# Name: Gaurav Jaiswal
# Reg no: 20BKT0028

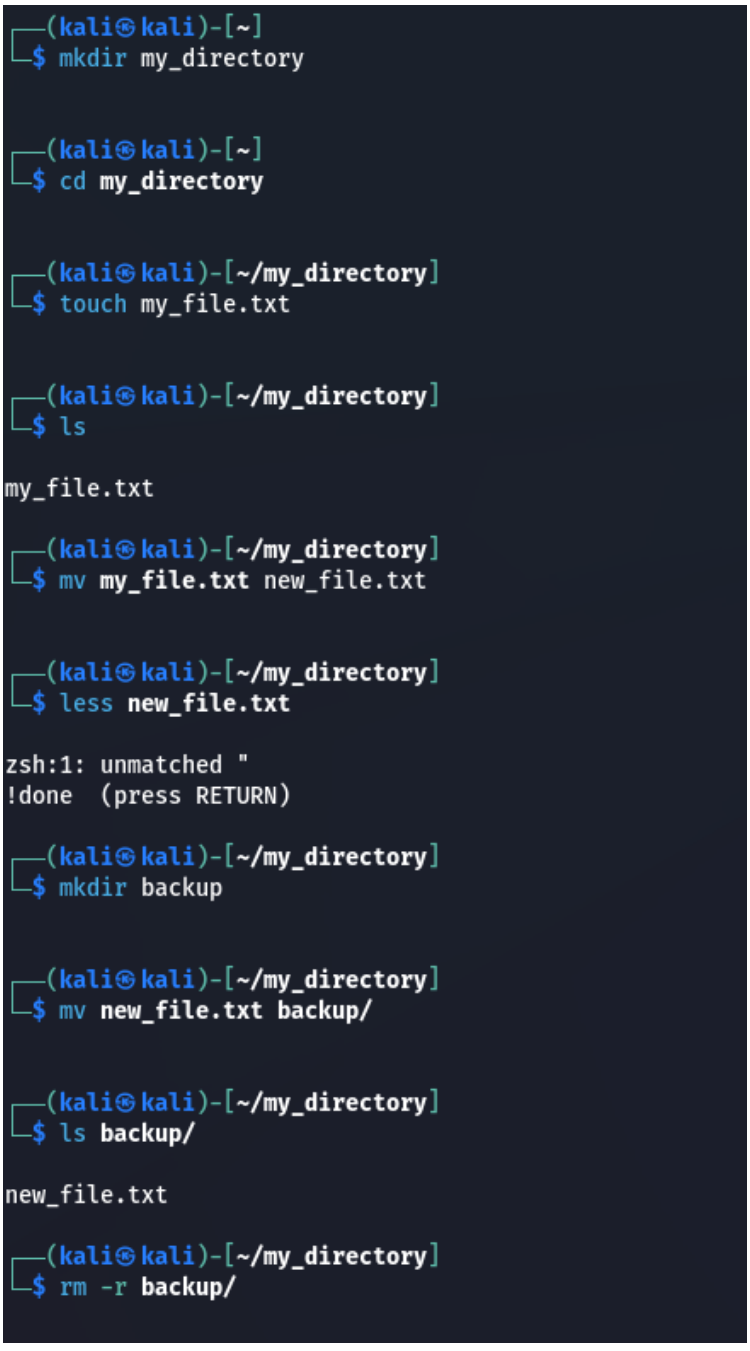
# CYBER SECURITY AND ETHICAL HACKING ASSESSMENT 2

### Task 1: File and Directory Manipulation

**COMMANDS:**

```
mkdir my_directory
cd my_directory
touch my_file.txt
ls
mv my_file.txt new_file.txt
less new_file.txt
echo "Hello, World!" >> new_file.txt
mkdir backup
mv new_file.txt backup/
ls backup/
rm -r backup/
```

**OUTPUT:**




**EXPLANATION:**

mkdir my_directory: This command creates a new directory called "my_directory" in the current location.

cd my_directory: This command allows you to navigate into the "my_directory" directory.

touch my_file.txt: This command creates an empty file called "my_file.txt" in the current directory.

ls: This command lists all the files and directories in the current directory.

mv my_file.txt new_file.txt: This command renames the file "my_file.txt" to "new_file.txt".

less new_file.txt: This command displays the content of "new_file.txt" using the pager tool "less". You can scroll through the content using the arrow keys, and press "q" to exit.

echo "Hello, World!" >> new_file.txt: This command appends the text "Hello, World!" to the file "new_file.txt".

mkdir backup: This command creates a new directory called "backup" within the "my_directory" directory.

mv new_file.txt backup/: This command moves the file "new_file.txt" to the "backup" directory.

ls backup/: This command lists the files and directories in the "backup" directory to verify that "new_file.txt" is now located there.

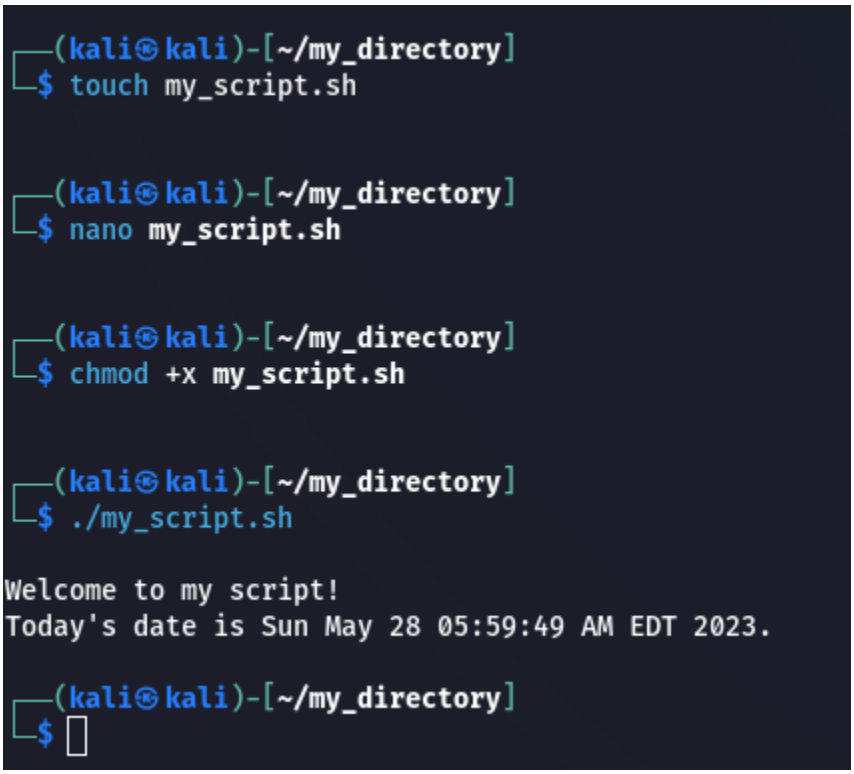
rm -r backup/: This command deletes the "backup" directory and all its contents recursively (the "-r" flag is used to remove directories).

## Task 2: Permissions and Scripting

**COMMANDS:**

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

**OUTPUT:**




**EXPLANATION:**

touch my_script.sh: This command creates a new file called "my_script.sh" in the current directory.

nano my_script.sh: This command opens the file "my_script.sh" in the nano text editor, allowing you to add and edit the script's content. You can use any text editor of your choice.

#!/bin/bash, echo "Welcome to my script!", echo "Today's date is $(date).": These lines are added to the "my_script.sh" file. The first line is a shebang that specifies the interpreter to be used (in this case, bash). The subsequent lines are commands to be executed when the script is run.

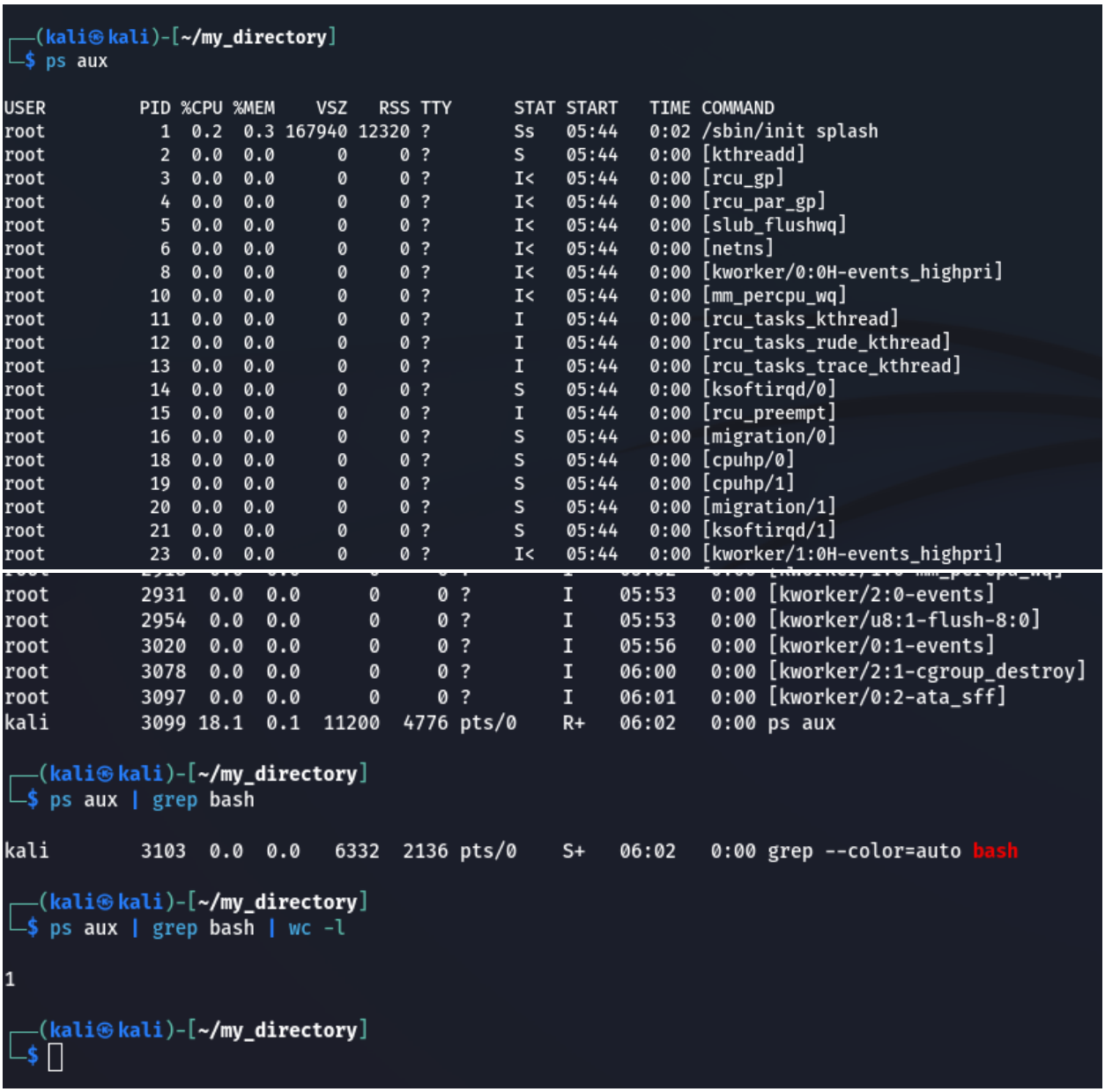
chmod +x my_script.sh: This command makes the "my_script.sh" file executable by adding execute permissions.

./my_script.sh: This command runs the "my_script.sh" script, executing the commands within it and displaying the output.

## Task 3: Command Execution and Pipelines

**COMMANDS:**
ps aux
ps aux | grep bash
ps aux | grep bash | wc -l

**OUTPUT:**



**EXPLANATION:**
ps aux: This command lists all the processes running on your system.

grep bash: This command filters the output of the previous command and displays only the processes that have "bash" in their name.

wc -l: This command counts the number of lines in the filtered output, providing the total count of processes with "bash" in their name.