VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

SMART BRIDGE – Ethical Hacking

Assessment - 2

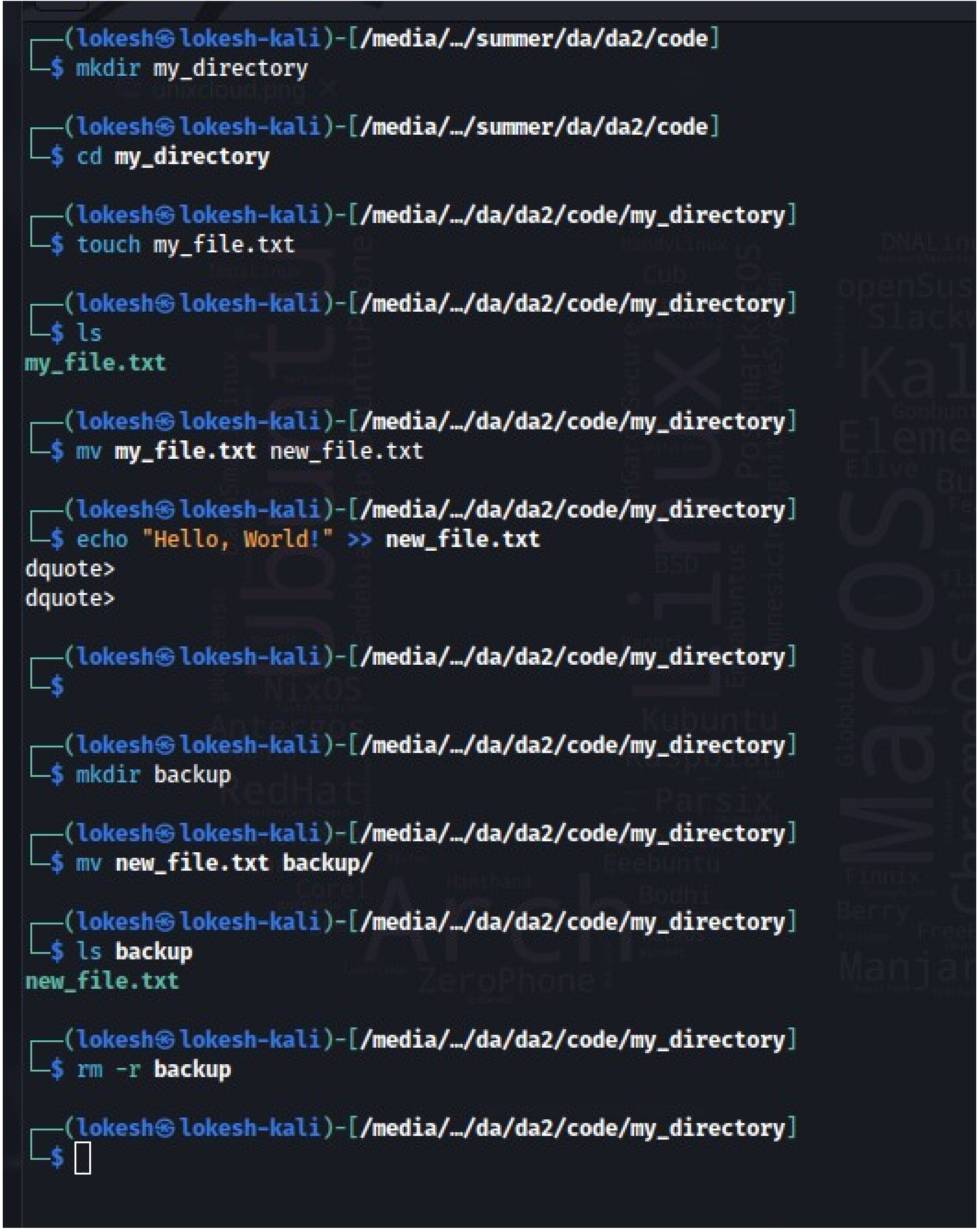
NAME	:	LOKESHWARAN M
REG NO	:	20BCE2599
DATE	:	22-5-2023
SCHOOL	:	SCOPE

Task 1: File and Directory Manipulation

Bash script :

```
#!/bin/bash

# Create a directory called "my_directory"
mkdir my_directory

# Navigate into the "my_directory"
cd my_directory

# Create an empty file called "my_file.txt"
touch my_file.txt

# List all files and directories in the current directory
ls

# Rename "my_file.txt" to "new_file.txt"
mv my_file.txt new_file.txt

# Display the content of "new_file.txt" using a pager tool (e.g., less)
less new_file.txt

# Append the text "Hello, World!" to "new_file.txt"
echo "Hello, World!" >> new_file.txt

# Create a new directory called "backup" within "my_directory"
mkdir backup

# Move "new_file.txt" to the "backup" directory
mv new_file.txt backup/

# Verify that "new_file.txt" is now located in the "backup" directory
ls backup

# Delete the "backup" directory and all its contents
rm -r backup
```

Output :



```
┌──(lokesh㉿lokesh-kali)-[/media/…/summer/da/da2/code]
└─$ mkdir my_directory

┌──(lokesh㉿lokesh-kali)-[/media/…/summer/da/da2/code]
└─$ cd my_directory

┌──(lokesh㉿lokesh-kali)-[/media/…/da/da2/code/my_directory]
└─$ touch my_file.txt

┌──(lokesh㉿lokesh-kali)-[/media/…/da/da2/code/my_directory]
└─$ ls
my_file.txt

┌──(lokesh㉿lokesh-kali)-[/media/…/da/da2/code/my_directory]
└─$ mv my_file.txt new_file.txt

┌──(lokesh㉿lokesh-kali)-[/media/…/da/da2/code/my_directory]
└─$ echo "Hello, World!" >> new_file.txt
dquote>
dquote>

┌──(lokesh㉿lokesh-kali)-[/media/…/da/da2/code/my_directory]
└─$

┌──(lokesh㉿lokesh-kali)-[/media/…/da/da2/code/my_directory]
└─$ mkdir backup

┌──(lokesh㉿lokesh-kali)-[/media/…/da/da2/code/my_directory]
└─$ mv new_file.txt backup/

┌──(lokesh㉿lokesh-kali)-[/media/…/da/da2/code/my_directory]
└─$ ls backup
new_file.txt

┌──(lokesh㉿lokesh-kali)-[/media/…/da/da2/code/my_directory]
└─$ rm -r backup

┌──(lokesh㉿lokesh-kali)-[/media/…/da/da2/code/my_directory]
└─$ ▯
```

Explanition :

This bash script performs the following tasks:

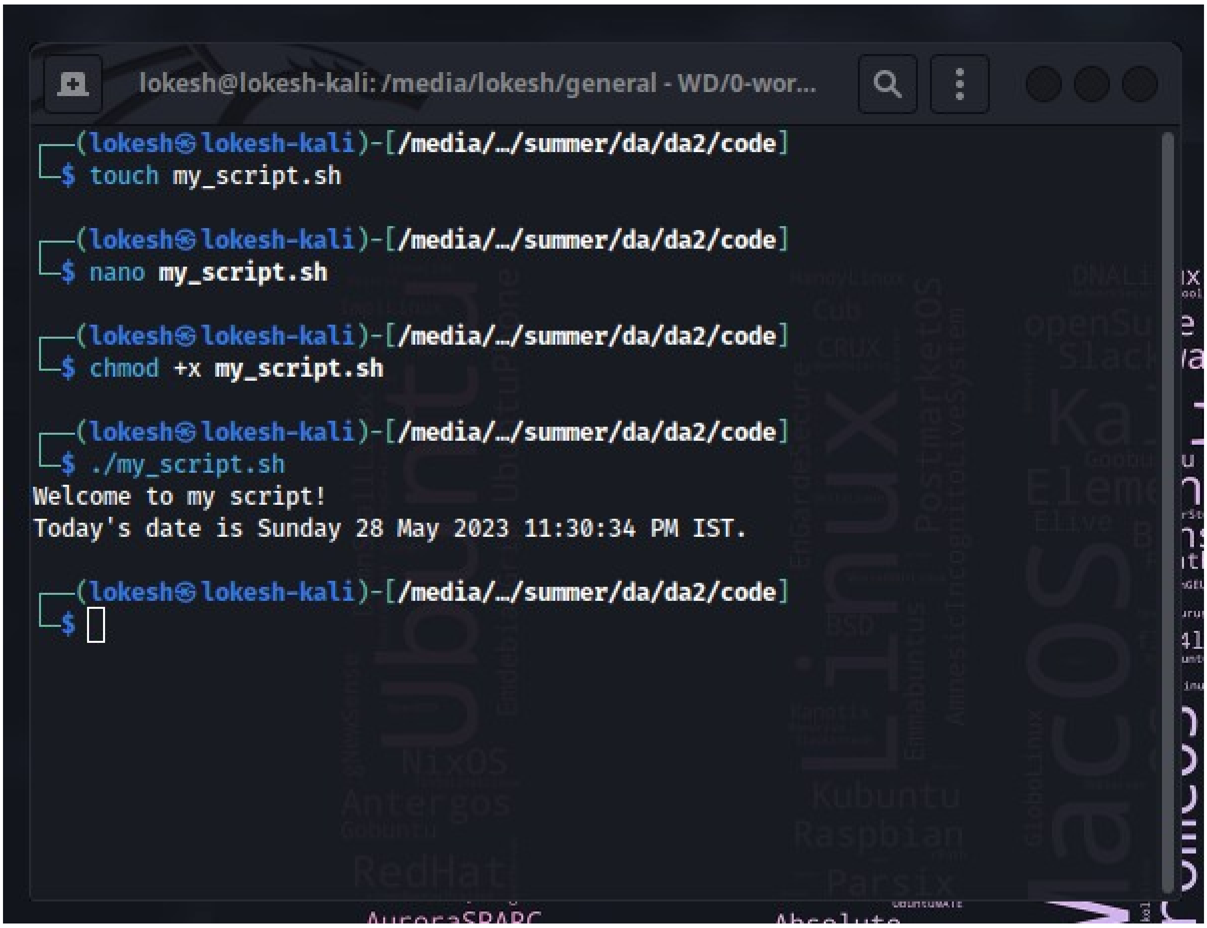
Creates a directory called "my_directory".
Navigates into the "my_directory".
Creates an empty file called "my_file.txt".
Lists all files and directories in the current directory.
Renames "my_file.txt" to "new_file.txt".
Displays the content of "new_file.txt" using a pager tool (in this example, less is used).
Appends the text "Hello, World!" to "new_file.txt".
Creates a new directory called "backup" within "my_directory".
Moves "new_file.txt" to the "backup" directory.
Verifies that "new_file.txt" is now located in the "backup" directory by listing the contents of the "backup" directory.
Deletes the "backup" directory and all its contents using the rm -r command.
Each task is executed in sequential order within the script.

Task 2: Permissions and Scripting

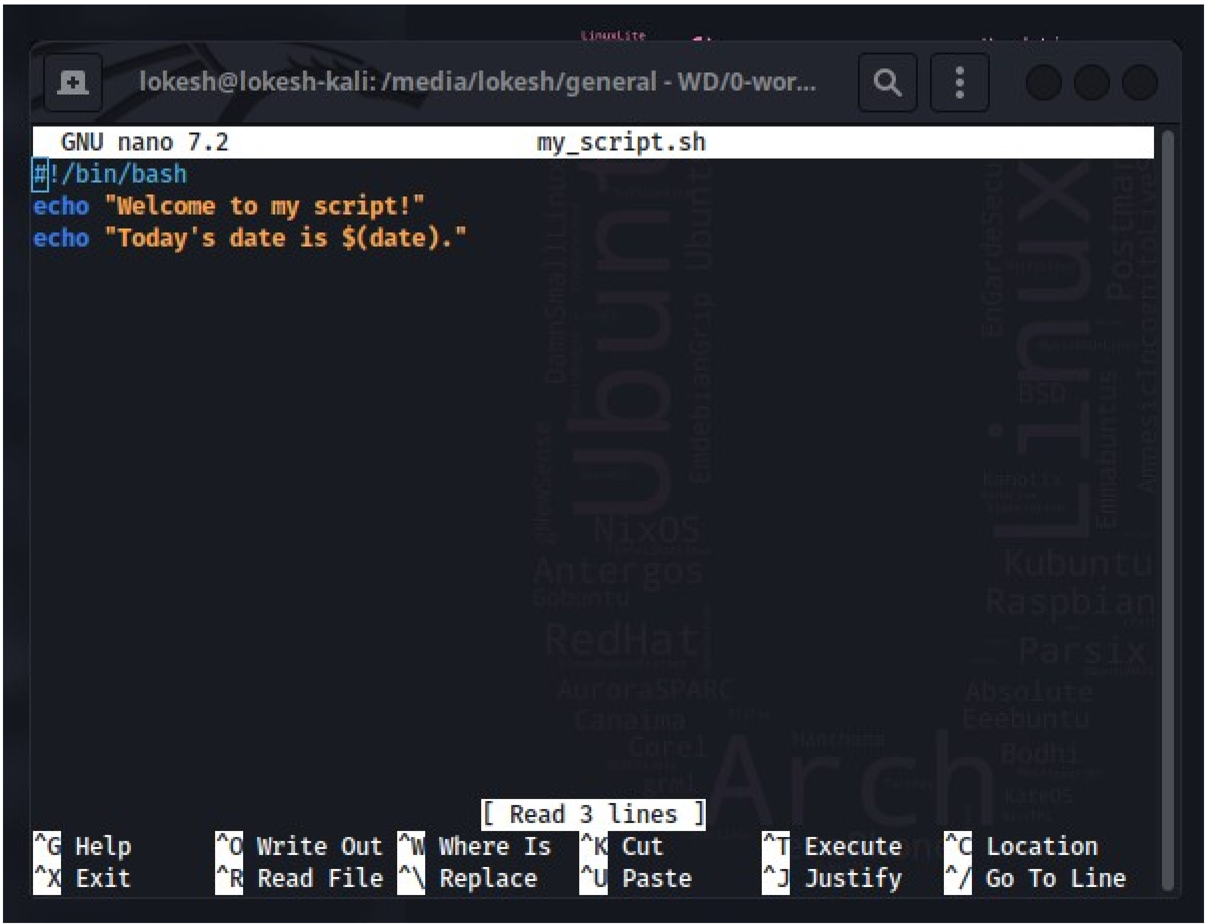
Bash script :

```
# Create a new file called "my_script.sh" using the touch command:
touch my_script.sh

# Edit "my_script.sh" using a text editor
nano my_script.sh

# Make "my_script.sh" executable using the chmod command
chmod +x my_script.sh

# Run "my_script.sh"
./my_script.sh
```

Output :



```
┌──(lokesh㉿lokesh-kali)-[/media/…/summer/da/da2/code]
└─$ touch my_script.sh

┌──(lokesh㉿lokesh-kali)-[/media/…/summer/da/da2/code]
└─$ nano my_script.sh

┌──(lokesh㉿lokesh-kali)-[/media/…/summer/da/da2/code]
└─$ chmod +x my_script.sh

┌──(lokesh㉿lokesh-kali)-[/media/…/summer/da/da2/code]
└─$ ./my_script.sh
Welcome to my script!
Today's date is Sunday 28 May 2023 11:30:34 PM IST.

┌──(lokesh㉿lokesh-kali)-[/media/…/summer/da/da2/code]
└─$ 
```



```
  GNU nano 7.2                      my_script.sh
#!/bin/bash
echo "Welcome to my script!"
echo "Today's date is $(date)."




                              [ Read 3 lines ]
^G Help        ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit        ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^/ Go To Line
```

Explanition :

Create a new file called "my_script.sh" using the touch command:

touch my_script.sh

Edit "my_script.sh" using a text editor of your choice (e.g., nano, vi, or vim):

nano my_script.sh

Add the following lines to "my_script.sh"

```
#!/bin/bash
echo "Welcome to my script!"
echo "Today's date is $(date)."
```

Save and exit the file (in nano, press Ctrl+X, followed by Y, and then Enter).

Make "my_script.sh" executable using the chmod command

chmod +x my_script.sh

Run "my_script.sh"

./my_script.sh

Verify that the output matches the expected result, which should be

Welcome to my script!
Today's date is [current date and time].

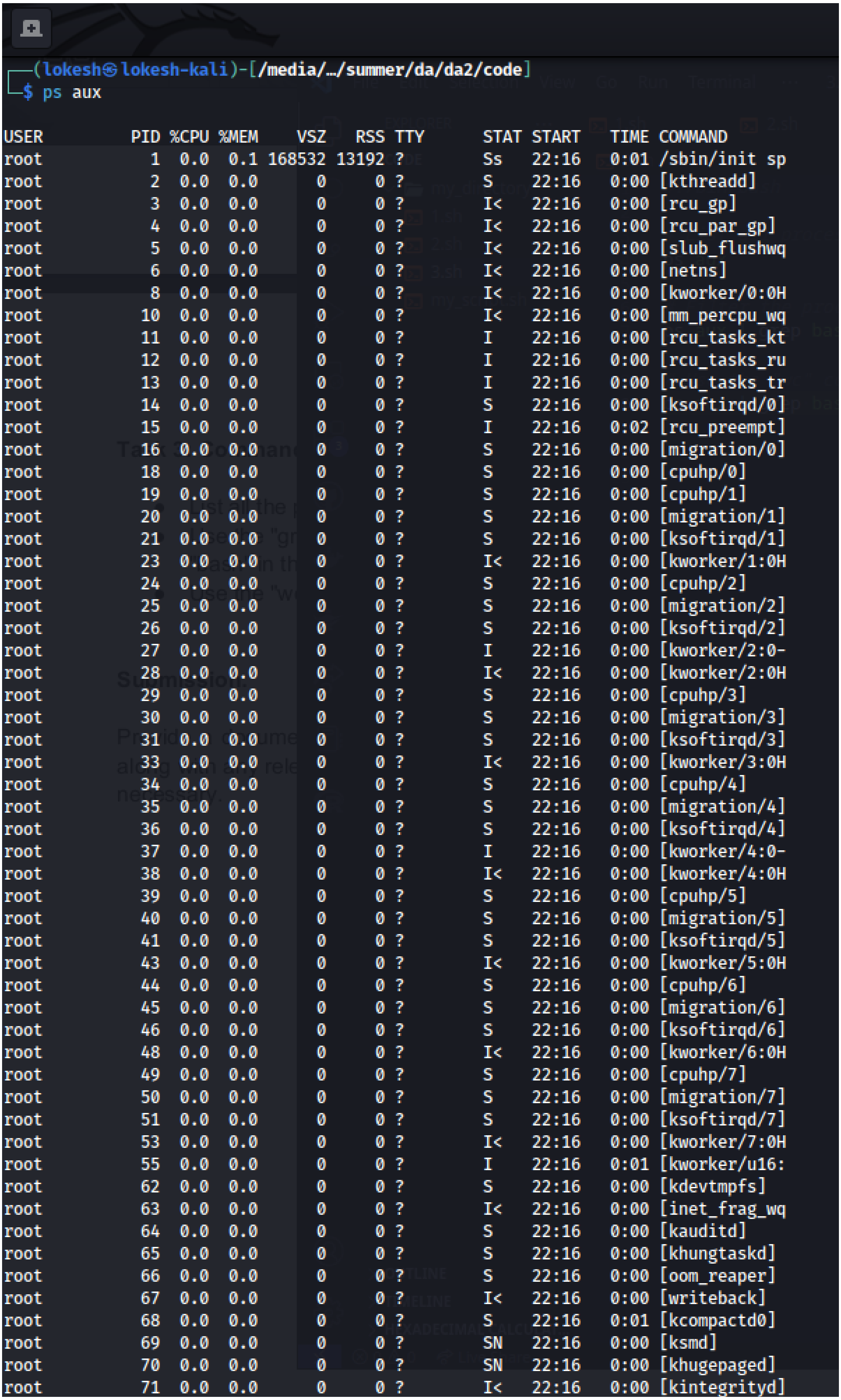
The script will display a welcome message and the current date and time when executed.

Task 3: Command Execution and Pipelines

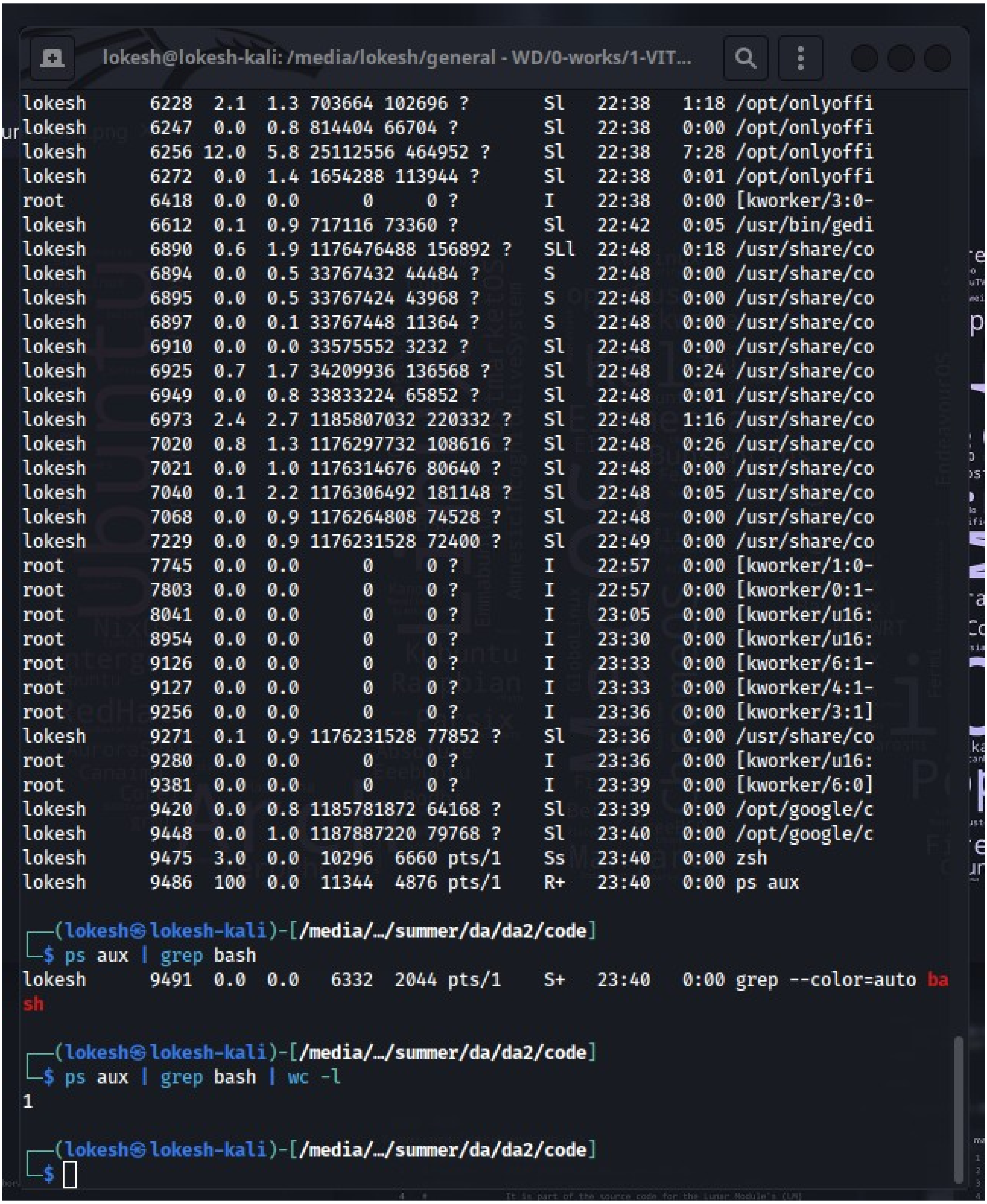
Bash script :

```
#!/bin/bash

# List all processes running on the system using the "ps" command
ps aux

# Filter the processes list and display only the processes with "bash" in their name using the "grep"
command
ps aux | grep bash

# Use the "wc" command to count the number of lines in the filtered output
ps aux | grep bash | wc -l
```

Output :



```
┌──(lokesh㉿lokesh-kali)-[/media/…/summer/da/da2/code]
└─$ ps aux

USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.0  0.1 168532 13192 ?        Ss   22:16   0:01 /sbin/init sp
root           2  0.0  0.0      0     0 ?        S    22:16   0:00 [kthreadd]
root           3  0.0  0.0      0     0 ?        I<   22:16   0:00 [rcu_gp]
root           4  0.0  0.0      0     0 ?        I<   22:16   0:00 [rcu_par_gp]
root           5  0.0  0.0      0     0 ?        I<   22:16   0:00 [slub_flushwq
root           6  0.0  0.0      0     0 ?        I<   22:16   0:00 [netns]
root           8  0.0  0.0      0     0 ?        I<   22:16   0:00 [kworker/0:0H
root          10  0.0  0.0      0     0 ?        I<   22:16   0:00 [mm_percpu_wq
root          11  0.0  0.0      0     0 ?        I    22:16   0:00 [rcu_tasks_kt
root          12  0.0  0.0      0     0 ?        I    22:16   0:00 [rcu_tasks_ru
root          13  0.0  0.0      0     0 ?        I    22:16   0:00 [rcu_tasks_tr
root          14  0.0  0.0      0     0 ?        S    22:16   0:00 [ksoftirqd/0]
root          15  0.0  0.0      0     0 ?        I    22:16   0:02 [rcu_preempt]
root          16  0.0  0.0      0     0 ?        S    22:16   0:00 [migration/0]
root          18  0.0  0.0      0     0 ?        S    22:16   0:00 [cpuhp/0]
root          19  0.0  0.0      0     0 ?        S    22:16   0:00 [cpuhp/1]
root          20  0.0  0.0      0     0 ?        S    22:16   0:00 [migration/1]
root          21  0.0  0.0      0     0 ?        S    22:16   0:00 [ksoftirqd/1]
root          23  0.0  0.0      0     0 ?        I<   22:16   0:00 [kworker/1:0H
root          24  0.0  0.0      0     0 ?        S    22:16   0:00 [cpuhp/2]
root          25  0.0  0.0      0     0 ?        S    22:16   0:00 [migration/2]
root          26  0.0  0.0      0     0 ?        S    22:16   0:00 [ksoftirqd/2]
root          27  0.0  0.0      0     0 ?        I    22:16   0:00 [kworker/2:0-
root          28  0.0  0.0      0     0 ?        I<   22:16   0:00 [kworker/2:0H
root          29  0.0  0.0      0     0 ?        S    22:16   0:00 [cpuhp/3]
root          30  0.0  0.0      0     0 ?        S    22:16   0:00 [migration/3]
root          31  0.0  0.0      0     0 ?        S    22:16   0:00 [ksoftirqd/3]
root          33  0.0  0.0      0     0 ?        I<   22:16   0:00 [kworker/3:0H
root          34  0.0  0.0      0     0 ?        S    22:16   0:00 [cpuhp/4]
root          35  0.0  0.0      0     0 ?        S    22:16   0:00 [migration/4]
root          36  0.0  0.0      0     0 ?        S    22:16   0:00 [ksoftirqd/4]
root          37  0.0  0.0      0     0 ?        I    22:16   0:00 [kworker/4:0-
root          38  0.0  0.0      0     0 ?        I<   22:16   0:00 [kworker/4:0H
root          39  0.0  0.0      0     0 ?        S    22:16   0:00 [cpuhp/5]
root          40  0.0  0.0      0     0 ?        S    22:16   0:00 [migration/5]
root          41  0.0  0.0      0     0 ?        S    22:16   0:00 [ksoftirqd/5]
root          43  0.0  0.0      0     0 ?        I<   22:16   0:00 [kworker/5:0H
root          44  0.0  0.0      0     0 ?        S    22:16   0:00 [cpuhp/6]
root          45  0.0  0.0      0     0 ?        S    22:16   0:00 [migration/6]
root          46  0.0  0.0      0     0 ?        S    22:16   0:00 [ksoftirqd/6]
root          48  0.0  0.0      0     0 ?        I<   22:16   0:00 [kworker/6:0H
root          49  0.0  0.0      0     0 ?        S    22:16   0:00 [cpuhp/7]
root          50  0.0  0.0      0     0 ?        S    22:16   0:00 [migration/7]
root          51  0.0  0.0      0     0 ?        S    22:16   0:00 [ksoftirqd/7]
root          53  0.0  0.0      0     0 ?        I<   22:16   0:00 [kworker/7:0H
root          55  0.0  0.0      0     0 ?        I    22:16   0:01 [kworker/u16:
root          62  0.0  0.0      0     0 ?        S    22:16   0:00 [kdevtmpfs]
root          63  0.0  0.0      0     0 ?        I<   22:16   0:00 [inet_frag_wq
root          64  0.0  0.0      0     0 ?        S    22:16   0:00 [kauditd]
root          65  0.0  0.0      0     0 ?        S    22:16   0:00 [khungtaskd]
root          66  0.0  0.0      0     0 ?        S    22:16   0:00 [oom_reaper]
root          67  0.0  0.0      0     0 ?        I<   22:16   0:00 [writeback]
root          68  0.0  0.0      0     0 ?        S    22:16   0:01 [kcompactd0]
root          69  0.0  0.0      0     0 ?        SN   22:16   0:00 [ksmd]
root          70  0.0  0.0      0     0 ?        SN   22:16   0:00 [khugepaged]
root          71  0.0  0.0      0     0 ?        I<   22:16   0:00 [kintegrityd]
```


lokesh@lokesh-kali: /media/lokesh/general - WD/0-works/1-VIT...

```
lokesh      6228  2.1  1.3 703664 102696 ?       Sl   22:38   1:18 /opt/onlyoffi
lokesh      6247  0.0  0.8 814404 66704 ?        Sl   22:38   0:00 /opt/onlyoffi
lokesh      6256 12.0  5.8 25112556 464952 ?     Sl   22:38   7:28 /opt/onlyoffi
lokesh      6272  0.0  1.4 1654288 113944 ?      Sl   22:38   0:01 /opt/onlyoffi
root        6418  0.0  0.0      0     0 ?        I    22:38   0:00 [kworker/3:0-
lokesh      6612  0.1  0.9 717116 73360 ?        Sl   22:42   0:05 /usr/bin/gedi
lokesh      6890  0.6  1.9 1176476488 156892 ?   SLl  22:48   0:18 /usr/share/co
lokesh      6894  0.0  0.5 33767432 44484 ?      S    22:48   0:00 /usr/share/co
lokesh      6895  0.0  0.5 33767424 43968 ?      S    22:48   0:00 /usr/share/co
lokesh      6897  0.0  0.1 33767448 11364 ?      S    22:48   0:00 /usr/share/co
lokesh      6910  0.0  0.0 33575552 3232 ?       Sl   22:48   0:00 /usr/share/co
lokesh      6925  0.7  1.7 34209936 136568 ?     Sl   22:48   0:24 /usr/share/co
lokesh      6949  0.0  0.8 33833224 65852 ?      Sl   22:48   0:01 /usr/share/co
lokesh      6973  2.4  2.7 1185807032 220332 ?   Sl   22:48   1:16 /usr/share/co
lokesh      7020  0.8  1.3 1176297732 108616 ?   Sl   22:48   0:26 /usr/share/co
lokesh      7021  0.0  1.0 1176314676 80640 ?    Sl   22:48   0:00 /usr/share/co
lokesh      7040  0.1  2.2 1176306492 181148 ?   Sl   22:48   0:05 /usr/share/co
lokesh      7068  0.0  0.9 1176264808 74528 ?    Sl   22:48   0:00 /usr/share/co
lokesh      7229  0.0  0.9 1176231528 72400 ?    Sl   22:49   0:00 /usr/share/co
root        7745  0.0  0.0      0     0 ?        I    22:57   0:00 [kworker/1:0-
root        7803  0.0  0.0      0     0 ?        I    22:57   0:00 [kworker/0:1-
root        8041  0.0  0.0      0     0 ?        I    23:05   0:00 [kworker/u16:
root        8954  0.0  0.0      0     0 ?        I    23:30   0:00 [kworker/u16:
root        9126  0.0  0.0      0     0 ?        I    23:33   0:00 [kworker/6:1-
root        9127  0.0  0.0      0     0 ?        I    23:33   0:00 [kworker/4:1-
root        9256  0.0  0.0      0     0 ?        I    23:36   0:00 [kworker/3:1]
lokesh      9271  0.1  0.9 1176231528 77852 ?    Sl   23:36   0:00 /usr/share/co
root        9280  0.0  0.0      0     0 ?        I    23:36   0:00 [kworker/u16:
root        9381  0.0  0.0      0     0 ?        I    23:39   0:00 [kworker/6:0]
lokesh      9420  0.0  0.8 1185781872 64168 ?    Sl   23:39   0:00 /opt/google/c
lokesh      9448  0.0  1.0 1187887220 79768 ?    Sl   23:40   0:00 /opt/google/c
lokesh      9475  3.0  0.0  10296  6660 pts/1    Ss   23:40   0:00 zsh
lokesh      9486  100  0.0  11344  4876 pts/1    R+   23:40   0:00 ps aux

┌──(lokesh㉿lokesh-kali)-[/media/…/summer/da/da2/code]
└─$ ps aux | grep bash
lokesh      9491  0.0  0.0   6332  2044 pts/1    S+   23:40   0:00 grep --color=auto ba
sh

┌──(lokesh㉿lokesh-kali)-[/media/…/summer/da/da2/code]
└─$ ps aux | grep bash | wc -l
1

┌──(lokesh㉿lokesh-kali)-[/media/…/summer/da/da2/code]
└─$ 
```

Explanition :

This bash script performs the following tasks:

Lists all processes running on the system using the "ps" command with the "aux" options.

Filters the processes list using the "grep" command to display only the processes with "bash" in their name.
Uses the "wc" command with the "-l" option to count the number of lines in the filtered output.

Each task is executed sequentially within the script. The filtered output will contain only the processes with "bash" in their name, and the final output will provide the count of those processes.