

Module 1 Assessment

Q1) Use the appropriate command to list all files larger than 1 MB in the current directory and save the output to a file.

Method 1: Using ls with -S flag and print with AWK

```
cmd: ls -l | awk -v dir=$PWD 'BEGIN {print "Printing the files greater than 1 MB in" dir } $5 > 1048000 {print $NF} END {print "The Filesgreater than 1 MB has been printed"}'
```

Output:

```
mysterychap@mysteryChap:~/embedUR-linux-practical/linux-assessment/module-1$ ls -l
total 1044
-rwxr-xr-x 1 root      root      1044931 Jan 29 04:48 module1.pdf
-rw-r--r-- 1 mysterychap mysterychap 644 Jan 29 04:43 q1.txt
-rw-r--r-- 1 mysterychap mysterychap 627 Jan 29 04:41 q2.txt
-rw-r--r-- 1 mysterychap mysterychap 1581 Jan 29 04:40 q3.txt
-rw-r--r-- 1 mysterychap mysterychap 1311 Jan 29 04:33 q4.txt
-rw-r--r-- 1 mysterychap mysterychap 1824 Jan 29 04:39 q5.txt
mysterychap@mysteryChap:~/embedUR-linux-practical/linux-assessment/module-1$ ls -l | awk -v dir=$PWD 'BEGIN {print "Printing the files greater than 1 MB in" dir } $5 > 104800 {print $NF} END {print "The Filesgreater than 1 MB has been printed"}'
Printing the files greater than 1 MB in /home/mysterychap/embedUR-linux-practical/linux-assessment/module-1
module1.pdf
The Filesgreater than 1 MB has been printed
mysterychap@mysteryChap:~/embedUR-linux-practical/linux-assessment/module-1$
```

Method 2: Using find command with exec flag

```
cmd: find . -type f -size +1M -exec ls -l {} +
```

Output:

```
mysterychap@mysteryChap:/proc$ sudo find . -type f -size +1M -exec ls -l {} +
find: './42263/task/42263/fdinfo/6': No such file or directory
find: './42263/fdinfo/5': No such file or directory
-r----- 1 root root 140737471594496 Jan 31 23:42 ./kcore
mysterychap@mysteryChap:/proc$
```

Q2) Replace all occurrences of "localhost" with "127.0.0.1" in a configuration file named config.txt, and save the updated file as updated_config.txt.

```
cmd: sed 's/localhost/127.0.0.1/g' module2/config.txt > module2/updated_config.txt
```

Output:

```
mysterychap@mysteryChap:~/embedUR-linux-practical/linux-assessment$ cat > module2/config.txt
This is the file with text Tocalhost.
I need to change all the localhost to 127.0.0.1
mysterychap@mysteryChap:~/embedUR-linux-practical/linux-assessment$ sed 's/localhost/127.0.0.1/g' module2/config.txt > module2/updated_config.txt
mysterychap@mysteryChap:~/embedUR-linux-practical/linux-assessment$ cat module2/updated_config.txt
This is the file with text 127.0.0.1.
I need to change all the 127.0.0.1 to 127.0.0.1
mysterychap@mysteryChap:~/embedUR-linux-practical/linux-assessment$
```

Q3) Use the appropriate command to search for lines containing the word "ERROR" in a log file but exclude lines containing "DEBUG". Save the results to a file named filtered_log.txt.

cmd: grep ERROR module2/log.txt | grep -v DEBUG > module2/filtered_log.txt

Output:

```
mysterychap@MysteryChap:~/embedUR-linux-practical/linux-assessment$ cat module2/log.txt
DEBUG: Starting the application initialization.
ERROR: Unable to connect to the database.
INFO: User 'admin' logged in successfully.
DEBUG: Fetching configuration settings from the server.
ERROR: DEBUG - Configuration settings could not be applied.
INFO: Scheduled job 'backup' completed successfully.
DEBUG: Connection to server timed out. Retrying...
ERROR: Failed to fetch data from API endpoint '/users'.
INFO: Maintenance mode activated.
ERROR: DEBUG - Query execution failed due to a syntax error.
DEBUG: Reloading application modules.
ERROR: Missing required parameter in the request.
INFO: Shutting down the system gracefully.
DEBUG: Closing unused network connections.
mysterychap@MysteryChap:~/embedUR-linux-practical/linux-assessment$ grep ERROR module2/log.txt | grep -v DEBUG > module2/filtered_log.txt
mysterychap@MysteryChap:~/embedUR-linux-practical/linux-assessment$ cat module2/filtered_log.txt
ERROR: Unable to connect to the database.
ERROR: Failed to fetch data from API endpoint '/users'.
ERROR: Missing required parameter in the request.
```

Q4) Write a code to identify the process with the highest memory usage and then terminate it.

cmd: sudo kill -SIGTERM \$(ps -eo pid --sort=-rss | sed -n '2p')

Observation: The process that consumed more memory was snapd which is actually a daemon. So it restarted again as soon as i terminated it.

Initial pid 260 -> After Kill command pid -> 22440

Output:

```
mysterychap@MysteryChap:~/embedUR-linux-practical/linux-assessment$ ps -aux --sort=-rss | head
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root      260  0.3  0.5 2514912 42676 ?
root       63  0.2  0.4 75004 36352 ?
root      305  0.0  0.2 112796 23808 ?
attended-upgrade-shutdown --wait-for-signal
root      267  0.0  0.1 68720 14976 ?
root      266  0.0  0.1 68716 13696 ?
root       1  0.5  0.1 22112 12896 ?
systemd+  221  0.0  0.1 21456 12672 ?
mystery+  586  0.0  0.1 20192 10856 ?
root     142  0.1  0.1 526756 10856 ?
ap/snapd/25935 -o ro,nodev,allow_other,suid
mysterychap@MysteryChap:~/embedUR-linux-practical/linux-assessment$ ps -eo pid --sort=-rss | head
PID
260
63
305
267
266
1
221
21784
586
mysterychap@MysteryChap:~/embedUR-linux-practical/linux-assessment$ ps -eo pid --sort=-rss | sed -n '2p'
260
```

Q5) Use the networking tool command and print all the gateway available in a sorted manner

cmd: netstat -r | awk '{print \$2}' | sed -n '3,\$p' | sort

Output:

```
mysterychap@MysteryChap:~$ netstat -r
Kernel IP routing table
Destination     Gateway         Genmask        Flags   MSS Window irtt Iface
default         MysteryChap.msh 0.0.0.0      UG        0 0          0 eth0
172.18.176.0    0.0.0.0        255.255.240.0  U         0 0          0 eth0
192.168.122.0   0.0.0.0        255.255.255.0  U         0 0          0 virbr0
mysterychap@MysteryChap:~$ netstat -rn | awk '{print $2}' | sort
IP
Gateway
172.18.176.1
0.0.0.0
0.0.0.0
mysterychap@MysteryChap:~$ netstat -r| awk '{print $2}' | sed -n '3,$p' | sort
0.0.0.0
0.0.0.0
MysteryChap.msh
mysterychap@MysteryChap:~$
```