**ASSESSMENT- 2**

**Name: Sachin Bharti**

**Mail:** [**sachin.bharti2020@vitstudent.ac.in**](mailto:sachin.bharti2020@vitstudent.ac.in)

**Reg. No: 20BCI0117**

**Assignment: Bash Shell Basics**

**Task 1: File and Directory Manipulation**

1. Create a directory called "my\_directory".

***mkdir my\_directory***

1. Navigate into the "my\_directory".

***cd my\_directory***

1. Create an empty file called "my\_file.txt".

***touch my\_file.txt***

1. List all the files and directories in the current directory.

***ls***

1. Rename "my\_file.txt" to "new\_file.txt".

***mv my\_file.txt new\_file.txt***

1. Display the content of "new\_file.txt" using a pager tool of your choice.

***less new\_file.txt***

1. Append the text "Hello, World!" to "new\_file.txt".

***echo "Hello, World!" >> new\_file.txt***

1. Create a new directory called "backup" within "my\_directory".

***mkdir backup***

1. Move "new\_file.txt" to the "backup" directory.

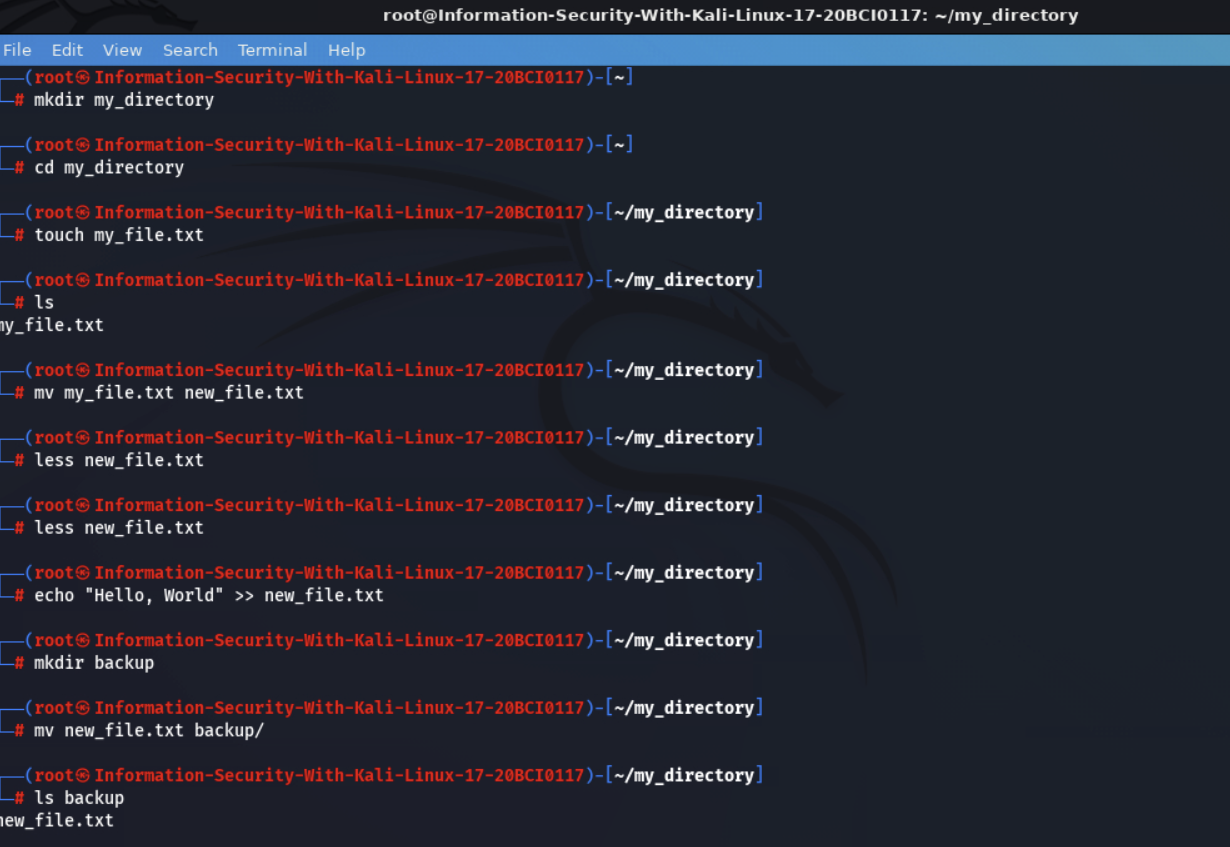
***mv new\_file.txt backup/***

1. Verify that "new\_file.txt" is now located in the "backup" directory.

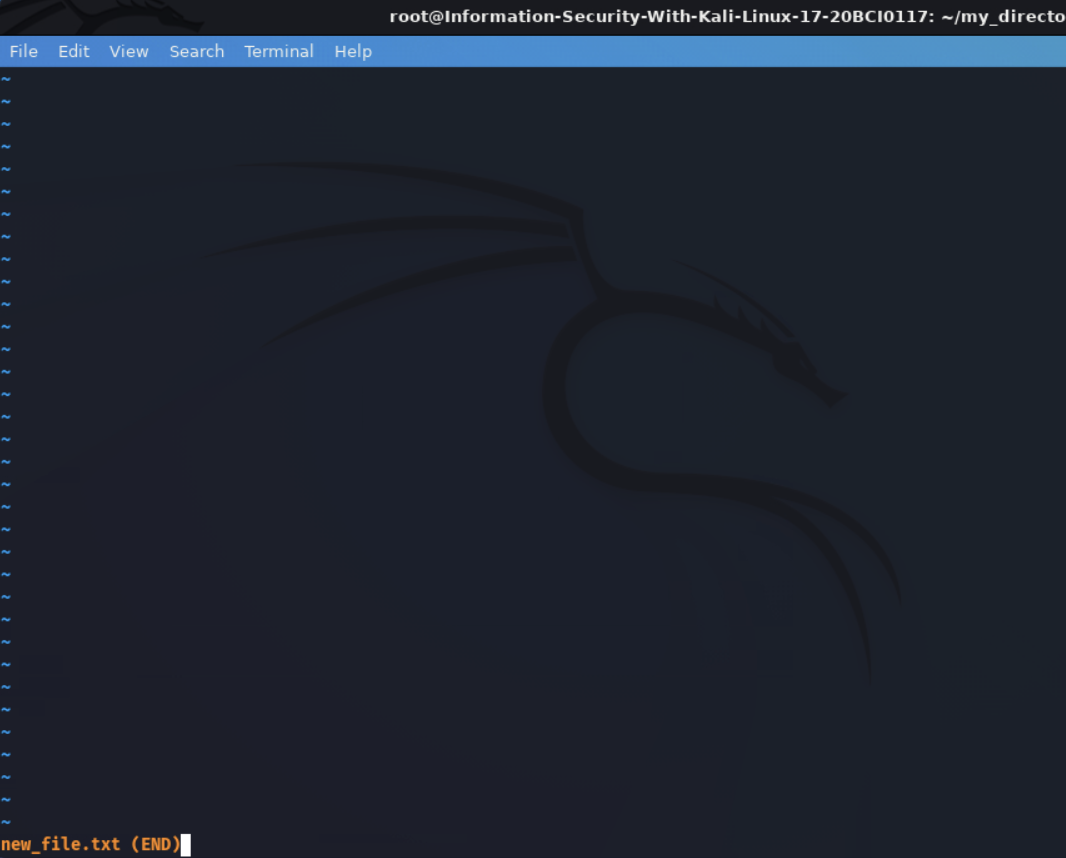
***ls backup***

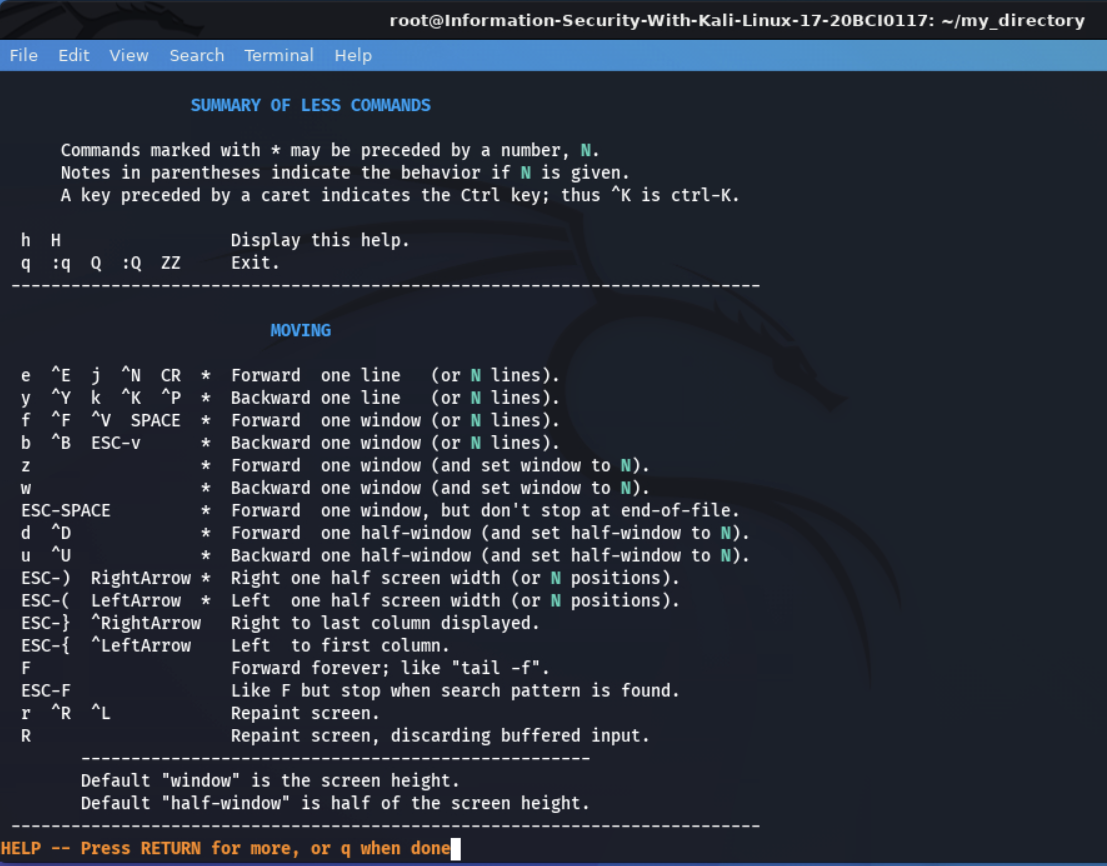
1. Delete the "backup" directory and all its contents.

***rm -r backup***

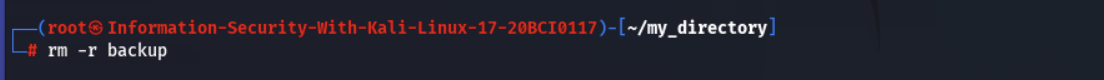


Output for *less new\_file.txt:*

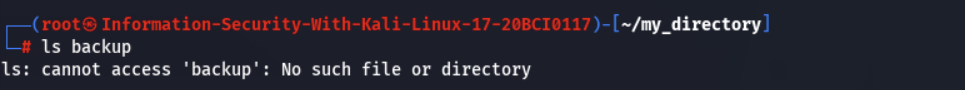




Deleting Backup:



Showing the proof of deleted backup:



**Task 2: Permissions and Scripting**

* Create a new file called "my\_script.sh".

***touch my\_script.sh***

* Edit "my\_script.sh" using a text editor of your choice and add the following lines:

**bash**

**#!/bin/bash**

**echo "Welcome to my script!"**

**echo "Today's date is $(date)."**

**Save and exit the file.**

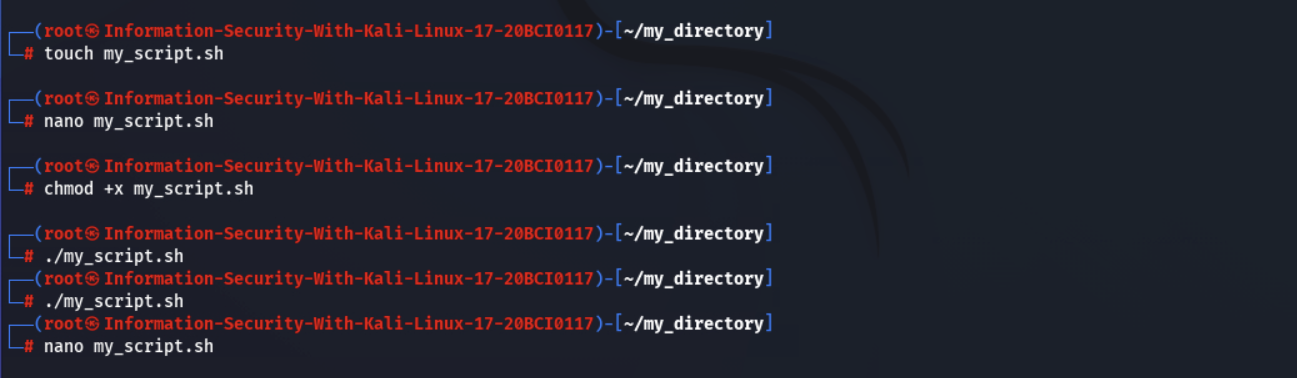
***nano my\_script.sh***

* Make "my\_script.sh" executable.

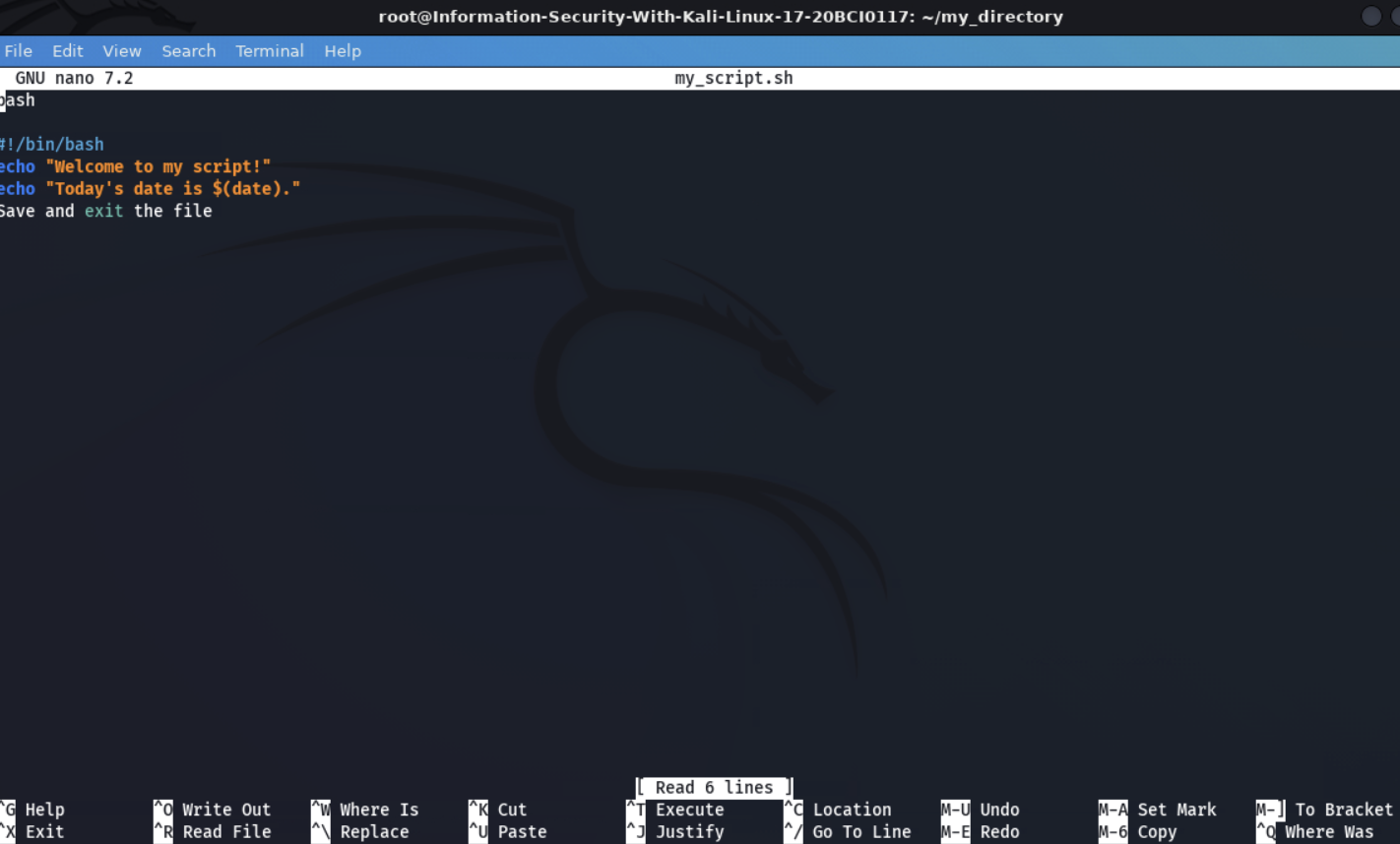
***chmod +x my\_script.sh***

* Run "my\_script.sh" and verify that the output matches the expected result.

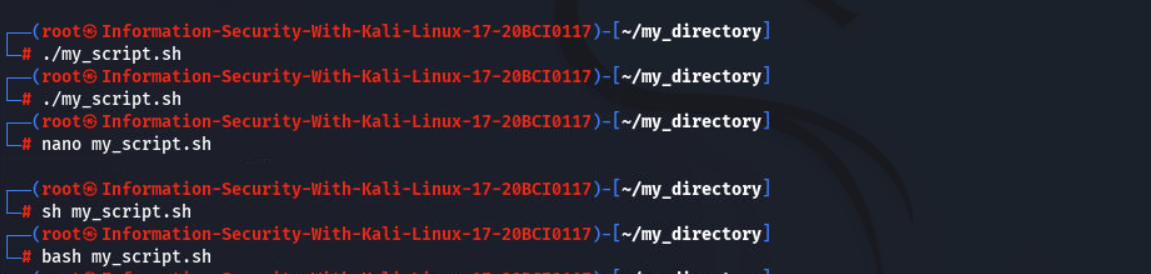
***./my\_script.sh***



**Writing inside “my\_script.sh” using *nano my\_script.sh***



**I tried all the commands to run the “my\_script.sh” but didn’t get any output.**

****

**Task 3: Command Execution and Pipelines**

* List all the processes running on your system using the "ps" command.

***ps aux***

* Use the "grep" command to filter the processes list and display only the processes with "bash" in their name.

***ps aux | grep bash***

* Use the "wc" command to count the number of lines in the filtered output.

***ps aux | grep bash | wc***

