

Task – 1

Commands of Kali Linux

Created By

Sanjay Sharma - 2065

1. pwd:

Description:

The `pwd` command displays the complete path of the current working directory. It helps users understand their present location in the Linux filesystem. This is useful when navigating complex directory structures.

Command:

```
(sanjay@Sanjay)-[~]  
$ pwd  
/home/sanjay
```

2. ls:

Description:

The `ls` command lists files and directories in the current location. It allows users to view available resources before performing operations. Different options can be used to display hidden or detailed file information.

Command:

```
(sanjay@Sanjay)-[~]  
$ ls  
Desktop Documents Downloads Music Pictures Public Templates Videos
```

3. cd:

Description:

The `cd` command is used to change the current working directory. It allows navigation between folders in the Linux filesystem. Proper use of this command is essential for file management.

Command:

```
(sanjay@Sanjay)-[~]  
$ cd Downloads
```

4. clear

Description:

The `clear` command removes all previous output from the terminal screen. It provides a clean workspace for executing new commands. This improves readability during long terminal sessions.

Command:

```
(sanjay@Sanjay)-[~/Downloads]  
$ clear
```

5. mkdir:

Description:

The `mkdir` command creates new directories in the filesystem. It helps organize files by grouping them into folders. Multiple directories can be created at once using options.

Command:

```
(sanjay@Sanjay)-[~/Downloads]  
$ mkdir Saikyo
```

6. tree:

Description:

The `tree` command displays directories in a hierarchical tree format. It visually represents folder structure and subdirectories. This is useful for understanding project layouts.

Command:

```
(sanjay@Sanjay)-[~/Downloads]  
$ tree  
.  
└─ Saikyo  
  
2 directories, 0 files
```

7. rmdir:

Description:

The `rmdir` command deletes empty directories. It prevents accidental deletion of directories containing files. This command ensures safe directory removal.

Command:

```
(sanjay@Sanjay)-[~/Downloads]
$ rmdir Saikyo

(sanjay@Sanjay)-[~/Downloads]
$ tree
.
0 directories, 0 files
```

8. touch:

Description:

The `touch` command creates a new empty file. It can also update timestamps of existing files. This command is often used to quickly create files.

Command:

```
(sanjay@Sanjay)-[~/Downloads]
$ touch CS CsGo
```

9. cp:

Description:

The `cp` command copies files or directories from one location to another. It is commonly used for backups and duplication. Options allow recursive copying and permission preservation.

Command:

```
(sanjay@Sanjay)-[~/Downloads]
$ cp CS CsGo
```

10. mv:

Description:

The `mv` command moves or renames files and directories. It helps reorganize the filesystem efficiently. The same command is used for both moving and renaming.

Command:

```
(sanjay@Sanjay)-[~/Downloads]  
$ mv CsGo CSGO
```

11. rm:

Description:

The `rm` command permanently deletes files or directories. It should be used carefully as deletion cannot be undone. Options allow recursive and forced deletion.

Command:

```
(sanjay@Sanjay)-[~/Downloads]  
$ rm CSGO
```

12. cat:

Description:

The `cat` command displays the contents of a file on the terminal. It is useful for viewing small text files. Multiple files can also be combined using this command.

Command:

```
(sanjay@Sanjay)-[~/Downloads]  
$ cat CS
```

13. more:

Description:

The `more` command shows file content one screen at a time. It helps users read large files without scrolling overflow. Navigation is limited to forward movement.

Command:

```
(sanjay@Sanjay)-[~/Downloads]  
$ more CS
```

14. less:

Description:

The `less` command provides an advanced file viewer. It allows forward and backward scrolling with search functionality. This command is preferred for large files.

Command:

```
(sanjay@Sanjay)-[~/Downloads]  
$ less CS
```

15. ls -la:

Description:

The `ls -la` command displays all files including hidden ones with details. It shows permissions, ownership, size, and modification time. This command is helpful for auditing files and directories.

Command:

```
(sanjay@Sanjay)-[~/Downloads]  
$ ls -la  
total 12  
drwxr-xr-x  3 sanjay sanjay 4096 Dec 29 18:58 .  
drwx----- 14 sanjay sanjay 4096 Dec 29 18:57 ..  
-rw-rw-r--  1 sanjay sanjay   0 Dec 29 18:57 CS  
drwxrwxr-x  2 sanjay sanjay 4096 Dec 29 18:48 Saikyo
```

16. uname -a:

Description:

The `uname -a` command displays complete system information. It includes kernel version, system name, and architecture. This command helps identify system configuration.

Command:

```
(sanjay@Sanjay) - [~/Downloads]
$ uname -a
Linux Sanjay 6.16.8+kali-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.16.8-1kali1 (2025-09-24) x86_64 GNU/Linux
```

17. whoami:

Description:

The `whoami` command displays the currently logged-in username. It is useful in multi-user environments. This command helps confirm user identity.

Command:

```
(sanjay@Sanjay) - [~/Downloads]
$ whoami
sanjay
```

18. id:

Description:

The `id` command displays user and group identification details. It shows UID, GID, and associated groups. This is helpful for permission and access management.

Command:

```
(sanjay@Sanjay) - [~/Downloads]
$ id sanjay
uid=1000(sanjay) gid=1000(sanjay) groups=1000(sanjay),24(cdrom),25(floppy),27(sudo),29(audio),30(dip),44(video),46(plugindev),100(users),101(netdev),103(scanner),116(blueetooth),119(lpadmin)
```

19. top:

Description:

The `top` command displays real-time system processes. It shows CPU, memory usage, and running tasks. This command helps monitor system performance.

Command:

```
(sanjay@Sanjay)~[~/Downloads]
$ top

top - 19:17:09 up 39 min, 1 user, load average: 0.10, 0.19, 0.18
Tasks: 227 total, 1 running, 224 sleeping, 0 stopped, 2 zombie
%Cpu(s):  4.1 us,  2.5 sy,  0.0 ni, 93.1 id,  0.0 wa,  0.0 hi,  0.3 si,  0.0 st
MiB Mem :  3127.3 total,   994.2 free,  1521.9 used,   849.1 buff/cache
MiB Swap:  2652.0 total,  2652.0 free,    0.0 used.  1605.4 avail Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM     TIME+ COMMAND
 2662 sanjay    20   0  530916 12392  7224 S   5.3   0.4   0:55.26 speech-dispatch
 1891 sanjay    20   0 4150612 556568 171316 S   5.0  17.4   2:37.39 gnome-shell
 1754 sanjay    9  -11  106076 15156   8948 S   0.7   0.5   0:15.16 pipewire
 1801 sanjay    9  -11  181972 21492   8792 S   0.7   0.7   0:07.42 pipewire-pulse
 3146 sanjay    20   0  569708 66228  53568 S   0.7   2.1   0:06.92 gnome-terminal-
  163 root       20   0     0     0     0 I   0.3   0.0   0:02.80 kworker/u512:4-events_unbound
  555 root       20   0  113732  9860   8280 S   0.3   0.3   0:03.38 vmtoolsd
 2004 sanjay    20   0  579656 105860  87936 S   0.3   3.3   0:03.51 vmtoolsd
 2678 sanjay    20   0  24528  9428   5680 S   0.3   0.3   0:01.07 sd_espeak-ng
14929 sanjay    20   0 1213592 84868  41000 S   0.3   2.7   0:05.49 orca
    1 root       20   0  24912 15088  10916 S   0.0   0.5   0:31.05 systemd
    2 root       20   0     0     0     0 S   0.0   0.0   0:00.02 kthreadd
```

20. htop:

Description:

The `htop` command is an enhanced version of `top`. It provides a color-coded and interactive interface. This makes process monitoring easier.

Command:

```
(sanjay@Sanjay)~[~/Downloads]
$ htop
```

21. ps aux:

Description:

The `ps aux` command lists all running processes. It provides detailed process information. This command is useful for troubleshooting.

Command:

```
(sanjay@Sanjay)~[~/Downloads]
$ ps aux

USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.4  24912 15088 ?        Ss   18:37   0:01 /sbin/init splash
root         2  0.0  0.0     0     0 ?        S    18:37   0:00 [kthreadd]
root         3  0.0  0.0     0     0 ?        S    18:37   0:00 [pool_workqueue_release]
root         4  0.0  0.0     0     0 ?        I<   18:37   0:00 [kworker/R-rcu_gp]
root         5  0.0  0.0     0     0 ?        I<   18:37   0:00 [kworker/R-sync_wq]
root         6  0.0  0.0     0     0 ?        I<   18:37   0:00 [kworker/R-kvfree_rcu_reclaim]
root         7  0.0  0.0     0     0 ?        I<   18:37   0:00 [kworker/R-slub_flushwq]
root         8  0.0  0.0     0     0 ?        I<   18:37   0:00 [kworker/R-netns]
root        13  0.0  0.0     0     0 ?        I<   18:37   0:00 [kworker/R-mm_percpu_wq]
```


22. kill:

Description:

The `kill` command terminates a process using its PID. It helps stop unresponsive or unwanted processes. Proper usage prevents system instability.

Command:

```
(sanjay@Sanjay)-[~/Downloads]  
$ kill 1
```

23. killall:

Description:

The `killall` command terminates processes by name. It stops all instances of a given process. This simplifies process management.

Command:

```
(sanjay@Sanjay)-[~/Downloads]  
$ killall root
```

24. apt update:

Description:

The `apt update` command refreshes the package list. It retrieves the latest package information. This is required before installing updates.

Command:

```
(sanjay@Sanjay)-[~]  
$ sudo apt update  
[sudo] password for sanjay:  
Notice: It seems that you don't have any APT data sources configured.  
Notice: You won't be able to update your system or install new packages.  
Notice: For more information, please refer to the online documentation at:  
Notice: https://www.kali.org/docs/general-use/kali-linux-sources-list-repositories/  
All packages are up to date.
```

25. apt upgrade:

Description:

The `apt upgrade` command updates installed packages. It installs newer versions of existing software. This helps maintain system security.

Command:

```
(sanjay@Sanjay)-[~]
$ sudo apt upgrade
The following packages were automatically installed and are no longer required:
binutils-mingw-w64-i686 gcc-mingw-w64-i686-win32 libkdb5-10t64 ndiff
binutils-mingw-w64-x86_64 gcc-mingw-w64-i686-win32-runtime libkdb5-dev python3-asciitree python3-lsassy python3-pyfiglet python3-tld unicornscan
bloodhound.py gcc-mingw-w64-x86_64-win32 liblinear4 python3-asn1tools python3-masky python3-pylnk3 python3-unicrypto urlscan
comerr-dev gcc-mingw-w64-x86_64-win32-runtime liblua5.1-2 python3-asyauth python3-minidump python3-pymisc python3-wapiti-arsenic wapiti
dnsmasq imagemagick python3-aardwolf python3-asysocks python3-minkbereros python3-pyppkatz python3-winacl
dniff imagemagick-7.q16 python3-aesedb python3-bitstruct python3-malmap python3-pyshodan python3-xmldict
ettercap-common krb5-multidev python3-alicad python3-browser-cookie3 python3-neo4j python3-pysnmp python3-yaswp
ettercap-graphical libaio1t64 python3-aiosmb python3-dploit python3-neotime python3-qasync rsh-redone-client
figlet libgssrpc4t64 python3-aiosmb python3-git python3-neotime python3-qrcode python3-user-enun
finger libkadm5clnt-mit12 python3-aiosmb python3-github python3-oscrypto python3-qrcode python3-user-enun
gcc-mingw-w64-base libkadm5srv-mit12 python3-aiosmb python3-gitdb python3-pefile python3-serial-asyncio
Use 'sudo apt autoremove' to remove them.

Summary:
Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 0
```

26. apt install:

Description:

The `apt install` command installs new software packages. It downloads packages from official repositories. Dependencies are handled automatically.

Command:

```
(sanjay@Sanjay)-[~]
$ sudo apt install nmap
```

27. apt remove:

Description:

The `apt remove` command uninstalls software packages. It removes the program but keeps configuration files. This helps free system space.

Command:

```
(sanjay@Sanjay)-[~]
$ sudo apt remove nmap
```

```
(sanjay@Sanjay)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.205.128 netmask 255.255.255.0 broadcast 192.168.205.255
    inet6 fe80::20c:29ff:fe8b:d916 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:8b:d9:16 txqueuelen 1000 (Ethernet)
    RX packets 1875 bytes 1888574 (1.8 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 455 bytes 34168 (33.3 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 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 15636 bytes 938120 (916.1 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 15636 bytes 938120 (916.1 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

30. ip a:

Description:

The `ip a` command shows detailed IP configuration. It is a modern replacement for `ifconfig`. This command provides advanced network information.

Command:

```
(sanjay@Sanjay)-[~]
$ ip a
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: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:8b:d9:16 brd ff:ff:ff:ff:ff:ff
    inet 192.168.205.128/24 brd 192.168.205.255 scope global dynamic noprefixroute eth0
        valid_lft 1752sec preferred_lft 1752sec
    inet6 fe80::20c:29ff:fe8b:d916/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

31. ping:

Description:

The `ping` command tests network connectivity. It sends packets to a target host. This helps verify network availability.

Command:

```
(sanjay@Sanjay)-[~]
$ ping google.com
PING google.com (142.251.221.238) 56(84) bytes of data:
64 bytes from pnbomb-bk-in-f14.1e100.net (142.251.221.238): icmp_seq=1 ttl=128 time=6.04 ms
64 bytes from pnbomb-bk-in-f14.1e100.net (142.251.221.238): icmp_seq=2 ttl=128 time=4.61 ms
64 bytes from pnbomb-bk-in-f14.1e100.net (142.251.221.238): icmp_seq=3 ttl=128 time=3.29 ms
64 bytes from pnbomb-bk-in-f14.1e100.net (142.251.221.238): icmp_seq=4 ttl=128 time=15.4 ms
64 bytes from pnbomb-bk-in-f14.1e100.net (142.251.221.238): icmp_seq=5 ttl=128 time=12.7 ms
64 bytes from pnbomb-bk-in-f14.1e100.net (142.251.221.238): icmp_seq=6 ttl=128 time=5.71 ms
64 bytes from pnbomb-bk-in-f14.1e100.net (142.251.221.238): icmp_seq=7 ttl=128 time=3.59 ms
64 bytes from pnbomb-bk-in-f14.1e100.net (142.251.221.238): icmp_seq=8 ttl=128 time=3.50 ms
64 bytes from pnbomb-bk-in-f14.1e100.net (142.251.221.238): icmp_seq=9 ttl=128 time=3.41 ms
64 bytes from pnbomb-bk-in-f14.1e100.net (142.251.221.238): icmp_seq=10 ttl=128 time=4.27 ms
64 bytes from pnbomb-bk-in-f14.1e100.net (142.251.221.238): icmp_seq=11 ttl=128 time=3.92 ms
64 bytes from pnbomb-bk-in-f14.1e100.net (142.251.221.238): icmp_seq=12 ttl=128 time=3.18 ms
```

32. netstat -tuln:

Description:

The `netstat -tuln` command displays active ports. It shows listening services and connections. This helps identify open network services.

Command:

```
(sanjay@Sanjay)-[~]
$ netstat -tuln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp6      0      0 :::1716                 :::*                    LISTEN
udp6      0      0 :::1716                 :::*                    LISTEN
```

33. ss -tuln:

Description:

The `ss -tuln` command shows socket statistics. It is faster and more efficient than `netstat`. This command helps analyze network connections.

Command:

```
(sanjay@Sanjay)-[~]
$ ss -tuln
Netid      State      Recv-Q     Send-Q     Local Address:Port      Peer Address:Port
udp        UNCONN     0           0             *:1716                  *:*
tcp        LISTEN     0           50           *:1716                  *:*
```

34. nmcli:

Description:

The `nmcli` command manages network connections. It interacts with NetworkManager services. This command is useful for configuring networks.

Command:

```
(sanjay@Sanjay)-[~]
$ nmcli dev status
DEVICE  TYPE      STATE      CONNECTION
eth0    ethernet  connected  Wired connection 1
lo      loopback  connected (externally)  lo
```

35. wget:

Description:

The `wget` command downloads files from the internet. It supports HTTP, HTTPS, and FTP protocols. This command is commonly used for file retrieval.

Command:

```
(sanjay@Sanjay)-[~]
$ wget https://example.com/file.zip
```

36. curl:

Description:

The `curl` command transfers data from URLs. It supports multiple protocols and options. This command is useful for API testing.

Command:

```
(sanjay@Sanjay)-[~]  
$ curl https://example.com
```

37. grep:

Description:

The `grep` command searches text patterns in files. It helps locate specific content quickly. This command supports regular expressions.

Command:

```
(sanjay@Sanjay)-[~]  
$ grep "error" log.txt
```

38. find:

Description:

The `find` command searches files in directories. It uses different conditions like name and size. This command is powerful for file discovery.

Command:

```
(sanjay@Sanjay)-[~]  
$ find / -name demo.txt
```

39. locate:

Description:

The `locate` command finds files quickly. It uses a prebuilt database for searching. Results may not include recent files.

Command:

```
(sanjay@Sanjay)-[~]  
$ locate demo.txt
```

40. which:

Description:

The `which` command shows the path of a command. It helps identify executable locations. This ensures correct command execution.

Command:

```
(sanjay@Sanjay)-[~]  
$ which nmap
```

41. chmod:

Description:

The `chmod` command modifies file permissions. It controls read, write, and execute access. This command is critical for security.

Command:

```
(sanjay@Sanjay)-[~]  
$ chmod 755 script.sh
```

42. chown:

Description:

The `chown` command changes file ownership. It assigns new user and group ownership. This helps manage access rights.

Command:

```
(sanjay@Sanjay)-[~]  
$ sudo chown kali:kali demo.txt
```


43. ln -s:

Description:

The `ln -s` command creates symbolic links. It acts as a shortcut to files or directories. This helps in file organization.

Command:

```
(sanjay@Sanjay)-[~]  
$ ln -s demo.txt shortcut.txt
```

44. echo:

Description:

The `echo` command prints text to the terminal. It is commonly used in scripts. This command displays variables and messages.

Command:

```
(sanjay@Sanjay)-[~]  
$ echo "Hello Kali Linux"  
Hello Kali Linux
```

45. wc:

Description:

The `wc` command counts file content. It shows line, word, and character counts. This is useful for text analysis.

Command:

```
(sanjay@Sanjay)-[~]  
$ wc -l demo.txt
```

46. sort:

Description:

The `sort` command arranges file content. It sorts lines alphabetically or numerically. This helps organize data.

Command:

```
(sanjay@Sanjay)-[~]  
$ sort names.txt
```

47. diff:

Description:

The `diff` command compares two files. It highlights differences line by line. This is useful for version comparison.

Command:

```
(sanjay@Sanjay)-[~]  
$ diff file1.txt file2.txt
```

48. sed:

Description:

The `sed` command edits text streams. It performs search and replace operations. This command is powerful for automation.

Command:

```
(sanjay@Sanjay)-[~]  
$ sed 's/Linux/Kali/' demo.txt
```

49. nmap:

Description:

The `nmap` command scans networks and hosts. It identifies open ports and services. This command is widely used in penetration testing.

Command:

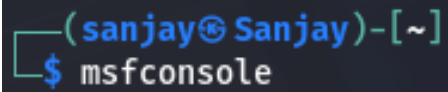
```
(sanjay@Sanjay)-[~]  
$ nmap -sV 192.168.1.1
```

50. msfconsole:

Description:

The `msfconsole` command launches Metasploit Framework. It provides tools for exploitation and testing. This command is essential for ethical hacking.

Command:

A terminal window with a dark background. The prompt is `(sanjay@Sanjay)-[~]` in green. Below it, the command `$ msfconsole` is entered, with the dollar sign in blue and the text in white.

```
(sanjay@Sanjay)-[~]  
$ msfconsole
```