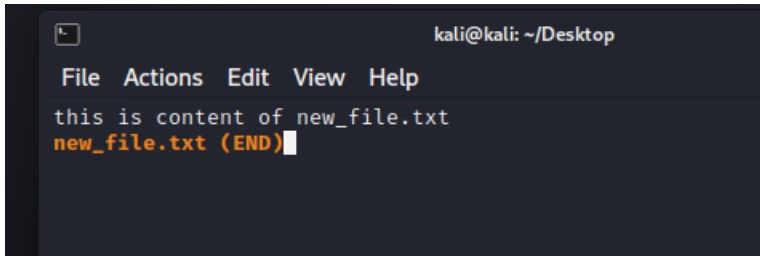
File Viewing and Editing:

1. **cat**: Concatenate and display file content Syntax: **cat [options] file_name** Example: **cat file.txt** (displays the content of "file.txt")



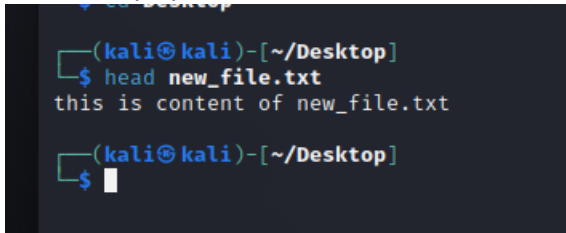
A terminal window with a dark background. On the left, there is a file icon labeled 'new_file.txt'. The terminal shows the prompt '(kali@kali)-[~/Desktop]' followed by the command '\$ cat new_file.txt'. The output is 'this is content of new_file.txt'. The next prompt shows the user is still in the same directory.

2. **less**: View file content with pagination Syntax: **less [options] file_name** Example: **less file.txt** (displays the content of "file.txt" with pagination)

A terminal window showing the output of the 'less' command. The window title is 'kali@kali: ~/Desktop'. The menu bar shows 'File Actions Edit View Help'. The content displays 'this is content of new_file.txt' followed by 'new_file.txt (END)' on the next line, with a cursor at the end of the second line.

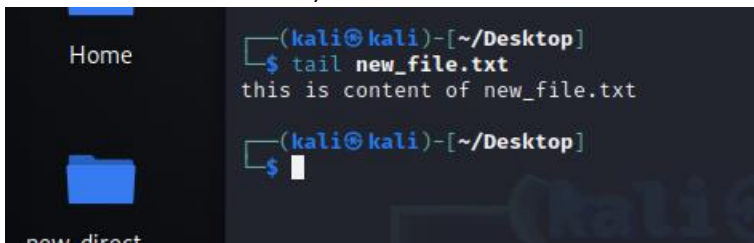
```
kali@kali: ~/Desktop
File Actions Edit View Help
this is content of new_file.txt
new_file.txt (END)
```

3. **head**: Display the beginning of a file Syntax: **head [options] file_name** Example: **head -n 10 file.txt** (displays the first 10 lines of "file.txt")

A terminal window showing the output of the 'head' command. The window title is '(kali@kali)-[~/Desktop]'. The prompt is '\$'. The command 'head new_file.txt' is entered, and the output is 'this is content of new_file.txt'. The prompt '\$' is shown again on the next line.

```
(kali@kali)-[~/Desktop]
$ head new_file.txt
this is content of new_file.txt
(kali@kali)-[~/Desktop]
$
```

4. **tail**: Display the end of a file Syntax: **tail [options] file_name** Example: **tail -n 5 file.txt** (displays the last 5 lines of "file.txt")

A terminal window showing the output of the 'tail' command. The window title is '(kali@kali)-[~/Desktop]'. The prompt is '\$'. The command 'tail new_file.txt' is entered, and the output is 'this is content of new_file.txt'. The prompt '\$' is shown again on the next line. On the left side of the terminal, there is a sidebar with a 'Home' button and a folder icon labeled 'new_direct'.

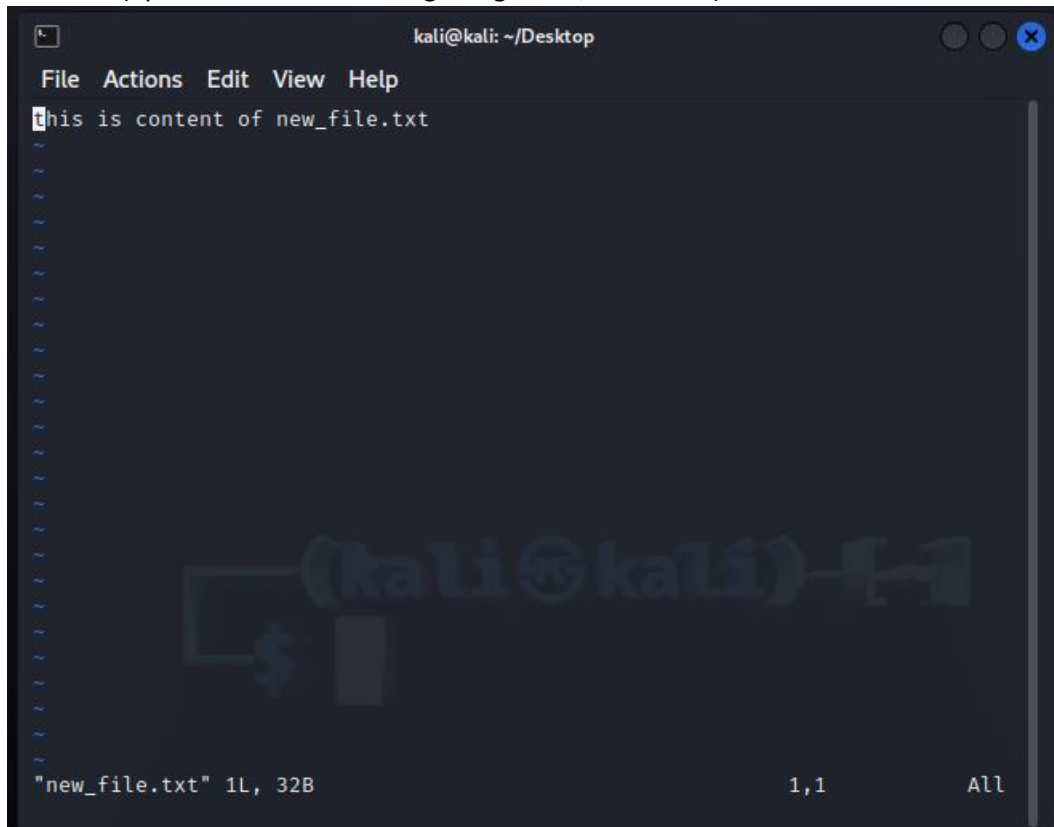
```
Home
(kali@kali)-[~/Desktop]
$ tail new_file.txt
this is content of new_file.txt
(kali@kali)-[~/Desktop]
$
new_direct
```

5. **nano**: Text editor for creating and editing files Syntax: **nano [options] file_name** Example: **nano new_file.txt** (opens "new_file.txt" for editing using the nano editor)

```
kali@kali: ~/Desktop
File Actions Edit View Help
GNU nano 7.2 new_file.txt
this is content of new_file.txt

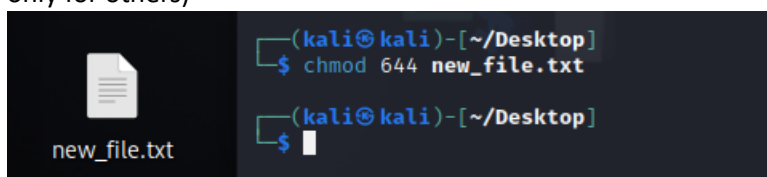
[ Read 1 line ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute
^X Exit      ^R Read File  ^^ Replace    ^U Paste      ^J Justify
```

6. **vi/vim**: Powerful text editor for experienced users Syntax: **vi/vim [options] file_name** Example: **vi file.txt** (opens "file.txt" for editing using the vi/vim editor)

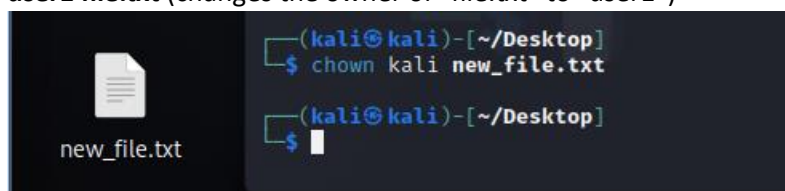


File Permissions:

1. **chmod**: Change file permissions Syntax: **chmod [options] permissions file_name** Example: **chmod 644 file.txt** (sets the file permissions of "file.txt" to read-write for the owner and read-only for others)



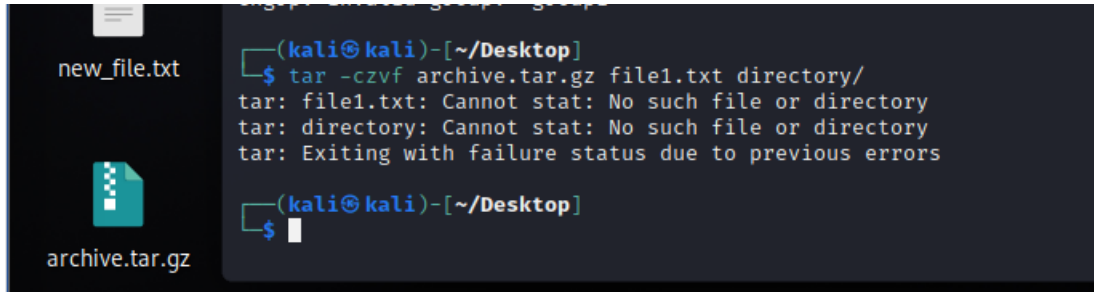
2. **chown**: Change file owner Syntax: **chown [options] new_owner file_name** Example: **chown user1 file.txt** (changes the owner of "file.txt" to "user1")



3. **chgrp**: Change file group Syntax: **chgrp [options] new_group file_name** Example: **chgrp group1 file.txt** (changes the group of "file.txt" to "group1")

File Compression and Archiving:

1. **tar**: Archive files Syntax: **tar [options] archive_file files/directories** Example: **tar -czvf archive.tar.gz file1.txt directory/** (creates a compressed tar archive named "archive.tar.gz" containing "file1.txt" and "directory/")

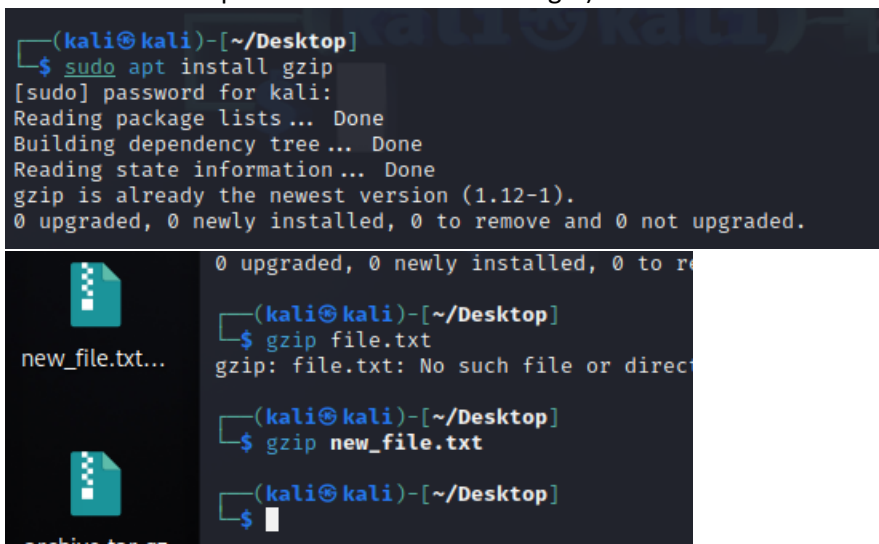


The terminal shows the execution of the `tar -czvf archive.tar.gz file1.txt directory/` command. The output indicates that the files `file1.txt` and `directory/` do not exist, resulting in a failure status. To the left of the terminal, there are icons for `new_file.txt` and the created `archive.tar.gz` file.

```
(kali㉿kali)-[~/Desktop]
$ tar -czvf archive.tar.gz file1.txt directory/
tar: file1.txt: Cannot stat: No such file or directory
tar: directory: Cannot stat: No such file or directory
tar: Exiting with failure status due to previous errors

(kali㉿kali)-[~/Desktop]
$
```

2. **gzip**: Compress files Syntax: **gzip [options] file_name** Example: **gzip file.txt** (compresses "file.txt" and creates a compressed file named "file.txt.gz")



The terminal shows the installation of `gzip` using `sudo apt install gzip`. It then shows attempts to use `gzip` on `file.txt` and `new_file.txt`, both of which result in "No such file or directory" errors. To the left, there are icons for `new_file.txt...` and `archive.tar.gz`.

```
(kali㉿kali)-[~/Desktop]
$ sudo apt install gzip
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
gzip is already the newest version (1.12-1).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.

(kali㉿kali)-[~/Desktop]
$ gzip file.txt
gzip: file.txt: No such file or directory

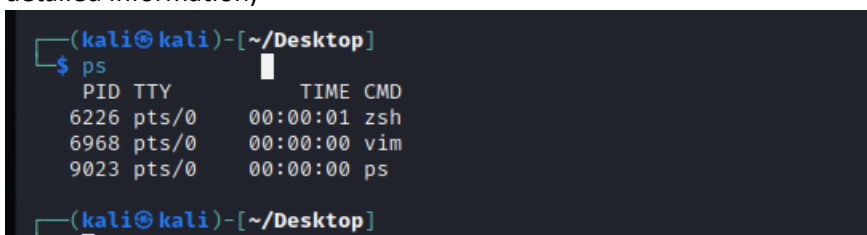
(kali㉿kali)-[~/Desktop]
$ gzip new_file.txt

(kali㉿kali)-[~/Desktop]
$
```

3. **unzip**: Extract files from a ZIP archive Syntax: **unzip [options] archive_file** Example: **unzip archive.zip** (extracts files from "archive.zip" into the current directory)

Process Management:

1. **ps**: List running processes Syntax: **ps [options]** Example: **ps aux** (lists all running processes with detailed information)

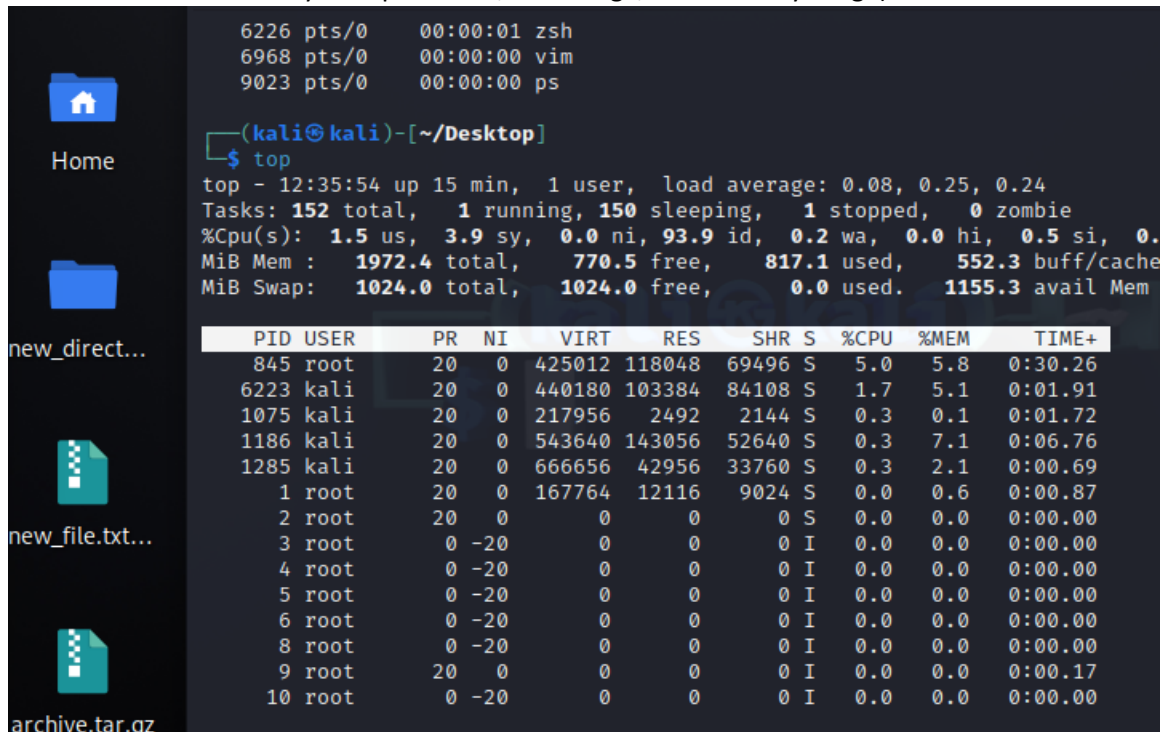


The terminal shows the output of the `ps` command, displaying a table of running processes with columns for PID, TTY, TIME, and CMD.

```
(kali㉿kali)-[~/Desktop]
$ ps
  PID TTY          TIME CMD
 6226 pts/0    00:00:01 zsh
 6968 pts/0    00:00:00 vim
 9023 pts/0    00:00:00 ps

(kali㉿kali)-[~/Desktop]
$
```

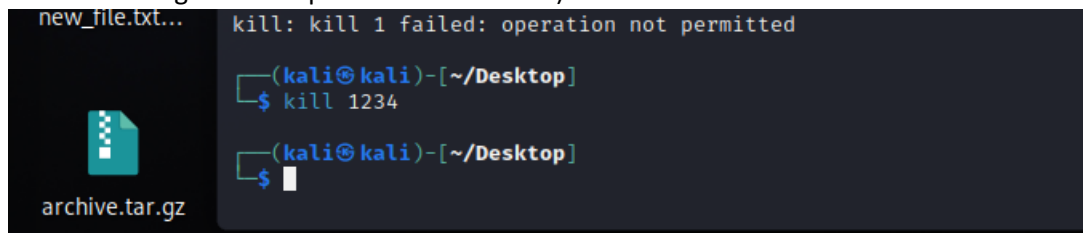
2. **top**: Display real-time system information and processes Syntax: **top** Example: **top** (displays real-time information about system processes, CPU usage, and memory usage)



```
(kali㉿kali)-[~/Desktop]
$ top
top - 12:35:54 up 15 min, 1 user, load average: 0.08, 0.25, 0.24
Tasks: 152 total, 1 running, 150 sleeping, 1 stopped, 0 zombie
%Cpu(s): 1.5 us, 3.9 sy, 0.0 ni, 93.9 id, 0.2 wa, 0.0 hi, 0.5 si, 0.
MiB Mem : 1972.4 total, 770.5 free, 817.1 used, 552.3 buff/cache
MiB Swap: 1024.0 total, 1024.0 free, 0.0 used. 1155.3 avail Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM     TIME+
    845 root        20   0  425012 118048 69496 S   5.0   5.8   0:30.26
    6223 kali       20   0  440180 103384 84108 S   1.7   5.1   0:01.91
    1075 kali       20   0   217956   2492   2144 S   0.3   0.1   0:01.72
    1186 kali       20   0   543640 143056 52640 S   0.3   7.1   0:06.76
    1285 kali       20   0   666656  42956  33760 S   0.3   2.1   0:00.69
      1 root        20   0   167764   12116   9024 S   0.0   0.6   0:00.87
      2 root        20   0         0         0         0 S   0.0   0.0   0:00.00
      3 root         0 -20         0         0         0 I   0.0   0.0   0:00.00
      4 root         0 -20         0         0         0 I   0.0   0.0   0:00.00
      5 root         0 -20         0         0         0 I   0.0   0.0   0:00.00
      6 root         0 -20         0         0         0 I   0.0   0.0   0:00.00
      8 root         0 -20         0         0         0 I   0.0   0.0   0:00.00
      9 root        20   0         0         0         0 I   0.0   0.0   0:00.17
     10 root         0 -20         0         0         0 I   0.0   0.0   0:00.00
```

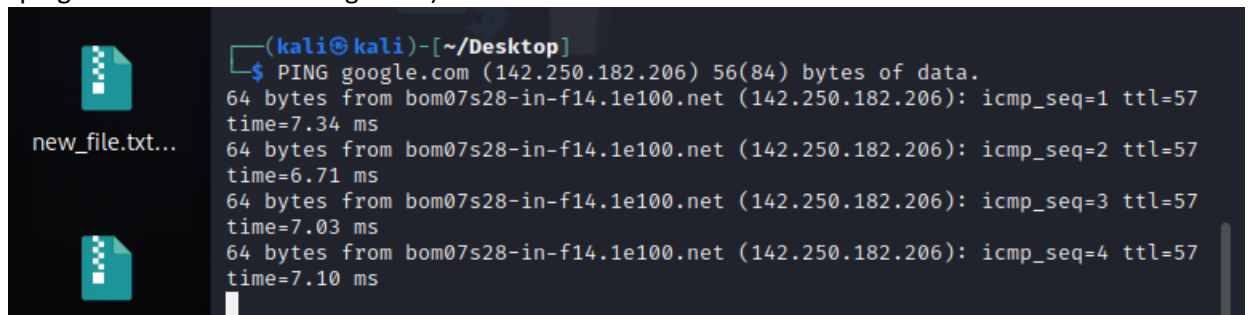
3. **kill**: Terminate processes Syntax: **kill [options] process_id** Example: **kill 1234** (sends a termination signal to the process with ID 1234)



```
(kali㉿kali)-[~/Desktop]
$ kill 1
kill: kill 1 failed: operation not permitted

(kali㉿kali)-[~/Desktop]
$ kill 1234
```

4. **bg**: Run processes in the background Syntax: **command &** Example: **ping google.com &** (runs the "ping" command in the background)



```
(kali㉿kali)-[~/Desktop]
$ PING google.com (142.250.182.206) 56(84) bytes of data.
64 bytes from bom07s28-in-f14.1e100.net (142.250.182.206): icmp_seq=1 ttl=57
time=7.34 ms
64 bytes from bom07s28-in-f14.1e100.net (142.250.182.206): icmp_seq=2 ttl=57
time=6.71 ms
64 bytes from bom07s28-in-f14.1e100.net (142.250.182.206): icmp_seq=3 ttl=57
time=7.03 ms
64 bytes from bom07s28-in-f14.1e100.net (142.250.182.206): icmp_seq=4 ttl=57
time=7.10 ms
```

5. **fg**: Bring background processes to the foreground Syntax: **fg %[job_id]** Example: **fg %1** (brings the background job with ID 1 to the foreground)

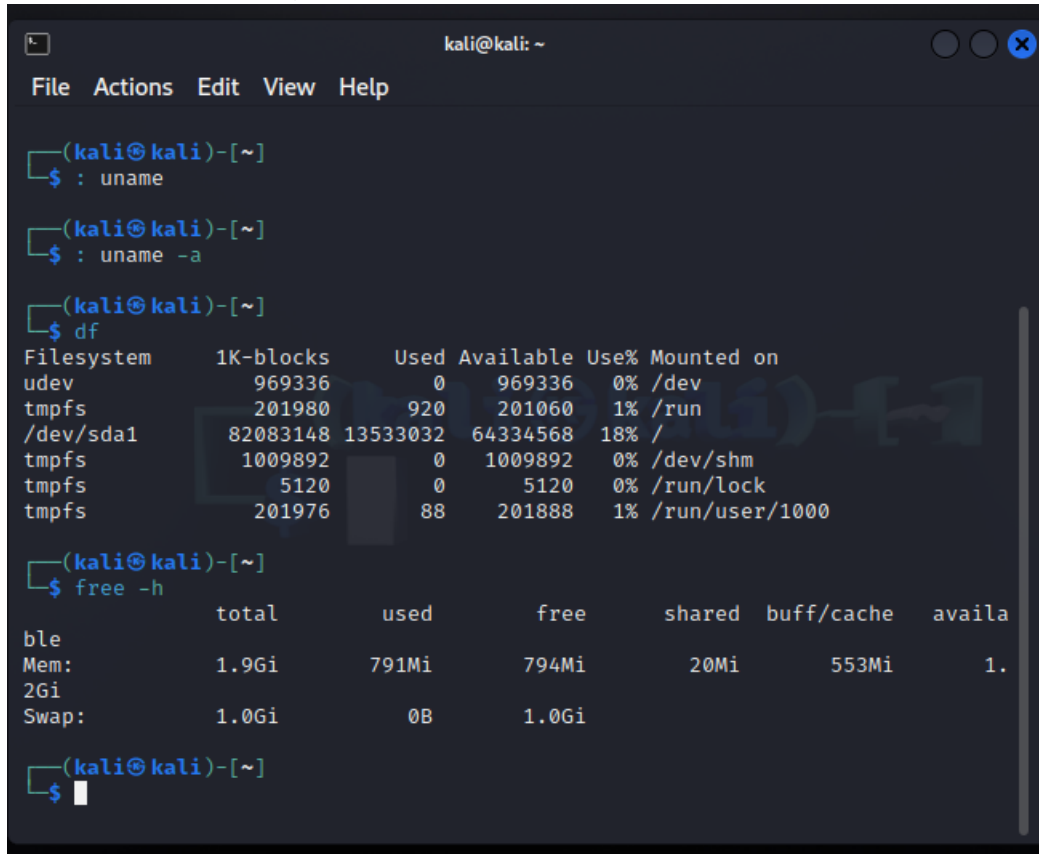
```
➔ fg %1
fg: %1: no such job

(kali@kali)-[~]
$ fg %112
fg: %112: no such job

(kali@kali)-[~]
```

System Information:

1. **uname**: Print system information Syntax: **uname [options]** Example: **uname -a** (displays all system information)
2. **df**: Display disk space usage Syntax: **df [options] [directory]** Example: **df -h** (displays disk space usage in a human-readable format)
3. **free**: Display memory usage Syntax: **free [options]** Example: **free -h** (displays memory usage in a human-readable format)



```
kali@kali: ~
File Actions Edit View Help

(kali@kali)-[~]
$ : uname

(kali@kali)-[~]
$ : uname -a

(kali@kali)-[~]
$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
udev             969336         0    969336   0% /dev
tmpfs            201980         920    201060   1% /run
/dev/sda1       82083148 13533032 64334568 18% /
tmpfs           1009892         0    1009892   0% /dev/shm
tmpfs             5120         0         5120   0% /run/lock
tmpfs           201976         88    201888   1% /run/user/1000

(kali@kali)-[~]
$ free -h
              total        used        free      shared  buff/cache   availa
ble
Mem:           1.9Gi       791Mi       794Mi         20Mi       553Mi       1.
2Gi
Swap:          1.0Gi          0B         1.0Gi

(kali@kali)-[~]
$
```

4. **uptime**: Show system uptime Syntax: **uptime** Example: **uptime** (displays the system uptime)
5. **who**: Display logged-in users Syntax: **who [options]** Example: **who** (displays information about currently logged-in users)

6. **w**: Display logged-in users and their activities Syntax: **w [options]** Example: **w** (displays a list of currently logged-in users and their activities)

```
(kali㉿kali)-[~]
$ uptime
12:39:07 up 18 min,  1 user,  load average: 0.11, 0.21, 0.23

(kali㉿kali)-[~]
$ who
kali    tty7      2023-05-28 12:20 (:0)

(kali㉿kali)-[~]
$ w
12:39:17 up 19 min,  1 user,  load average: 0.25, 0.23, 0.24
USER      TTY      FROM          LOGIN@   IDLE   JCPU   PCPU WHAT
kali      tty7      :0            12:20    18:59  35.04s  0.31s xfce4-sessio

(kali㉿kali)-[~]
$
```

Networking:

1. **ifconfig**: Configure network interfaces Syntax: **ifconfig [interface] [options]** Example: **ifconfig eth0** (displays the configuration of the "eth0" network interface)

2. **ping**: Send ICMP echo requests to a network host Syntax: **ping [options] host** Example: **ping google.com** (sends ICMP echo requests to the host "google.com" to check network connectivity)

```
kali tty7 :0 12:20 18:59 35.04s 0.31s xfce4-sessio

(kali@kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::879c:8be5:9681:8a4c prefixlen 64 scopeid 0<link>
    ether 08:00:27:c7:e1:36 txqueuelen 1000 (Ethernet)
    RX packets 153 bytes 35996 (35.1 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 185 bytes 17858 (17.4 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 4 bytes 240 (240.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 4 bytes 240 (240.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(kali@kali)-[~]
$ ping google.com
PING google.com (142.250.182.206) 56(84) bytes of data:
64 bytes from bom07s28-in-f14.1e100.net (142.250.182.206): icmp_seq=1 ttl=57
time=8.13 ms
64 bytes from bom07s28-in-f14.1e100.net (142.250.182.206): icmp_seq=2 ttl=57
time=6.02 ms
64 bytes from bom07s28-in-f14.1e100.net (142.250.182.206): icmp_seq=3 ttl=57
time=6.75 ms
^C
— google.com ping statistics —
3 packets transmitted, 3 received, 0% packet loss, time 2007ms
rtt min/avg/max/mdev = 6.020/6.967/8.128/0.873 ms
```

3. **ssh**: Securely connect to a remote system Syntax: **ssh [options] [user@]host** Example: **ssh user1@192.168.0.1** (establishes a secure SSH connection to the remote host with the IP address 192.168.0.1 using the username "user1")
4. **scp**: Securely copy files between systems Syntax: **scp [options] source_file destination** Example: **scp file.txt user1@192.168.0.1:/home/user1/** (copies "file.txt" from the local system to the remote system at "/home/user1/" using SSH)
5. **wget**: Download files from the web Syntax: **wget [options] URL** Example: **wget https://example.com/file.txt** (downloads the file "file.txt" from the specified URL to the current directory)

System Administration:

1. **sudo**: Execute commands with superuser privileges Syntax: **sudo command** Example: **sudo apt-get install package_name** (runs the "apt-get install" command with superuser privileges to

install a package)

```
(kali㉿kali)-[~]
$ sudo apt-get install package_name
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
E: Unable to locate package package_name
(kali㉿kali)-[~]
```

2. **apt-get**: Package management for Debian-based distributions Syntax: **apt-get [options] command package_name** Example: **apt-get update** (updates the package list on a Debian-based distribution)

```
(kali㉿kali)-[~]
$ apt-get update
Reading package lists... Done
E: List directory /var/lib/apt/lists/partial is missing. - Acquire (13: Permission denied)
W: Problem unlinking the file /var/cache/apt/pkgcache.bin - RemoveCaches (13: Permission denied)
W: Problem unlinking the file /var/cache/apt/srcpkgcache.bin - RemoveCaches (13: Permission denied)
```

3. **yum**: Package management for Red Hat-based distributions Syntax: **yum [options] command package_name** Example: **yum install package_name** (installs a package using the yum package manager)
4. **systemctl**: Manage system services Syntax: **systemctl [options] command service_name** Example: **systemctl start service_name** (starts a system service)
5. **crontab**: Schedule recurring tasks Syntax: **crontab [options]** Example: **crontab -e** (opens the crontab file for editing to schedule recurring tasks)

```
(kali㉿kali)-[~]
$ crontab -e
no crontab for kali - using an empty one

Select an editor. To change later, run 'select-editor'.
 1. /bin/nano          ← easiest
 2. /usr/bin/vim.basic
 3. /usr/bin/vim.tiny

Choose 1-3 [1]:
```

6. **useradd**: Add a new user Syntax: **useradd [options] username** Example: **useradd user1** (creates a new user with the username "user1")

```
(kali㉿kali)-[~]
$ useradd user1
useradd: Permission denied.
useradd: cannot lock /etc/passwd; try again later.
(kali㉿kali)-[~]
$
```

7. **passwd**: Change user password Syntax: **passwd [options] username** Example: **passwd user1** (changes the password for the user "user1")