Basic Linux Commands

S M Abu Rafi Sadat **ID: 170042082** **Date: 5th April, 2021**

The report will follow this format to describe briefly what each command does:

"Command -> what it does"
"Here will be a screenshot supporting above description."
**Anything inside a third bracket will be considered optional.

1. pwd -> Prints the current path or directory.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ pwd
/home/sadat_11
```

2. ls -> Prints the files and directories existing in the current path except the hidden ones.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5  arr.sh  basic.sh  boolopr.sh  case.sh  case2.sh  ch0.
txt  ch1.txt  food.txt  hello.sh  loop.sh  opr.sh  sampl
e  sample.c  stropr.sh
```

3. ls -a -> Prints the files and directories existing in the current path including the hidden ones.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ ls -a
.                         .viminfo    food.txt
..                        5           hello.sh
.array.sh.swp             arr.sh      loop.sh
.bash_history             basic.sh    opr.sh
.bash_logout              boolopr.sh  sample
.bashrc                   case.sh     sample.c
.profile                  case2.sh    stropr.sh
.sudo_as_admin_successful ch0.txt
.test.sh.swp              ch1.txt
```

4. cd <directory path> -> Take you to the given path <directory path>

```
sadat_11@LAPTOP-4Q7A9CT8:~$ cd sample_dir/inner_dir
sadat_11@LAPTOP-4Q7A9CT8:~/sample_dir/inner_dir$
```

5. cd .. -> Takes you back to the previous directory.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ cd sample_dir
sadat_11@LAPTOP-4Q7A9CT8:~/sample_dir$ cd ..
sadat_11@LAPTOP-4Q7A9CT8:~$
```

6. cd -> Takes you back to the home directory from wherever you are.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ cd sample_dir/inner_dir
sadat_11@LAPTOP-4Q7A9CT8:~/sample_dir/inner_dir$ cd
sadat_11@LAPTOP-4Q7A9CT8:~$
```

7. mkdir <directory name> -> Creates directory according to the given name <directory name>.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5          boolopr.sh  ch0.txt   hello.sh  sample
arr.sh     case.sh     ch1.txt   loop.sh   sample.c
basic.sh   case2.sh    food.txt  opr.sh    stropr.sh
sadat_11@LAPTOP-4Q7A9CT8:~$ mkdir sample_dir
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5           case.sh   food.txt  sample
arr.sh      case2.sh  hello.sh  sample.c
basic.sh    ch0.txt   loop.sh   sample_dir
boolopr.sh  ch1.txt   opr.sh    stropr.sh
```

8. mkdir -p <directory path> Force create path and then create directory if a path is not already there.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ mkdir -p  sample_dir/inner_d
ir
sadat_11@LAPTOP-4Q7A9CT8:~$ cd sample_dir/inner_dir
sadat_11@LAPTOP-4Q7A9CT8:~/sample_dir/inner_dir$ cd
```

9. [rmdir or rm -d] <directory name> -> Removes an empty directory.

```
sadat_11@LAPTOP-4Q7A9CT8:~/sample_dir$ ls
innerdir
sadat_11@LAPTOP-4Q7A9CT8:~/sample_dir$ rmdir innerdir
sadat_11@LAPTOP-4Q7A9CT8:~/sample_dir$ ls
sadat_11@LAPTOP-4Q7A9CT8:~/sample_dir$
```

10. rm <file name -> removes files.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5           case.sh   food.txt  sample
arr.sh      case2.sh  hello.sh  sample.c
basic.sh    ch0.txt   loop.sh   sample_dir
boolopr.sh  ch1.txt   opr.sh    stropr.sh
sadat_11@LAPTOP-4Q7A9CT8:~$ rm ch1.txt
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5         boolopr.sh  ch0.txt   loop.sh  sample.c
arr.sh    case.sh     food.txt  opr.sh   sample_dir
basic.sh  case2.sh    hello.sh  sample   stropr.sh
sadat_11@LAPTOP-4Q7A9CT8:~$
```

11. touch <file name> -> Creates a file.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5         boolopr.sh  ch0.txt   loop.sh  sample.c
arr.sh    case.sh     food.txt  opr.sh   sample_dir
basic.sh  case2.sh    hello.sh  sample   stropr.sh
sadat_11@LAPTOP-4Q7A9CT8:~$ touch ch1.txt
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5           case.sh   food.txt  sample
arr.sh      case2.sh  hello.sh  sample.c
basic.sh    ch0.txt   loop.sh   sample_dir
boolopr.sh  ch1.txt   opr.sh    stropr.sh
sadat_11@LAPTOP-4Q7A9CT8:~$
```

12. man <command> or <command> --help -> Will provide you with information regarding the command. Here's an interesting example when use one in another.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ man --help
Usage: man [OPTION...] [SECTION] PAGE...

  -C, --config-file=FILE     use this user configuration file
  -d, --debug                emit debugging messages
  -D, --default              reset all options to their default values
      --warnings[=WARNINGS]  enable warnings from groff

 Main modes of operation:
  -f, --whatis               equivalent to whatis
  -k, --apropos              equivalent to apropos
  -K, --global-apropos       search for text in all pages
  -l, --local-file           interpret PAGE argument(s) as local filename(s)
  -w, --where, --path, --location
                             print physical location of man page(s)
  -W, --where-cat, --location-cat
                             print physical location of cat file(s)

  -c, --catman               used by catman to reformat out of date cat pages
  -R, --recode=ENCODING      output source page encoded in ENCODING

 Finding manual pages:
  -L, --locale=LOCALE        define the locale for this particular man search
  -m, --systems=SYSTEM       use manual pages from other systems
  -M, --manpath=PATH         set search path for manual pages to PATH

  -S, -s, --sections=LIST    use colon separated section list

  -e, --extension=EXTENSION  limit search to extension type EXTENSION

  -i, --ignore-case          look for pages case-insensitively (default)
  -I, --match-case           look for pages case-sensitively

      --regex                show all pages matching regex
      --wildcard             show all pages matching wildcard

      --names-only           make --regex and --wildcard match page names only,
                             not descriptions
```

13. cp <original file> <copy file> -> Copies <original file> to <copy file>. If the <copy file> does not exist it creates one and then copies into it.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ cat ch0.txt
a b c d
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5  arr.sh  basic.sh  boolopr.sh  case.sh  case2.sh  ch0.
txt  food.txt  hello.sh  loop.sh  opr.sh  sample  sample
.c  sample_dir  stropr.sh
sadat_11@LAPTOP-4Q7A9CT8:~$ cp ch0.txt ch1.txt
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5  arr.sh  basic.sh  boolopr.sh  case.sh  case2.sh  ch0.
txt  ch1.txt  food.txt  hello.sh  loop.sh  opr.sh  sampl
e  sample.c  sample_dir  stropr.sh
sadat_11@LAPTOP-4Q7A9CT8:~$ cat ch1.txt
a b c d
```

14. mv <old> <new> -> Renames the <old> directory or file to <new>.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5  arr.sh  basic.sh  boolopr.sh  case.sh  case2.sh  ch0.txt  ch1.txt  food.txt  hello.sh  ju
st_dir  loop.sh  opr.sh  sample  sample.c  stropr.sh
sadat_11@LAPTOP-4Q7A9CT8:~$ mv just_dir sample_dir
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5       basic.sh    case.sh   ch0.txt  food.txt  loop.sh  sample    sample_dir
arr.sh  boolopr.sh  case2.sh  ch1.txt  hello.sh  opr.sh   sample.c  stropr.sh
sadat_11@LAPTOP-4Q7A9CT8:~$
```

15. cat <file name> Shows the content of the <file name>.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ cat ch1.txt
a b c d
```

16. echo <something> -> prints <something> into the console or writes in a file and adds a new line after <something>.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ cat ch0.txt
a b c d
sadat_11@LAPTOP-4Q7A9CT8:~$ echo e f g h >> ch0.txt
sadat_11@LAPTOP-4Q7A9CT8:~$ cat ch0.txt
a b c d
e f g h
sadat_11@LAPTOP-4Q7A9CT8:~$ echo i j k l
i j k l
sadat_11@LAPTOP-4Q7A9CT8:~$
```

17. vi <file name> -> open <file name> in a vim editor.

pressing ‘i’ will enable the INSERT mode.

press ‘ESC’ to get out of the INSERT mode and then type in ‘:wq’ to save the file and exit the editor respectively. You can also do that in separately, like- ‘:w’ and then ‘:q’.

18. sudo <something> -> ‘sudo’ is used to get the root access or privilege to accompish the task that a normal user cannot. But it might require root-user's password.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ sudo bash
[sudo] password for sadat_11:
root@LAPTOP-4Q7A9CT8:~# _
```

19. df -> it shows you the available disc space of the filesystem(in KB).

```
root@LAPTOP-4Q7A9CT8:~# df
Filesystem     1K-blocks      Used Available Use% Mounted on
rootfs         160004092 133910964  26093128  84% /
none           160004092 133910964  26093128  84% /dev
none           160004092 133910964  26093128  84% /run
none           160004092 133910964  26093128  84% /run/lock
none           160004092 133910964  26093128  84% /run/shm
none           160004092 133910964  26093128  84% /run/user
tmpfs          160004092 133910964  26093128  84% /sys/fs/cgroup
C:\            160004092 133910964  26093128  84% /mnt/c
D:\             59768828  50078104   9690724  84% /mnt/d
E:\            472063996 361041776 111022220  77% /mnt/e
F:\            210156540 194791588  15364952  93% /mnt/f
G:\             52222972  30682644  21540328  59% /mnt/g
root@LAPTOP-4Q7A9CT8:~# _
```

‘-m’ will cause the result to be in megabytes.

20. du [<filesystem>] -> It shows disc usage of the filesystem and can take an argument to show the disc usage of a directory in the system.

```
root@LAPTOP-4Q7A9CT8:~# du
0       ./sample_dir
32      .
root@LAPTOP-4Q7A9CT8:~# du sample_dir
0       sample_dir
```

21. ls –lah -> shows the usage of disc by all the listed files and folders.

```
root@LAPTOP-4Q7A9CT8:~# ls -lah
total 32K
drwxr-xr-x 1 sadat_11 sadat_11 4.0K May  4 19:08 .
drwxr-xr-x 1 root     root     4.0K Jan 22  2020 ..
-rw-r--r-- 1 sadat_11 sadat_11  12K Jul  8  2020 .array.sh.swp
-rw------- 1 sadat_11 sadat_11  11K May  4 01:33 .bash_history
-rw-r--r-- 1 sadat_11 sadat_11  220 Jan 22  2020 .bash_logout
-rw-r--r-- 1 sadat_11 sadat_11 3.7K Jan 22  2020 .bashrc
-rw-r--r-- 1 sadat_11 sadat_11  807 Jan 22  2020 .profile
-rw-r--r-- 1 sadat_11 sadat_11    0 Jan 22  2020 .sudo_as_admin_successful
-rw-r--r-- 1 sadat_11 sadat_11  12K Feb  5  2020 .test.sh.swp
-rw------- 1 sadat_11 sadat_11 9.9K May  4 19:08 .viminfo
-rw-rw-rw- 1 sadat_11 sadat_11    0 Jul  9  2020 5
-rwxrwxrwx 1 sadat_11 sadat_11  119 Jul  9  2020 arr.sh
-rwxrwxrwx 1 sadat_11 sadat_11   40 Jul  9  2020 basic.sh
-rwxrwxrwx 1 sadat_11 sadat_11  311 May  4 13:28 boolopr.sh
-rwxrwxrwx 1 sadat_11 sadat_11  180 May  4 13:20 case.sh
-rwxrwxrwx 1 sadat_11 sadat_11  133 Jul 10  2020 case2.sh
-rw-r--r-- 1 sadat_11 sadat_11   24 May  4 19:08 ch0.txt
-rw-r--r-- 1 sadat_11 sadat_11    8 May  4 18:50 ch1.txt
-rw-rw-rw- 1 sadat_11 sadat_11  115 Jul  8  2020 food.txt
-rwxrwxrwx 1 sadat_11 sadat_11   73 May  4 13:14 hello.sh
-rwxrwxrwx 1 sadat_11 sadat_11  183 Jul 10  2020 loop.sh
-rwxrwxrwx 1 sadat_11 sadat_11  518 Jul  8  2020 opr.sh
-rwxrwxrwx 1 sadat_11 sadat_11 8.3K Jul  8  2020 sample
-rw-rw-rw- 1 sadat_11 sadat_11  107 Jul  8  2020 sample.c
drwxr-xr-x 1 sadat_11 sadat_11 4.0K May  4 18:30 sample_dir
-rwxrwxrwx 1 sadat_11 sadat_11  208 Jul  9  2020 stropr.sh
root@LAPTOP-4Q7A9CT8:~#
```

22. tar <options> <archive-file> [file or directory to be archived] -> compress according to the given option and names it <archive-file>. Stores that in the mentioned directory if given. Options like ‘–cvf’ is used to compress into and ‘-xvf’ is to extract from a given file <archive-file>. You can use regex expression like ‘*.txt’ to compress all txt file of that directory. If no directory is explicitly mentioned current directory is implicitly understood by the terminal.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ tar -cvf file.tar *.txt
ch0.txt
ch1.txt
food.txt
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5       basic.sh    case.sh   ch0.txt  file.tar  hello.sh  opr.sh  sample.c    stropr.sh
arr.sh  boolopr.sh  case2.sh  ch1.txt  food.txt  loop.sh   sample  sample_dir
sadat_11@LAPTOP-4Q7A9CT8:~$
```

```
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5       basic.sh    case.sh   file.tar  loop.sh  sample    sample_dir
arr.sh  boolopr.sh  case2.sh  hello.sh  opr.sh   sample.c  stropr.sh
sadat_11@LAPTOP-4Q7A9CT8:~$ tar -xvf file.tar
ch0.txt
ch1.txt
food.txt
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5       basic.sh    case.sh   ch0.txt   file.tar  hello.sh  opr.sh  sample.c    stropr.sh
arr.sh  boolopr.sh  case2.sh  ch1.txt   food.txt  loop.sh   sample  sample_dir
sadat_11@LAPTOP-4Q7A9CT8:~$
```

23. sudo apt-get [option] <package> -> installs or upgrade <package> according to instruction.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ sudo apt-get install zip
[sudo] password for sadat_11:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libdumbnet1 libfreetype6
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  unzip
The following NEW packages will be installed:
  unzip zip
0 upgraded, 2 newly installed, 0 to remove and 3 not upgraded.
Need to get 334 kB of archives.
After this operation, 1196 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

24. zip or unzip <file> -> zips or unzips <file>.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ zip file.zip *.txt
  adding: ch0.txt (stored 0%)
  adding: ch1.txt (stored 0%)
  adding: food.txt (deflated 10%)
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5       basic.sh    case.sh   ch0.txt   file.tar  food.txt  loop.sh  sample    sample_dir
arr.sh  boolopr.sh  case2.sh  ch1.txt   file.zip  hello.sh  opr.sh   sample.c  stropr.sh
sadat_11@LAPTOP-4Q7A9CT8:~$
```

```
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5       basic.sh    case.sh   file.tar  hello.sh  opr.sh   sample.c    stropr.sh
arr.sh  boolopr.sh  case2.sh  file.zip  loop.sh   sample   sample_dir
sadat_11@LAPTOP-4Q7A9CT8:~$ unzip file.zip
Archive:  file.zip
 extracting: ch0.txt
 extracting: ch1.txt
  inflating: food.txt
sadat_11@LAPTOP-4Q7A9CT8:~$ ls
5       basic.sh    case.sh   ch0.txt   file.tar  food.txt  loop.sh  sample    sample_dir
arr.sh  boolopr.sh  case2.sh  ch1.txt   file.zip  hello.sh  opr.sh   sample.c  stropr.sh
sadat_11@LAPTOP-4Q7A9CT8:~$
```

25. uname [option] -> shows distro information.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ uname -a
Linux LAPTOP-4Q7A9CT8 4.4.0-19041-Microsoft #488-Microsoft Mon Sep 01 13:43:00 PST 2020 x86
_64 x86_64 x86_64 GNU/Linux
sadat_11@LAPTOP-4Q7A9CT8:~$
```

26. hostname [option] -> shows hostname and ip address.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ hostname
LAPTOP-4Q7A9CT8
sadat_11@LAPTOP-4Q7A9CT8:~$ hostname -I
192.168.0.101
```

27. ping <server> -> check connection with server <server>. Press ‘Ctrl’ + ‘c’ to stop pinging and get statistics.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ ping google.com
PING google.com (142.250.195.206) 56(84) bytes of data.
64 bytes from maa03s42-in-f14.1e100.net (142.250.195.206): icmp_seq=1 ttl=115 time=33.2 ms
64 bytes from maa03s42-in-f14.1e100.net (142.250.195.206): icmp_seq=2 ttl=115 time=32.8 ms
64 bytes from maa03s42-in-f14.1e100.net (142.250.195.206): icmp_seq=3 ttl=115 time=33.3 ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2000ms
rtt min/avg/max/mdev = 32.832/33.128/33.344/0.262 ms
```

28. chmod <permission> <filename> -> It enables a user to change a file’s permissions.

for –x make it executable, -w makes it editable, -r read makes it readable. u,o and g are used to represent user owner and group’s access to that file.

```
sadat_11@LAPTOP-4Q7A9CT8:~$ ls -l sample.c
-rw-rw-rw- 1 sadat_11 sadat_11 107 Jul  8  2020 sample.c
sadat_11@LAPTOP-4Q7A9CT8:~$ chmod u=w sample.c
sadat_11@LAPTOP-4Q7A9CT8:~$ ls -l sample.c
--w-rw-rw- 1 sadat_11 sadat_11 107 Jul  8  2020 sample.c
sadat_11@LAPTOP-4Q7A9CT8:~$
```

here user’s permission changed to only writable.

We can, thus, use the above commands to perform basic linux terminal operations.